A Machine Learning Approach to a Multidetector Array Response Function for Nuclear Search

David Fobar, W. J. Vanderlip, William Koch, *Member, IEEE*, and Peter Chapman

***Abstract*— During nuclear search operations, the localization of radioactive sources can be a time-consuming process that requires mapping relative radiation intensity in a large area to determine the position of a source. This article introduces the use of machine learning, specifically a temporal convolutional network (TCN), to estimate the direction between a detector array and a static ^{137}Cs source. This application of machine learning provides a directional vector in 4π with a 90% confidence of 5.6° and a 99% confidence within 11.2°. With the use of low-cost NaI(Tl) detectors, the effects of self-shielding within the array creates gamma-ray shadows depending on the orientation to the source. Using the convolved detector array response function, we apply supervised machine learning with a neural network to predict a unit vector that points toward the observed source. The directional vector is expected to reduce search times once implemented in future work.**

***Index Terms*— 3-D sensors, detector arrays, neural networks.**

I. Introduction

As RADIATION detector technology continues to develop, sensor packages continue to shrink in size. This evolution, combined with advances in battery power, autonomous vehicle capabilities, and embedded processor power, has created opportunities to improve methods to estimate radiation source location. Recent work in this field has combined data from other sensors, such as Light Detection and Ranging (LIDAR) and Global Positioning System (GPS), in a capability called Scene Data Fusion (SDF) [1]. The combination of information using SDF creates new challenges in managing large volumes of data. This article introduces the use of a machine learning algorithm, specifically an artificial neural network (ANN), to perform calculations necessary to estimate a unit vector pointing toward an expected radiation source. This work uses a single static ^{137}Cs source to train an ANN; in subsequent work, the ANN will be further refined for multiple sources and dynamic environments using a technique called transfer learning [2]. Ultimately this work will be incorporated onto a tracked robotic platform whose task is to semi-autonomously traverse a potentially hazardous environment and map the radiation field and localize point sources of radiation.

Manuscript received 23 March 2022; revised 19 May 2022; accepted 13 June 2022. Date of publication 30 June 2022; date of current version 16 August 2022. This work was supported by the Defense Threat Reduction Agency's (DTRA) Nuclear Science and Engineering Research Center (NSERC), West Point, NY.

The authors are with the Department of Physics and Nuclear Engineering, United States Military Academy, West Point, NY 10996 USA (e-mail: david.fobar@westpoint.edu; william.vanderlip@westpoint.edu; william.koch@westpoint.edu; peter.chapman@westpoint.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNS.2022.3186831.

Digital Object Identifier 10.1109/TNS.2022.3186831

A. Detector Array Directional Response

It has been previously demonstrated that an array of three to four detectors can be used to provide a general direction to a radiation source. These works compare different detector count rates in the array to estimate a direction using a maximum likelihood estimate (MLE) algorithm [3], [4]. Other efforts have relied on multiple passes in an environment to map the radiation field and identify areas likely to have a radiation source [5], [6]. Compton imaging is another technique that can provide a directional response [7], [8]. However, this response is based on individual event analysis, rather than integrating counts as done in this study. Directional detection systems whose localization is not based on mapping the radiation field can take advantage of dynamic reconstruction to improve their directionality performance [9]. It is not clear if Britt's work [4] leverages dynamic reconstruction for the static detector and moving source test case where localization is reported to achieve ($14° \pm 20°$). This work will use Britt's MLE algorithm static detector and static source, with a mean absolute error of ($24.9° \pm 0.5°$) in directionality prediction, as a benchmark for comparison.

II. General Approach

A. Detector Array and Its Response

We have arranged five thallium-doped sodium iodide [NaI(Tl)] detectors, four in the cardinal directions, with one central detector recessed below the others (4+1). The use of a central detector attempts to increase the vertical sensitivity beyond that of a basic four-detector array. The outer detectors are 2" × 4" × 4" NaI(Tl), and the center detector is a 3" × 3" NaI(Tl). The four detectors are Bicron 2 × 4M6/2 detectors, and the central detector is a Bicron 802-3W. Each of the detectors have the photomultiplier tube integrated with the crystal. Bridgeport Instruments eMorpho USB MCAs are attached to each detector to acquire data from the array. Each detector is gain matched prior to data collection and has between 8% and 11% energy resolution at 662 keV.

Using an MCNP-PoliMi simulation [11] for a proof of principle, Fig. 1 demonstrates 1e6 simulated particles that originate from two different ^{137}Cs point sources 45 cm from the center of the array. The self-shielding within the detector

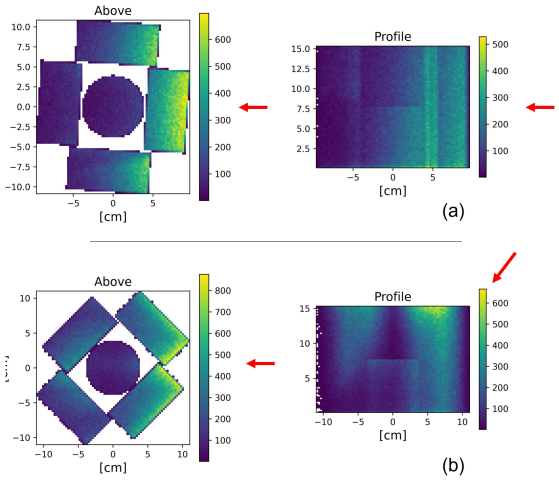

Fig. 1. Simulated density of interactions within the 4+1 detector array shown with the source in two different positions, indicated by the red arrow. In both (a) and (b), the left image is a top-down view of the detector array while the right image is viewing the array from the side. Image (b) has rotated the detector array by approximately 45° while raising the source so that it is no longer horizontal with the array.

array is demonstrated by the variation in density of photon interactions causing more than 10 keV of energy deposition. Image (a) demonstrates when a source is mostly exposed to a single detector, while image (b) has the source raised such that some of the photons have a clear path to the previously shielded detectors.

This 4+1 detector arrangement creates a complicated array response function where each detector has a unique response strongly dependent on the source's angular position in relation to the array. Fig. 2 shows how a single detector's absolute efficiency changes as the source is moved about the array. The two plots demonstrate the 662-keV photopeak relative count rates from a 30 μCi ^{137}Cs source held at many positions in 4π about the array. The individual detector's efficiency is at a maximum when openly exposed to the source from the (0°, 0°) direction (red arrow) and lower when "shadowed" by the other detectors in the array. The remainder of this work uses the entire energy deposition spectrum in order to estimate the source direction, not just the photopeak count rate.

While Fig. 2 demonstrates a single detector, the strength of the presented system is multiple detectors being taken into account with each prediction. Fig. 3 maps how the source in a static position is observed by each detector in the 4+1 array. As the source is moved about the detector array, the composite image of each detector's energy deposition histogram provides the information necessary to determine the source direction.

B. Machine Learning

1) ANNs for Directionality: ANNs have become an increasingly popular way to solve challenging problems. It is common for ANNs to be trained to provide either discrete response (particle classification) or a regression response that can be further manipulated to provide a meaningful continuous output. The choice between classification and regression is problem-dependent [12]. In the case of predicting directionality, a continuous output function is desired, thus we chose a regression output. Additionally, there are many ways to structure an ANN. Historically, long short-term memory (LSTM) [13] or gated-recurrent units (GRU) [14] have been used on problems with a temporal component. Recently, temporal convolutional networks (TCNs) have demonstrated competitive performance with a reduced training cost [15]. Through training, a TCN is able to take into account previous observations when making a prediction on the currently presented information [16]. As work on this project continues, the inclusion of temporal information is expected to enhance the networks ability to provide a direction vector toward the source while the detector array is in motion.

Initial thoughts for applying machine learning were to have a neural network observe real-time list mode data (nanosecond timestamp resolution and 16-bit pulse height value), in the form of a LSTM or GRU network. While this network architecture provided reasonable results, the amount of data and computation required for training and prediction were daunting, which led to the use of the TCN architecture. Using a TCN, we are able to present the network with a single sample that provides the last 15 s of energy deposited in the form of 1 s histograms for each detector. Another way to describe the network input; we provide a 2-D image each second and the network remembers the previous 14 s worth of events. Each detector's histogram is displayed as a row of pixels, thus if the energy deposited histogram has 100 bins, the images are in the shape of 5×100 pixels. Therefore, one could consider the network input as being a 15 s video with a 1 Hz refresh rate. The value of 15 s allows for many unique ground truth samples for each 60 s of experimental data collection. Testing was only completed using the ground truth samples, while the training data were randomly generated as discussed in the experimental setup section of this article. It is anticipated that this refresh rate will need to become dependent on the detector count rate when sources are either not present, or far away.

The output of the network results in a Cartesian unit vector, described by the x, y, and z points on a unit sphere. The use of a Cartesian as opposed to spherical coordinate system allows for seamless mapping across 4π. The use of spherical coordinates (azimuthal θ and polar ϕ angles) as the ANN output layer incorrectly trains the network as if there is a significant physical difference between the azimuthal angles of 1° and 359°, along the prime meridian. This is remedied by mapping spherical to Cartesian coordinates (x, y, and z) within a range of [−1, 1] using (1)–(3). ϕ is defined as the north and south poles being 90° and −90°, respectively

$$x = \cos(\theta)\cos(\phi) \tag{1}$$

$$y = \sin(\theta)\cos(\phi) \tag{2}$$

$$z = \sin(\phi). \tag{3}$$

2) Network Design: As depicted in Fig. 4, each of the 15 time-steps being evaluated are passed through identical parallel networks comprised of two hidden dense layers with rectified linear unit (ReLU) activation functions [17] using the Keras library [18] for Python [19]. The first layer has 50 nodes while the second has ten nodes. At this stage, only a single set of trained weights and biases are applied

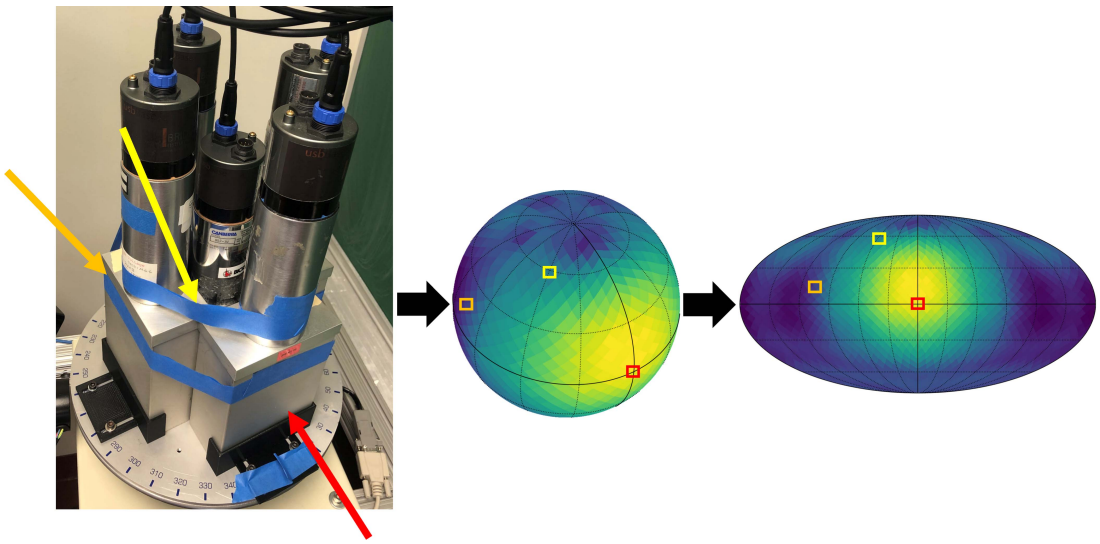

Fig. 2. Single detector's relative full energy deposition angular response as a result of the 4+1 detector array. Three different source angular positions are highlighted with three different color arrows/boxes. The oval plot is the complete 4π angular response plotted onto a sphere, which is then flattened in a way to distort the surface in a way that maintains constant cell area regardless of latitudinal position, commonly described as a Mollweide projection [10].

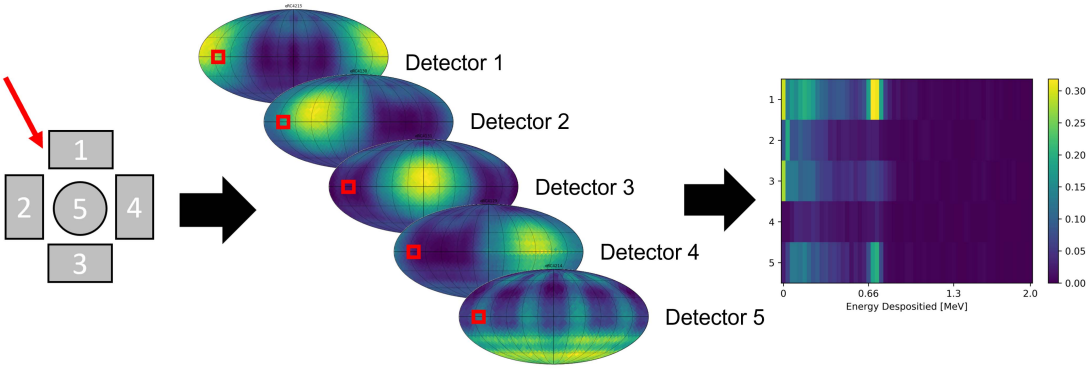

Fig. 3. Demonstration of the array's directional response as observed by the ANN. The red box on each Molleweide projection highlights where from the source is being observed for each detector. Each detector's complete energy deposition histogram is combined as a 2-D array to be passed to the neural network.

to each time-step of the sample. Next, the TCN portion of the network applies separate weights and biases for each time step. Each node of the TCN also uses a ReLU activation function. This allows for the network to create a weighted convolution of previous information to inform the prediction of the current time step. The TCN layer provides the network prediction (output) through three linear regression nodes, each one trained to provide the x, y, and z coordinates of the source on a unit sphere.

III. Experimental Procedures

The largest challenge facing most supervised machine learning problems is the collection of sufficient labeled data. In order to provide the data, a 4π data collection apparatus was constructed.

A. 4π Experiment Setup

The experiment test stand, shown in Fig. 5, has two axes of rotation about the detector array. The first axis is the rotation table that the detector array rests on; it is capable of rotating 360° in the horizontal plane. The second axis controls the source arm; it has an axis of rotation perpendicular to the rotational axis of the table. The source arm allows the 30 μCi ^{137}Cs source to be placed anywhere along a 180° arc at a

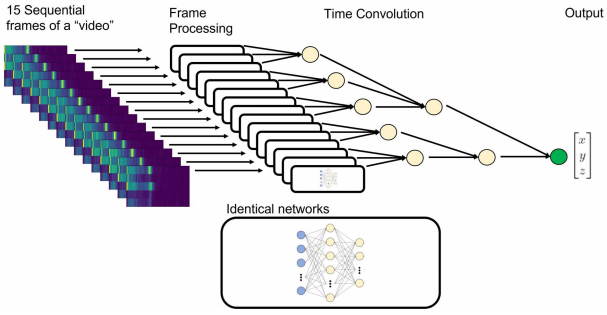

Fig. 4. Sequence of 15 frames being processed by the ANN. Each frame is processed by identical networks before entering the time convolution. The output of the network results in a regression along three axes, x, y, and z.

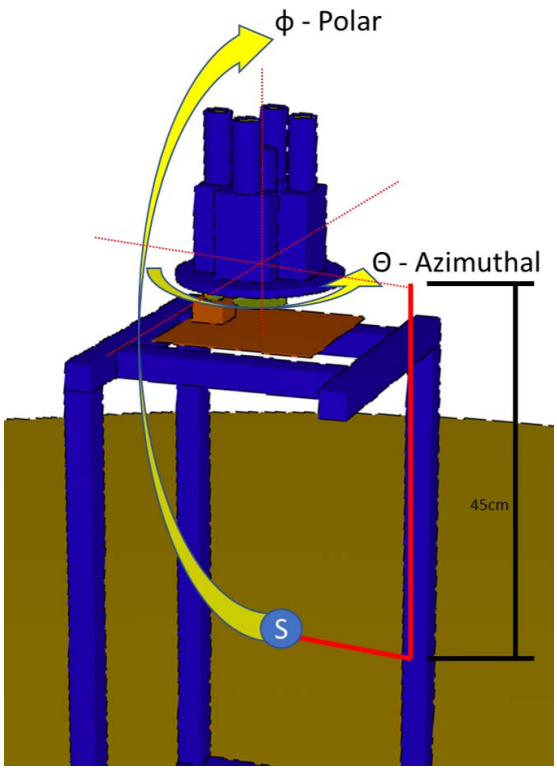

Fig. 5. This image demonstrates the experimental setup used to gather data in all 4π about the detector array. Two axes are used to rotate the detector array and articulate the source arm. The source is fixed to the red rotary arm at the position marked "S." Through an azimuthal rotation of the array and polar rotation of the source, the source can be placed at any angular position on a unit sphere about the detector array. The two rotational axes are controlled through stepper motors and an Arduino that interacts with a python script.

constant 45 cm distance from the base/center of the rotation table. Through the rotational motion of the detector table, and the positioning of the source, we are able to place the source at any fixed angular position in 4π about the detector array. The two separate stepper motors are controlled by an Arduino Uno. The microcontroller receives position instructions from a Python script, which also records incoming list mode detector data. The positioning of the source is determined using the HEALPix [20] algorithm implemented by the healpy python library [21]. The healpy variable *nside* is set to 12 for generating the data collection source locations, which results in 1728 unique source positions for data collection. Each position is represented by the cells shown on the surface of the sphere in Fig. 2. The source was held at a positions described by the unit vector pointing toward the center of those cells.

While the samples provided to the neural network are histograms, the data recorded by the eMorpho MCAs is list-mode. This method of data collection allows for additional random sampling of prerecorded data to generate unique 1-s histograms, which may then be collated into the 15-s training samples. The random sampling process provides a large training dataset without compromising the performance of the ANN. Validation and testing of the ANN strictly uses ground truth data where no random sampling takes place.

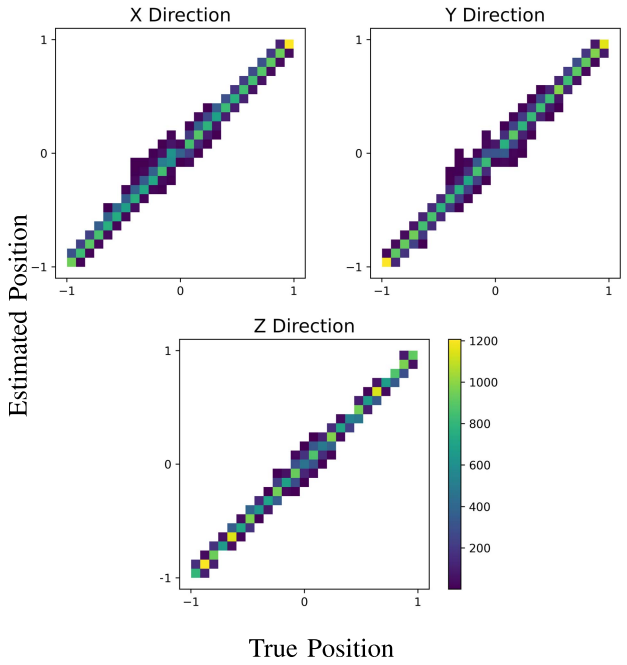

Fig. 6. This series of images demonstrates the ability of the network to correctly predict (y-axis) the sources true position (x-axis) in each of the Cartesian coordinates. The vertical axis indicates the model's position estimate and the horizontal axis indicates the true position.

B. Assessment of Accuracy and Uncertainty

In order to determine accuracy, 172 800 unique ground truth 15-s sample were presented to the model, 100 samples from each experimental source location. The network provided a unit vector estimate ($\hat{r}$) to the source location and this estimate was compared to the true unit vector ($\hat{R}$) to provide an absolute angle displacement ($\Delta\Phi$) using the following equation:

$$\Delta\Phi = \cos^{-1}\left(\hat{r} \cdot \hat{R}\right). \tag{4}$$

IV. Results

Fig. 6 demonstrates the output of the ANN as a function of the sources true position. The linear nature of the plots demonstrate that the ANN predicts a sources true position using Cartesian coordinates.

A histogram showing the azimuthal and polar errors is shown in Fig. 7. With a Gaussian function fit to each distribution, it is shown that the ANN is able to predict within 3.5° in either direction with 95% accuracy (2σ).

An alternative method of demonstrating the accuracy is shown in the following angular displacement $\Delta\Phi$ histogram and cumulative distribution plots in Fig. 8. The cumulative distribution enables an assessment on the overall accuracy of the ANN. The presented network averages accuracy of 3.1°, with 90% of the predictions being within 5.6° and 99% within 11.2°. It is important to note that this accuracy is limited by the resolution of the data gathered. With only 1728 different positions, there are about 4.9° difference between neighboring source positions.

However, the global response of angular displacement shown in Fig. 8 fails to capture which directions the ANN

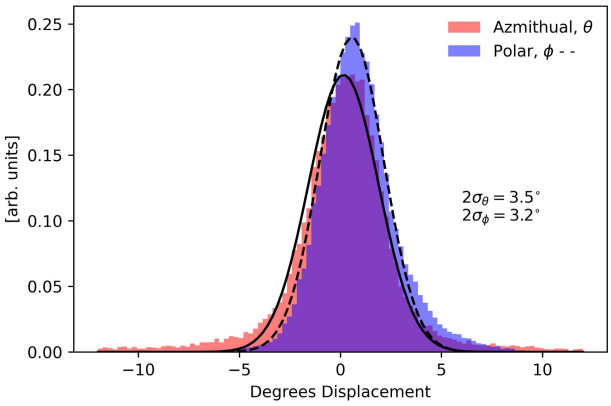

Fig. 7. Histograms representing the angular displacement in strictly the azimuthal and polar directions with a Gaussian distribution applied to provide statistical information on accuracy.

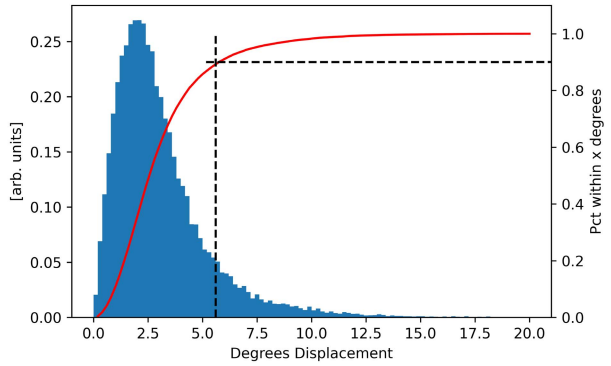

Fig. 8. This plot demonstrates the histogram of angular displacement between the predicted position and the true position as well as a cumulative distribution of angular displacement.

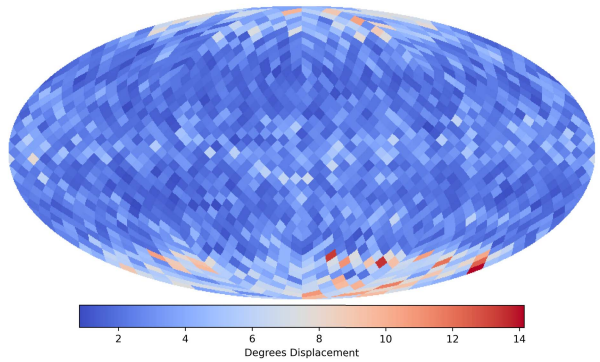

Fig. 9. This plot demonstrates the average predicted angular displacement from the sources true position.

struggles to accurately predict the direction. Fig. 9 aids in this understanding by showing the regions in which the network accuracy is challenged. The ANN's prediction capabilities is degraded about the equator and at the polar regions of the array, as highlighted in Fig. 9. The larger displacement error along the south pole likely results from the detectors resting on a circular metal plate, which causes scatters prior to radiation reaching the detector array.

V. Discussion

It is thought that the northern regions of the arrays directionality response is degraded by the limited shadowing by adjacent detectors. Along the equator, the limited performance is due to the models reduced ability to predict the azimuthal angle rather than the polar angle, this is demonstrated in Fig. 9. This likely results from small changes in azimuthal angle that do not result in significant changes in the shadowing of adjacent detectors. As for the southern polar region, the poor performance most certainly results from scattering that occurs within the test stand. The detectors rest on an aluminum plate that is supported by a steel plate. Additionally, there is a stepper motor directly under the detector array. These dense physical objects can cause gamma-rays to inadvertently enter the detector array from angles other than the sources true position, at a rate significantly higher than background. Recalling that the intended use of this array is on a tracked robotic platform, limited performance at the southern pole is not of significant concern as this region of response would only be used if the robot is directly over a point source.

During the early stages of this project, the intent was to train the model with either a combination of experimental and simulated data or solely simulated data. While the simulated directional response produces full energy deposition plots that look similar to those shown in Fig. 3, the simulated environment did not accurately include the detector housings, PMTs, and MCA bases. The poor performance at the southern pole emphasizes the importance to train our model with the most realistic conditions possible, and further demonstrates why the simulated environment struggled to effectively provide training data.

VI. Conclusion and Future Work

The presented neural network solution provides accurate position predictions for a static source with an update every second of observation. The next step in this campaign is to use a virtual test bed (a simulated environment built on SCALE, developed by Oak Ridge National Laboratory [22]). The virtual test bed uses the arrays directional response to radiation flux rather than tracking the energy deposition of individual particles through the detector array. This may provide sufficient simulated data that can be used for training. Additionally, a higher activity source is to be used from a further distance to mitigate the potential of the ANN developing its prediction from solid-angle losses. Once a new experimental environment is realized and training supplemented by simulation attained, a robust training dataset is to be developed where the source is moved through a virtual environment that includes the detector array. Finally, once the ANN is able to predict the position of a nonstatic source, the direction vector and timestamp of prediction will feed into a SDF algorithm for source localization.

Acknowledgment

This work was performed under the auspices of the Defense Threat Reduction Agency's (DTRA) Nuclear Science and Engineering Research Center (NSERC), West Point, NY, USA.

REFERENCES

[1] R. Barnowski, A. Haefner, L. Mihailescu, and K. Vetter, "Scene data fusion: Real-time standoff volumetric gamma-ray imaging," *Nucl. Instrum. Methods Phys. Res. Sec. A, Accel., Spectrom. Detectors Assoc. Equip.*, vol. 800, pp. 65–69, Nov. 2015.

[2] Y. Bengio. (2011). *Deep Learning of Representations for Unsupervised and Transfer Learning*. [Online]. Available: http://proceedings.mlr.press/v27/bengio12a.html

[3] M. J. Willis, S. E. Skutnik, and H. L. Hall, "Detection and positioning of radioactive sources using a four-detector response algorithm," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 767, pp. 445–452, Dec. 2014.

[4] C. Britt, X. Wen, H. Qi, and J. P. Hayward, "Directionality for wearable, closely packed radiation detector arrays," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 986, Jan. 2021, Art. no. 164708.

[5] K. Vetter, "Multi-sensor radiation detection, imaging, and fusion," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 805, pp. 127–134, Jan. 2016.

[6] K. Vetter *et al.*, "Gamma-ray imaging for nuclear security and safety: Towards 3-D gamma-ray vision," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 878, pp. 159–168, Jan. 2018.

[7] F. Terzioglu, P. Kuchment, and L. Kunyansky, "Compton camera imaging and the cone transform. A brief overview," *Inverse Problems*, vol. 34, no. 5, 2018.

[8] T. Baca *et al.*, "Gamma radiation source localization for micro aerial vehicles with a miniature single-detector Compton event camera," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2021, pp. 338–346.

[9] B. K. Horn, R. C. Lanza, J. T. Bell, and G. E. Kohse, "Dynamic reconstruction," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 1, pp. 193–205, Feb. 2010.

[10] J. P. Snyder, "Map projections: A working manual," *USGS Numbered Ser.-Prof. Paper*, vol. 1395, p. 385, 1987.

[11] S. A. Pozzi, E. Padovani, and M. Marseguerra, "MCNP-PoliMi: A Monte-Carlo code for correlation measurements," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 513, no. 3, pp. 550–558, Nov. 2003.

[12] A. Liaw and M. Wiener. (2002). *Classification and Regression by RandomForest*. [Online]. Available: http://www.stat.berkeley.edu/

[13] R. Dey and F. M. Salem, "Gate-variants of gated recurrent unit (GRU) neural networks," in *Proc. IEEE 60th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2017, pp. 1597–1600.

[14] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proc. SSST 8th Workshop Syntax, Semantics Struct. Stat. Transl.*, 2014, pp. 103–111.

[15] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in *Proc. Eur. Conf. Comput. Vis.* in Lecture Notes in Computer Science, vol. 9915, Aug. Sep. 2016, pp. 47–54.

[16] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.

[17] A. M. F. Agarap. (2019). *Deep Learning Using Rectified Linear Units (ReLU)*. [Online]. Available: https://github.com/AFAgarap/relu-classifier

[18] C. François. (2015). *Keras: The Python Deep Learning Library*. Keras.Io. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2018ascl.soft06022C/abstract

[19] *The Python Language Reference Python 3.10.4 Documentation*. Accessed: Mar. 2022. [Online]. Available: https://docs.python.org/3/reference/

[20] K. M. Gorski *et al.*, "HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere," *Astrophys. J.*, vol. 622, no. 2, pp. 759–771, Apr. 2005. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2005ApJ..622.759G/abstract

[21] A. Zonca *et al.*, "Healpy: Equal area pixelization and spherical harmonics transforms for data on the sphere in Python," *J. Open Source Softw.*, vol. 4, no. 35, p. 1298, Mar. 2019, doi: 10.21105/joss.01298.

[22] W. A. Wieselquist, R. A. Lefebvre, and E. M. A. Jessee, *SCALE Code System*. Oak Ridge, TN, USA: Oak Ridge National Laboratory, Radiation Safety Information Computational Center, 2019. [Online]. Available: https://www.ornl.gov/scale