Vertex Reconstruction of Neutrino Interactions using Deep Learning

Adam M. Terwilliger*, Gabriel N. Perdue†, David Isele‡, Robert M. Patton§ and Steven R. Young§
*Grand Valley State University, Allendale, MI, USA
†Fermi National Accelerator Laboratory, Batavia, IL, USA
‡University of Pennsylvania, Philadelphia, PA, USA
§Oak Ridge National Laboratory, Oak Ridge, TN, USA
terwilla@mail.gvsu.edu, perdue@fnal.gov, isele@seas.upenn.edu, {pattonrm, youngsr}@ornl.gov

***Abstract*—Deep learning offers new tools to improve our understanding of many important scientific problems. Neutrinos are the most abundant particles in existence and are hypothesized to explain the matter-antimatter asymmetry that dominates our universe. Definitive tests of this conjecture require a detailed understanding of neutrino interactions with a variety of nuclei. Many measurements of interest depend on vertex reconstruction – finding the origin of a neutrino interaction using data from the detector, which can be represented as images. Traditionally, this has been accomplished by utilizing methods that identify the tracks coming from the interaction. However, these methods are not ideal for interactions where an abundance of tracks and cascades occlude the vertex region. Manual algorithm engineering to handle these challenges is complicated and error prone. Deep learning extracts rich, semantic features directly from raw data, making it a promising solution to this problem. In this work, deep learning models are presented that classify the vertex location in regions meaningful to the domain scientists improving their ability to explore more complex interactions.**

Introduction

Neutrinos are elementary particles with little mass, no charge, and small interaction cross sections - about 100 trillion neutrinos pass through a human body per second without notice. Neutrinos are generated in stars and nuclear power plants via weak beta decay but are also produced through subatomic particle disintegration. It is hypothesized that high precision measurements of neutrino interactions will allow us to discover the physics responsible for our matter dominated universe [1].

MINERvA (Main Injector Experiment for v-A) [2] is a neutrino scattering experiment at Fermi National Accelerator Laboratory (Fermilab). The detector, seen in Figure 1, is exposed to the NuMI (Neutrinos at the Main Injector) neutrino beam [3]. Energy values collected from the detector may be mapped to pixel values in an image and subsequently analyzed. Example images are shown in Figure 2.

This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

An important feature for many measurements is vertex reconstruction–identifying the point in space where the neutrino interaction occurred. Vertex reconstruction is the focus of this work because it is a well-studied problem with clear benchmarks for performance characterization. The current state of the art method [2] used as a baseline for this work leaves room for improvement. This baseline method relies on identifying the linear tracks emanating from the vertex and calculating their intersection point.

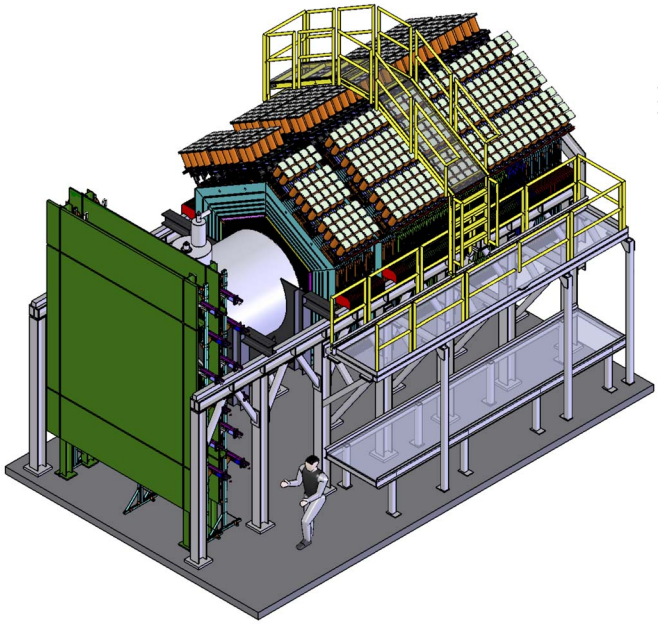

Fig. 1: The MINERvA detector. The neutrino beam is incident from the left, roughly parallel to the large veto shield in front of the detector.

For this study, millions of simulated neutrino-nucleus scattering events were created and represented as images. The core classification task is to classify the horizontal region (group of pixel columns) in each image where the interaction originated. The baseline method struggles to classify the vertex region when tracks are non-linear or the number of tracks is so great that individual tracks are difficult to identify. Deep learning

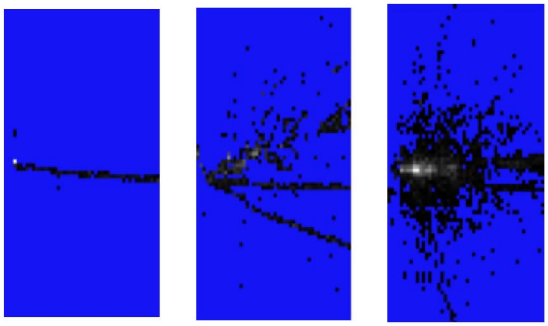

Fig. 2: Examples of neutrino interactions ranging from simple on the far left to complex on the far right. Note that only the X-view is displayed. Every 0.0 value is imputed as a blue pixel.

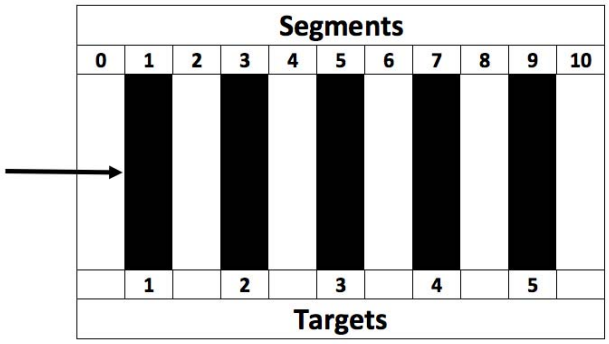

Fig. 3: Simple illustration of the MINERvA detector. There are five main targets of interest of differing metal compositions. Segments as defined by the target regions are the regions between the targets and the regions up and downstream of the targets.

(DL) can infer "deeper" underlying patterns from these images and demonstrate improved classification performance.

In addition to improving scientific results, this work showcases a novel application area of deep learning in particle physics and demonstrates a classification problem that is atypical for deep learning. Vertex reconstruction, the classification objective presented in this work, is unique in that it combines aspects of both region detection and object recognition. The core classification task is focused on region detection–finding the vertex origin horizontally in the image. However, vertex reconstruction also utilizes elements of traditional object recognition like CIFAR-10 [4], because tracks in each region of the image will appear slightly different due to the varying composition of the MINERvA detector. Thus, successfully classifying the vertex depends on learning both the location and characteristics of the tracks.

This classification objective poses challenges that motivate our unique models. For instance, traditional methods are targeted specifically towards natural images where objects exist at different scales and translation invariance is desired. In our data, all examples are at the same scale and position within the image is important as each pixel corresponds to a physical location in the detector. We address the challenge of designing a CNN that can handle images of differing angles for the same observation by utilizing three independent columns of convolutional layers. This dataset combines object classification and localization in a way that does not occur in datasets that consist of photographic images.

The remainder of this paper will provide additional details of the dataset, methods that leverage the structure of the detectors used in the experiment, and the vertex reconstruction results. The significance of these results and future work will be discussed. An appendix is provided to give additional background on the MINERvA detector for those unfamiliar with its details.

Related Work

One area of deep learning [5] that has seen great success is supervised learning with convolutional neural networks (CNNs). The defining characteristics of CNNs are shared weights and pooling between neighboring pixels. These characteristics give CNNs powerful invariance to shifts of features within images. CNNs have achieved state of the art results in the domains of object recognition [6] using the ImageNet [7] dataset and facial recognition [8] using the Labeled Faces in the Wild (LFW) [9] dataset. These are long-standing, difficult computer vision problems that have seen dramatic improvement through utilizing deep learning.

The field of high-energy physics has seen recent work utilizing deep learning neural networks [10], [11], [12], but much of this work has focused on smaller networks operating hand-engineered features. In contrast, this work utilizes CNNs that can extract features from the raw image data generated by the detector. NOvA [13], another particle physics group at Fermilab, utilized detector data to develop models for more traditional object recognition problems.

Dataset

The dataset is comprised of $1,701,439$ simulated events generated using the GENIE Neutrino Monte carlo Generator [14]. Each event is represented as three two-dimensional arrays of energy values, one for each view as shown in Figure 4. The values in the images are normalized, such that the greatest energy response from the detector (i.e. brightest pixel) is set to a value of 1.0 for each image.

This dataset was split into $1,396,000$ training images and $168,200$ testing images for each view. The X view images are 127×50 pixels, and the U and V images are 127×25 pixels. These images are truncated views of the full MINERvA detector extending slightly beyond the main target regions.

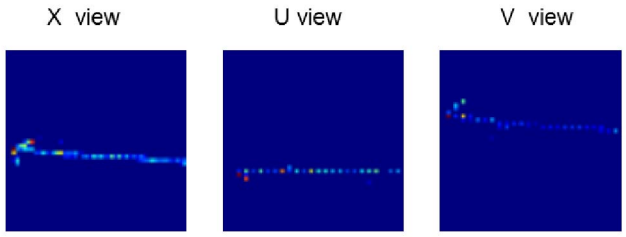

Fig. 4: An example of an event with all 3 views displayed.

The images are oriented, such that the left side of the image is the side the beam enters the detector. "Truth" data is provided for every image at multiple granularities: a segment ranging from 0 to 10 encoding the vertex region, as seen in Figure 3 and as a floating point Z value, ranging from 0 to 8000 corresponding to the precise vertex location. Therefore, the problem may be cast as both a segment classification task and a regression task.

In both cases, the core objective is to identify the vertex location horizontally in the image. In Figure 2, the vertex region in all three images is on the far left, segment 0, but each image has very different levels of complexity, due to the number and variety of tracks emanating from the vertex. The primary motivation for deep learning in vertex reconstruction lies in the difficulty of applying the previous state of the art method to accurately locate the vertex in complex events such as the image on the far right in Figure 2. Deep learning can infer more complex patterns which can lead to better overall classification accuracy.

Numerous challenges were evident within this dataset. One such challenge is due to the targets being more dense material than material in the other segments. As such, events are more likely to originate in these regions, which results in the class imbalance found in Table I. Backwards-tracks (i.e. tracks traveling in the opposite direction of the beam) are also difficult to handle. The largest residual challenge is when multiple neutrino interactions occur in a single image. This occurs when multiple interactions "pile up" inside MINERvA's timing resolution for resolving individual events. It is a problem because there is one event of interest tagged by a track in a subsequent detector, but it may be ambiguous which hits in the target region correspond to the labeled event when multiple events are present.

TABLE I: Class distribution

Target	1	2	3	4	5
Distribution	12.9%	13.8%	11.4%	8.4%	10.8%
Segment	1	3	5	7	9

Segment	0	2	4	6	8	10
Distribution	2.4%	4.7%	4.8%	13.5%	1.2%	16.0%

METHOD

The deep learning networks used in this work are convolutional neural networks and are implemented using Caffe [15]. The best-performing network used for segment classification, as shown in Figure 5, consists of three columns of convolutional and pooling layers, followed by two fully connected layers that combine the features from the 3 columns. The columnar structure is utilized since each view of the detector is from a different angle, resulting in images that cannot be combined into channels of a single image without losing spatial locality. The weights of the convolutional layers in each column are learned independently. Each column contains four convolutional layers, four max pooling layers, a fully connected layer. A rectified linear unit (ReLU) activation function is applied after each convolutional and fully connected layer. Dropout (at 0.50 rate) is utilized after each fully connected layer, and the network utilizes a softmax loss layer for classification. The number of outputs increase from 12 to 20 to 28 to 36 after each convolutional layer.

Additionally, the model in Figure 5, utilizes convolutional layers with kernels ranging in size from 8×3 in the first convolutional layer, 7×3 in the second convolutional layer, and 6×3 in the third and fourth convolutional layers. Stride of 1 pixel and padding of 0 was chosen for convolutional layers, while stride of 2x1 pixels was chosen for max pooling layers.

The best-performing network, from Figure 5, was trained using the Adagrad [16] solver with a mini-batch size of 500 for 200k iterations. Improvements in performance were found using the stochastic gradient descent (SGD) solver with a fixed learning rate of 0.0025 trained for 100k iterations and 0.00025 for another 100k iterations. At the end of 200k iterations, the test classification accuracy is reported.

This work also explores solving vertex reconstruction through regression. The final fully connected layer has a single output whose target is the true Z location with the loss function implemented using Caffe's euclidean loss layer. Two methods of encoding the Z location as a target value were explored: (1) naive encoding of the Z location as the position within the detector in millimeters and (2) removing sparse or empty regions from the data and normalizing the remaining regions between 0 and 1.

RESULTS

Using the network shown in Figure 5, there were $9,948$ out of $168,200$ test images incorrectly classified. This corresponded to an overall 94.09% test classification accuracy. Table II shows the value of deep learning in vertex reconstruction as improvement in the test classification accuracy is found in all five target regions over the previous state of the art method. Minuscule improvements to slight decreases in accuracy are seen in the even-numbered segments not containing targets, which can be inferred from the class imbalance from Table I. However, as shown by the 95% binomial proportion confidence intervals, the decrease in performance is only statistically significant for segments 8 and 10.

Fig. 5: Top performing model that consists of four convolutional layers, four max pooling layers, and three fully connected layers. Not shown are intermediary ReLU layers after each convolutional and fully connected layer. Colors represented: blue (size of the input images), red (size of convolutional kernels (hxw)), orange (size of the max pooling kernels (hxw)), purple (size of fully connected layers), green (dropout layers at 50%), cyan (final classification accuracy/loss).

TABLE II: Per-class classification accuracy and 95% binomial proportion confidence interval using best performing four convolutional layer network from Figure 5 compared to previously used method. This confidence interval accounts for sampling error in the testing set.

Segment	Target	Baseline	DL
1	1	92.2 ± 0.357%	**96.4 ± 0.248%**
3	2	91.5 ± 0.359%	**96.4 ± 0.240%**
5	3	91.2 ± 0.401%	**95.4 ± 0.297%**
7	4	89.1 ± 0.514%	**93.4 ± 0.409%**
9	5	88.8 ± 0.459%	**94.9 ± 0.320%**

Segment	Baseline	DL
0	78.9 ± 1.25%	78.1 ± 1.27%
2	88.4 ± 0.706%	88.5 ± 0.703%
4	88.6 ± 0.693%	89.2 ± 0.677%
6	95.1 ± 0.281%	95.1 ± 0.281%
8	**73.7 ± 1.92%**	61.3 ± 2.12%
10	**98.0 ± 0.167%**	96.8 ± 0.210%

TABLE III: Overall classification accuracy using three convolutional layers.

Model	Accuracy
Previous – Baseline	91.90%
Single view – X-view only	80.42%
Three views – Single channel	88.71%
Three views – Multiple columns	**93.58%**

Table III illustrates the need for the three columnar network architecture in Figure 5. Deep learning with only a single-view shows a significant decrease in performance from the baseline method. Utilizing all three views as a channel provides an improvement from the single x-view, but still fails to surpass the traditional method. Treating each view as a channel in a single column assumes a spatial locality that does not exist across the views. This lack of spatial locality is due to each view representing a different angle of the interaction, hence, a pixel location in one view does not correspond to that same pixel location in another view. The networks that utilized independently trained columns

TABLE IV: Exploration of data encoding and solvers using three convolutional layers.

Model	Accuracy
8-bit PNG + Adagrad	91.24%
8-bit PNG + SGD	**93.58%**
32-bit FP + Adagrad	92.10%
32-bit FP + SGD	87.92%

TABLE V: Comparison of network architectures using SGD

# convLayers	Kernel Sizes ($\{h\}\times w$)	Accuracy
Three	$\{6, 6, 3\} \times 3$	93.58%
Four	$\{\mathbf{8}, \mathbf{8}, \mathbf{7}, \mathbf{6}\} \times \mathbf{3}$	**94.09%**
Five	$\{8, 7, 7, 3, 3\} \times 3$	93.55%

for each view were able to demonstrate an improvement in test classification accuracy over the baseline method. It also important to note that GoogleNet [17], which does not treat the views independently, achieves comparable performance (93.9%). However, GoogleNet is much more computationally expensive to train. By exploiting our knowledge about the orientation of the views, we are able to drastically reduce the computation required to achieve these results. Using a Titan X Pascal GPU, the GoogleNet network required 74 hours to train to convergence on this dataset, while the more efficient three column network took only 10 hours to train to convergence.

Choosing optimal deep learning solver parameters is non-trivial. The difference in training between Adagrad and SGD solvers is elucidated in Table IV. Another important factor in training was using 8-bit unsigned encoding. In order to reduce the size of our dataset on disk, we explored storing the images as 8-bit PNG images instead of 32-bit floating point (FP) arrays. This reduction in precision came with over a 2% increase in test classification accuracy with the SGD solver, likely by reducing the ability of the network to overfit to the training data. Additional exploration is needed to better understand the decrease in test classification accuracy from 32-bit floating point to 8-bit PNG encoding when using the Adagrad solver.

Different network architectures were explored in an attempt to boost performance, as seen in Table V. Three convolutional layer networks using kernel sizes of 6×3, 6×3, and 3×3 in the first, second, and third convolutional layers, respectively. This network is slightly worse performing than the four convolutional layer network. We also explored a five convolutional layer network using kernel sizes of 8×3, 7×3, 7×3, 3×3, 3×3, in the first through fifth convolutional layers, respectively. This network again showed slightly worse performance than the four convolutional layer network.

Figure 6 represents the original naive regression model ($R^2 = 0.55$) which is biased by upstream events (segment 0) and the water target (segment 6). To address these biases, normalization techniques were applied, far upstream events were removed, and empty regions were collapsed. This resulted in well-sampled target space with an ($R^2 = 0.95$) found Figure 7. Mapping this result back to the original space in Figure 8 results in an ($R^2 = 0.96$), but shows areas for improvement with right-skewed segments 0/1 and significant misses in segment 6 on either side of the water target.

Discussion

There are three main categories of misclassification: boundaries (Figure 9 - left), backwards-tracks (Figure 9 - right), and multiple-event (Figure 10). Boundary misclassifications, Figure 9 - left, are difficult to correct and may be limited by the resolution of the detector. A non-expert could easily make the same mistake the network is making in Figure 9 - right, as the trajectory is traced all the way back to the predicted red region. Additionally, Figure 10 shows how multiple-event pile-up affects classification. Again, a non-expert would not be able to determine the "correct" trajectory in each of the

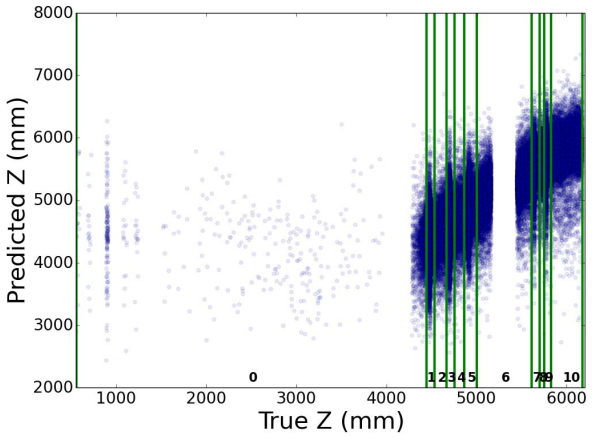

Fig. 6: Naive regression model ($R^2 = 0.55$).

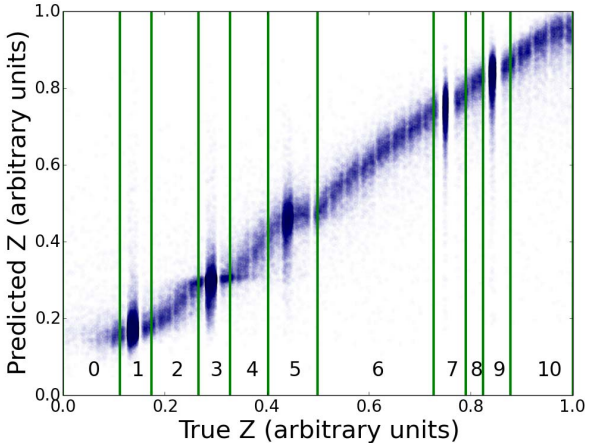

Fig. 7: Normalized regression model ($R^2 = 0.95$).

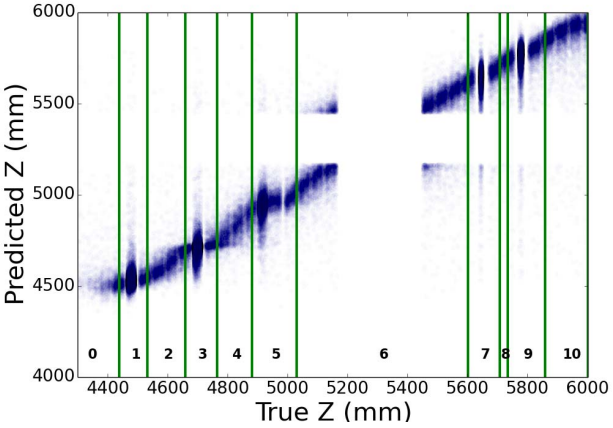

Fig. 8: Normalized regression model mapped back to original space ($R^2 = 0.96$).

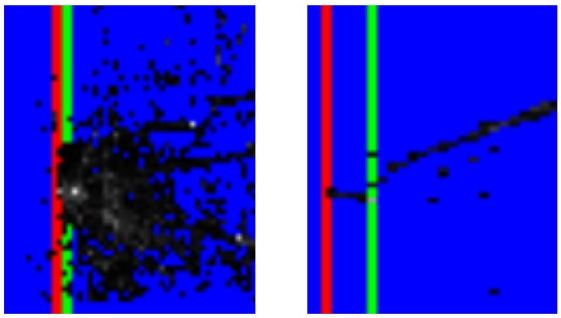

Fig. 9: Misclassified boundary image (left) and misclassified backwards-track image (right). Red indicates the incorrectly predicted segment and green indicates the true segment.

Figure 10 examples due to two or more interactions appearing in each image.

One idea to address backwards-track and multiple event misclassification is to leverage additional data as input to the network, such as timing information (i.e. when was each pixel activated) or the vertex location predicted by non-DL methods. Additionally, the work presented in this paper was based purely on simulation data, which will likely be the only source of large amounts of labeled data. Even though these simulations are extremely rigorous and are good representations of real data, applying these trained networks to real data will likely involve addressing any bias that exists in the simulation.

The regression results, while improved by normalizing the data and ignoring events from low probability regions, are still insufficient. Mapping these regression results back to the segment classification problem results in a segment classification accuracy of less than 30%. Alternate loss functions are likely needed to improve regression results. Attempting to classify the segments using this single output may also make it difficult to learn the features that are unique to each segment (i.e. target segment vs. non-target segment).

Conclusion

This work explored a novel application of deep learning to particle physics. Additionally, a unique classification problem was uncovered that combined aspects of both region detection and object recognition. The deep learning models presented in this work demonstrated state of the art performance accuracy for this dataset. This improvement in accuracy can offer physicists the ability reduce uncertainties in their calculations and to utilize data from more complex events that previously could not be reconstructed with confidence. Future work will involve investigating methods for applying these networks trained on simulation data to data collected from the physical detector while managing any bias that exists in the simulation data.

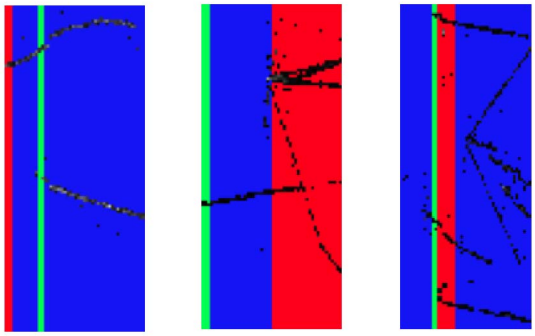

Fig. 10: Examples of misclassified images containing multiple-events. Red indicates the incorrectly predicted segment and green indicates the true segment.

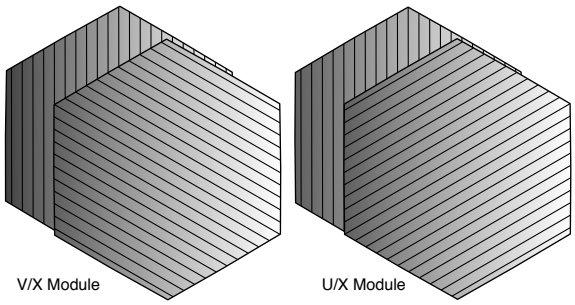

Fig. 11: MINERvA detector modules contain pairs of planes with scintillator strips at different angles – vertical, or so-called "X" planes and planes at $\pm 60^0$, or "U" and "V" planes. Each module in the tracking regions of the detector contains one X and one of either a U or V planes. Modules then alternate in type through the detector, UX, VX, UX, etc. The alignment of planes is skewed here to make the X plane visible beneath the U and V planes.

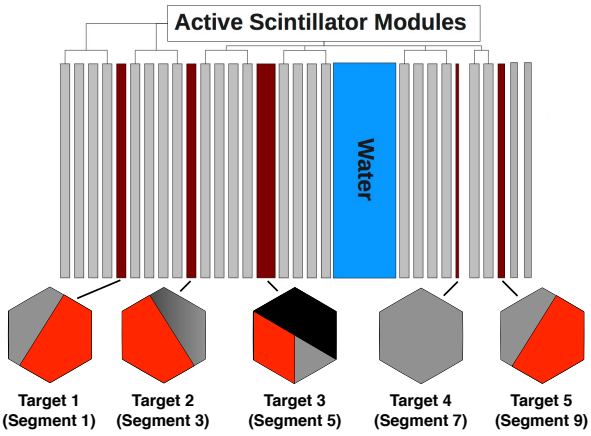

Fig. 12: The passive targets. Note that the water target is empty in this work. Each active scintillator module contains two planes, as shown in Figure 11.

APPENDIX A - MINERvA EXPERIMENT

Neutrinos are produced by striking a graphite target with very high energy protons, focusing the spray of produced subatomic particles with magnetic horns, and allowing them to decay in a long beam pipe into neutrinos (other particles are filtered with hundreds of meters of rock).

The primary active elements in MINERvA are long scintillator strips that provide a linear response to charged particles. Strips are bundled into planes and stacked roughly perpendicular to the neutrino beam, as shown in Figure 1. Figure 11 shows how the detector modules are arranged, with the strips in each plane oriented at one of three different angles (or views) to provide a stereoscopic event image. The number of X view planes is equal to the sum of the number of U and V view planes, so the image "density" is twice as high in X as it is in U or V. Additional details about the MINERvA detector can be found elsewhere [2], [18]. The so-called "nuclear target region" near the front of the detector contains solid, passive planes of iron, lead, and carbon, as seen in Figure 12 The suite of targets allow for ratio measurements with reduced systematic uncertainties [19]. These targets are of particular interest to this work.

The geometry of the nuclear target region is complex and the passive targets themselves further interfere with event reconstruction using traditional clustering and tracking algorithms. However, accurate placement of the neutrino vertex is crucial for a physics program that measures nuclear A-dependent effects, thereby motivating this work.

ACKNOWLEDGEMENTS

This work was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internship program.

Research sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U. S. Department of Energy.

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

We would like to thank the MINERvA collaboration for the use of their simulated data and for many useful and stimulating conversations. MINERvA is supported by the Fermi National Accelerator Laboratory under US Department of Energy contract No. DE-AC02-07CH11359 which included the MINERvA construction project. MINERvA construction support was also granted by the United States National Science Foundation under Award PHY-0619727 and by the University of Rochester. Support for participating MINERvA physicists was provided by NSF and DOE (USA), by CAPES and CNPq (Brazil), by CoNaCyT (Mexico), by CONICYT (Chile), by CONCYTEC, DGI-PUCP and IDI/IGIUNI (Peru), and by Latin American Center for Physics (CLAF).

REFERENCES

[1] R. Acciarri *et al.*, "Long-Baseline Neutrino Facility (LBNF) and Deep Underground Neutrino Experiment (DUNE)," 2016.

[2] L. Aliaga *et al.*, "Design, calibration, and performance of the {MINERvA} detector," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 743, pp. 130 – 159, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900214000035

[3] P. Adamson *et al.*, "The NuMI Neutrino Beam," *Nucl. Instrum. Meth.*, vol. A806, pp. 279–306, 2016.

[4] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 05 2015. [Online]. Available: http://dx.doi.org/10.1038/nature14539

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[8] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[9] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008.

[10] P. Baldi, P. Sadowski, and D. Whiteson, "Searching for exotic particles in high-energy physics with deep learning," *Nat Commun*, vol. 5, 07 2014. [Online]. Available: http://dx.doi.org/10.1038/ncomms5308

[11] P. J. Sadowski, D. Whiteson, and P. Baldi, "Searching for higgs boson decay modes with deep learning," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2393–2401. [Online]. Available: http://papers.nips.cc/paper/5351-searching-for-higgs-boson-decay-modes-with-deep-learning.pdf

[12] P. Sadowski, J. Collado, D. Whiteson, and P. Baldi, "Deep learning, dark knowledge, and dark matter," *Journal of Machine Learning Research*, 2015.

[13] A. Aurisano, A. Radovic, D. Rocco, A. Himmel, M. D. Messier, E. Niner, G. Pawloski, F. Psihas, A. Sousa, and P. Vahle, "A Convolutional Neural Network Neutrino Event Classifier," *JINST*, 2016.

[14] C. Andreopoulos *et al.*, "The GENIE Neutrino Monte Carlo Generator," *Nucl. Instrum. Meth.*, vol. A614, pp. 87–104, 2010.

[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[16] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[18] G. N. Perdue *et al.*, "The MINERνA Data Acquisition System and Infrastructure," *Nucl. Instrum. Meth.*, vol. A694, pp. 179–192, 2012.

[19] J. Mousseau *et al.*, "Measurement of Partonic Nuclear Effects in Deep-Inelastic Neutrino Scattering using MINERvA," *Phys. Rev.*, vol. D93, no. 7, p. 071101, 2016.