

PAPER • OPEN ACCESS

Efficient antihydrogen detection in antimatter physics by deep learning

To cite this article: P Sadowski *et al* 2017 *J. Phys. Commun.* **1** 025001

View the [article online](#) for updates and enhancements.

Related content

- [Physics with antihydrogen](#)
W A Bertsche, E Butler, M Charlton *et al.*
- [Cold Antihydrogen Atoms](#)
J Walz
- [Simulations of antihydrogen formation in a nested Penning trap](#)
S Jonsell, D P van der Werf and M Charlton

Recent citations

- [Machine learning at the energy and intensity frontiers of particle physics](#)
Alexander Radovic *et al*



PAPER

OPEN ACCESS

RECEIVED
6 June 2017REVISED
23 July 2017ACCEPTED FOR PUBLICATION
3 August 2017PUBLISHED
6 September 2017

Original content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](#).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



Efficient antihydrogen detection in antimatter physics by deep learning

P Sadowski¹, B Radics^{2,3}, Ananya¹, Y Yamazaki² and P Baldi¹¹ Department of Computer Science, University of California, Irvine, CA 92617, United States of America² Atomic Physics Laboratory, RIKEN, Saitama 351-0198, Japan³ ETH Zurich, Institute for Particle Physics, CH-8093 Zürich, SwitzerlandE-mail: bradics@phys.ethz.ch

Keywords: antimatter, antihydrogen, deep learning

Abstract

Antihydrogen is at the forefront of antimatter research at the CERN Antiproton Decelerator. Experiments aiming to test the fundamental CPT symmetry and antigravity effects require the efficient detection of antihydrogen annihilation events, which is performed using highly granular tracking detectors installed around an antimatter trap. Improving the efficiency of the antihydrogen annihilation detection plays a central role in the final sensitivity of the experiments. We propose deep learning as a novel technique to analyze antihydrogen annihilation data, and compare its performance with a traditional track and vertex reconstruction method. We report that the deep learning approach yields significant improvement, tripling event coverage while simultaneously improving performance in terms of AUC by 5%.

1. Introduction

In recent years numerous experiments have successfully trapped, or formed a beam of, antihydrogen atoms. These experiments aim to test CPT symmetry by comparing the properties of the antihydrogen atom to those of the hydrogen atom [1–4], or test the effects of gravity on antimatter [5–7]. Antihydrogen is produced either by injecting antiproton and positron plasmas into cryogenic electro-magnetic traps where three-body recombination takes place ($\bar{p} + 2e^+ \rightarrow \bar{H} + e^+$), or by the charge exchange processes between positronium, antiproton, and antihydrogen ($\text{Ps}^{(*)} + \bar{p} \rightarrow \bar{H} + e^-$ and $\bar{H} + \text{Ps} \rightarrow \bar{H}^+ + e^-$). In most cases, antihydrogen production is detected by its annihilation signature whereby several charged pions are emitted from the annihilation vertex [8]. These are distinguished from cosmic or instrumental background events using the hit multiplicity and the inferred positions of the tracks and annihilation vertex.

Previous experiments have detected trapped antimatter annihilation events using algorithms specifically developed for tracking and vertex-finding [9–15], predominantly adopted from high-energy physics (HEP). However, antimatter trap experiments are not designed for high-resolution and high-efficiency tracking performance; the necessary layers of vacuum chamber walls, multi-ring electrodes, and thermal isolation limits the resolution of the detector and the effectiveness of these algorithms. Therefore, the vertex-reconstruction approach may not be optimal for detecting antihydrogen event signals in this setting.

We present a machine learning approach for analyzing data from antimatter trap experiments that does not require explicit vertex reconstruction. Specifically, we propose *deep learning* with artificial neural networks as a strategy to identify antihydrogen annihilation signatures from low-level detector data alone. This approach has the advantage of being able to exploit subtle and previously-unrecognized characteristics in the data, as demonstrated in Higgs-search and Higgs-decay studies from HEP [16–18]. We train our system using Monte Carlo simulated data to distinguish between antihydrogen-like signal and antiproton background annihilation events, and compare the performance to a traditional track and vertex reconstruction method.

The specific use case presented in this paper is the ASACUSA antihydrogen experiment [4], whose aim is to directly compare the ground-state hyperfine transition frequency of hydrogen with that of antihydrogen—a test

of the CPT-invariance principle. Measurements of the antihydrogen hyperfine transition frequency are performed by measuring the rate of antihydrogen atom production in a Penning–Malmberg trap while controlling the antiproton injection conditions, the overlap time while mixing antimatter plasmas, and other key parameters which allow us to constrain the three-body recombination yield and the level population evolution time of the formed antiatoms [19]. Thus, the sensitivity of the experiment depends directly on the efficient detection of antihydrogen annihilations.

Other antihydrogen experiments face similar challenges. Although the background processes or event topology might be different, classifying annihilations in a cryogenic environment with significant material budget is a common problem. Therefore, advances in single annihilation event classification in ASACUSA could potentially lead to performance improvements in other experiments.

2. Monte Carlo simulation and annihilation event reconstruction

The ASACUSA experiment traps charged particles (antiprotons and positrons) on the central axis using electromagnetic fields. When a neutral antihydrogen atom is produced (predominately via three-body recombination of antiprotons and positrons), it may escape these trapping fields and annihilate on the inner wall of the trap, emitting charged pions that can be detected. However, continuous annihilation of trapped antiprotons on residual gas atoms produces background annihilation events that emit the same pions. The latter tend to be emitted from the central axis, rather than the inner wall, so we can distinguish between antihydrogen and antiproton events if the position of the annihilation can be inferred from the detector data.

The pions are detected by the Asacusa Micromegas Tracker (AMT) [15], which consists of a full-cylinder layer of plastic trigger scintillator bars sandwiched in between two, half-cylinder, curved micro-strip pattern gaseous detector layers of Micromegas technology [20, 21]. In the typical vertex-finding approach, each trigger event is processed with a track and vertex reconstruction algorithm, proceeding as follows. First, the single detector channels are searched for entries above threshold, and neighboring channels are iteratively clustered to form hits. The detected hits are used to form and fit tracks, using a Kalman-filtering algorithm [22]. After fitting all hit pair combinations in an event, the track candidates are filtered by their compatibility with the known antiproton cloud position. The filtered track candidates are then paired, and their three-dimensional point of closest approach is assigned as the vertex position. Finally, event classification is performed based on the radial distance of the reconstructed vertex closest to the central axis.

In order to test the ability of this algorithm to discriminate between signal and background annihilations, Monte Carlo simulations were performed to produce event data that reflect the realistic conditions in the ASACUSA experiment. The Geant4 toolkit [23] was used to simulate the antihydrogen and antiproton annihilation processes, the trap environment, and the AMT. The full material budget of the cryogenic vacuum trap was used in order accurately model the effects of multiple Coulomb scattering on the emitted pions [24] before they reached the surrounding detector. Micromegas single hit resolution was simulated with Garfield++, Magboltz, and Heed [25, 26] for the response of charged pions of $p \simeq 100$ MeV/c momentum, and was found to be much better ($\sigma_{\text{hit}} \simeq 250$ μm) than the amount of smearing of the pion track positions ($\sigma_{\text{Coul.}} \simeq 2$ mm) due to Coulomb scattering when the pions reach the AMT detector. Using the detected hits, the reconstructed vertex position resolution was estimated to be $\sigma_{\text{vx}} \simeq 1$ cm.

The result of vertex-reconstruction on the simulated data is illustrated in figure 1. In the ASACUSA experiment, tracking detector layers are only present in the upper half of the trap, which is reflected in the distribution of reconstructed vertices. The shape of the distribution also reflects the smearing of the position by 1 cm due to the Coulomb scattering of the charged pions in the trap material, and the heavy overlap in the class-conditional distributions of the radial distance $R = \sqrt{X^2 + Y^2}$. For the experiments performed in section 4, probabilistic classification of events is performed by a simple neural network model with a single input, R , one hidden layer of 100 hyperbolic tangent units, and a logistic output unit that predicts the probability that an event was an antihydrogen atom versus an antiproton.

3. Deep learning approach

We propose a deep learning approach in which an artificial neural network is trained to classify annihilations directly from the raw detector data. This automated, end-to-end learning strategy can potentially identify discriminative information in the raw data that is typically discarded by the vertex-reconstruction algorithm. For example, this approach will provide predictions for events on which vertex-reconstruction fails, such as when only a single track can be reconstructed.

Two design choices were made regarding the neural network architecture that address the geometry of the detector. First, data from micro-strips situated along the azimuth (ϕ) and axial (Z) dimensions are processed

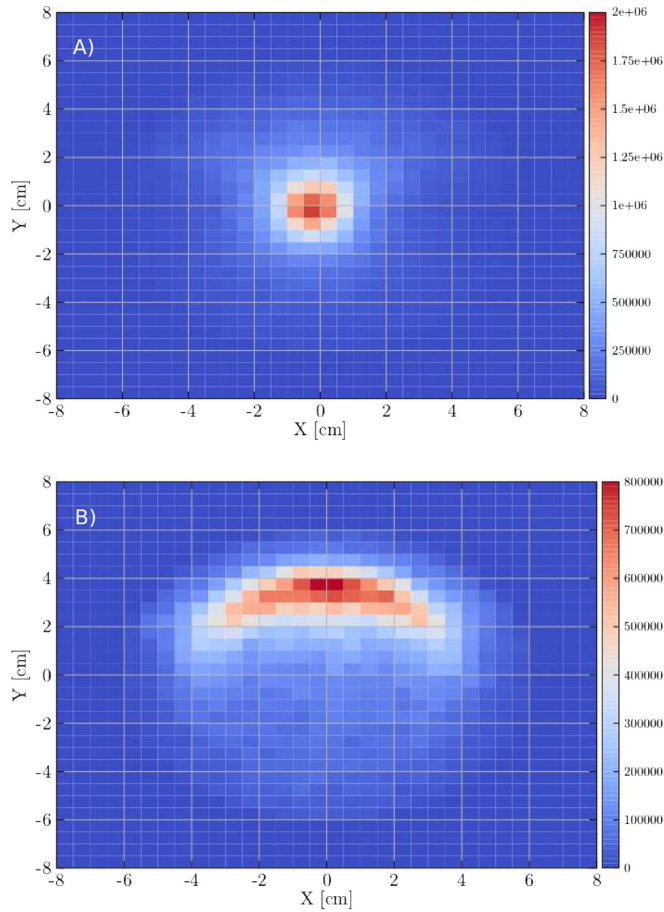


Figure 1. Reconstructed annihilation vertex positions using the traditional track and vertex finding method for (A) antiproton background events confined to the center of the trap, and (B) antihydrogen signal events distributed on the multi-ring electrode wall at $R = 4$ cm. The color scale indicates the number of vertices found at a given position.

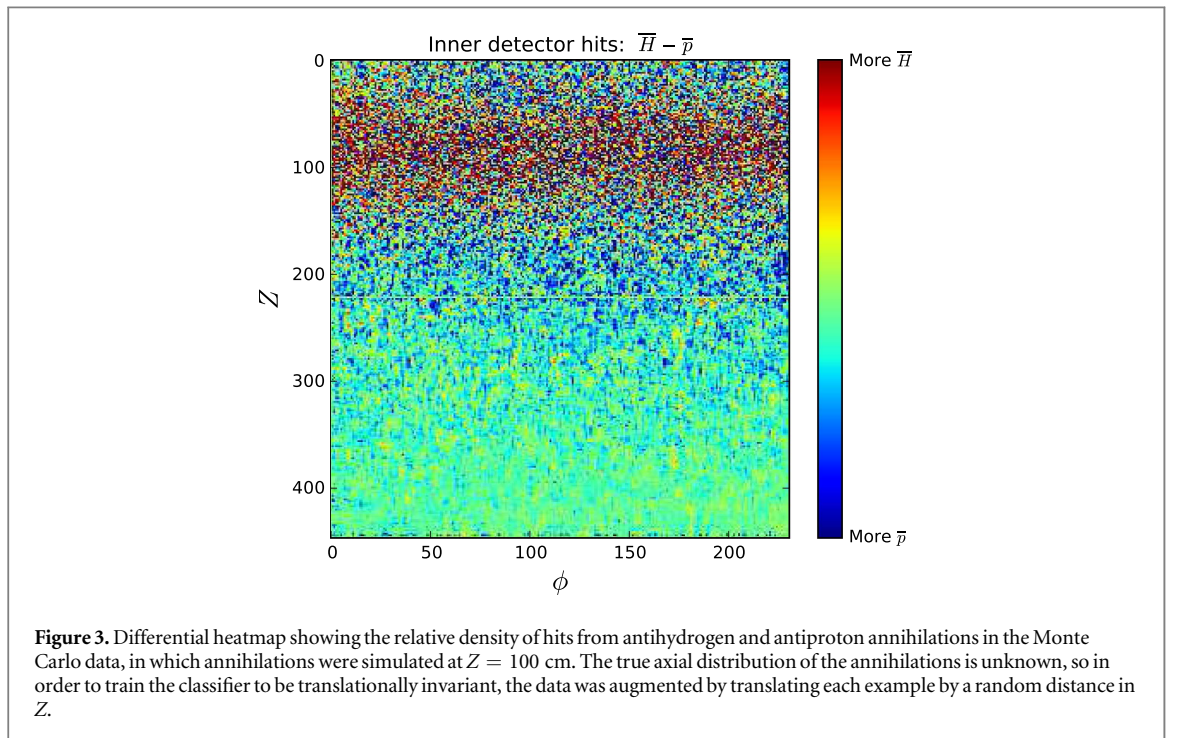
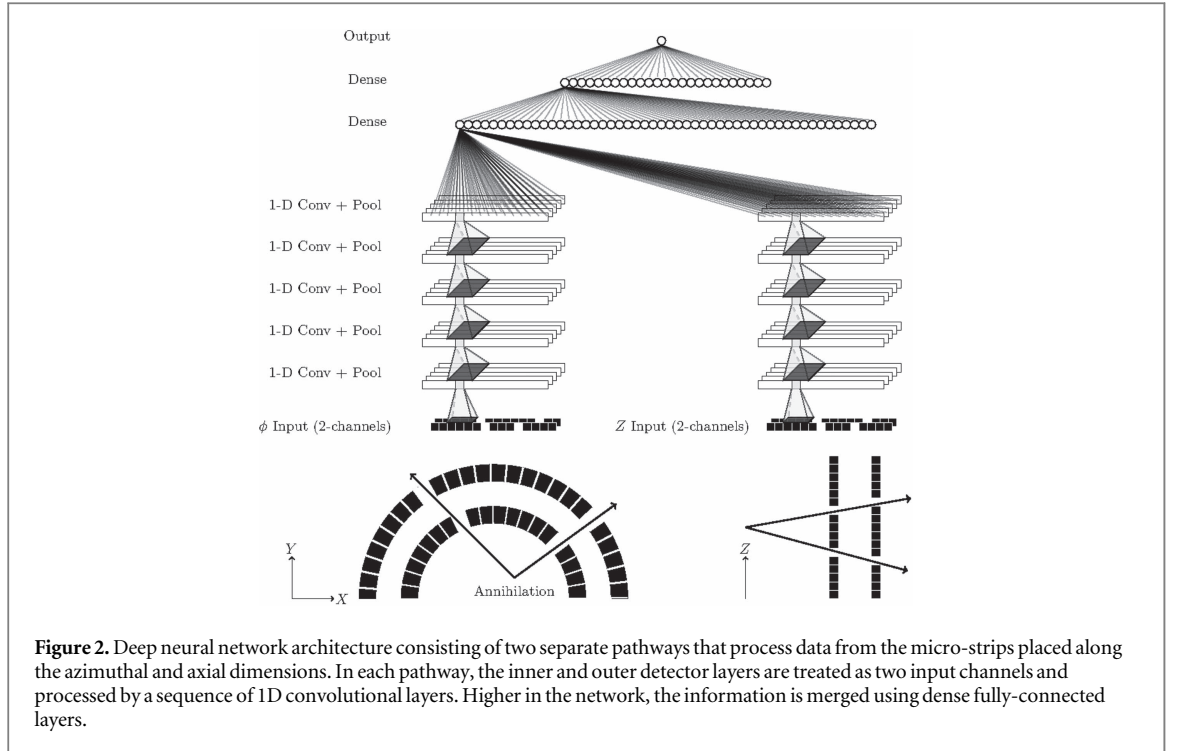
along separate pathways that are then merged by concatenating the hidden representations higher in the architecture. Second, each pathway consists of alternating 1D convolution and max-pooling layers, which repeat the same feature detectors (via weight-sharing) and pool activity from adjacent neurons. This design accounts for the translational invariances of the detector and reduces the total number of network parameters.

The raw detector data consists of 1430 binary values from the two detector layers and two micro-strip orientations: 246 inner and 290 outer azimuthal strips, and 447 inner and 447 outer axial strips. The inner and outer detector layers are treated as two associated ‘channels’, where the outer azimuthal layer is downsampled with linear interpolation to match the dimensionality of the inner layer. A diagram of the network architecture is shown in figure 2.

4. Results

One million annihilation events of each type were simulated and randomly divided into training (60%), validation (20%), and test subsets (20%). Because the detector only covers half of the trap, many pions escaped undetected, and the vertex finding (VF) algorithm failed on 75% of these events: 20% did not result in any detector hits; 7% had a hit in only one detector; and 48% had hits in both detectors but a vertex could not be reconstructed. Vertex reconstruction failed because either (1) there were not enough distinct hits to infer the presence of at least two tracks, or (2) the point of closest approach of the reconstructed tracks was greater than the threshold value of 1 cm. Thus, a direct performance comparison was only possible on the 25% of the test set events for which the VF algorithm succeeded, even though the deep learning approach provides a prediction for every event.

Due to the complexity of the non-neutral plasma physics and atomic scattering processes, there is no physics model to our knowledge that would be able to predict the true axial distribution of the antihydrogen atoms in realistic experimental conditions. In the future, characterization of this distribution may lead to improved performance, but for the present analysis we train the classifier in such a way that the output is invariant to



translations in the axial dimension. This was achieved by augmenting the training data, in which simulated antihydrogen annihilations occurred at $Z = 100$ cm (see figure 3), by randomly translating each event during training by t ‘pixels’ in the Z dimension, where t is sampled from the uniform distribution $t \sim U(-100, 347)$. The same augmentation was used on the validation and test sets to evaluate performance, but was the same on non-augmented data, indicating that the network learned Z -invariant features.

The neural network was trained with a variety of hyperparameter combinations in order to optimize generalization performance on the validation set. These hyperparameters included the specifics of the network architecture shape. The best architecture has five 1D convolutional layers with kernel sizes 7-3-3-3-3 (the size of the receptive fields for neurons in each layer), channel sizes 8-16-32-64-128 (the number of distinct feature detectors in each layer), and rectified linear activation [27]. In order to account for translational invariance, each

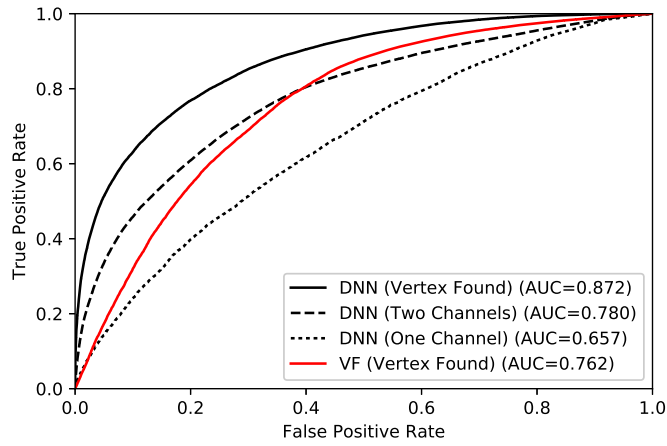


Figure 4. ROC curves for the vertex finding algorithm (VF) and a deep convolutional neural network (DNN) on the subset of events for which an annihilation vertex can be reconstructed ('Vertex Found', 25% of all events). Also shown are the curves for DNN predictions on the subset of events in which both the inner and outer detector channels are hit, but no vertex could be reconstructed (48% of all events), and on events in which only one channel is hit (7% of all events).

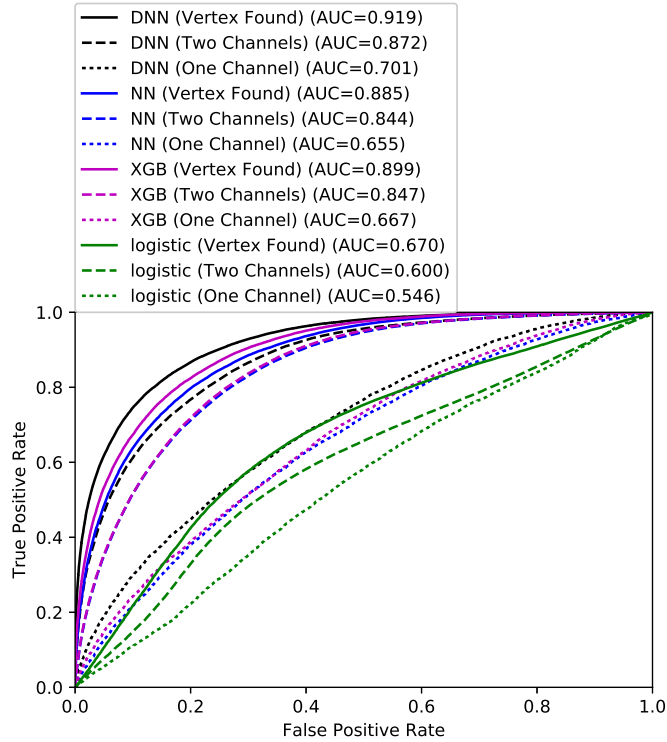


Figure 5. ROC curves for four classification models trained on the non-augmented detector-level data set, with annihilations simulated at $Z = 100$: the deep convolutional architecture (DNN) from figure 4, a shallow neural network with a single hidden layer of 2000 units (NN), XGBoost (XGB), and a logistic regression model (equivalent to a neural network with no hidden layer). Predictions are made on the three event subsets: events for which a vertex can be reconstructed (Vertex Found), events in which both detector channels are hit but no vertex can be reconstructed (Two Channels), and events in which only one channel is hit (One Channel). The discriminative capability of the logistic regression model is predominately a result of the Z -dependence shown in figure 3, where the antihydrogen distribution has a tighter spread around $Z = 100$.

convolution layer is followed by a max-pooling layer with pool size 2 and stride length 2 [28]. The flattened representations from the two pathways are then concatenated and followed by two fully-connected layers of 50 and 25 rectified linear units, then a single logistic output unit with a relative entropy loss. During training, 50% dropout was used in the top two fully-connected layers to reduce overfitting [29, 30]. The model weights were initialized from a scaled normal distribution as suggested by He *et al* [31], then trained using the Adam optimizer [32] ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e - 08$) with mini-batch updates of size 100 and a learning rate that was initialized to 0.0001 and decayed by 1% at the end of each epoch. Training was stopped when the validation

objective did not improve within a window of three epochs. The models were implemented in KERAS [33] and THEANO [34], and trained on a cluster of Nvidia Titan Black processors.

Performance comparisons were made by plotting the receiver operating characteristic curve and summarizing the discrimination by calculating the area under the curve (AUC). Figure 4 shows that the deep neural network classifier outperforms the VF approach by a large margin on the test events for which a vertex could be reconstructed (0.87 versus 0.76 AUC). Remarkably, the deep neural network achieves 0.78 AUC on a disjoint set of events for which a vertex could *not* be reconstructed, and 0.82 AUC on the union set containing all events for which both detector layers were hit (not shown). This effectively triples the event coverage—to 73% of all events—while *simultaneously* improving the AUC by more than 0.05. These results clearly demonstrate that useful information contained in the raw data is being discarded by the VF algorithm, and that the deep learning approach is able to use this information to improve discrimination.

Additional experiments evaluated the advantage of deep learning compared to ‘shallow’ machine learning with boosted decision trees and neural networks with only a single hidden layer. These experiments were performed on the non-augmented data, where the convolution and pooling layers provide *less* of an advantage for the DNN. Figure 5 shows that XGBoost [35] and shallow neural networks both perform better than the VF algorithm using the raw detector data (0.89 and 0.90 AUC on the same test events), but not as well as the DNN (0.92 AUC). The shallow network had 2000 hidden rectified linear units and was trained with 70% dropout in the hidden layer and the Adam optimizer ($\alpha = 0.0001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e - 8$); the hidden layer size, dropout probability, and learning rate were optimized based on validation performance. The XGBoost classifier had 100 trees with a maximum depth of 100 and minimum child weight of 10, and was trained with $\eta = 0.1$, L2 regularization factor of 0.1, and feature sampling rate 0.5.

5. Conclusion

In summary, we report the first application of deep learning for the identification of antihydrogen annihilation events using realistic Monte Carlo simulations of both antihydrogen-like signals and antiproton background events. Moreover, we address the scenario in which classification must be translationally-invariant in the axial dimension to address systematic uncertainty in the experiment. The results demonstrate significant performance improvements compared to track and vertex reconstruction approaches in a scenario with limited detector resolution. Vertex reconstruction necessarily discards statistical information when fitting tracks, and the approach fails completely on all but 25% of events in our experiments. With the deep learning approach, an end-to-end neural network model extracts additional statistical information from the raw data, improving AUC from 0.76 to 0.87 on the same event subset. It also has far better coverage, achieving 0.82 AUC on the 73% of events in which both detector layers were hit.

While we expect more modest performance gains in detectors with better resolution, the deep learning approach proposed here can easily be deployed to other ongoing antimatter experiments with different instruments. Furthermore, the ability to detect complex signals in high-dimensional data without relying on explicit track reconstruction potentially offers a new direction in the design of future detectors.

Finally, the data used in this paper, comprising two million events, are publicly available from the machine learning in physics web portal: <http://mlphysics.ics.uci.edu>.

Acknowledgments

The work of BR was supported by the Grant-in-Aid for Specially Promoted Research (no. 24000008) of the Japanese Ministry of Education, Culture, Sports, Science and Technology (Monbukakagu-sho), Special Research Projects for Basic Science of RIKEN, Pioneering Project of RIKEN. The work of PB and PS was in part supported by NSF grants IIS-1321053 and IIS-1550705, as well as an NVIDIA Corporation hardware award, to PB.

References

- [1] Andresen G *et al* 2010 *Nature* **468** 673
- [2] Andresen G B *et al* (ALPHA Collaboration) 2011 *Nat. Phys.* **7** 558
- [3] Gabrielse G *et al* (ATRAP Collaboration) 2012 *Phys. Rev. Lett.* **108** 113002
- [4] Kuroda N *et al* (ASACUSA Collaboration) 2014 *Nat. Commun.* **5** 3089
- [5] Aghion S *et al* (AEGIS Collaboration) 2014 *Nat. Commun.* **5** 4538
- [6] Pérez P *et al* (GBar Collaboration) 2015 *Hyperfine Interact.* **233** 21
- [7] Amole C *et al* (Alpha Collaboration) 2013 *Nat. Commun.* **4** 1785
- [8] Hori M, Yamashita K, Hayano R and Yamazaki T 2003 *Nucl. Instrum. Methods A* **496** 102

- [9] Amoretti M et al 2004 *Nucl. Instrum. Methods A* **518** 679
- [10] Amole C et al 2014 *Nucl. Instrum. Methods A* **735** 319
- [11] Zhang Z, Oelert W, Grzonka D and Sefzick T 2009 *Chin. Sci. Bull.* **54** 189
- [12] Storey J et al 2013 *Nucl. Instrum. Methods A* **732** 437
- [13] Corradini M et al 2013 *Nucl. Instrum. Methods A* **711** 12
- [14] Corradini M et al 2015 *Hyperfine Interact.* **233** 53
- [15] Radics B et al 2015 *Rev. Sci. Instrum.* **86** 083304
- [16] Baldi P, Sadowski P and Whiteson D 2014 *Nat. Commun.* **5** 4308
- [17] Baldi P, Sadowski P and Whiteson D 2015 *Phys. Rev. Lett.* **114** 111801
- [18] Sadowski P, Collado J, Whiteson D and Baldi P 2015 *Journal of Machine Learning Research, Workshop and Conf. Proc.* vol 42, p 81
- [19] Radics B, Murtagh D J, Yamazaki Y and Robicheaux F 2014 *Phys. Rev. A* **90** 032704
- [20] Giomataris Y, Rebourgeard P, Robert J P and Charpak G 1996 *Nucl. Instrum. Methods A* **376** 29–35
- [21] Cazaux S, Lerch T and Aune S 2014 Detecteur courbe de particules gazeux *Patent App.* EP20,130,188,550
- [22] Strandlie A and Frühwirth R 2010 *Rev. Mod. Phys.* **82** 1419
- [23] Agostinelli S et al 2003 *Nucl. Instrum. Methods A* **506** 250
- [24] Shibata M, Mohri A, Kanai Y, Enomoto Y and Yamazaki Y 2008 *Rev. Sci. Instrum.* **79** 015112
- [25] Smirnov I 2005 *Nucl. Instrum. Methods A* **554** 474
- [26] Schindler H and Veenhof R Garfield++, <http://cern.ch/garfieldpp>
- [27] Nair V and Hinton G E 2010 *Proc. 27th Int. Conf. on Machine Learning (ICML-10)* ed J Furnkranz and T Joachims (Omnipress) pp 807–14
- [28] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge, MA: MIT Press)
- [29] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 *J. Mach. Learn. Res.* **15** 1929
- [30] Baldi P and Sadowski P 2014 *Artif. Intell.* **210C** 78
- [31] He K, Zhang X, Ren S and Sun J 2015 *The IEEE Int. Conf. on Computer Vision (ICCV)* (<https://doi.org/10.1109/iccv.2015.123>)
- [32] Kingma D P and Ba J 2014 *Proc. 3rd Int. Conf. on Learning Representations (ICLR)*
- [33] Chollet F 2015 Keras (GitHub)
- [34] Theano Development Team 2016 arXiv:[abs/1605.02688](https://arxiv.org/abs/1605.02688)
- [35] Chen T and Guestrin C 2016 *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)* (San Francisco, CA) **785–94**