

Deep Learning in Physics exemplified by the Reconstruction of Muon-Neutrino Events in IceCube

The IceCube Collaboration[†]

[†] http://icecube.wisc.edu/collaboration/authors/icrc17_icecube

E-mail: mirco.huenefeld@tu-dortmund.de

Recent advances, especially in image recognition, have shown the capabilities of deep learning. Deep neural networks can be extremely powerful and their usage is computationally inexpensive once the networks are trained. While the main bottleneck for deep neural networks in the traditional domain of image classification is the lack of sufficient labeled data, this usually does not apply to physics where millions of Monte Carlo simulations exist.

The IceCube Neutrino Observatory is a Cherenkov detector deep in the Antarctic ice where the reconstruction of muon-neutrino events is one of the key challenges. Due to limited computational resources and the high data rate, only simplified reconstructions limited to a small subset of data can be run on-site at the South Pole. However, in order to perform online analysis and to issue real-time alerts, a fast and powerful reconstruction is necessary.

This paper demonstrates how deep learning techniques such as those used in image recognition can be applied to IceCube pulses in order to reconstruct muon-neutrino events. These methods can be generalized to other physics experiments.

Corresponding authors: Mirco Hünnefeld^{*1}

¹ *Dept. of Physics, TU Dortmund University, D-44221 Dortmund, Germany*

35th International Cosmic Ray Conference

10-20 July, 2017

Bexco, Busan, Korea

*Speaker.

1. Introduction

The IceCube Neutrino Observatory is a neutrino detector located at the South Pole instrumenting a cubic kilometer of glacial ice. The detector consists of 5160 digital optical modules (DOMs) installed on 86 vertical strings at depths between 1450m and 2450m. These strings are deployed on an approximately triangular grid with a string-to-string spacing of about 125 m. [1]

In order to perform online analyses at the South Pole and to enable rapid follow-up observations, a realtime analysis framework for the IceCube neutrino observatory was implemented. The online processing and filtering system [1] calibrates the recorded DOM waveforms, extracts pulses from the waveforms and performs some basic event reconstructions. With the help of these reconstructions about 1% of all triggered events are selected by dedicated online filters for further on-site processing (Online L2 Selection). One of the multiple online filters in place is the Level2 Online Muon Filter which aims to select well reconstructed track-like events of muons produced in charged current interactions of muon neutrinos. Given the hardware limitations at the South Pole, events from the Online L2 Selection must be processed within about 30s to prevent pileup. The limited computational resources and the high data rate pose a key challenge to obtaining an accurate event reconstruction. While more sophisticated reconstruction methods exist [2], their usage on-site becomes intractable when taking their runtime into account which can require minutes to hours for a single event. [3]

One possible solution to obtain powerful event reconstructions, while keeping the computational complexity low, is the use of deep learning techniques. Recent advances in image recognition [4] have shown the capabilities of deep convolutional networks. These networks belong to the class of representation-learning methods [5]. They are capable of processing raw data and of creating an abstract level of representation. Once the networks are trained, their usage is computationally inexpensive, while the reconstruction accuracy can be improved compared to conventional methods. The network performs a set amount of mathematical operations on the input data resulting in a very stable runtime that is essentially independent of the input. In comparison, runtimes of currently used on-site reconstructions are highly dependent on the input. These characteristics make deep learning methods an excellent candidate for powerful and fast on-site reconstructions.

2. Hexagonally shaped Data and Network Architecture

An IceCube muon-neutrino simulation dataset is used with a simulated neutrino spectrum of E^{-1} and an energy range from 100GeV to 10PeV. For the results shown in this paper, the neutrino spectrum is reweighed to an unbroken power-law flux with $E^{-2.13}$ according to [6]. Only track-like events are used, where the muon resulting from the charged-current neutrino interaction comes within 60m of the detector. Approximately 34,000 events are used for each of the test and validation set, while the training dataset consists of approximately 3 million events. The framework Tensorflow[7] is used in combination with the programming language Python¹. Training of the convolutional neural network is performed on an NVIDIA Tesla P40² and takes about a week.

¹<https://www.python.org/>

²<https://images.nvidia.com/content/pdf/tesla/184427-Tesla-P40-Datasheet-NV-Final-Letter-Web.pdf>

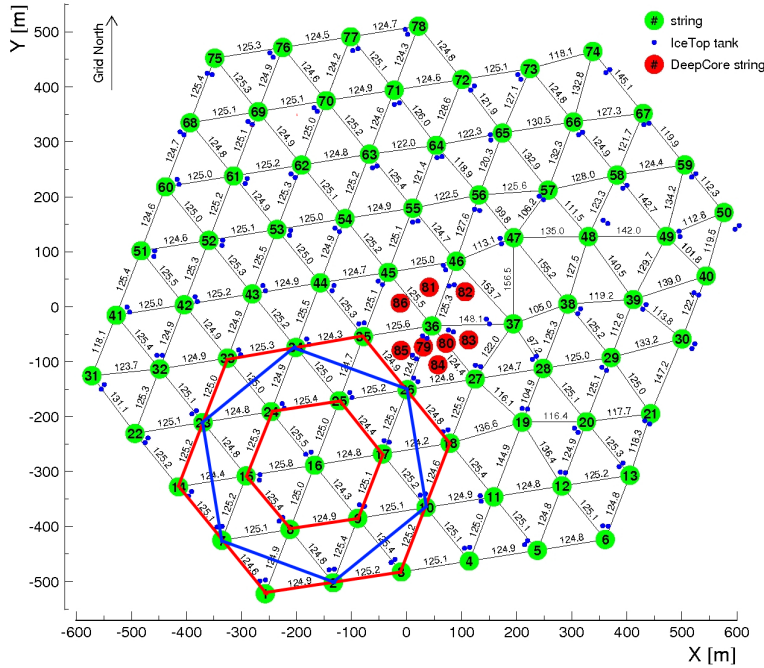


Figure 1: An on-top view of the IceCube detector is shown. The in green depicted 78 strings are on an approximately triangular grid, while the DeepCore strings painted in red are installed in a denser configuration. The hexagons around String 16 show the shape of hexagonal convolution kernels with 7, 13 and 19 included strings.

Figure 1 illustrates the geometry of the IceCube detector. The 86 strings of the IceCube detector, which each hold 60 DOMs, are divided into two detector parts. The first 78 strings are arranged in an hexagonal configuration. In the typical use cases of convolutional layers, the convolution is performed on orthogonal coordinate systems. The convolutional operators in Tensorflow therefore only support convolutions on such coordinate systems. In order to use the convolutional operators, the input data needs to be transformed. For this purpose, the hexagonally shaped data is transformed as described in Fig. 2. In a similar fashion, hexagonally shaped convolutional filters as shown in Fig. 1 are obtained and used for the convolutional layers.

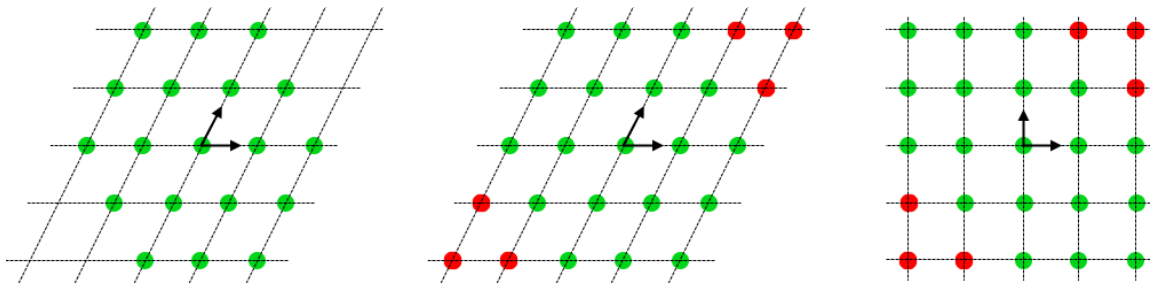


Figure 2: Hexagonally shaped data on the left (green dots) can be transformed into an orthogonal grid on the right by padding with zeros (red dots) and then shifting the rows, so that they align. Hexagonally shaped convolution filters are obtained with the same transformation.

The input data consists of one temporal and three spatial dimensions. For every DOM there is a series of pulses of arbitrary length. Each of these pulses consists of an arrival time and a charge. The pulse series of arbitrary length are condensed into 7 variables consisting of: sum of all charges, number of pulses, time of first pulse, time of last pulse, average time of pulses,

standard deviation of pulse times and highest pulse charge. The first 78 strings are mapped onto an orthogonal 10×10 grid via the described transformation. No transformation is applied to the DeepCore strings. Instead, the x- and y-dimensions of the DeepCore strings are reduced to a single dimension representing the string number. As a result, the input tensor has dimensions $(10 \times 10 \times 60 \times 7)$ and $(8 \times 60 \times 7)$ for the hexagonally arranged strings 1 through 78 and the 8 DeepCore strings, respectively. A different approach with a binning in time, where each bin holds the sum of all charges of the pulses within that time bin, achieves a similar accuracy as the results shown in section 3.

The input of the 78 hexagonally arranged strings is fed into six 3D convolutional layers with hexagonally shaped kernels (Fig. 1), while the DeepCore input is processed by four 2D convolutional layers. The 3D convolution is performed over the three spatial dimensions $(10 \times 10 \times 60)$. Similarly, the 2D convolution is applied on the two spatial dimensions (8×60) of the DeepCore input. The output of these convolutional layers is flattened, combined, and used as input to seven fully connected layers. The last fully connected layer has an output node for every target label. Currently, 14 different labels are being reconstructed such as the neutrino energy, the muon energy and the arrival direction of the muon.

In addition, the input data as well as the target labels are normalized to have a mean of 0 and a standard deviation of 1. Since the use of batch normalization [8] makes the training of arbitrary activation functions tractable, a combination of different activation functions is used. The motivation for this is that each activation function has a specific use case for which it is well suited. A combination of different activation functions might, therefore, be better suited to approximate complex problems. During training of the neural network, the learning rate of the gradient descent minimizer and the dropout rate are gradually decreased from 10^{-2} to 10^{-7} and from 0.2 to 0.05, respectively. Dropout[9] is a regularization technique, where a certain rate (dropout rate) of neurons are dropped during training. This prevents overfitting and makes the network more robust, as it cannot rely on the output of a single neuron. In connection with the high amount of training data and the comparably low complexity of the model, only a small amount of dropout is needed to prevent overfitting.

It should be noted that the architecture and settings for the neural network as described above are not yet fully optimized. Improvements through further optimization are expected.

3. Runtime and Performance

As mentioned in section 1, one of the key challenges for an on-site reconstruction is the runtime requirement. The Level2 Online Muon Filter, which includes several track and energy reconstructions, is one of the multiple filters being run on the Online L2 Selection. Its average runtime is about 273ms/event. However, the runtime is highly dependent on the event, with a standard deviation of 605ms/event. Single, very luminous events can require several seconds to process. In comparison, the neural network as described in section 2 has a runtime of about 20ms/event on an NVIDIA Quadro M1000M and a runtime of approximately 2ms/event on an NVIDIA Tesla P40. For the NVIDIA Tesla P40 the bottleneck appears not to be the prediction itself, but the time needed to feed the data into the network. A direct comparison between the runtimes, however, is difficult because the Level2 Online Muon Filter is performed on a CPU. In addition, the runtime

of the neural network is very stable with a standard deviation of less than 5ms/event. The network always performs the same number of mathematical operations regardless of how luminous the event is. Therefore, the reconstruction by the neural network could be added to the Level2 Online Muon Filter without violating the runtime restrictions. Moreover, if more severe limitations were required, the network could easily be pruned with only little effects on the reconstruction performance.

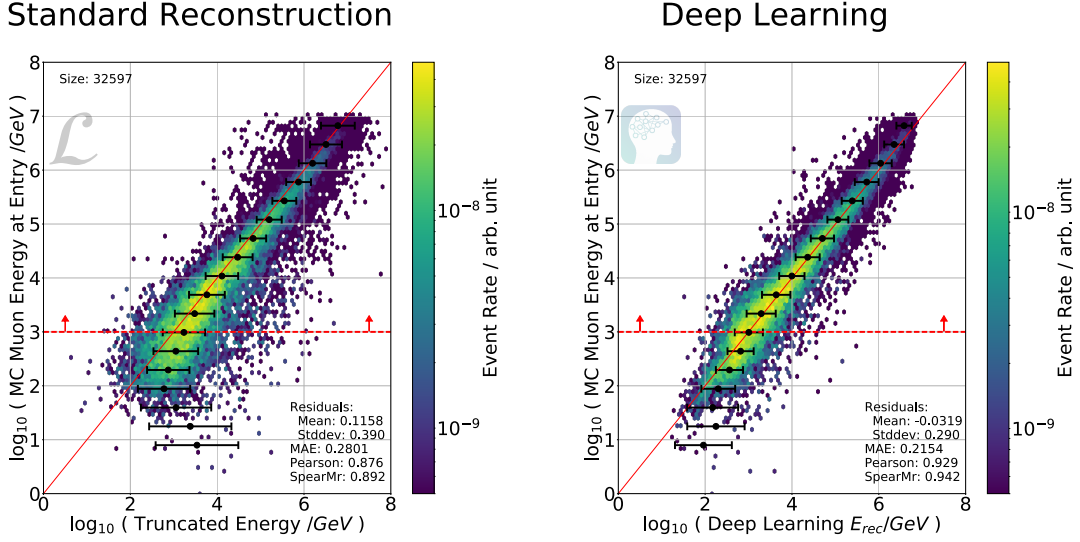


Figure 3: The correlation plots between Monte Carlo truth (y-Axis) and reconstructed energy (x-Axis) are shown for the current standard reconstruction (Truncated Mean [2, p. 18]) and the deep learning approach. The deep learning approach significantly improves the resolution.

One of the target labels reconstructed is the muon energy at its entry into the detector. This is currently the most widespread used proxy for the energy of the primary neutrino. In Fig. 3, the correlation plots are shown for the current standard reconstruction (Truncated Mean [2, p. 18]) and the deep learning approach. It should be noted that the results obtained here are for events of the Online L2 Selection, while additional quality cuts are applied in [2, 10]. The resolution of the reconstruction can be measured in many different ways. For a high resolution, a low standard deviation of the residuals and a high correlation between reconstruction and Monte-Carlo-Truth is expected, as well as a low mean absolute error (MAE). In comparison to the current standard, the deep learning approach can reduce the MAE and the standard deviation of the residuals by over 20%. The standard reconstruction uses the energy loss dE/dx to predict the muon energy. Therefore, it is not well suited for energies below about 1 TeV, where dE/dx is no longer proportional to the muon energy. The deep learning approach does not have this limitation. To obtain a fair comparison, only events with a muon energy above 1 TeV, indicated by the dashed line in Fig. 3, are used to calculate the resolution. The reconstruction of other energy related target labels also yields significant improvements. The directional reconstruction of events, however, currently still remains a challenge. This is further discussed in section 5.

In addition, it should be noted that the deep learning approach is able to reconstruct all events.

The standard reconstructions are only performed if certain quality criteria are fulfilled. The events on which the standard reconstruction is not performed or when it fails are often events which are hard to reconstruct. This could, for instance, be due to events clipping the instrumented volume or to low energetic and dim events. The deep learning approach is still able to reconstruct these events with a resolution comparable to the resolution of the events successfully reconstructed by the standard method.

4. Uncertainty Estimation

In order to get an estimate of the uncertainty on the reconstruction, the network architecture described in section 2 is extended by a small neural network of three fully connected layers. The first fully connected layer of the original architecture is used as input for this new, smaller network. Additionally, a gradient stop is applied to the input, so that the training of the new, smaller network will not affect the original network. The last layer of the new, smaller network has one output node for every target label. It predicts the absolute deviation between the reconstructed and the true value.

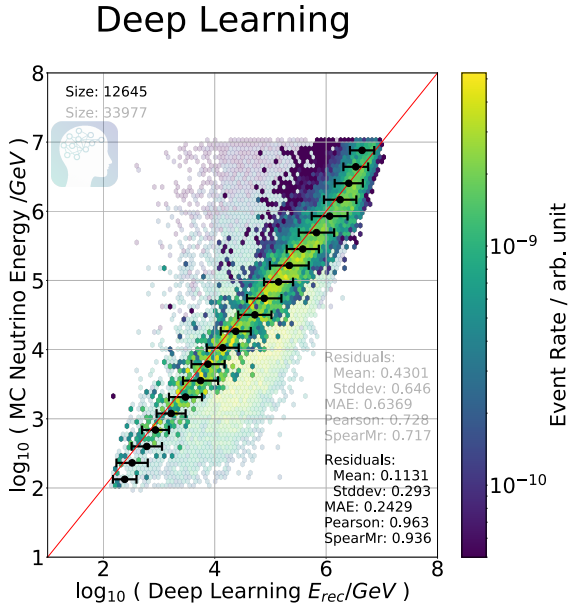


Figure 4: Only the top 10% of events are chosen, in which the neural network estimated the smallest uncertainty. The original sample is shown in the background. In the top left corner, the number of unweighted events is shown for each sample. The uncertainty estimation performed by the neural network can be used to choose a subsample of well reconstructed events, improving the resolution of the neutrino energy by more than 50%, while keeping a tenth of all events.

Although this is just a first guess of the uncertainty, the results are very promising. This can be demonstrated by using the uncertainty estimation performed by the network to select a subsample of well reconstructed events. An example is shown in Fig. 4. It should be emphasized that the reconstruction of the primary neutrino energy is an inherently difficult task.

Moreover, the estimated uncertainty for the deep learning reconstruction can also be used for the standard reconstruction as shown in Fig. 5. Events reconstructed well by the deep learning approach are also well reconstructed by the standard reconstruction. A direct estimation of the uncertainty specifically for the standard reconstruction can be performed to further improve the results. This is of special interest for the standard track reconstruction performed on-site in regard to real time alerts and follow-up observations.

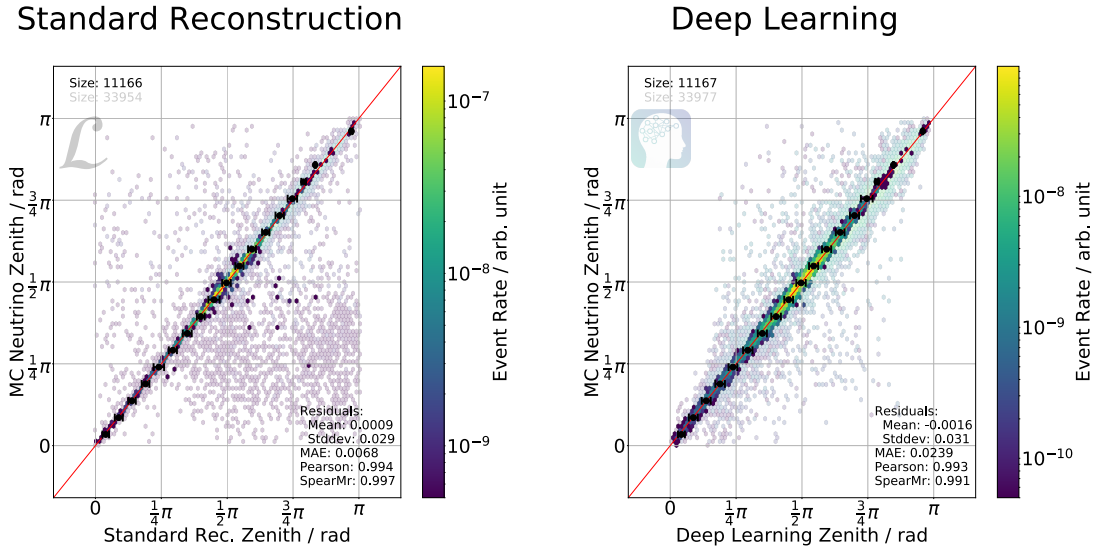


Figure 5: The uncertainty estimation of the neural network for its prediction on the neutrino zenith angle is used to select a sample of well reconstructed events. Although the uncertainty estimation is performed for the deep learning reconstruction, it is also applicable to the standard reconstruction.

5. Directional Reconstruction

The directional reconstruction of events is difficult for various reasons. In contrast to the reconstruction of energy related labels, the directional reconstruction heavily depends on the timing information in addition to the spatial dimensions. Moreover, target labels like the azimuth angle are cyclic. This is difficult in practice with regard to gradient descent minimization. One way of circumventing this is by reconstructing the coordinates of a direction vector separately.

In IceCube, the first pulse registered by a DOM is the most useful for a directional reconstruction, since the photons will have scattered the least. This fact is also used by the standard directional reconstruction currently used on-site which only uses the first pulse as well as the total number of pulses [11, pp. 35 ff]. Therefore, a deep learning approach using these values as input essentially reduces the dimensionality of the problem and should be able to reproduce comparable results. However, while the deep learning approach is able to better reconstruct the direction on average, it currently can not compete with the median and 68%-quantile resolution of the standard reconstruction. Adjusting the loss function from mean squared error to something less outlier sensitive only partially improves the result.

One possible reason for this might be the deviations of the 78 approximately hexagonally arranged strings from an exact hexagonal grid as shown in Fig. 1. The positions of the DOMs deviate up to 35m and about 10m on average off of the perfect virtual hexagonal grid. A convolutional layer assumes symmetry over the input data. However, the deviations break this symmetry which might lead to a worse reconstruction. Investigations are currently underway to validate this assumption. If this holds true, measures can be taken before the convolution in order to correct for these deviations.

6. Conclusions and Outlook

One of the key challenges for an on-site reconstruction is the hardware limitation. The deep learning approach presented here is able to significantly improve the energy reconstruction while reducing the runtime. In addition, the neural network is able to estimate the uncertainty of its prediction as well as the uncertainty of other reconstruction methods. It is therefore well suited for an application at the South Pole. Moreover, the network architecture is not yet fully optimized, but already generates very promising results. Improvements are expected through further optimization.

Currently, a study investigating the effects of an imperfect hexagonal grid is underway. Depending on the results of this study, measures to correct the pulse times for the actual DOM positions can be undertaken. A full integration of the time dimension via a 4D convolution or a combination of a 3D convolution over the spatial dimensions and a recurrent neural network to handle the time dimension might result in an even better performance. Additionally, the third part of the IceCube detector, IceTop, can be integrated as well. Moreover, experience has shown that in most cases training on a specific subset of data with a similar topology converges faster and better than training the same, or a more complex model, on a bigger subset with more event topologies. This leads to the idea of performing an online classification into different topologies. A dedicated network for each topology could then be executed after the classification. This will effectively reduce the complexity of the problem.

7. Acknowledgements

Mirco Hünnefeld acknowledges the support of the Martin-Schmeißer-Stiftung, Germany.

References

- [1] **IceCube** Collaboration, M. Aartsen, et al., *JINST* **12** (2017) P03012.
- [2] **IceCube** Collaboration, M. Aartsen, et al., *JINST* **9** (2014) P03009.
- [3] **IceCube** Collaboration, M. Aartsen, et al., *Astropart. Phys.* **92** (2017) 30 – 41.
- [4] G. Hinton, A. Krizhevsky, et al. in *Advances in Neural Information Processing Systems 25*, pp.~1097–1105. Curran Associates, Inc., 2012.
- [5] Y. LeCun, Y. Bengio, and G. Hinton, *Nature* **521** (2015) 436–444.
- [6] **IceCube** Collaboration, M. Aartsen, et al., *The Astrophysical Journal* **833** (2016) 3.
- [7] M. Abadi, et al. 2016. Software available from tensorflow.org.
- [8] S. Ioffe and C. Szegedy, *CoRR* [abs/1502.03167](https://arxiv.org/abs/1502.03167) (2015).
- [9] N. Srivastava, et al., *J. Mach. Learn. Res.* **15** (Jan., 2014) 1929–1958.
- [10] **IceCube** Collaboration, R. Abbasi et al., *Nucl. Instrum. Meth.* **A703** (2013) 190–198.
- [11] K. Schatto. PhD thesis, [Mainz U.](https://www.uni-mainz.de/), 2014-06-02.