Technical Report

RISE Germany Internship: Generative Neural Network Reconstruction of IceCube Cascade-like Events

Anthony Flores, Mirco Huennefeld

08/2018

technische universität dortmund

SFB 876 Verfügbarkeit von Information durch Analyse unter Ressourcenbeschränkung

Part of the work on this technical report has been supported by Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center SFB 876 "Providing Information by Resource-Constrained Analysis", project C3.

Speaker: Prof. Dr. Katharina Morik
Address: TU Dortmund University
Joseph-von-Fraunhofer-Str. 23
D-44227 Dortmund
Web: http://sfb876.tu-dortmund.de

1 Introduction

The IceCube Neutrino Observatory is located at the South Pole and encompasses ~1km^3 of polar ice [1]. About 5000 Digital Optical Modules (DOMs) located about 1.5 - 2.5 km beneath the surface of the ice are designed to capture the Cherenkov radiation emitted by charged secondary particles produced in neutrino interactions as they travel through the detector. Two types of events can be resolved by the detector: tracks and cascades [2]. Track-like patterns in the detector occur when a neutrino-induced muon passes through the ice. Cascade events – the subject of this report – are spherical light patterns produced by particle showers that are either hadronic or electromagnetic in nature. Of particular interest in IceCube data analysis is event reconstruction, or obtaining the parameters needed to describe the particle as it interacts in the detector. For cascade-like events, these parameters include the vertex (x, y, z coordinates and the interaction time t) of the particle as it first interacts with the detector medium, the initial direction of the particle (as described by zenith and azimuthal angles), and the energy of the particle for a total of seven parameters. For track-like events, the distinctive path of the muon allows current IceCube algorithms to reconstruct the initial direction up to an angular resolution of $\sim 1°$ [2]. For cascade events, which resemble a point source in the detector, the initial direction is much more difficult to reconstruct than the vertex or the energy.
In this report, Section 2 will detail the creation and validation of a new Neural Network technique to reconstruction. Section 3 will present the results of this network when attempting to reconstruct cascade events, and Section 4 will present a discussion of these results before a brief conclusion.

2 Methods

2.1 Loss Functions

In order to properly train our generative neural network, we needed to select a loss function to evaluate the performance of our network when compared to a validation set of simulations. The proposed loss functions were as follows:

1. χ^2 statistic,
$$\chi^2 = \frac{(s-d)^2}{s/n_s^2 + d/n_d^2} \tag{1}$$
where s is the DOM hit count for generated simulations, d is the DOM hit count of the test or validation data, and n$_s$ and n$_d$ are the number of trials of simulation or data.

2. MSE statistic,
$$MSE = \frac{1}{n}\sum(s-d)^2 \tag{2}$$
essentially an average, unnormalized χ^2

3. Likelihood statistic [3]

$$\mathcal{L} = \left(\frac{\mu}{s/n_s}\right)^s \cdot \left(\frac{\mu}{d/n_d}\right)^d \tag{3}$$

a poissonian likelihood ratio assuming the detection of photons can be considered a counting experiment. s, d, n_s, and n_d are defined as above, with μ being the expected rate of counts

2.2 Small Scale Training

Before constructing a network that could simulate events throughout the entire IceCube detector, we began by simplifying the training set and input hypothesis. In this case, we simulated events inside a detector that had only 27 DOMs arranged in a three-layer cube with side length 100 m. The cascade hypothesis was simplified to contain only the three parameters that described the vertex of the event (x, y, and z coordinates), and furthermore the z-coordinate of simulated cascades was fixed to be 50m, or the middle layer of the detector. This allowed us to generate cascade events in a two dimensional plane. Selecting an event from our test data, we calculated the three loss statistics between this event and the generating cascade, which resulted in a 2-D landscape of loss values. We simulated events and calculated these statistics at 0.5 increments from 0-100m in the x- and y-directions, yielding 2601 total events. We present the loss landscape in figure 1. Through brute-force minimization of the 2-D array, we retrieve the x- and y-values that represent the minimum loss, and therefore the closest simulated event to our chosen data event. The χ^2 and MSE loss landscapes were much more well-defined than the Likelihood landscape, so these two loss functions were selected to be used in training the full sized network moving forward.

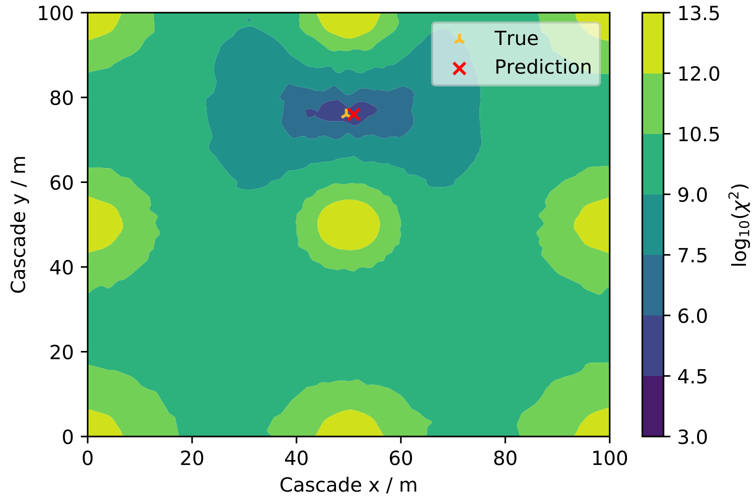
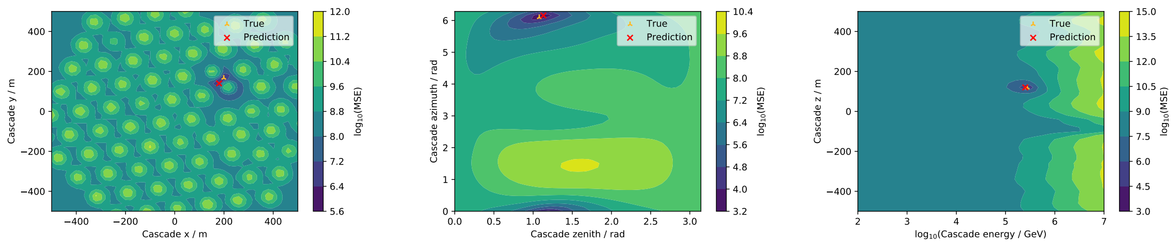


Figure 1: Chi Square loss landscape for toy simulations. Vertex of event compared to generated DOM response marked by red cross. Note the nine DOMs in the layer clearly visible at their positions every 50m.

2.3 Full Network

Based on the success of the generative network using the toy simulations, the network was adapted to train on events with a six-parameter hypothesis (three for vertex, two for direction, and one for energy). The limitation on the z-coordinate was removed, and all DOMs in the full IceCube observatory were included. In order to assess the success of this network, we took an approach similar to the toy network. A specific test event was selected from IceCube simulations, and its true cascade parameters were recorded. Of these six parameters, 4 were fixed to the true value, and the remaining two were varied between 0 and their maximum value to create 2601 events as before. The trained network simulated DOM responses for each of these 2601 events and created a loss landscape just as before. Through the same 2-D brute-force minimization, the parameter values at the minimum were obtained. We performed this validation by varying x and y while holding the z-coordinate, directional angles, and energy constant, by varying the zenith and azimuthal directional angles and keeping the vertex and energy parameters fixed, and by varying energy and the z-coordinate while fixing the x and y vertex components and directional angles. The resulting landscapes for a particular neural network with MSE loss function, 5 hidden convolutional layers (to efficiently share weights), and a final fully connected layer is presented in Figure 2. To further test the success of the networks, the resolution of the landscapes was increased by generating 2601 events but only in the region surround the true minimum. These fine resolution landscapes are presented in Figure 3.

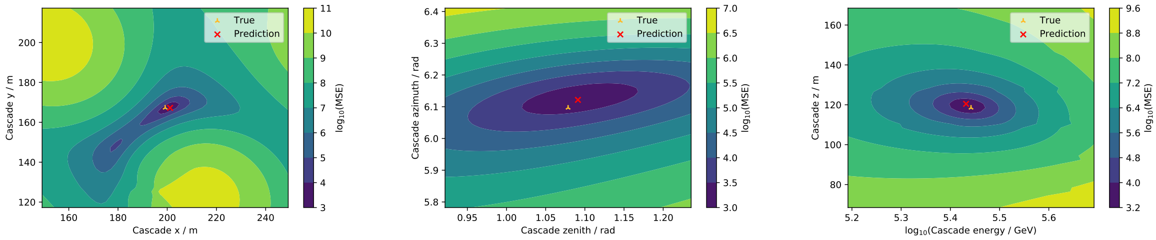


(a) x, y Plane (b) Zenith, Azimuth Plane (c) Energy, z Plane

Figure 2: 2-D loss landscapes for the generated network. In each landscape, two parameters were varied while the other four were fixed to their true value. Notice that in the x, y landscape (left), the DOM placement of the IceCube detector is clearly visible as it was in figure 1. Also notice in the zenith, azimuth landscape (middle), that the loss values wrap around in the azimuthal direction between 0 and 2pi. The true parameters of the selected event are marked with a yellow triangle, and the predicted parameter from the minimum are marked with a red cross.

2.4 Reconstruction

Having trained a generative neural network to simulate DOM responses, we were able to apply it to a novel reconstruction method. This method allowed for the selection of



(a) x, y Plane (fine) (b) Zenith, Azimuth Plane (fine) (c) Energy, z Plane (fine)

Figure 3: Increased resolution landscapes for the same event as Figure 2. DOMs surrounding event are still clear in the x, y plane (left), and the landscape for both angles and energy is still smooth at this increased resolution.

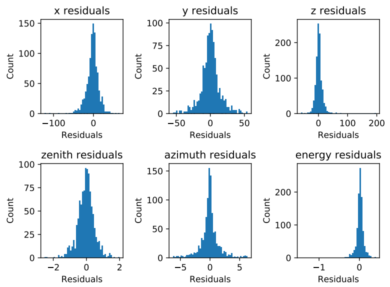
optimization tool, either through the use of the *Scipy Optimize* package or *Tensorflow*. If the Tensorflow method was specified, an initial cascade hypothesis would be passed to the built-in Tensorflow optimizer. If the Scipy option was selected, the 6 cascade parameters were wrapped in an MSE loss function, which was passed directly to the Scipy minimizer. Additionally, a second reconstruction method was available in the Scipy option that fit parameters iteratively. Similar to the landscape approach to validation, any number of parameters could be fixed by specification in a config file, and only the remaining parameters would be reconstructed. The best fit from this reconstruction would be passed to the next reconstruction. Any number or combination of parameters could be fit during any iteration. For either the Scipy or Tensorflow option, an option was available to provide an initial reconstruction seed as an initial point for optimization.

3 Results

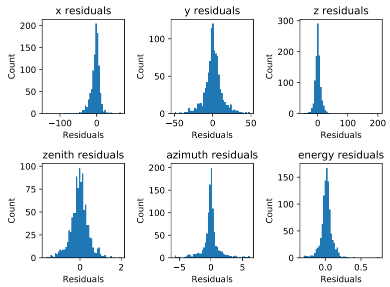
All three reconstruction options (Tensorflow, Scipy, and Scipy Iterative) were employed to reconstruct a test set of IceCube data. In all cases, an initial seed was provided by a previously trained Deep Neural Network as detailed in [4]. While not used directly for reconstruction, the network was designed to yield the six cascade parameters from the ∼5000 DOM responses in the detector. The residuals between the reconstructed parameters and the true values as well as the opening angles due to reconstruction as a function of energy are provided in Figures 4 and 5. The standard deviations of residuals for the three methods are presented in Table 1.
The Tensorflow method performed the best reconstruction in terms of both residuals and opening angle. Additionally, it was the fastest method, reconstructing more than 1000 events in ∼2 minutes. The iterative method performed the worst, being both the slowest, and having the largest standard deviation of residuals and average opening angle.

Parameter	Scipy st dev.	Tensorflow st dev.	Iterative st dev.
x (m)	15.705	13.335	16.004
y (m)	13.994	11.558	15.044
z (m)	12.990	13.084	13.489
Zenith (rad)	0.499	0.432	0.530
Azimuth (rad)	1.604	1.473	1.646
Energy (log10 GeV)	0.098	0.080	0.104

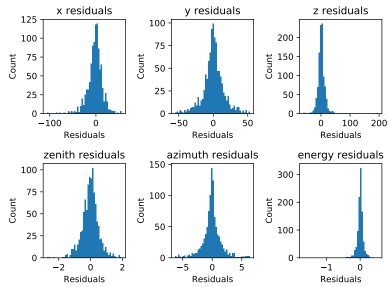
Table 1: Standard deviations for the residual plots found in Figure 4. Note that the Tensorflow reconstruction method has the smallest standard deviations in five of six parameters.



(a) Scipy Residuals



(b) Tensorflow Residuals

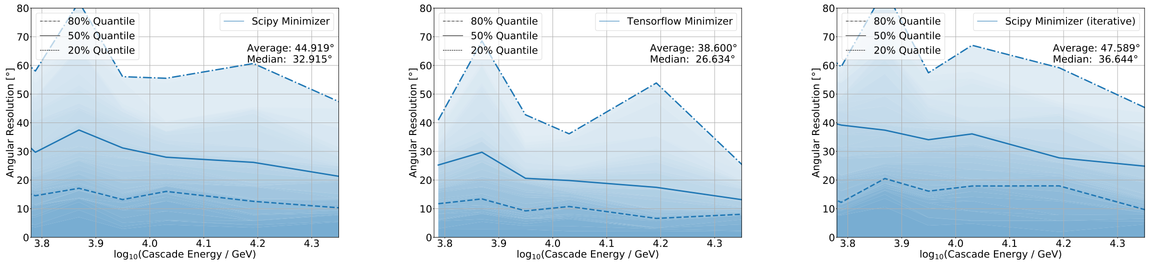


(c) Scipy (Iterative) Residuals

Figure 4: Residuals for all six reconstructed parameters for the three reconstruction methods. Standard deviations for these residual plots found in Table 1.

4 Discussion

The results presented here are evidence for a solid step in finalizing a novel reconstruction methodology. While the speed of the Tensorflow method and the deterministic nature of a neural network are attractive, there is still work that can be performed to improve the accuracy of reconstruction, especially in terms of the angular resolution. The generative network is performing as expected. Given that the characteristic spacing of DOMs is 125m [1], and the residuals for vertex are all under 20m, local minima in the x, y, z, and energy landscapes do not seem to greatly affect reconstruction of those parameters. In contrast, the angular resolution is heavily influenced by local minima. Because of this, while vertex and energy reconstruction are performing decently, the failure to reconstruct initial direction causes our method to perform along the same lines as the DNN Seed used as an initial guess (presented in Figure 6). The directional reconstruction of cascade events has always been more difficult than muon tracks based on their shape [2]. As the final DOM response of the cascade appears spherical in the detector, the angle of entry may not be immediately clear to the network, which could indicate that timing information is key to reconstructing the direction. New generative networks are training using time bins to account for charge progressively collected by the DOMs, and initial results show



(a) Scipy (b) Tensorflow (c) Scipy (Iterative)

Figure 5: Opening angle (angular resolution) plots for the three minimization techniques. Tensorflow reconstruction method (middle) had best results with both lowest average and median opening angles.

the networks beginning to learn this information. Once the training is complete, new reconstructions will be performed with the aim of improving angular resolution from that of the current networks and DNN seed.

5 Conclusions

A generative neural network was trained with a six parameter hypothesis to simulate cascade neutrino events in the IceCube detector. After successful validation, this network was used to reconstruct neutrino events to obtain the best-fit parameters for vertex, initial direction, and energy based on an initial seed from a separately trained deep neural network. Several options were provided for minimization techniques in the reconstruction, and the built-in Tensorflow optimizer provided the fastest and most accurate minimization. The reconstruction of vertex and energy parameters is satisfactory, but the nature of cascade events makes directional reconstruction difficult, especially at lower energies. While this reconstruction is only marginally better than the seed provided by the DNN, the future inclusion of timing information may drastically improve the reconstruction of zenith and azimuth directional angles.

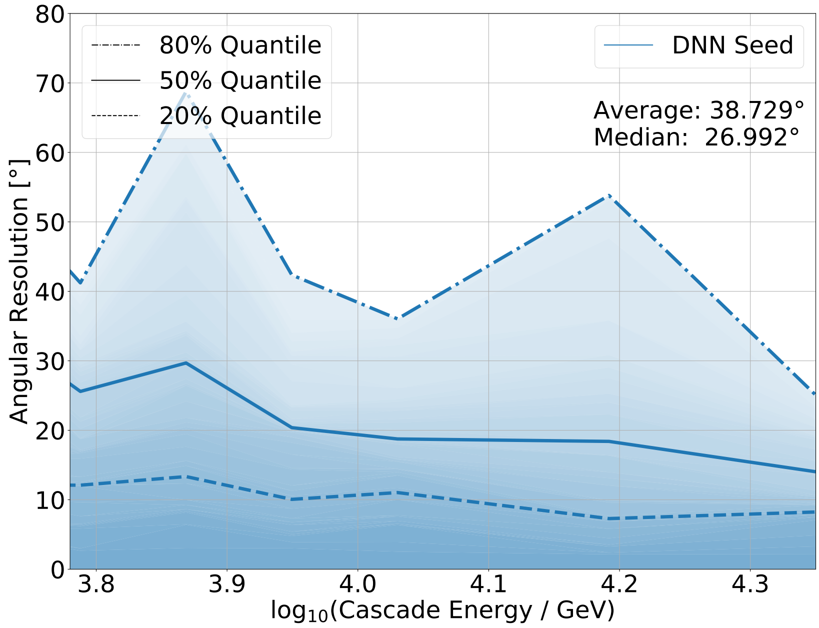


Figure 6: Opening angle (angular resolution) plot for the Deep Neural Network (DNN) seed without any additional reconstruction. Note that the average and median opening angle are only slightly larger than those from the Tensorflow reconstruction method. It is hoped that the inclusion of timing information will more clearly separate these two methods.

References

[1] A. Achterberg *et al.*, *First year performance of the IceCube neutrino telescope*, Astroparticle Phyiscs, vol. 26, Issue 3, pp. 155-173, October 2006.

[2] M. G. Aartsen *et al.*, *First Observation of PeV-Energy Neutrinos with IceCube*, Physical Review Letters , vol. 111, Issue 2, July 2013.

[3] D. Chirkin, *Likelihood description for comparing data with simulation of limited statistics*, arXiv:1304.0735 [astro-ph.IM], April 2013.

[4] M. Huennefeld (IceCube Collaboration), *Deep Learning in Physics exemplified by the Reconstruction of Muon-Neutrino Events in IceCube*, PoS ICRC2017 (1057), 2017