

Study Cosmic Ray Mass Composition using Deep Learning in Telescope Array Surface Array Detector

Oleg Kalashev *

Institute for Nuclear Research of the Russian Academy of Sciences, Moscow, 117312, Russia

E-mail: kalashev@inr.ac.ru

Mikhail Kuznetsov †

Institute for Nuclear Research of the Russian Academy of Sciences, Moscow, 117312, Russia

E-mail: kuznetsov@inr.ac.ru

The ultra-high-energy cosmic rays mass composition study with the Telescope Array surface detector is discussed. We present the new analysis based on deep convolutional neural network using detector signal time series as an input and trained on a large Monte-Carlo dataset. We compare the sensitivity of the new technique and the previously presented boosted decision tree multivariate analysis built upon 14 observables. Possible systematic errors of the method are discussed.

36th International Cosmic Ray Conference -ICRC2019-

July 24th - August 1st, 2019

Madison, WI, U.S.A.

*For the Telescope Array Collaboration

†Speaker.

1. Introduction

The most reliable and established methods proposed for the ultra-high-energy cosmic ray (UHECR) composition study are based on the extensive air showers (EAS) longitudinal shape observations by the fluorescence telescopes. This method uses the depth of the shower maximum X_{\max} as a composition-sensitive observable [1]. The lateral shower density profiles on ground level, observed by the network of surface detector (SD) stations, can also be used for the study, although no single observable is known that has a comparable to X_{\max} sensitivity to the mass composition. Recently an advanced approach to measure the mass composition in the energy range $10^{18.0} - 10^{20.0}$ eV using solely the surface detector data was proposed [2], which benefits longer than 95% duty cycle of the SD [3]. The method is based on the multivariate boosted decision tree (BDT) technique and uses 14 synthetic composition-sensitive variables obtained in SD event reconstruction as input.

Modern experiments record the full time-resolved signal of each SD station (in the case of the Telescope Array in each of the two layers of the scintillator). One may benefit from the enhanced analysis based on the full signal compared to the traditional methods based mostly on the values that could be measured by the detectors of the previous generation: the arrival time of the first particle and the integral signal of each detector. The successes of the machine learning methods, in particular, deep convolutional neural networks in image recognition and many related tasks including challenges in astroparticle and particle physics made it possible to develop this kind of enhanced analysis. We describe the proposed event reconstruction method in detail in Sec 2 and discuss its application to the UHECR mass composition study in Sec. 3.

2. Event Reconstruction

We use existing standard SD event reconstruction [4] as a first approximation. In the standard reconstruction the event geometry is recovered using the arrival times of the shower front particles measured by the triggered (> 0.3 MIP) detectors. Shower front is approximated with empirical functions proposed in Ref. [5] and later modified in AGASA experiment [6]. The pulse heights in the detectors together with the event geometry information are used to calculate the shower lateral distribution profile normalization S_{800} [7].

In order to determine the Linsley front curvature parameter [5] the joint fit of shower front and lateral distribution function is performed with 7 free parameters: $x_{\text{core}}, y_{\text{core}}, \theta, \phi, S_{800}, t_0, a$ [8]:

$$\begin{aligned}
 t(r) &= t_0 + t_{\text{plane}} + a \times (1 + r/R_L)^{1.5} LDF(r)^{-0.5}, \\
 S(r) &= S_{800} \times LDF(r), \\
 f(r) &= \left(\frac{r}{R_m}\right)^{-1.2} \left(1 + \frac{r}{R_m}\right)^{-(\eta-1.2)} \left(1 + \frac{r^2}{R_1^2}\right)^{-0.6}, \\
 t_{\text{plane}}^i &= \frac{1}{c} \vec{n} \cdot (\vec{R}_i - \vec{R}_{\text{core}}), \\
 R_m &= 90.0 \text{ m}, R_1 = 1000 \text{ m}, R_L = 30 \text{ m}, \\
 \eta &= 3.97 - 1.79(\sec(\theta) - 1),
 \end{aligned}$$

$$r = \sqrt{(\vec{R}_i - \vec{R}_{core})^2 - (\vec{n}(\vec{R}_i - \vec{R}_{core}))^2},$$

where \vec{R}_{core} is the location of the shower core, \vec{R}_i are the locations of each station of an event, obtained from the pre-defined coordinate system of the array centered at the Central Laser Facility (CLF) [9], t_{plane} is the arrival timing of the shower plane at the distance r , \vec{n} – unit vector towards the direction of arrival of a primary particle, c is a speed of light and a is the Linsley front curvature parameter. When the fit is done the primary particle energy is estimated as function

$$E = E_{SD}(S_{800}, \theta)$$

of density S_{800} at distance 800 m from the shower core and zenith angle θ using lookup table obtained by Monte Carlo simulation [4].

In both standard and machine learning based reconstruction procedures the full Monte-Carlo simulation of TA SD events is used. We sample p, He, N and Fe nuclei in equal fractions and primary energies distributed assuming the spectrum of HiRes experiment [10]. The events are generated by CORSIKA [11] with EGS4 [12] model for electromagnetic interactions, QGSJET II-03 [13] and FLUKA [14] for high and low energy hadronic interactions. Below we also investigate the effect of high energy hadronic interaction model choice using proton set generated with QGSJET II-04 [15]. The event sets are sampled assuming isotropic primary flux with zenith angles $\theta < 45^\circ$. We apply the reconstruction quality cuts described in the previous composition study [2] to the simulated events in the same way they are applied to data.

2.1 Deep Learning-based Event Reconstruction

The generic idea of the new method is to use the raw observables, i.e. the time-resolved signals for the set of adjacent triggered detectors to recover primary particle properties instead of (or in addition to) synthetic observables used in [2]. The full detector Monte Carlo simulation [4] of the TA observatory is used to produce the time-resolved signals from the two layers of SD stations. The simulation gives us the raw observables as a function of primary particle properties and what we aim to obtain is the inverse function. A multilayer feed-forward artificial neural network (NN) is an appropriate tool for that, since it is known to be able to approximate any continuous function with any finite accuracy [16]. In practice, the achievable accuracy is often limited by stochastic nature of the problem introducing unavoidable bias. The systematic uncertainty, e.g. hadronic interaction model at ultra-high energies does not constrain the NN accuracy nominally, however it should be taken into account when interpreting NN model output. Other than that, the accuracy of NN model is limited just by the complexity of the network, i.e. the number of trainable weights, with more complex networks requiring more data to train.

For our purpose we utilize convolutional NN, a widely used special kind of feed-forward NN optimized for image and sequence processing [17]. In Fig. 1 we show an example of NN architecture used in this work. The typical UHECR event triggers 5 to 10 neighbour stations. The readouts from 4×4 grids¹ are used as input with each pixel corresponding to a particular detector. There are two time-resolved signals, one per layer, for each detector. The typical length of signal recorded with 20 ns time resolution does not exceed 256 points. The overall dimensionality of raw

¹we also tried 6×6 grid configurations which appeared to be less efficient for mass composition study

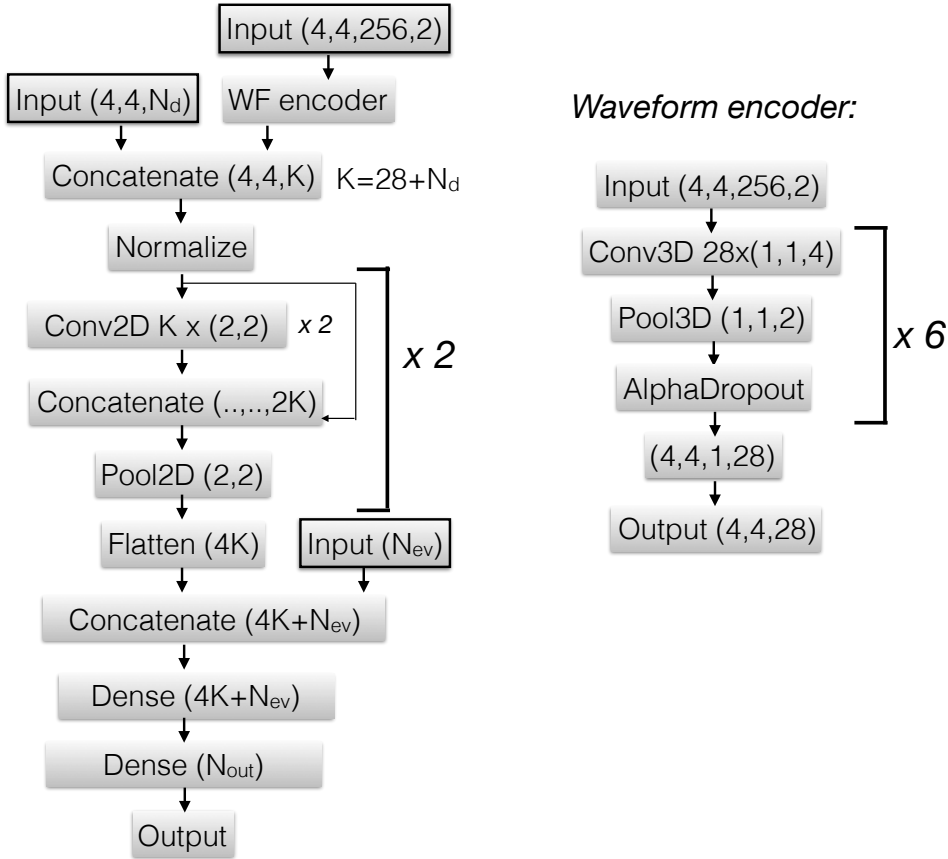


Figure 1: Neural network architecture

input data is therefore $4 \times 4 \times 256 \times 2$. The grid is centered at the event core. An approximate position of the later is estimated using the standard reconstruction.

The full time-resolved signal from two scintillator layers of each surface detector is first converted to 28-dimensional feature vector using waveform encoder, consisting of six convolutional layers followed by max-pooling and α -dropout [18]. The later helps to avoid overfitting, while keeping mean and variance of inputs to their original values by randomly setting activations to the negative saturation value. The scaled exponential linear units are used as activations to achieve the self-normalizing property [18].

The extracted features from the waveforms are treated as a multichannel image using sequence of 2D convolution layers. We also add some extra detector (or detector signal) properties N_d to the set of extracted detector signal features before feeding them to 2D convolution model. These properties include exact detector position (more precisely, altitude and offset from rectangular grid), detector state (on/off/saturated) and optionally standard reconstruction parameters (integral signal, timing relative to plane front).

In principle, one could utilize 3D convolutions to analyze the raw signal from the grid with time as third dimension, however in this case it wouldn't be as easy to include extra detector properties as input to the model. We tried this approach and found it to be less effective, perhaps

since signals from neighbor detectors are weakly correlated.

To properly take into account missing or switched off detectors, we have introduced special *Normalize* layer, somewhat similar to *Dropout* [19]. Namely, it drops the pixels corresponding to missing detectors and multiplies the activations of the present detectors by factor of $N_{total}/(N_{total} - N_{missing})$. This trick enhances the explained variance by few percent.

The 2D-convolutional model part consists of two blocks built from two convolutional layers with 2×2 kernels followed by 2×2 max-pooling. Before pooling the original signal is concatenated with the convolved one. The 2D convolutional model outputs a feature vector which is then processed by two fully connected layers. Before passing the feature vector to the first dense layer it is concatenated with extra event properties. At this step we add a set of 14 composition sensitive synthetic observables used in previous analysis [2] as well as season and day time to account for seasonal variations in atmosphere properties.

The NN can be used either as regression model for predicting primary nucleon mass $\log(A)$ directly or as classification model. In the later case NN outputs vector of probabilities for each class, corresponding to particular nucleon. The mean square error loss function is used for regression and cross-entropy loss is used for classification. The weights are optimized using adaptive learning rate method Adadelta [20]. The optimization is performed for 400 epochs with conditional early stop i.e. unless the loss function on validation data is not improving for more than 10 epochs.

The raw waveform data is converted to log scale before passing it as an input to the model. The rest of input and output data are normalized to have zero mean and unit variance. We also use log scale for output energy and particle mass.

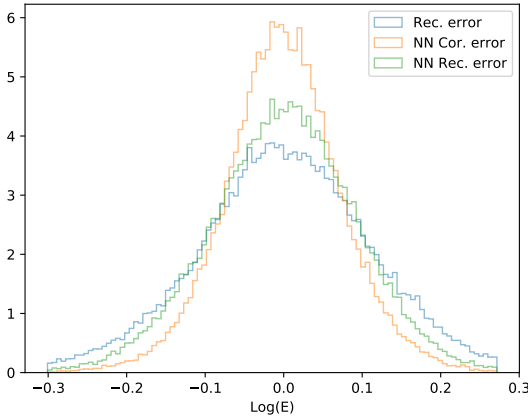


Figure 2: $\log_{10}(E)$ reconstruction error distribution for standard reconstruction scheme (blue curve), and pure NN reconstruction (green curve), NN-based reconstruction using classical reconstruction results (yellow curve).

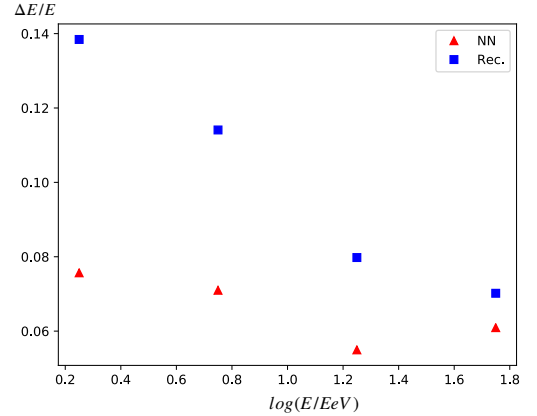


Figure 3: Relative standard deviation of energy reconstruction error $\sigma(E)/E$ for classic reconstruction scheme (blue squares) and CNN regression (red triangles) as a function of the absolute energy.

For the energy and arrival direction reconstruction we found it useful to predict correction to standard reconstruction instead of its absolute value. Fig. 2 illustrates the effect of the approach on primary particle energy E reconstruction. We plot $\log_{10}E$ error distribution for standard reconstruction (blue curve) along with NN-based reconstruction aiming to predict either absolute value

of $\log_{10}(E)$ (green curve) or the correction to the result of standard reconstruction (yellow curve). The later method produces the most accurate results.

The NN model is implemented in *Python* using *Keras* [21] library with *Tensorflow* backend. We also use *hyperopt* package [22] to optimize model hyperparameters, such as dropout rate, dimensionality of waveform encoder output, $L2$ -regularization coefficient.

3. Composition Study

For the composition study we train the NN to predict $\log_{10}(A)$. We call ξ the NN prediction as function of raw observables. The distribution of ξ variable for any fixed composition is quite wide due to similarity of the showers produced by different nuclei and stochastic nature of EAS development. In Fig. 4 we show the distribution for several energy bins assuming either proton or iron primary particles.

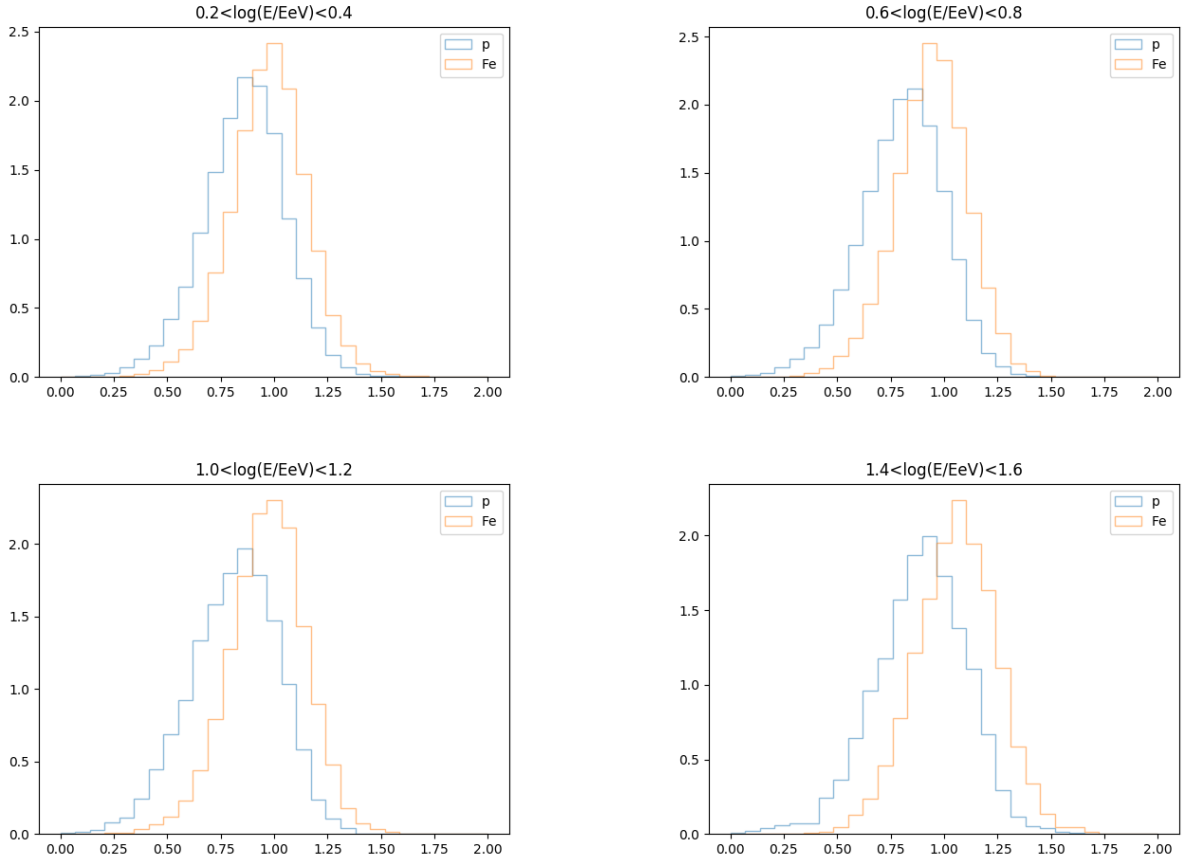


Figure 4: Distribution of neural network function ξ for proton and iron primary particles

To estimate average $\log(A)$ we follow the procedure proposed in previous analysis [2]. Namely, we first fit ξ distribution with mixture of p and Fe using method described in Ref. [23] and implemented in *TFractionFitter* ROOT package. Then as first approximation we use

$$\langle \ln A \rangle^{(1)} = \varepsilon_p \times \ln(M_p) + (1 - \varepsilon_p) \times \ln(M_{Fe}),$$

$\langle \ln A \rangle$ error	BDT	This work
Statistical	0.1	0.05
Systematic	0.44	0.14
Hadronic IM	0.74	0.75

Table 1: $\langle \ln A \rangle$ uncertainties in BDT (Ref. [2]) method and this work.

where ε_p is fraction of protons and M_p, M_{Fe} are atomic masses of proton and iron nuclei respectively. As a second step, we apply the above procedure to the ξ distributions obtained for pure *He* and *N* nuclei primaries which gives the error of the first approximation. Then we introduce 2-nd order polynomial correction to compensate the above error:

$$\langle \ln A \rangle^{(2)} = P_2(\langle \ln A \rangle^{(1)})$$

In Fig. 5 we illustrate the power of the method when applied to mono-type composition. The statistic error was calculated via uncertainty of ε_p by setting the amount of generated UHECR events to the one available after 9 years of exposure.

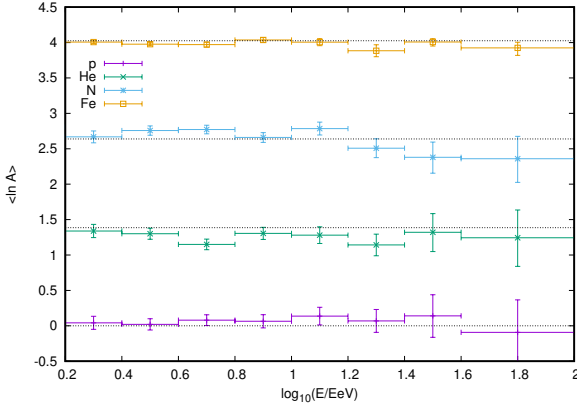


Figure 5: Model $\langle \ln A \rangle$ estimates along with statistical errors for MC sets generated assuming monotype composition and energy spectrum of TA.

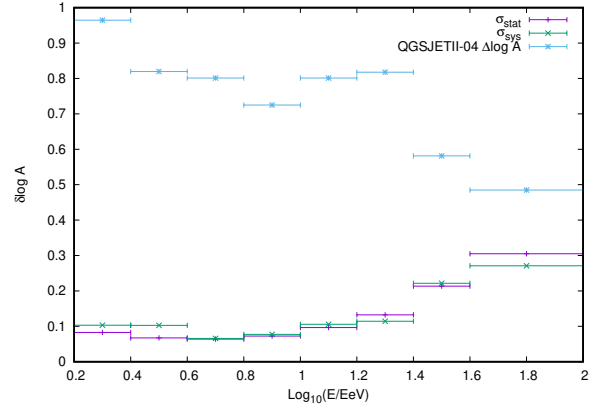


Figure 6: Energy dependence of $\langle \ln A \rangle$ uncertainties: statistical (purple), systematic (green) and hadronic interaction model related (blue)

To estimate the systematic error, 1000 Monte Carlo event sets were sampled in each bin with expected event count following TA SD energy spectrum and varying mass composition while keeping $\langle \ln A \rangle = 2$ using the result obtained in the previous study [2].

Finally, we estimate the uncertainty due to unknown high energy hadronic interaction model by calculation of $\langle \ln A \rangle$ for proton ($\ln A = 0$) set generated with QGSJET II-04 model [15]. In Fig. 6 we compare statistical, systematic and hadronic interaction model related errors in the energy range of interest. To summarize our results, in Table 1 we compare $\langle \ln A \rangle$ uncertainties in the method presented above with the previous study based on the boosted decision tree technique.

Acknowledgments

The work is supported by the Russian Science Foundation grant 17-72-20291.

References

References

- [1] Gaisser T K *et al.* (HiRes) 1993 *Phys. Rev.* **D47** 1919–1932
- [2] Abbasi R U *et al.* (Telescope Array) 2019 *Phys. Rev.* **D99** 022002 (*Preprint* 1808.03680)
- [3] Abu-Zayyad T *et al.* (Telescope Array) 2013 *Nucl. Instrum. Meth.* **A689** 87–97 (*Preprint* 1201.4964)
- [4] Abu-Zayyad T *et al.* (Telescope Array) 2013 *Astrophys. J.* **768** L1 (*Preprint* 1205.5067)
- [5] Linsley J and Scarsi L 1962 *Phys. Rev.* **128**(5) 2384–2392 URL <https://link.aps.org/doi/10.1103/PhysRev.128.2384>
- [6] Teshima M *et al.* 1986 *J. Phys.* **G12** 1097
- [7] Takeda M *et al.* 2003 *Astropart. Phys.* **19** 447–462 (*Preprint* astro-ph/0209422)
- [8] Abu-Zayyad T *et al.* (Telescope Array) 2013 *Phys. Rev.* **D88** 112005 (*Preprint* 1304.5614)
- [9] Takahashi Y *et al.* (Telescope Array) 2011 *AIP Conf. Proc.* **1367** 157–160
- [10] Abbasi R U *et al.* (HiRes) 2008 *Phys. Rev. Lett.* **100** 101101 (*Preprint* astro-ph/0703099)
- [11] Heck D, Knapp J, Capdevielle J N, Schatz G and Thouw T 1998
- [12] Nelson W R, Hirayama H and Rogers D W O 1985
- [13] Ostapchenko S 2006 *Nucl. Phys. Proc. Suppl.* **151** 143–146 (*Preprint* hep-ph/0412332)
- [14] Fasso A *et al.* 2003 *eConf* **C0303241** MOMT005 (*Preprint* hep-ph/0306267)
- [15] Ostapchenko S 2011 *Phys. Rev.* **D83** 014018 (*Preprint* 1010.1869)
- [16] Hornik K 1991 *Neural Networks* **4** 251 – 257 ISSN 0893-6080 URL <http://www.sciencedirect.com/science/article/pii/089360809190009T>
- [17] LeCun Y, Boser B, Denker J S, Henderson D, Howard R E, Hubbard W and Jackel L D 1989 *Neural Computation* **1** 541–551 ISSN 0899-7667
- [18] Klambauer G, Unterthiner T, Mayr A and Hochreiter S 2017 *CoRR* **abs/1706.02515** (*Preprint* 1706.02515) URL <http://arxiv.org/abs/1706.02515>
- [19] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 *Journal of Machine Learning Research* **15** 1929–1958 URL <http://jmlr.org/papers/v15/srivastava14a.html>
- [20] Zeiler M D 2012 *CoRR* **abs/1212.5701** (*Preprint* 1212.5701) URL <http://arxiv.org/abs/1212.5701>
- [21] Chollet F *et al.* 2015 Keras <https://github.com/fchollet/keras>
- [22] Bergstra, J, D Yamins, and D D Cox 2013 *JMLR Workshop and Conference Proceedings* **28(1)** 115–123 URL <https://dash.harvard.edu/handle/1/12561000>
- [23] Barlow R and Beeston C 1993 *Computer Physics Communications* **77** 219 – 228 ISSN 0010-4655 URL <http://www.sciencedirect.com/science/article/pii/001046559390005W>