Identification of Direct Cherenkov Pixels using Boosted Decision Trees

Robert Stein, Atilla Abramowski, Dieter Horns

May 30, 2016

Abstract

When Cosmic Rays travel through the atmosphere, the primary particle will often emit some Direct Cherenkov light before the much brighter Extended Air Shower is generated. Unlike the broader air shower component of telescope images, all of the Direct Cherenkov light will usually be concentrated in a single pixel. A new method for identifying these Direct Cherenkov pixels was developed, relying on supervised machine learning. A set of Cosmic Ray telescope images was simulated, both with and without the Extended Air Shower background, for use in classifier training. With reference to the background-free telescope images, the Direct Cherenkov pixel in each corresponding full-shower telescope image could be reliably identified. A Boosted Decision Tree to identify these Direct Cherenkov pixels was trained, using data from the full-shower training images. The Boosted Decision Tree performance was then tested on a second set of full-shower telescope images, and compared to existing methods of Direct Cherenkov pixel identification. After applying cuts to the test sample, 37.3% of all Direct Cherenkov pixels were accepted and correctly identified, representing a sevenfold increase in acceptance rates when compared to existing methods. The corresponding sample purity after cuts was 87.7%, a slight increase on existing methods. When only high multiplicity Boosted Decision Tree events were considered, 11.8% of all Direct Cherenkov pixels were identified, with a sample purity of 90.9%, while the existing method was not able to find a single comparable high-multiplicity event.

1 Introduction

Cosmic Rays will often emit Direct Cherenkov (DC) light in the upper atmosphere, before generating an Extended Air Shower (EAS) through interaction with the lower atmosphere. Analysis of the DC light from a shower can indicate primary particle energy, charge and position. It is consequently very useful for event reconstruction to measure the quantity and direction of DC light emission. There are numerous Telescope Arrays which currently image Cherenkov Light emission from Cosmic Rays in the atmosphere, such as the HESS, MAGIC and VERITAS experiments. In these telescope images, the DC light is usually concentrated in a single 'DC pixel'. Identifying this pixel is challenging, because the brighter EAS Cherenkov light background often overlaps with the DC pixel.

We define the variable $Q_{DC} = \frac{Intensity_{N.N.max}}{Intensity}$ as the ratio of the largest neighbouring pixel intensity to the intensity of a given pixel. At present, the DC pixel candidate can be identified by applying a number of cuts to pixels in an image [1], and from the subset of pixels passing the specified cuts, selecting the pixel with the smallest Q_{DC} as the 'DC candidate'. Due to the low pass rate for cuts, we obtain a small low-contamination dataset, while the majority of telescope images are left without a DC pixel candidate. An improved method would aim to increase the number of correctly identified DC pixels, while still enabling cuts which discriminate well between correctly and incorrectly identified DC pixels.

Classifiers provide an alternative method of identification, making use of supervised machine learning to find rules for categorising pixels. To train a classifier, we require a set of training pixels, as well as the correct class for each pixel. Once trained, a classifier can then be used to predict the class of a pixel. In this case, the classifier must learn to distinguish between non-DC and DC pixels.

The CORSIKA package [3] was used to generate Cosmic Ray events, and the Sim_telarray package [2] was used to generate corresponding HESS array telescope images. Simulation with EAS background was used to produce training pixel sets, while corresponding simulation without EAS background was used to find the true DC pixel in each training image. A Boosted Decision Tree (BDT) classifier was trained with the data to identify DC pixels. Once trained, the BDT was applied to a separate 'testing' set of simulated telescope images. As before, the true DC pixel was determined from a second EAS-free simulation. Thus, the accuracy of BDT identification for test telescope images was calculated.

2 Image Simulation

The full simulation of air showers was performed using the CORSIKA package, with a standard atmospheric profile derived from measurements conducted at the HESS site in Namibia. In total, 2000 training events and a further 2000 testing events were simulated. The simulated particles were Fe^{56}, within the Energy Range of $35-135$ TeV and a spectrum $\phi(E) \propto E^{-2.7}$. For each set of simulated event, 4 unique random number seeds were used to generate the shower. An altitude of 1800m was assumed, again corresponding to the HESS site. The simulated zenith angle ranged from $0° < \theta < 2°$, while the simulated azimuth angle ranges from $-2° < \phi < 2°$. The four smaller HESS-phase-1 telescopes were arranged in a cross along the x/y axis with the larger HESS-phase-2 'CT5' telescope placed at the center. The length of each cross arm was $85m$. The simulated target region of the cores was chosen to be a square centered on CT5, with each 300m-long side bisecting the x/y axis. Due to hardware differences between CT5 and the original HESS 1 telescopes, only images from HESS-phase-1 were considered.

To determine the true class of each pixel, a simulation was initially run with an energy cut of 10 PeV on all muons and electrons. Because this cut exceeded the primary particle energy, neither daughter muons and electrons, nor the photons they would have emitted, were simulated. Thus the hadronic Cherenkov Light from the primary particle and daughter fragments, but not the EAS light, was present in the camera image. A second identical 'EAS Simulation' was run including the same random seeds, but without the energy cut on muons and electrons. This gave a complete EAS image including identical DC light.

With the sim_telarray package, the expected HESS hardware response to each air shower was simulated. Among other things, the program accounts for atmospheric transmission and density, mirror positions, sizes and reflectivities, camera shadowing and triggering, quantum efficiency and pulse responses. For the full-shower image, the night sky background was also simulated by sim_telarray. Due to the comprehensive and detailed nature of these hardware simulations, the resultant images can be considered 'realistic'. However, sim_telarray introduces various sources of random noise to the simulation, leading to some divergence in the DC light between the EAS-free and full-shower images.

The various pixel entry variables were found from the sim_telarray output. The HESS telescope pixels have a high gain Channel 0 and a low gain Channel 1, with both voltages undergoing a Flash Analogue-to-Digital Conversion (FADC). The simulated value of the FADC Voltage for each channel was found. Using the pedestal and gain, the quantity $Intensity = (FADC - Pedestal) \times Gain$ was calculated for each channel. Due to possible saturation of the high gain FADC, only the low gain $Intensity$ was used. Sim_telarray also derives various Hillas whole-image parameters. These include the image width and length measured in degrees, from which the aspect ratio $A.R = \frac{width}{length}$ was calculated. The reconstructed shower direction and the shower center of gravity were also calculated, as positions in azimuth and zenith. Additionally the estimated energy and distance from each telescope to core, r_{core}, were found.

For every pixel, in addition to the $Intensity$, its location within the telescope image was determined using the standard HESS layout. The variables $\Delta_{C.o.G}$, $\Delta_{Direction}$ and Δ_{Line} were defined as the distance from the pixel to the shower center of gravity, shower direction, and the line joining those two points. Furthermore, the nearest neigbouring pixel IDs were calculated for every pixel position, enabling the $Intensity$ in each neighbouring pixel to be found. The largest neighbouring intensity was identified, and the ratio $Q_{DC} = \frac{Intensity_{N.N.max}}{Intensity}$ was derived. Similarly the largest neighbouring FADC was found, and the ratio $raw_Q = \frac{FADC_{N.N.max}}{FADC}$ was calculated. In addition, the Nearest Neighbour Mean Intensity $Mean_{N.N}$ was recorded. The variable $DC_{Signal} = Intensity - Mean_{N.N}$ was defined as an rough guess of the 'DC signal' component in the pixel. Lastly the Image Amplitude I_{tot}, defined as the total image intensity after the default tail cuts have been applied to the image.

3 DC Pixel Identification

As a basis for comparison, the original HESS cuts listed in Table 1 were replicated for the set of test data. For every image, the total image amplitude I_{tot} was used alongside the zenith angle θ to determine a dynamic cut, $Q_{DC} < 0.14 \times \log(\frac{I_{tot}}{161 \times \cos\theta})$. Among those pixels passing all cuts, the one with the smallest Q_{DC} was selected as the DC pixel candidate for the image. Because many images had no pixel that passed all cuts, the Q_{DC} method was frequently unable to identify a DC pixel. In the original analysis, an additional cut r_{core}>40m was applied. However, the uncertainty in determining the core position through Hillas Analysis is typically of the order of $\pm 30m$. Consequently, this particular cut was omitted.

The candidates were checked against the true DC pixels identified in the EAS-free images. From

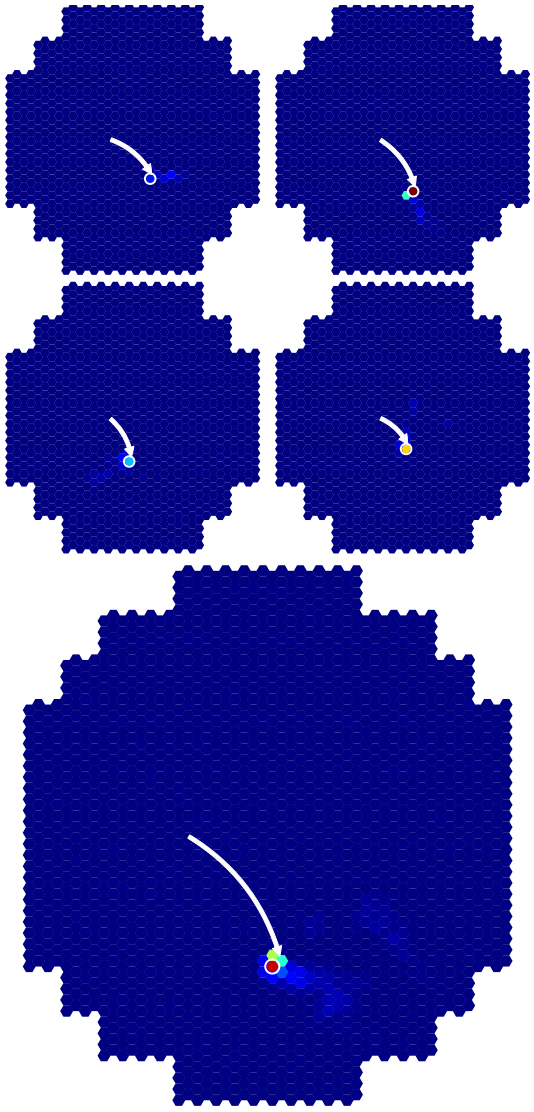

Figure 1: A typical camera image without the EAS shower. The DC light is visible in every telescope, indicated by the white arrow. The DC pixel is circled in white. The largest telescope image is from CT5, but was not used in analysis.

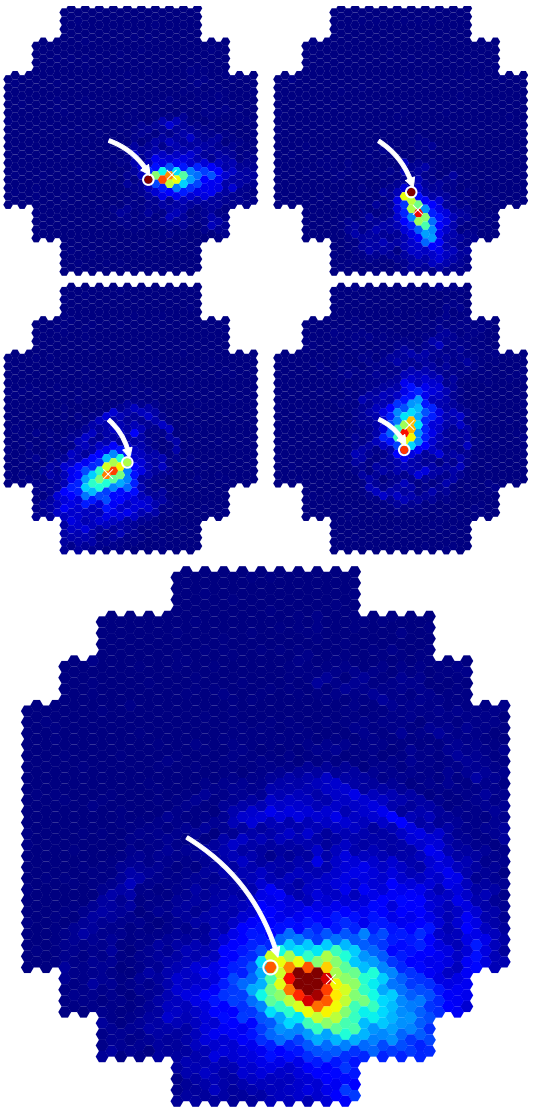

Figure 2: The same shower as in Figure 1 is shown here with the inclusion of the EAS shower. The DC light is pixels indicated with a white arrow and circle. The shower center of gravity is marked by a white cross.

Table 1: Cuts applied to image pixel sets, used by HESS collaboration [1]

Variable	Cut
$\Delta_{C.o.G}$	>0.17
$\Delta_{C.o.G}$	<0.91
$\Delta_{Direction}$	<0.45
Δ_{Line}	<0.23
Aspect Ratio	<0.75
Q_{DC}	$<0.14\times \log(\frac{I_{tot}}{161\times\cos\theta})$

the testing sample, 5.2% of all images were correctly identified and passed the cuts. Once misidentified events were considered, the post-cuts sample was 86.2% accurate, as shown in Figure 3. These values served as a benchmark for BDT performance.

The BDT was trained with the Scikit Learn Python package [4]. The training set of 2000 CORSIKA events was randomly split further, with 90% in a learning subset and 10% in a subset to check for overtraining. Within the learning subset, every HESS 1 image was used, provided it was triggered in both EAS-free and full-shower simulations. For each of the 4.7 million triggered image pixels, an entry was formed with the variables listed in Table 2. A class of 0 was assigned to every non-DC pixel, and a class of 1 was assigned to every DC pixel. Having created a dataset, the BDT was then trained with a maximum depth of 8, and with 100 trees generated. The data was provided in the form of individual pixel entries, rather than as discrete sets for images or events.

Table 2: Relative Feature Importance in HESS-1 BDT training

Variable	Relative Importance
DC_{Count}	0.34
$Mean_{N.N}$	0.26
$\Delta_{Direction}$	0.11
Image Amplitude	0.10
Q_{DC}	0.10
raw_Q	0.05
Δ_{Line}	0.02
$Intensity$	0.02

The relative importance of each 'feature' is automatically calculated by the Scikit Learn package, and is also recorded in Table 2. The variable DC_{Count} was consistently the most importance variable across many combinations of included variables and BDT training parameters. It was found that, under the conditions listed above, the BDT was 99.94 % accurate for the learning pixel subset, and 99.93 % accurate for the overtraining-check pixel subset. This indicates that the BDT was not significantly overtrained, which would otherwise be manifested by a large divergence in accuracy between learning and overtraining-check data.

Having trained the BDT successfully, it was then applied to the same test dataset as for the classic Q_{DC} identification. In each camera image, the event with the largest BDT score was deemed to be 'most signal-like', and thus selected as the DC pixel candidate. A cut was applied, requiring $P_{signal} > 0.5$ for the DC candidate to be accepted. A second cut requiring $DC_{Count} > 150$ removed many incorrectly identified events. Application of this combined cut greatly increases the successful identification rate. From the testing sample, 37.3% of all images were correctly identified and passed the cuts. The BDT was found to be 87.7% accurate in identifying DC pixels which passed the cuts. This represents a very significant improvement in pixel identification efficiency, as well as a minor increase in accuracy after cuts.

For event reconstruction, we often require events to have a high telescope multiplicity. If we only consider events in which all four DC pixels were identified, we can determine the high-multiplicity BDT performance. In this case, the fraction of correctly identified DC pixels falls to 11.8%, while the fraction of incorrectly identified pixels passing the cuts falls to 1.2%. The sample purity increases slightly to 90.9%. The relatively high fraction of passing events, in excess of the random expectation of $0.372^4 = 1.9\%$, suggests that DC pixel identification between different telescope images is strongly correlated.

However, out of 1245 triggered events, there were none in which the Q_{DC} method identified four DC pixels. Using poissonian statistics, we can place an upper limit of $< 0.24\%$ on the rate of accepted, high-multiplicity Q_{DC} events. Thus, as well as BDT accuracy slightly improving, the performance gap over the Q_{DC} method is vastly increased for high-multiplicity events. The results are summarised in Table 3.

Table 3: Comparison of Q_{DC} and BDT Performance

	Q_{DC}	BDT
Pixels Accepted and Correctly Identified (%)	5.2	37.2
Pixels Accepted and Incorrectly Identified (%)	0.8	5.2
Sample Purity (%)	86.2	87.7
High Multiplicity Pixels Accepted and Correctly Identified (%)	< 0.24	11.8
High Multiplicity Pixels Accepted and Incorrectly Identified (%)	< 0.24	1.2
Sample Purity (%)	/	90.9

4 Conclusion

The use of BDT identification has been shown to be superior to the traditional Q_{DC} method, by providing many more correctly identified DC pixels without a corresponding loss of accuracy. For all simulated HESS 1 images, the BDT yielded a sevenfold increase in the expected number of DC pixels. In the case of high-multiplicity events, the relative performance improves dramatically, representing at least a fifty-fold increase in the rate of correctly identified pixels passing all cuts. BDT performance improves with increasing training data size, so we can assume that the successful ID rate for pixels could continue to be increased further. Due to the good degree of separability of the BDT signal probability, a further increase in sample purity would certainly be possible at the expense of harsher P_{signal} cuts reducing the size of the datasets. In short, the BDT method produces not just superior performance but increased flexibility for balancing sample purity and size. Adoption of BDT identification for DC pixel could prove invaluable to future event reconstruction for large Cherenkov Telescopes, both currently operating, and those planned for the future.

References

[1] F. Aharonian et al. First ground based measurement of atmospheric Cherenkov light from cosmic rays. *Phys. Rev.*, D75:042004, 2007.

[2] Konrad Bernlohr. Simulation of Imaging Atmospheric Cherenkov Telescopes with CORSIKA and sim_telarray. *Astropart. Phys.*, 30:149–158, 2008.

[3] D. Heck, G. Schatz, T. Thouw, J. Knapp, and J. N. Capdevielle. CORSIKA: A Monte Carlo code to simulate extensive air showers. 1998.

[4] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

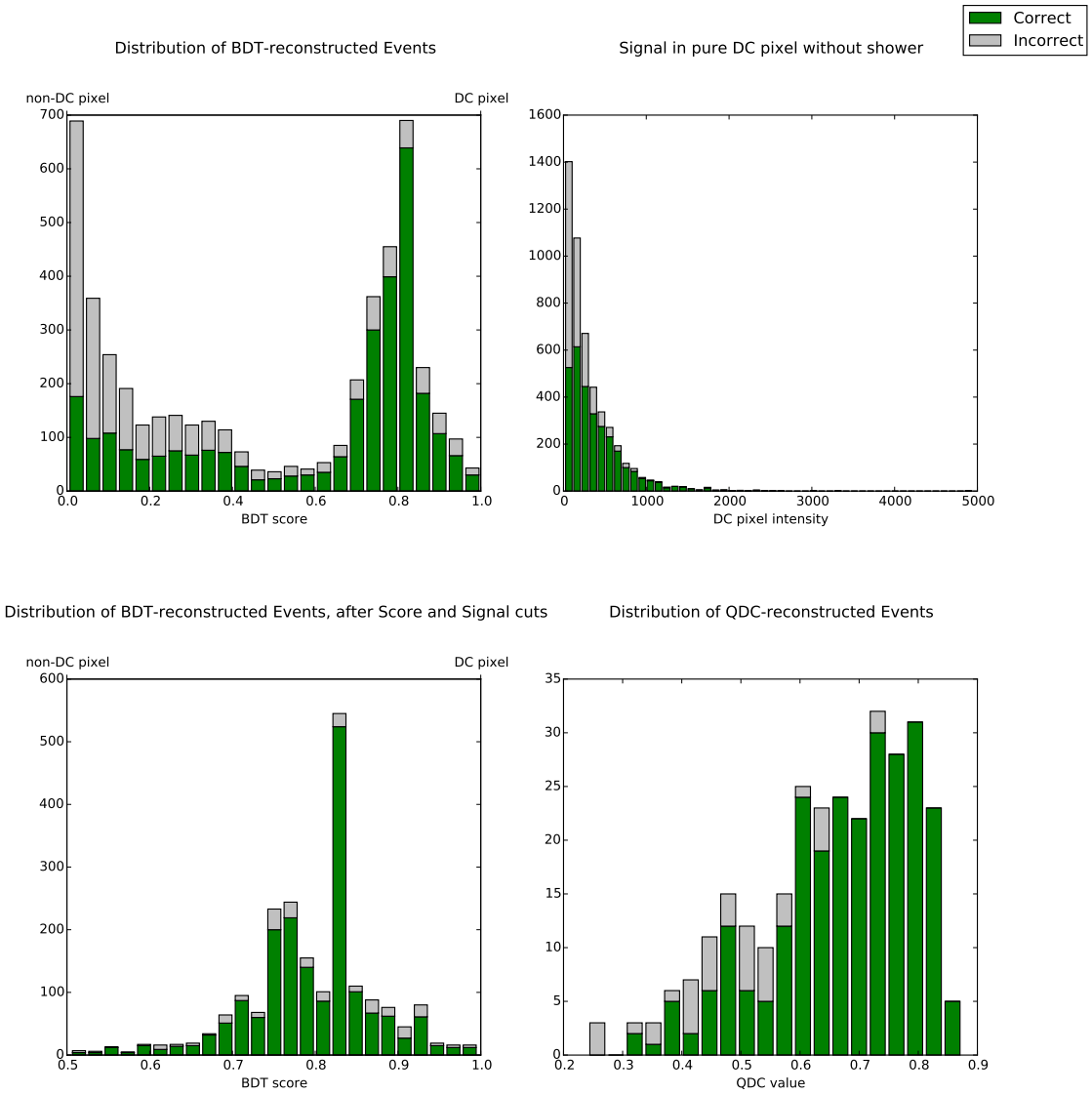

Figure 3: A study of 2000 simulated events is displayed above. For all graphs, the green represents correctly identified pixels, while the grey indicates incorrectly identified pixels. We desire many green pixels, few grey pixels, and a good degree of separability between the two groups. In the top left, the P_{signal} (BDT score) distribution is shown before any cuts. A low score indicates a non-DC pixel, and a high score indicates a DC pixel. Most incorrectly identified events are found at lower P_{signal} values. In the top right, the true intensity $True_{DC}$ in the EAS-free image is shown, with a broad exponential decay in count as DC intensity increases. As might be expected, events with a larger $True_{DC}$ are more likely to be correctly identified, as they have clearer signal. A cut is applied to the calculated value DC_{Count}, which is strongly correlated to $True_{DC}$. In the bottom left, we see the same BDT score distribution after both DC_{Count} and P_{signal} cuts are applied. The proportion of incorrectly identified events is visibly decreased. In the bottom right, a comparative distribution of Q_{DC} values for pixels that have passed all the Q_{DC}-related cuts is shown. The sample purity is similar to the BDT-reconstructed distribution, but the raw number of correctly identified events is significantly smaller.