0.1 Reading from raw data files

0.1.1 Low level classes

The basic classes used to read the raw data files generated by the trdbox server were provided by Jan Albert's 20?? Honours project report [ref]. The two python classes o32reader and adcarray work together to provide the basic interface. The constructor of o32reader takes in a path to a single data file which may or may not contain zero-suppression. The o32reader is iterable and as such can be used to loop through individual events in a datafile. Then using an instance of the adcarray class, one may use adcarray.analyse_event to convert a single event as provided by o32reader to a numpy array. The resulting numpy array may be accessed via the attribute adcarray.data. There are however a couple quirks which must be considered. First of all, the trdbox is designed to be able to output data from up to 16 pad rows however the detector present at UCT only has 12 pad rows. So it is recommended that the first step after analysing an event is to slice adcarray.data from 0 to 12 in the first axis and only dealing with this subarray. Second, the first event after configuring the chamber is always a special event known as a 'configuration event'. Instead of outputing ADC data, the chamber will simply return the configuration register values for each ADC unit. Therefore it is recommended that the first event of any data file is disregarded in case it happens to be the first event after a configuration. Third, when using zero-suppression, there is a 10-15% chance that a given event is incorrectly written to the datafile. This occurs if the time between two events is not long enough to allow the electronics to output their payload. When analysing such an event, adcarray will throw a datafmt_error which the user should use to skip over the current event. Finally, with the current setup, the two ROB boards which lay in the centre of the x-dimension are very vulnerable to voltage sag which renders their data useless. One may recognise this if the data contains hundreds of ADC values around 522. The way to deal with this is to simply apply a mask which sets all values corrosponding to the problematic ROBs to 0. Due to consistant bad behevaiour from these ROBs (the two in the detector and the one Lees :p, remove all this), the middle two ROBs were masked to 0 in all the results in section ????.

0.1.2 Initial processing

The aforementioned o32reader and adcarray classes provide an interface to read the raw data files produced by the trdbox; however this process is relatively slow and forces any program using the data to engage with a complicated reading procedure which may include undesireable events. As such it is recommended that the resulting adcarray.data numpy arrays are filtered and stored using numpy.save so that the they may be recalled much more quickly by other programs using numpy.load.

For the scope of this project, a script named interesting_event_extractor was created for this purpose. The main premise was to define a data_is_interesting function which would classify an event as 'interesting' or not. Then once any other pre-processing is complete, the resulting array would be saved to a directory for later use. The critera used for an event to be 'interesting' was simply for it to contain a single ADC value above a given threshold (the value used in this report was 300. Further explination given in section ????) and for all values above a certain threshold (taken to be 100) to lay in a single pad row. This ensured that only events which contained a high energy interaction that occured prodiminently in one single pad row would be accepted. The reason for the single pad row critereon was that the pad columns are much more tightly packed and therefore the detector has a much higher spatial resolution in the pad column direction. Many assumptions used in the analysis may break down when considering tracklets which cross over pad rows.

1 Results and data analysis

The various scripts used in this project may be broadly classified into 2 catagories, those which are used to validate the contents of a data file and those which assume valid data and perform higher level functions on this data. We will first take a look at the diagnosis tools which in general will inspect raw data files and help the user assess whether the chamber setup is conduce to valid data. Then we will look at some higher level programs which will use the output of interesting_event_extractor to make various inferences about the events which took place in the chamber.

1.1 Diagnosis tools

1.1.1 Background noise of the TRD

The first test of the chamber would be to sample the output of the detector with randomly triggered events. With this mode of operation it is unlikely that a given event has captured the path of a muon and thus such events can be used to measure the baseline value and influence of noise on the pads. This will give an indication that all the pads are functioning correctly and any pads with an abnormal baseline or level of noise can be isolated and investigated.

Here we will inspect the baseline values of a run containing ????? randomly triggered events with an anode voltage of 1500V and a drift voltage of -1400V. The mean value of a pad was calculated across all events and time bins and the standard deviation of each pad was calculated in the same way.

Figures ?? a) and b) reveal the baseline value and standard deviation of each pad. There is no clear relationship between the position of a pad and its baseline value in plot a), however plot b) demonstrates that there is a source of noise in certain regions of the TRD which are likely due to the readout electronics which sit on top of the chamber approximately above the affected zones.

Figure ?? and table ?? reveal that the observed baseline value accross all pads agrees with the specifications outlined in the TRD technical design report [ref] which hovers around 9.6 with a standard deviation of 5.3×10^{-2}.

1.1.2 ADC spectrum

The next test of the chamber involves taking a histogram of all ADC values across many events so that any high energy anomalies can be found and investigated. Once triggering has been implemented, this process may be used as an initial test to see that the incidence of high-energy interactions increases. This may indicate that the triggering process has been correctly implemented.

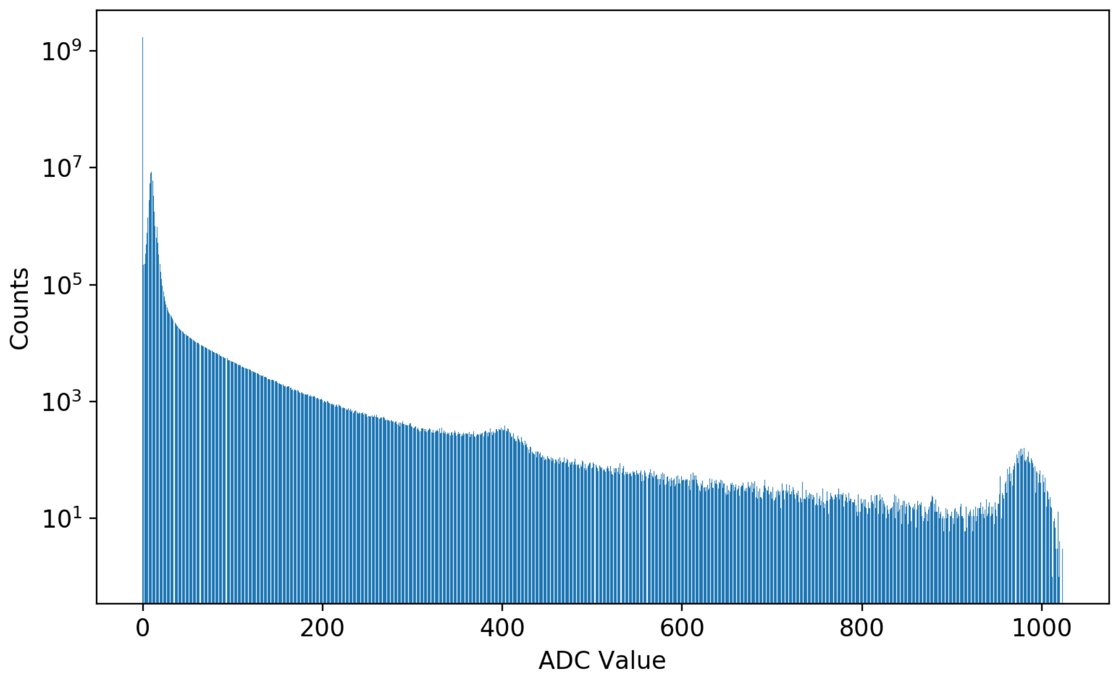


Figure 1: Histogram of approximately 285×10^6 ADC values accross 5500 events for a -1500V anode and 1400V drift voltage.

The above plot was constructed using 5500 triggered and zero-suppressed events with an anode voltage of -1500V and drift voltage of 1400V. The bump around 400 appears only once the triggering has been correctly implemented and corrosponds to the expected ADC value of a pad which measured the trace of a muon which passed close to the centre of the pad. This is also how the threshold value of 300 was chosen for the interesting_event_extractor as described in section ????.

@Jezza

1.1.3 Time bin sum

Once triggering yields reasonable results as per the previous tests, one may want to begin identifying which events and which data files may contain actual muon tracks. One useful test for this is to take a histogram of the sum of the ADC values across the time bins (also reffered to as a tb-sum) for one or more events. The utility of this is that the ADC spectrum is vulnerable to localized high energy spots which one would not anticipate if an event is to capture the entire path of a muon. On the other hand, a high tb-sum value indicates that a pad saw activity across multiple time values. This likely corrosponds to a muon track in the region of the given pad.

INSERT SOMETHING FROM WIAN HERE.

1.2 Higher level investigations

Once the triggering was correctly configured and the captured data passed all initial tests, all relevant raw data files were passed through the interesting_event_extractor and the following features were investigated.

1.2.1 Linear fits to event data

Having captured the track of a muon, one very useful property one might extract from such an event is the ray which the muon took through the detector. This report took a simplified approach to determining the approximate track of muon by taking a weighted linear fit of the x and y components of the track as a function of distance travelled through the detector and only considered ADC values with a value greater than 100. The ADC value (with baseline subtracted) of each point was taken to be its weighting in the fit and the z component of a point was assumed to be a linear function of the time bin to which it belongs. This is a reasonable assumption to make since the drift velocity of the electrons in the gas is approximately constant in the drift region and therefore the z-position of a ADC value is approximately linear in the time bin value. However taking the weight of a point in the fit to be its ADC value will tend to be disproportionately biased towards points which were slightly closer to a muon's true position than other points. Nevertheless this assumption still worked well as a first approximation.

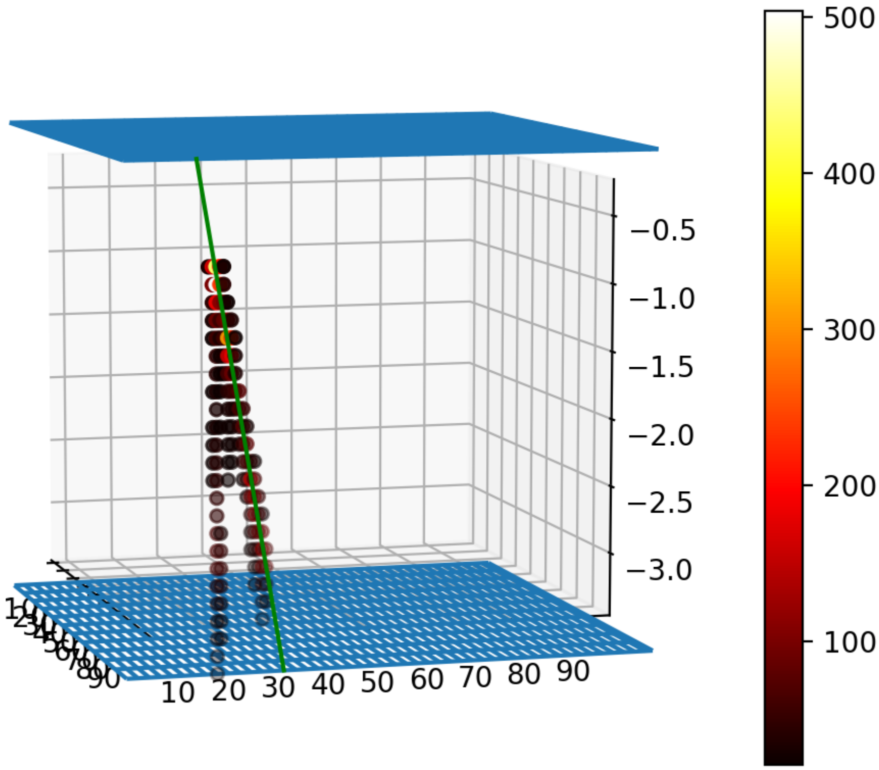


Figure 2: Example of the track of a single muon passing through the detector along with the corresponding linear fit. The anode voltage used was −1500V and the drift velocity was 1400V. Only points with ADC values greater than 30 are shown.

Parameter	Value
x-intercept	1.10
x-inclination	0.00
y-intercept	19.01
y-inclination	0.78
Approximate inclination angle (see ?.?.?)	77.7°

Table 1: Linear fit parameters for Figure ??

Figure ?? demonstrates the output of such a fit and Table ?? provides the fit parameters through index space of the pads. As expected, the ray is aligned in the y-direction (it lays in one pad row) and it is evident that the influence of the ray reaches accross multiple pad columns in a single time-slice. The long faint tails below the uppermost points demonstrate how a tail of noise can follow a legitimate measurement from a padwhich is likely due to internal capacitance and other electrical affects. This affect was not accounted for in this report.

1.2.2 Pulse height spectrum

The algorithm for determining the linear fit parameters can be used to produce what is known as the pulse height spectrum for the detector and its configurations. The pulse height spectrum is designed to give insight into the rate at which electrons approach the pads as a function of drift time. The first step in producing a pulse height spectrum is to take an event and produce the linear fit for the track. Then by taking the average ADC value in a neighbourhood around the track in the y-direction for each time slice (keep in mind that we are selecting for y-aligned tracks), we obtain an approximation of the intensity of the track at each time slice. Then by averaging the intensity at each time slice accross many events we obtain the pulse height spectrum of the detector and its configurations.

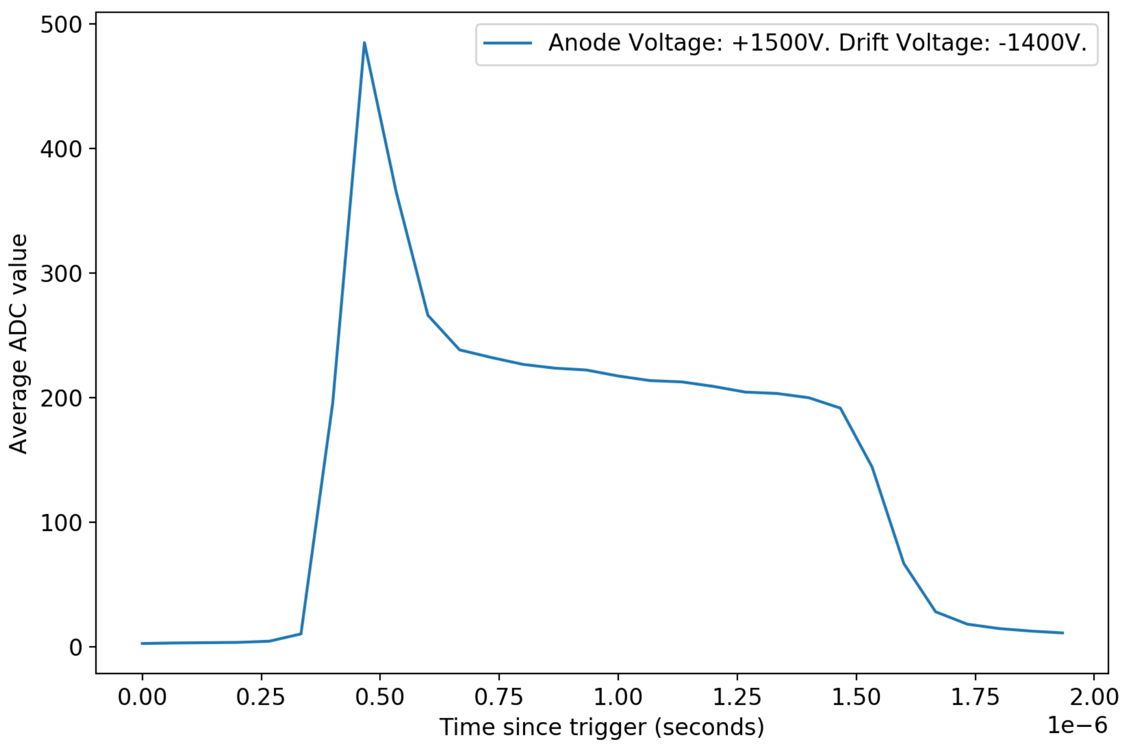
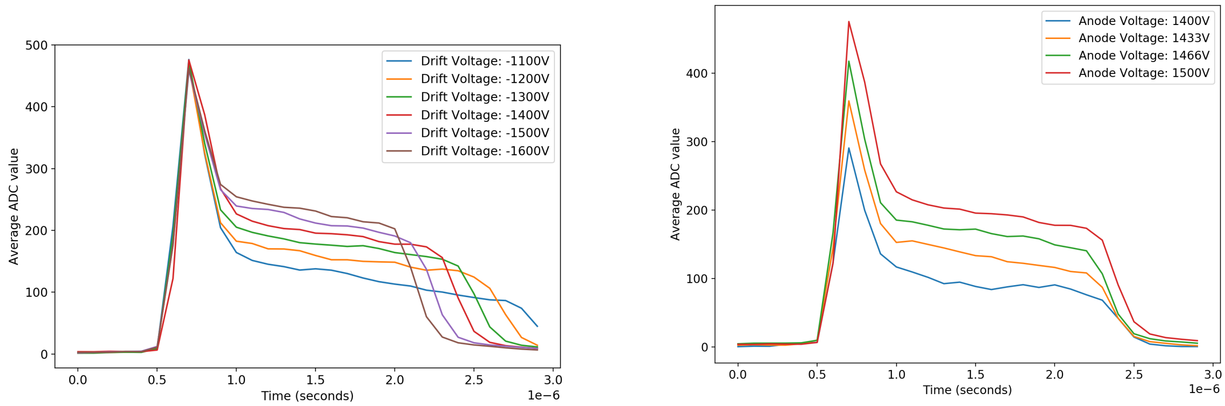


Figure 3: The pulse height spectrum for 20000 muon events with anode voltage −1500V and drift voltage 1400V.

Figure ?? shows a pulse height spectrum for the detector. There are various components to this plot which



(a) (b)

Figure 4: The pulse height spectrum for the TRD for various anode and drift configurations. Sub-figure a) has a constant anode voltage of 1500V and Sub-figure b) has a constant drift voltage of -1400V.

contain interesting information. For a generic single track, the leading peak is caused by the collection of electrons which were produced in the anode zone and are subsequently absorbed all at once, resulting in a spike. The proceeding plateau corrosponds to the consistent trickle of electrons from the drift zone into the anode zone. Ideally one would hope to see a perfectly horizontal plateau which would imply homogenous electron production as a function of distance, constant drift velocity and no loss of electrons due to re-absorbtion as the drift towards the pads. However with the current detector setup there appears to be a loss of intensity with increasing distance, which is likely due to impurities in the gas which absorb the ionization electrons as they drift. Notice however that we may extract the drift velocity of the electrons from the pulse height spectrum since the width of the plateau corrosponds to the time taken for the last electrons to drift from the bottom of the detector to the pads at the top. Since we know how far up the pads are situated, we may calculate the speed at which the electrons drift. Although this calculation is not performed in this report, it is key to recognise this connection.

One further relationship one may want to verify is the impact of the drift and anode voltages on the pulse height spectrum. Figure ?? demonstrates how increasing the drift voltage has the effect of increasing drift speed, since the plateau shortens, but also a small increase in height which means keeps the integral under the curve constant. This is to be expected since the drift voltage does not affect the gain of the detector and so the total number of electrons which reach the pads is constant (which is correlated with the ADC value) . Varying the anode voltage on the other hand does not change the width of the plateau but does affect the amplification of the electrons and thus increases the net area underneath the graph as shown in Figure ??.

1.2.3 Angular Distribution

Using the pulse height spectra obtained in section ?.?.?, one can infer how the time bin of an ADC value relates to the z-component of where the electrons that induce its potential were originally deposited. For a drift velocity of 1400V we may read off Figure ?? that it takes approximately 11 time bins for the electrons to move from the bottom of the detector to the top. Then using the z-dimension of the detector as described in the technical design report [ref], we may infer that the i'th time bin corrosponds to approximately a depth of $\frac{i-5}{11} * 0.036$m. Finally, using the pad spacings from the technical design report, we may calculate the inclination angle of a given track. This is how the inclination angle of the sample track in section ?.?.? was calculated. This is of course a rough calculation which ought to be made with more care and uncertainty analysis in the future.

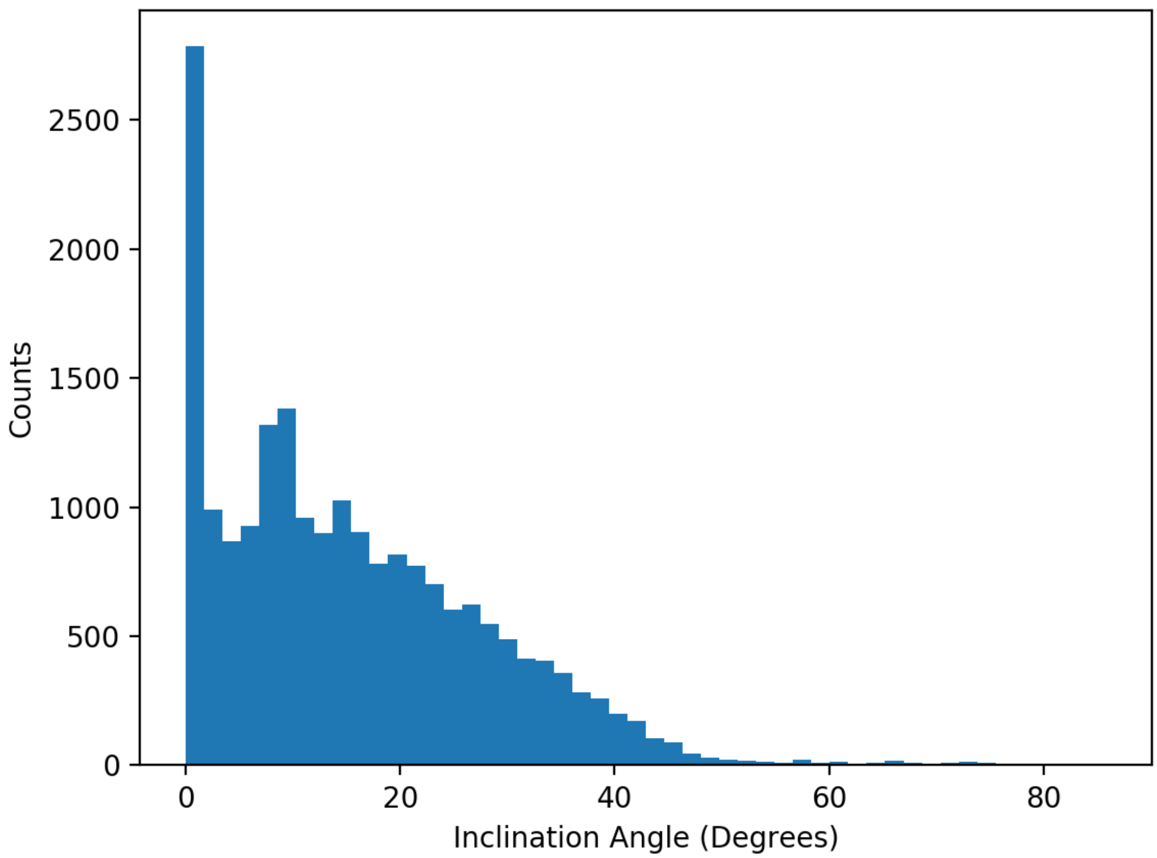


Figure 5: Distribution of estimated inclination angle for 20000 events with an anode velocity of 1500V and drift velocity −1400V.

Figure ?? shows the distribution of inclination angles for 20000 events. The bounds of the distribution are consistant with what was expected since the cross section of the trigger detector, TRD and their co-incidence rapidly decreases as the angle passes 45°. The overall shape is qualatively similar to the expected angular incidence of muons as explored in Section ?.?.?.

1.2.4 Pad column position resolution

The final property of the detector explored in this report was the position resolution in the y-direction of the detector within a single time slice. As a first order approximation, the position of a muon in a given time slice was approximated as the centre of 'mass' of a neighboorhood around where the linear fit crosses the time-slice which included approximately 4 pads to either side of the fit point. The mass of a point was simply taken to be its ADC value once the baseline has been subtracted. The value of the centre of mass y-coordinate was then compared to the value of the linear fit y-coordinate which was taken to be a far more accurate measure of the true position of the muon in that time slice. The difference between these two values was then histogrammed accross 20000 events with an anode velocity of 1500V and drift velocity -1400V and the result is shown in Figure ??

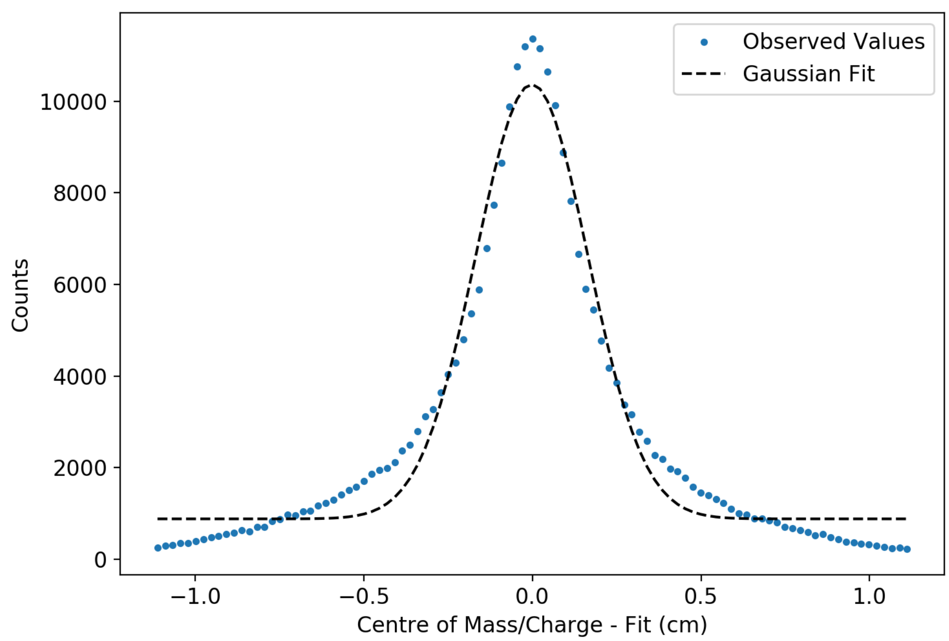


Figure 6: Histogram of the difference between the centre of mass and linear fit positions in each time slice accross 30000 events.

Parameter	Value
Scale	5.2×10^3
Mean	-1.4×10^{-3}
Standard Deviation	1.6×10^{-1}
Vertical Shift	8.8×10^2

Table 2: Best fit parameters for the Gaussian in Figure ??.

Table ?? gives the best fit parameters for the Gaussian and tells us that the position resolution of the detector is approximately 0.16cm (0.22 in pad width units) with a systematic bias which cannot be discerned from 0 at the given resolution.