Clustering update

Patrick for team CCCD
Bug fix

- Old algorithm iterated over a vector of unclustered above threshold pixels
- If a pixel was found to be close to an already clustered pixel it got erased from the unclustered vector
- This erase operation messed up the iteration over the vector, so clustering wasn’t working properly
- Whilst fixing also worked out how to speed up algorithm
Algorithm improvement

• Previous implementation took a vector of all pixels above threshold and checked to see if any of them were close to each pixel already in the cluster
• Changed to using a 2D array (just a TH2) of as yet unclustered pixels which allows you to only check the charge of the entries near the already clustered pixels
• 10s of minutes per event down to 0.5s
• Checked on a couple of events pixel by pixel and it’s behaving as expected
• 1) Make a vector of all the above seedthreshold pixel x,y positions and keep original TH2
• 1a) zero out all pixels in TH2 that are below skirtthreshold
• 2) Sort the vector of seed pixels by charge
• 3) Take the highest charge pixel, add it to a vector of x,y positions of things in the cluster
• 4) Loop over all pixels in the cluster vector
• 4a) Inside loop ask TH2 if pixels within nearneighdistance are above skirtthreshold, if they are add them to cluster vector and zero them in the histogram
• 4b) go back to 4 until no more pixels above threshold near pixels already in cluster
• 5) Get rid of any pixels in the seed pixel vector that are in the cluster you just made, if yes remove them from the seed pixel vector
• 6) Go to step 3 if there are seed pixels left, if not end
Summary

• Clustering algorithm now runs in a reasonable time period and seems to be behaving reasonably

• Ready for Will to process over all the good runs and some dummy images with known behaviour when he’s back and get information on number of clusters and size of clusters

• Aspect ratio cut temporarily removed for debugging, will add back in soon
Backup