GPU implementation of HGCal 2D Clustering (Day 2)

Z. Chen, T. Di Pilato, B. Kilian, 15-17 April, 2019

Day 2

CPU `calculateLocalDensity` (Tony & Ziheng):

1. fully synchronized with current kdTree version

GPU `calculateLocalDensity` (Ben):

2. Resolved data structure issues from previous hackathon (never pass vectors to a kernel)
3. Can now run our GPU kernel and access meaningful data

Plan for Day 3

1. complete the implementation of `calculateLocalDensity` on GPU and verify GPU results with CPU code

2. eventually start to implement the new `calculateDistanceToHigher` function on CPU

Starting Point

1. Latest CPU kdTree-based version of HGCal 2D clustering algorithm in CMSSW_10_6_0_pre2
2. Input data structure -- 2D histogram
3. Code of this effort from previous Hackathon

Goal

1. CPU histogram-based clustering
2. GPU histogram-based clustering
3. Get same result as CPU kdTree-based clustering

Day 1

1. Established starting point: setup and run.
2. Checked the result from existing code from previous Hackathons. Discrepancy in density and clusters between CPU kdTree-based
3. Started to develop our functions
    a. `calculateLocalDensity`

Plan for Day 2

1. Complete `calculateLocalDensity` for CPU and GPU histogram-based functions. Fully synchronized with CPU kdTree-based one.
2. Work on `calculateDistanceToHigher` CPU and GPU histogram-based functions. And compare the results with CPU kdTree-based one