Cluster study in a single ALPIDE chip

Jan Schöngarth
Allpix$^2$ set up

- A single ALPIDE Chip in the origin
- particle source at $z=-500\mu m$
- 10,000 Events, one particle per event
- beam direction $(0,0,1)$
- particles used: electron, positron, photon
- TTree output from CaloOutputWriter module
Cluster size comparison, $e^-$ and $e^+$

- one simulation with 10,000 Events for each particle and energy
- electrons and positrons produce a similar cluster size distribution
Cluster size comparison, $e^-$ and $e^+$

- bigger clusters at around 100keV
- to do: create a mean cluster size plot
Cluster size comparison, photon

• only at low energies photons produce clusters
Cluster size comparison, photon

- only at low energies photons produce clusters
- peak around 5keV
Cluster size comparison

- at 1keV there are no clusters from electrons
- at low energies most clusters are from photons
Cluster size comparison

- at higher energies electrons and positrons create similar distributions
Particle interaction with matter

- Ionization peak possibly at 100keV in Silicon?
- Photoelectric Effect peak possibly at 5keV in Silicon?
Algorithm idea

- Pixel coordinates as a Matrix

\[ C_1 = \begin{pmatrix} 511 & 512 & 512 \\ 256 & 256 & 257 \end{pmatrix} \]
Algorithm idea

• Pixel coordinates as a Matrix

\[ C_1 = \begin{pmatrix} 511 & 512 & 512 \\ 256 & 256 & 257 \end{pmatrix} \]

• substract the minimum x, maximum y from each x, y

• \( \min_x = 511, \ \max_y = 257 \)

\[ C_2 = \begin{pmatrix} 0 & 1 & 1 \\ -1 & -1 & 0 \end{pmatrix} \]
Algorithm idea

• $C_2 = \begin{pmatrix} 0 & 1 & 1 \\ -1 & -1 & 0 \end{pmatrix}$

• take absolute values

$\Rightarrow C_2' = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$
Algorithm idea

• $C'_2 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$

• use the columns of $C_2$ as indices for Shape Matrix

• $S = \begin{pmatrix} S_{00} & \cdots & S_{0n} \\ \vdots & \ddots & \vdots \\ S_{n0} & \cdots & S_{nn} \end{pmatrix} = 0$

• $n = n_{Pixel}$
Algorithm idea

\[ C_2' = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \]

\[ S = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \]
Algorithm idea

• $C_2' = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}$

$S = \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{pmatrix}$
Algorithm idea

• $C_2' = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$

➢ $S = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$
Algorithm idea

\[ C_2' = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \]

\[ S = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \]
Shape identification

• use `std::vector<vector<int>>` of size `clustersize × clustersize`
• compare shape vector to a list of vectors
• return integer which represents bin number of the shape
Problems

• Events with disconnected pixel hit locations
• Events with 5 or more pixel hits
Shape distribution, electron 1MeV
Shape distribution, electron 1MeV
Shape distribution, electron 1MeV
Shape distribution, electron 1MeV
Shape distribution, electron 1MeV
Shape distribution, electron 1MeV
Shape distribution, Electron 1MeV
Shape distribution, Electron 1MeV

3 pixel

vertical/horizontal line shapes
Shape distribution, Electron 1MeV

3 pixel

Events with disconnected pixel hits
Shape distribution, Electron 1MeV