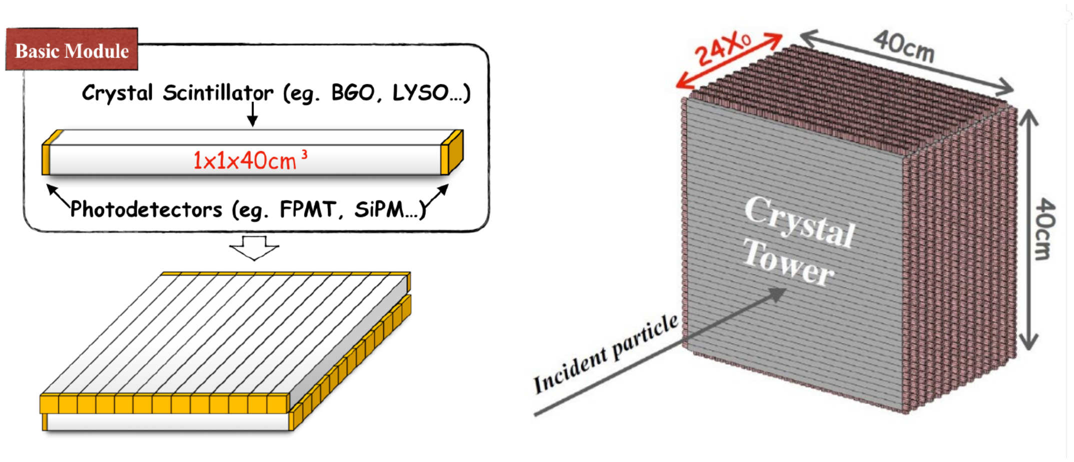
Photon Recognition Algorithm

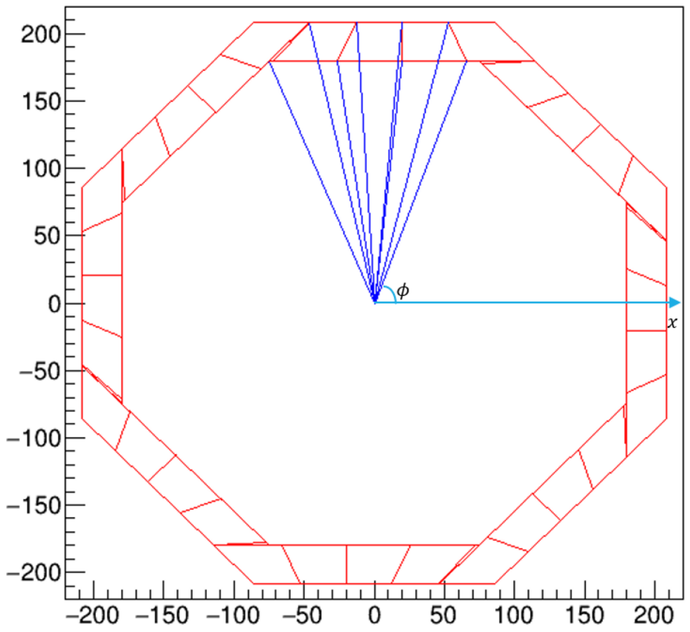
Zhang Yang

2023, Jan, 4th

Long Crystal Bar ECAL

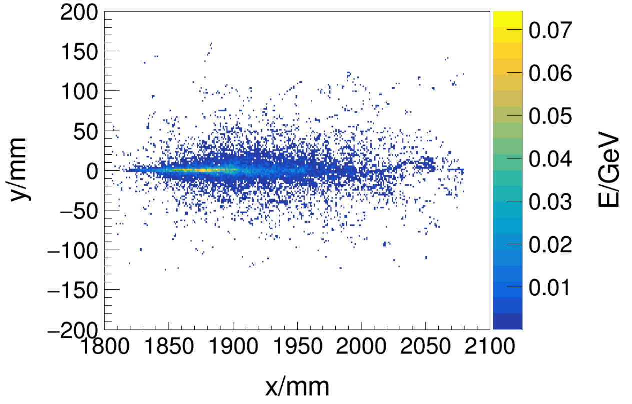
- Crystal bar
  - BGO: $X_0 = 1.12\ \mathrm{cm},\ \ R_M = 2.23\ \mathrm{cm},\ \lambda_I = 22.7\ \mathrm{cm}$
  - Size in simulation: $1 \times 1 \times 40{\sim}60\ \mathrm{cm}^3$
- Tower
  - $\sim 40 \times 60 \times 24X_0\ \mathrm{cm}^3$
- Detector:
  - $R = 1.86\ \mathrm{m}, L = 6.6\ \mathrm{m}, H = 28\ \mathrm{cm}$
  - 8 same trapezoidal staves
  - Avoid gaps point to IP

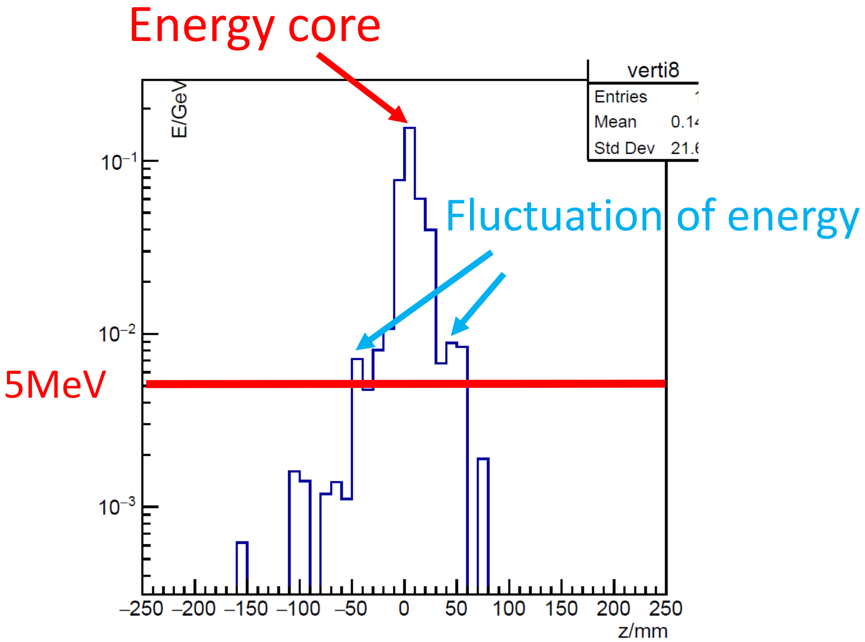




Recognition of Energy "Core"

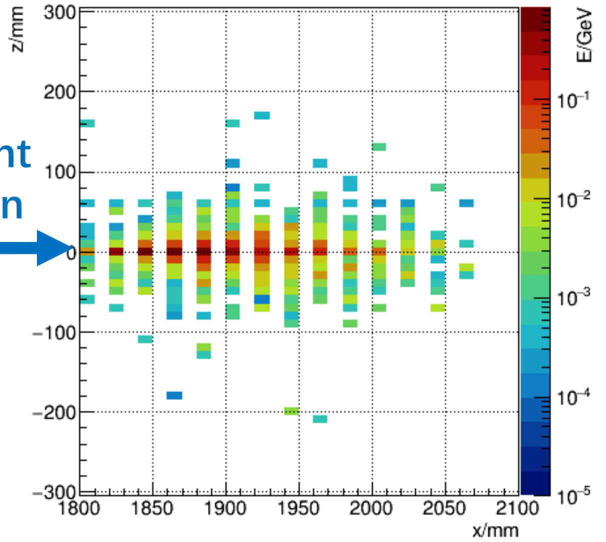
- Cluster recognition ➔ Energy "Core" recognition
  - Reduce the negative effects due to wider longitudinal and lateral developments of clusters.
- In each layer: local maxima



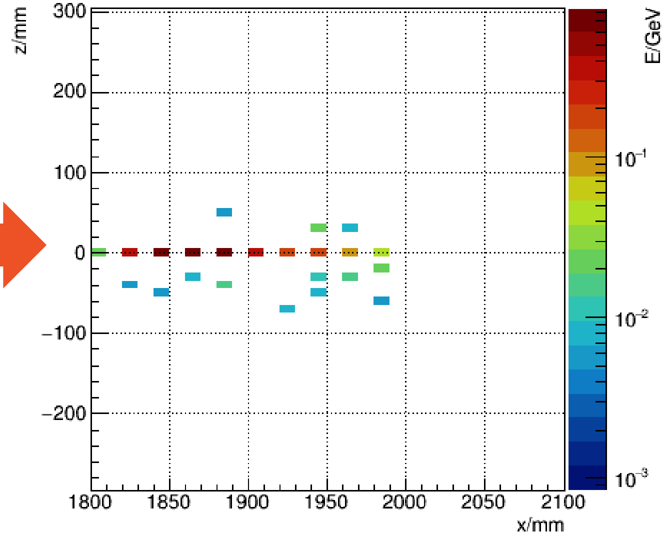


Energy deposits in bars of one layer

incident photon



Energy deposits in crystals



Local Maximum Distribution

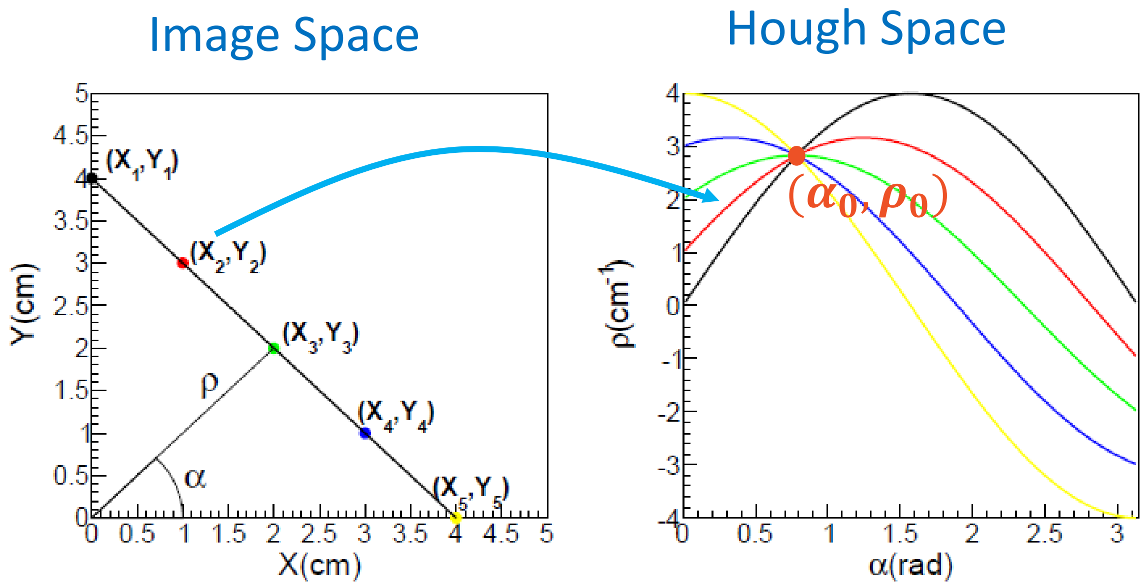
Principle of Hough Transformation

- A feature extraction method for detecting simple shapes (e.g. lines) in an image.

- For straight lines:

$$\rho = x\cos\alpha + y\sin\alpha$$

  - Each point (x, y) in image space is transformed to a curve in Hough space.
  - If several points (x_i, y_i) are collinear, their curves intersect at a point (α_0, ρ_0) in Hough space.
  - α_0 and ρ_0 are parameters of the straight line that pass through these points (x_i, y_i)



(x_i, y_i) ➡ $\rho = x_i\cos\alpha + y_i\sin\alpha$

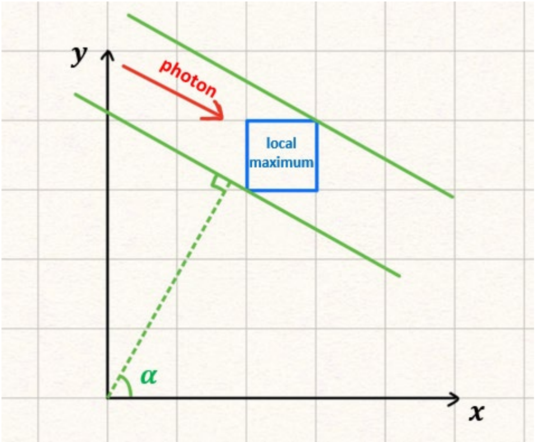
Series of Points — Series of Curves

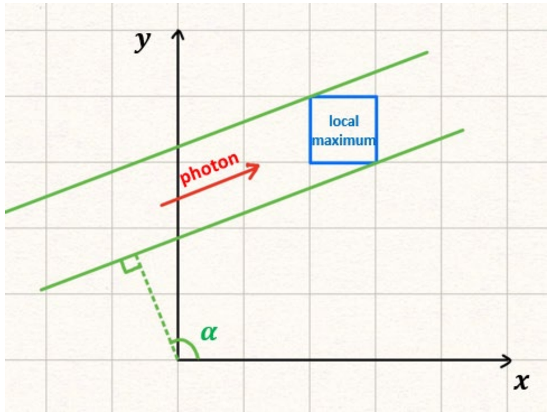
$\rho_0 = x\cos\alpha_0 + y\sin\alpha_0$ ⬅ (α_0, ρ_0)

Line — Point

Hough Transformation in ECAL

- Crystal in image space ➔ **band** in Hough space.
- Cluster recognition in horizontal and vertical projection spaces respectively.
- Each peak (overlap region of band) in Hough space is chosen as a cluster candidate.





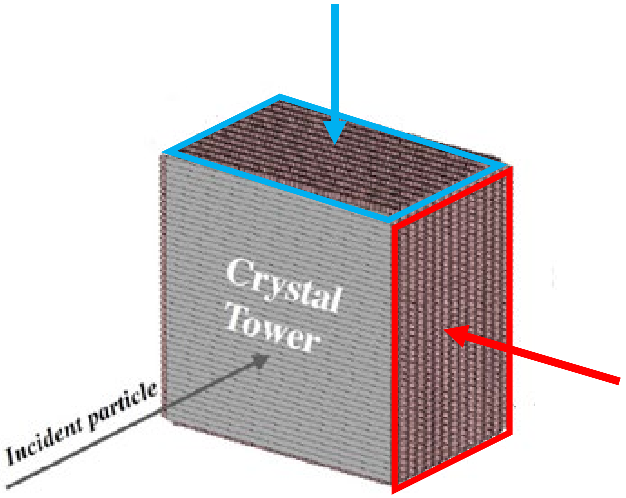
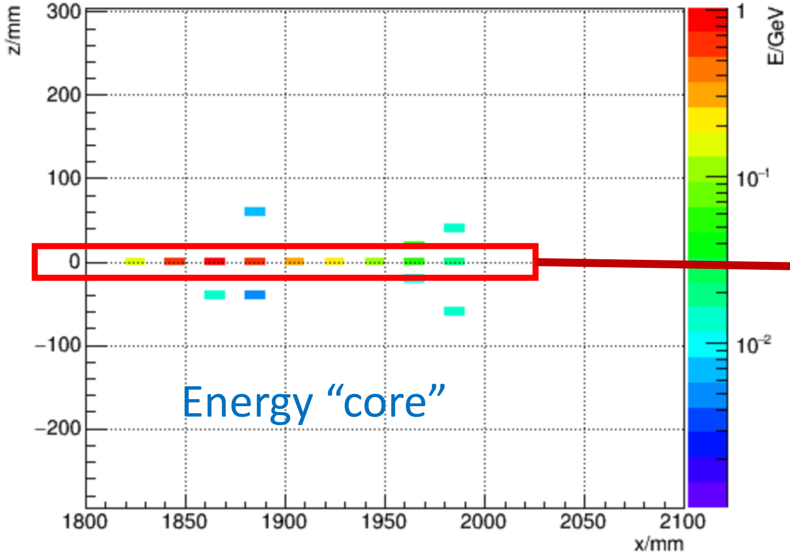
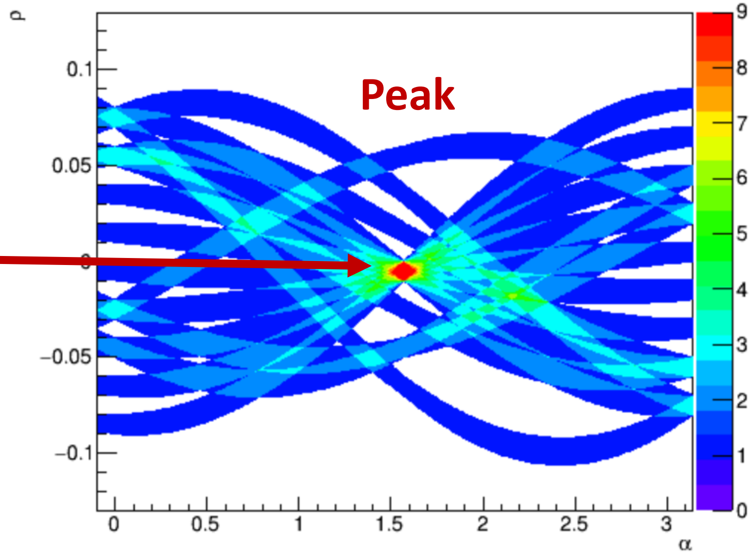


Image Space

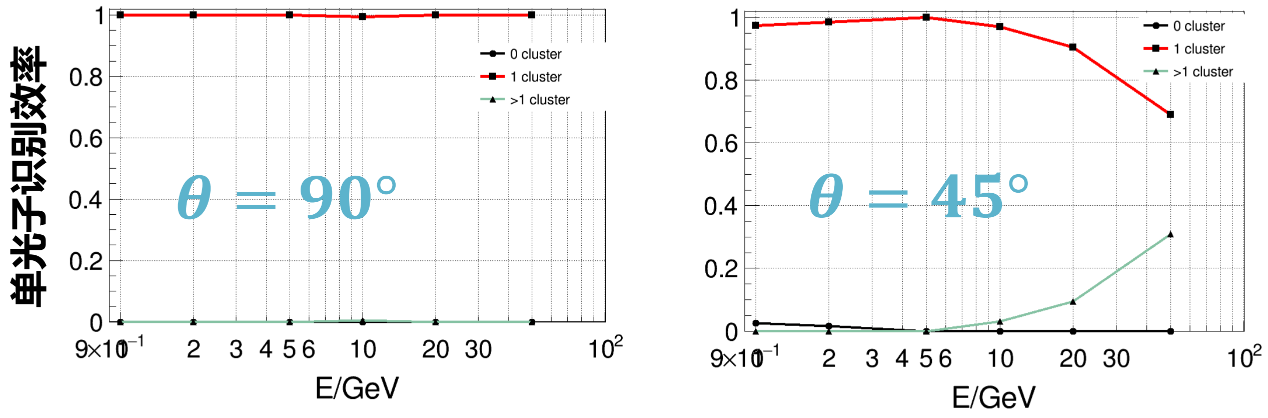


Hough Space



Performance check

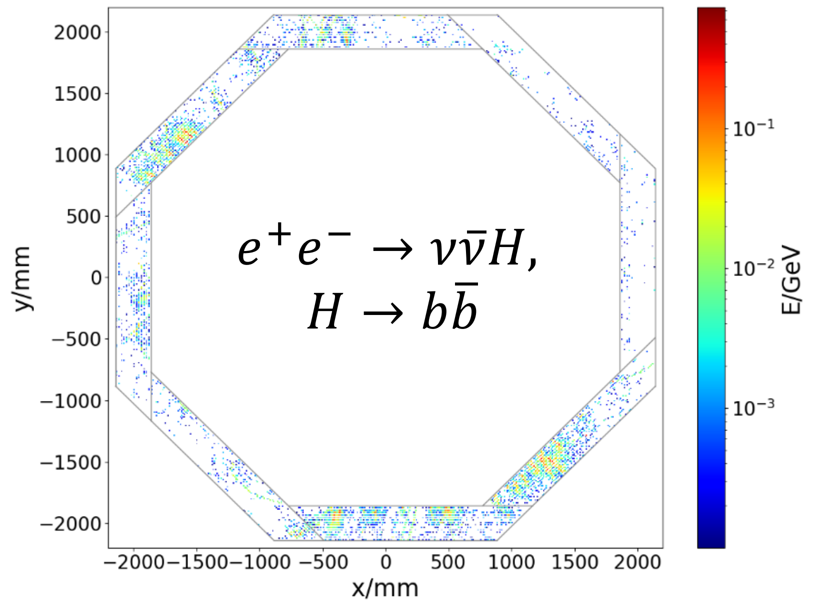
- Single photon recognition efficiency in a **local coordinate** (one tower)
  - Photon energy: $E = 1, 2, 5, 10, 20, 50\ \mathrm{GeV}$

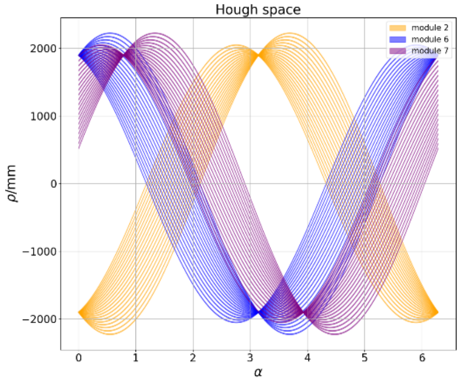


- Low energy or small $|\cos\theta|$: One & only one cluster
- High energy and large $|\cos\theta|$: >1 clusters
  - Fluctuations of energy deposits increase fake cluster

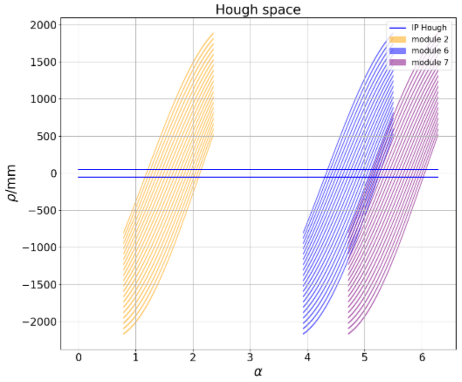
Generalization of the algorithm

- Motivation
  - Identify photons in jets → A global recognition algorithm is required
- Challenge
  - Much larger Hough space → Much larger memory & more computation required
- Solution
  - Limitation on Hough space
  - Using sparse matrix for Hough space





Original Hough space



After limitation

0	0	0	1	0
0	1	0	1	0
0	0	2	0	0
0	0	3	0	0
0	1	0	0	0
0	1	0	0	0

Original 2D matrix

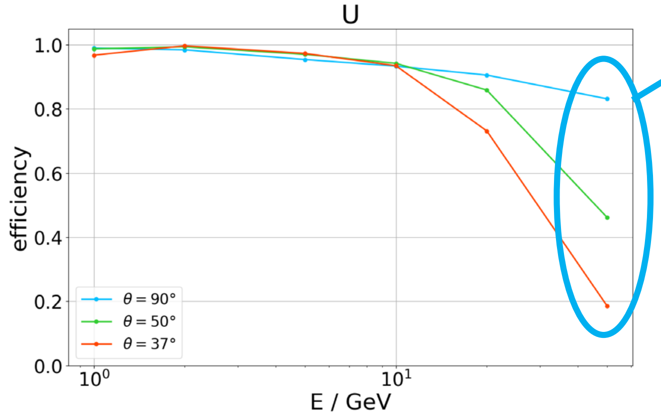
Row	Column	Value
1	4	1
2	2	1
2	4	1
3	3	2
4	3	3
5	2	1
6	2	1

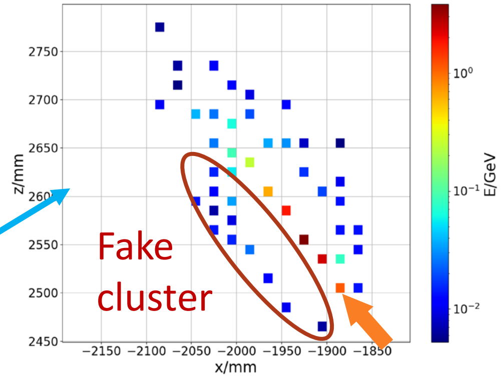
Sparse matrix

Optimization of High Energy Photon Recognition

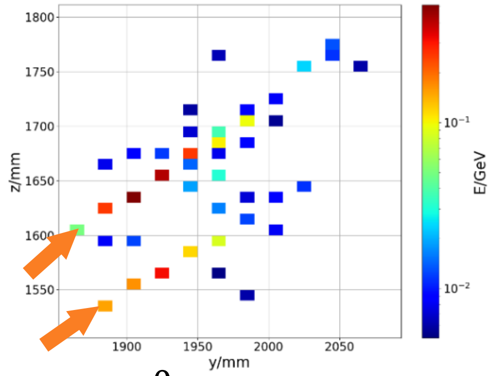
- Motivation
  - Fluctuations may be recognized as fake clusters
  - These fake clusters should be removed

- Solution
  - Cut on distances and energy ratio

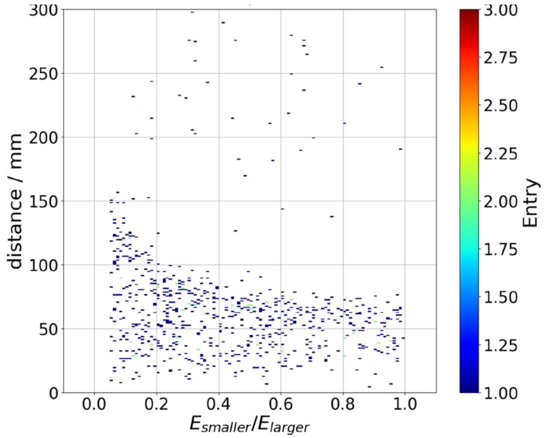




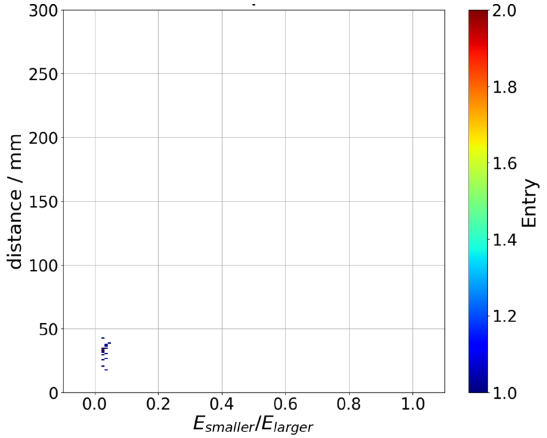
High energy photons



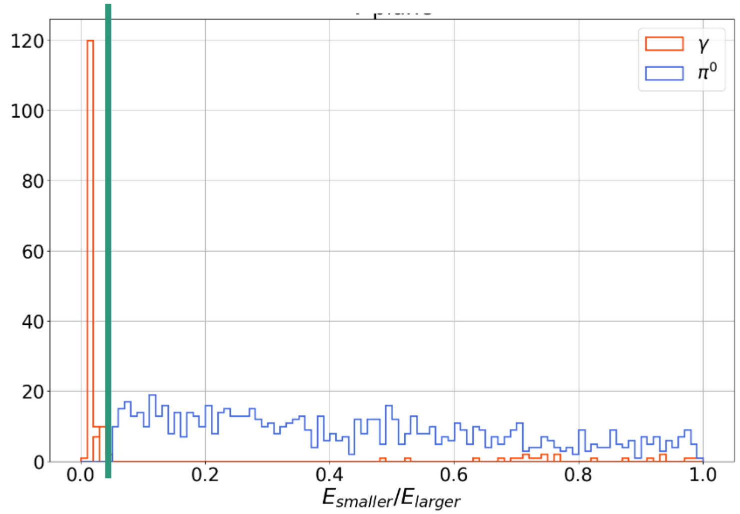
$\pi^0 \to \gamma\gamma$



$\pi^0 \to \gamma\gamma$



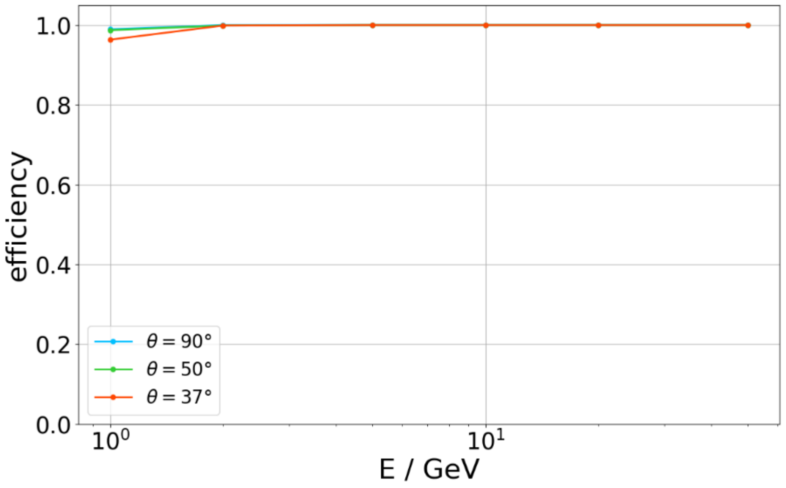
High energy photons



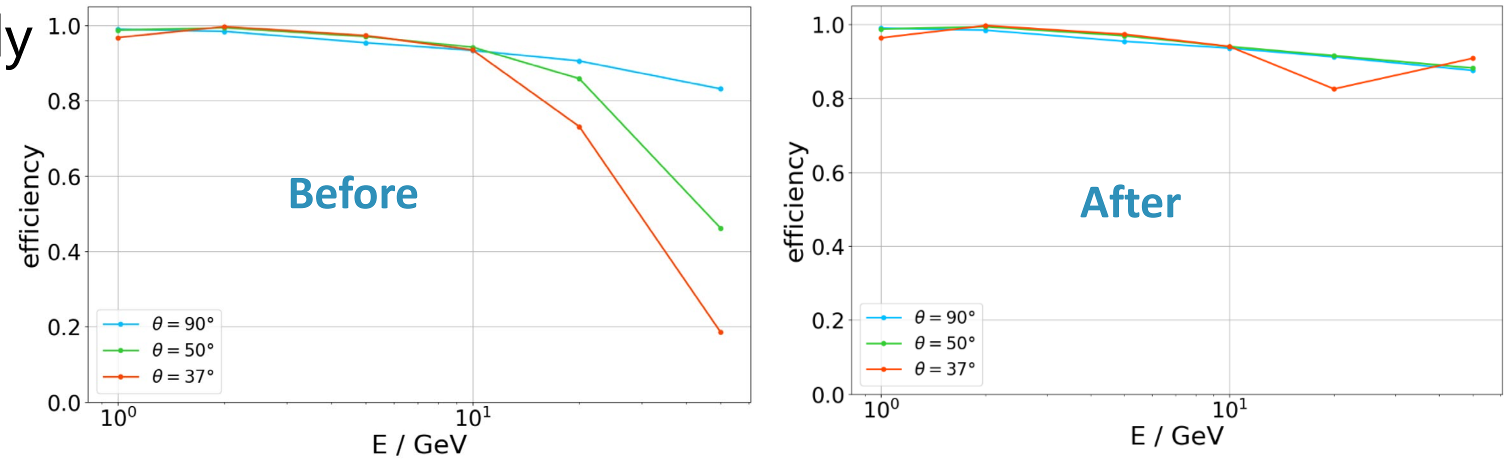
Performance check

- Single photon events
  - $E = 1, 2, 5, 10, 20, 50\ \mathrm{GeV}$
  - $\theta = 90°, 50°, 37°$
  - $\phi = 0°\sim360°$
- Performance
  - Photon recognition efficiency ~100%
  - Fake cluster fraction significantly reduced

Recognition efficiency

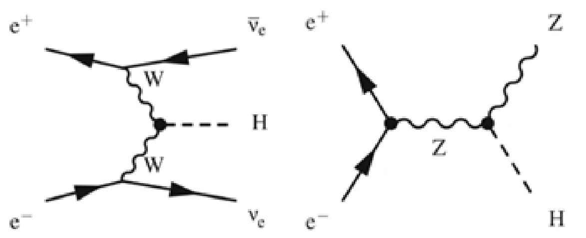


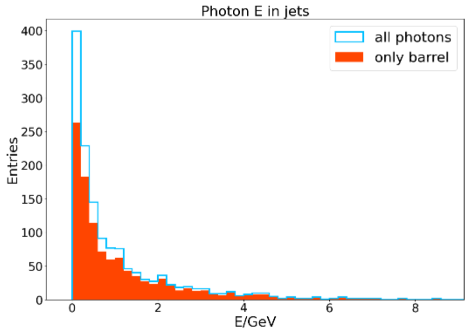
Recognition efficiency without fake cluster

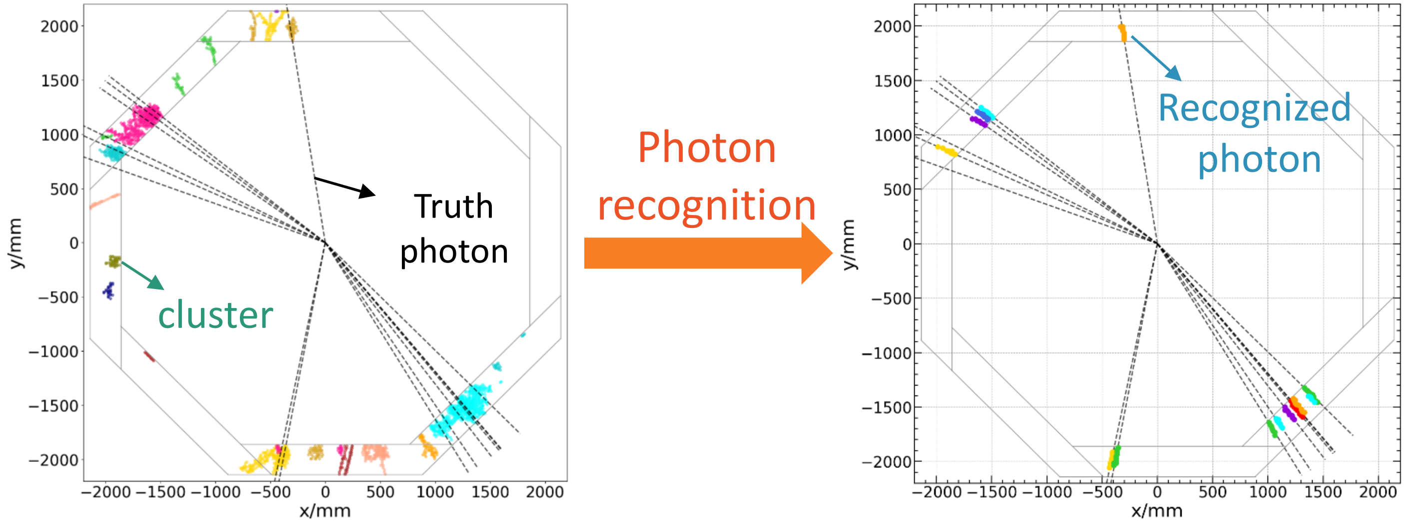


Performance check

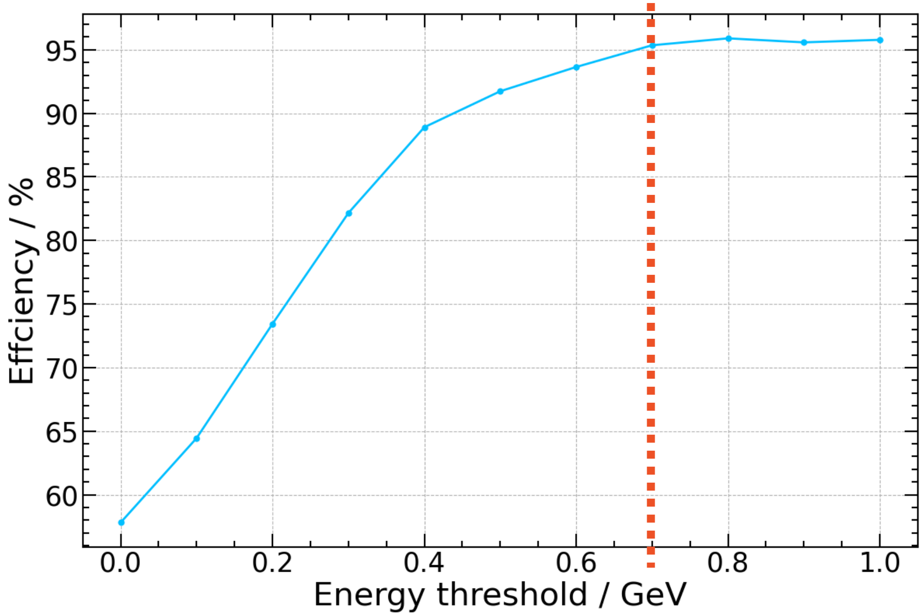
- Jet events
  - $e^+e^- \to \nu\bar{\nu}H,\ H \to b\bar{b}$







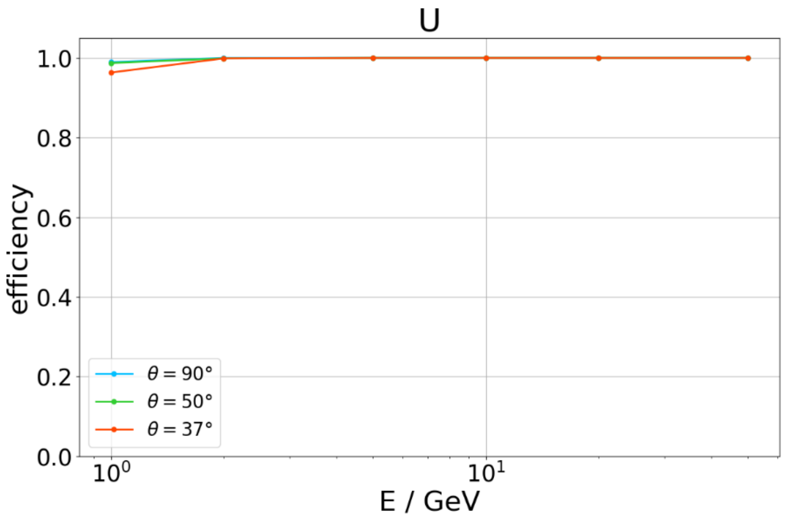
- Performance
  - Recognition efficiency >95% for photons with $E > 0.7\ \mathrm{GeV}$
  - Photons with lower energy will be recognized using other method

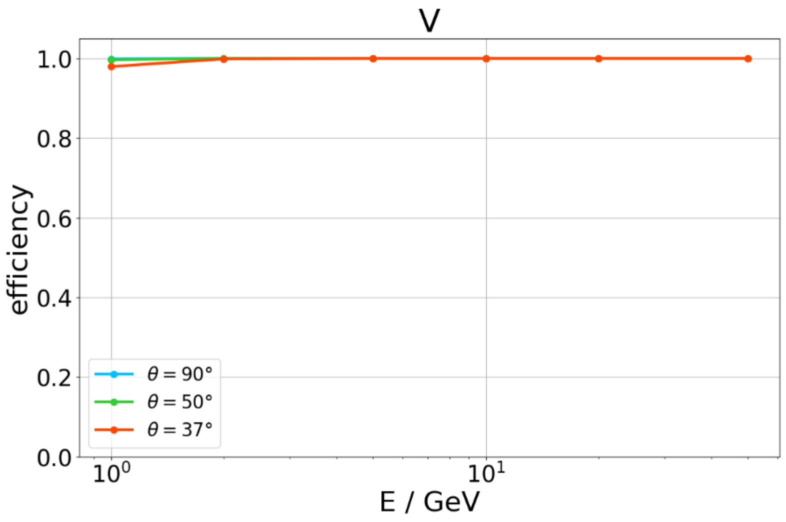


Backup

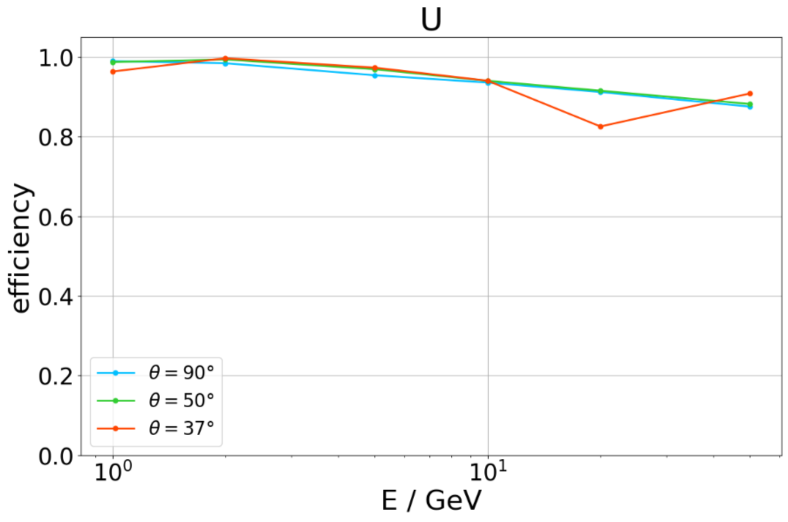
performance for γ

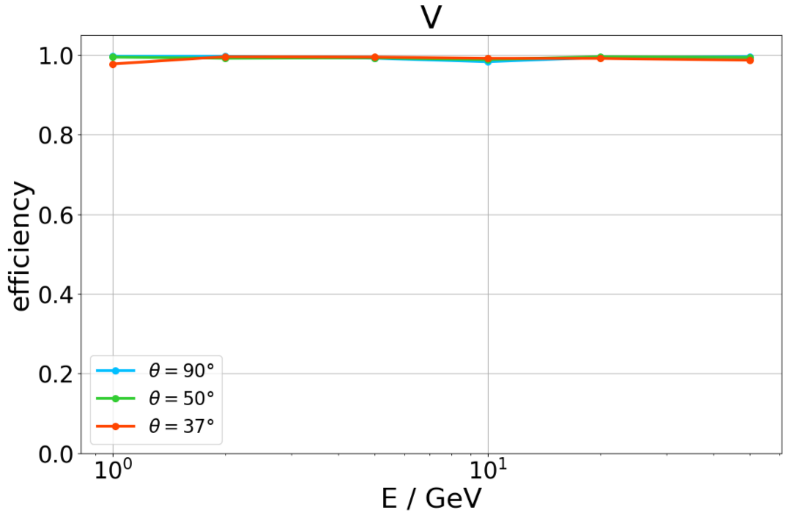
- number of cluster > 0





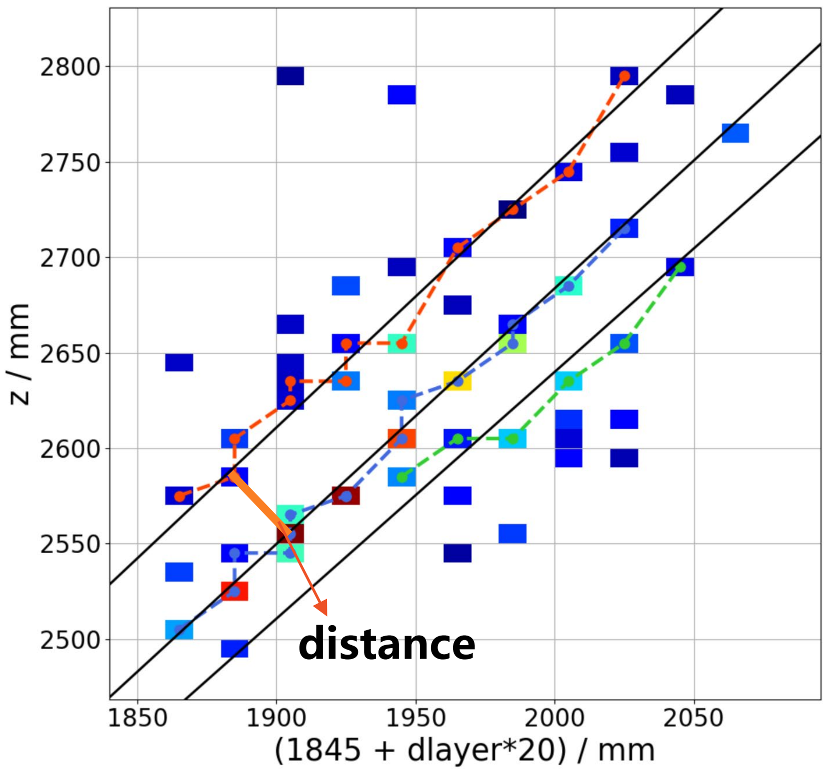
- number of cluster = 1





Distance & E_{small}/E_{max}

- 目的：区分两个相近的 Hough cluster 是否有涨落导致的假光子
- 通过对比这些 Hough clusters 之间的距离以及能量之比，去除假光子本底，保留双光子事例



左图有三个 Hough clusters:

$$\rho_1 = x\cos\alpha_1 + y\sin\alpha_1$$

$$\rho_2 = x\cos\alpha_2 + y\sin\alpha_2$$

$$\rho_3 = x\cos\alpha_3 + y\sin\alpha_3$$

两两做对比，以 $cluster_1, cluster_2$ 为例。设 $E_1 > E_2$

则 $cluster_1$ 中能量最大的 local max 坐标为 (x_m, y_m)

两个cluster的**距离**：过 (x_m, y_m) 垂直于 cluster1 的直线与两个cluster的交点的距离

垂线的方程：$\begin{cases} x = x_m + t\cos\alpha_1 \\ y = y_m + t\sin\alpha_1 \end{cases}$，

(x_m, y_m) 到两个cluster的距离：

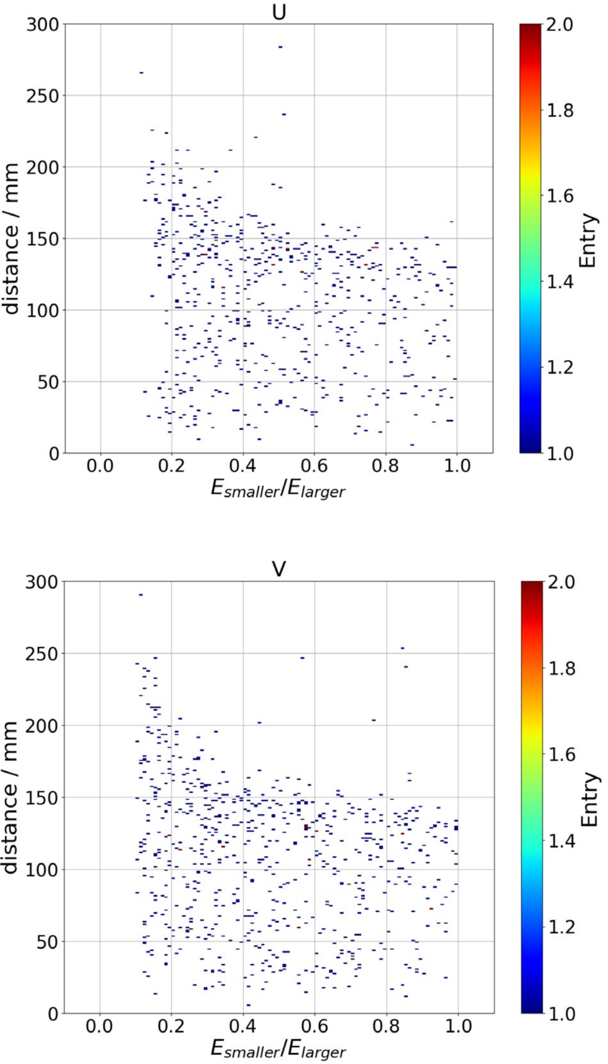
$$t_1 = \rho_1 - x_m\cos\alpha_1 - y_m\sin\alpha_1$$

$$t_2 = \frac{\rho_2 - x_m\cos\alpha_2 - y_m\sin\alpha_2}{\cos\alpha_1\cos\alpha_2 + \cos\alpha_1\cos\alpha_2}$$

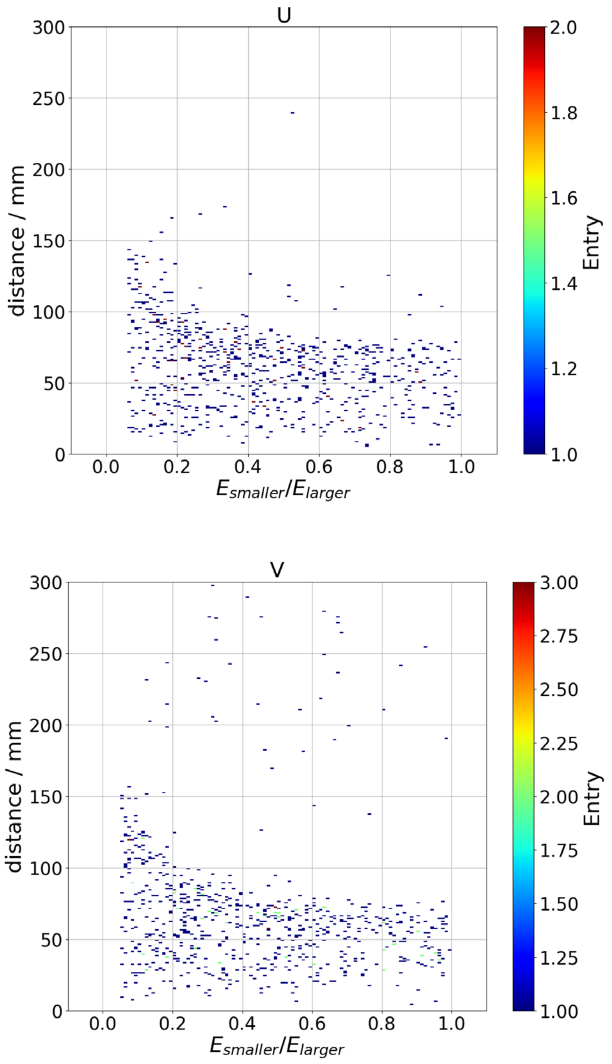
距离=$|t_1 - t_2|$

Distance & E_{small}/E_{max}

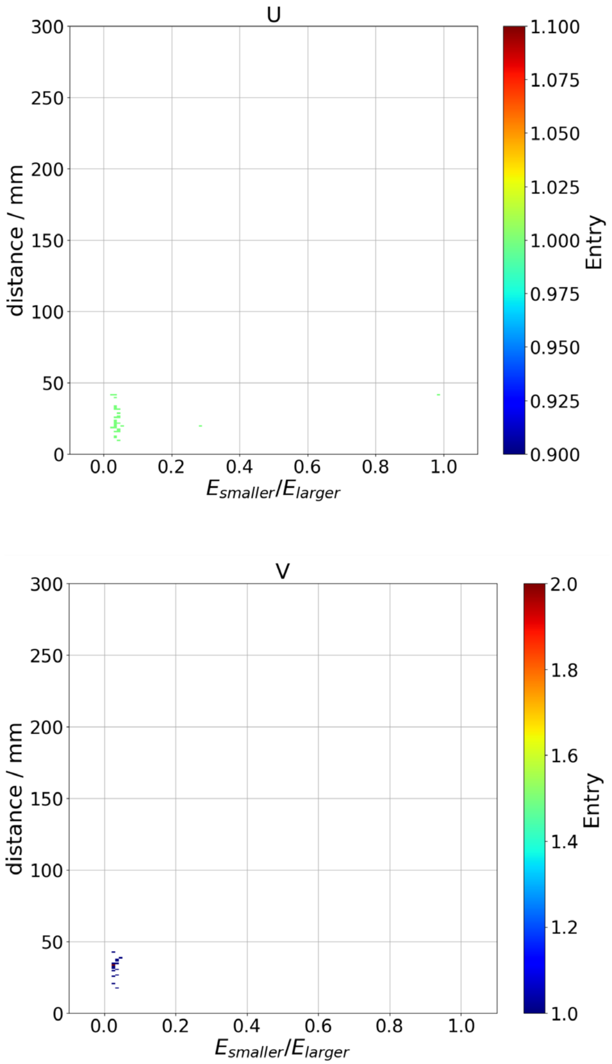
$\pi^0, E = 5GeV, \theta = 50°$



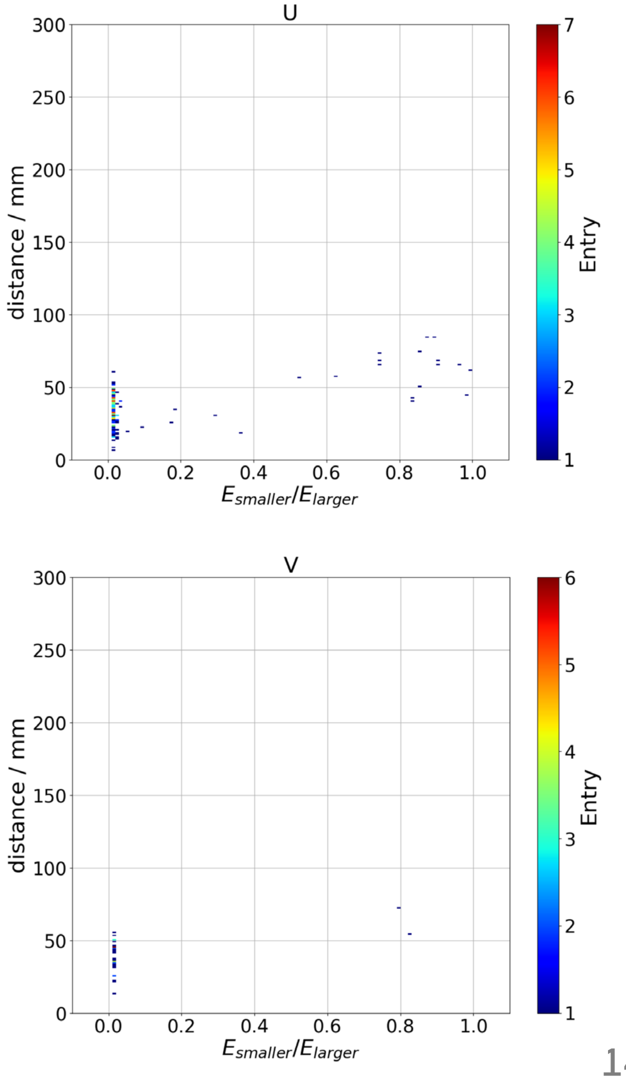
$\pi^0, E = 10GeV, \theta = 50°$



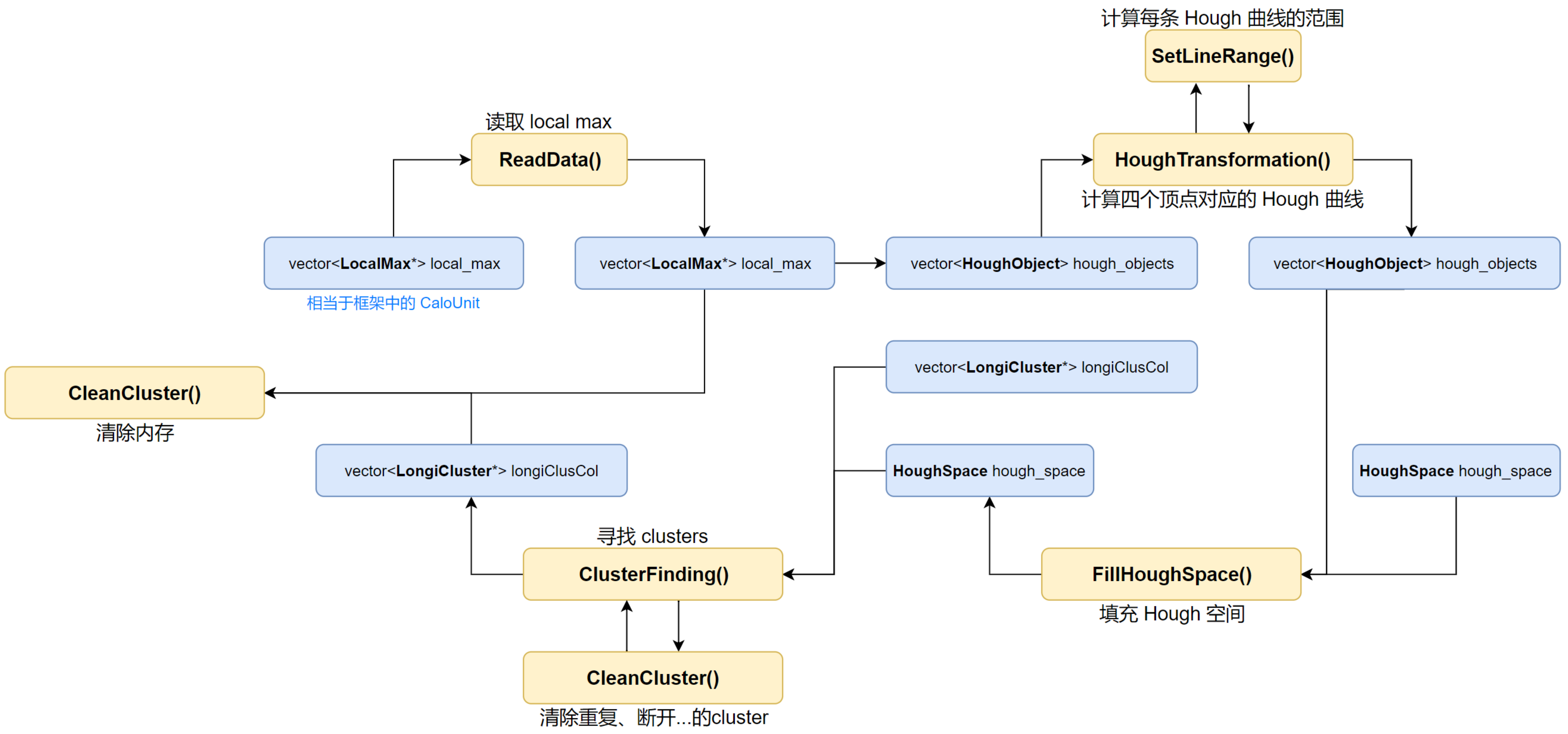
$\gamma, E = 20GeV, \theta = 50°$



$\gamma, E = 50GeV, \theta = 50°$



Workflow



计算每条 Hough 曲线的范围
SetLineRange()
读取 local max
ReadData()
HoughTransformation()
计算四个顶点对应的 Hough 曲线
vector<LocalMax*> local_max
相当于框架中的 CaloUnit
vector<LocalMax*> local_max
vector<HoughObject> hough_objects
vector<HoughObject> hough_objects
vector<LongiCluster*> longiClusCol
CleanCluster()
清除内存
vector<LongiCluster*> longiClusCol
HoughSpace hough_space
HoughSpace hough_space
寻找 clusters
ClusterFinding()
FillHoughSpace()
填充 Hough 空间
CleanCluster()
清除重复、断开...的cluster