GPU Pixel Tracks at HLT doublets filtering

Clustering & Machine Learning

Adriano Di Florio

INFN Sezione di Bari & Università degli Studi di Bari

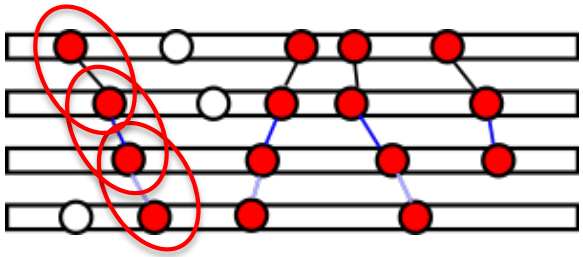
Tracking workflow

- *Current* Future Tracking Workflow plan

Tracking workflow

- *Current* Future Tracking Workflow plan

Doublets construction based mainly on geometry



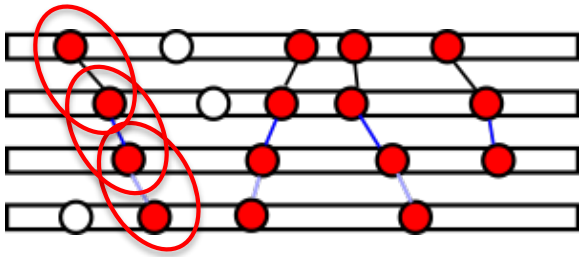
Tracking workflow

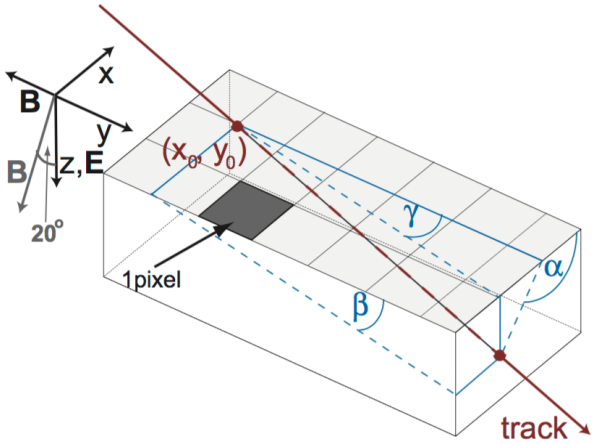
- *Current* Future Tracking Workflow plan

Doublets construction based mainly on geometry

***BUT** we can get some further information form the RECO Hits*

Hit Pixel Clusters





x is azimuthal direction in the *barrel detector* and radial direction in the *forward detectors*

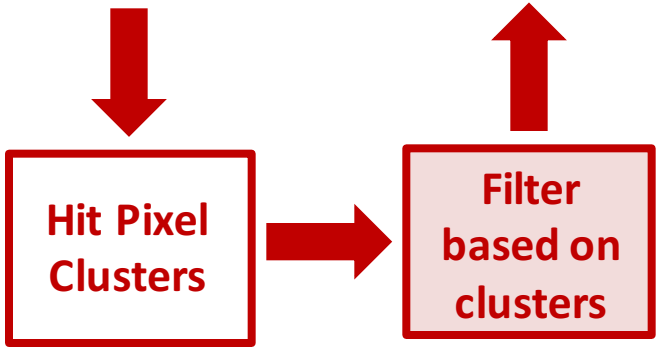
How is a Pixel Cluster represented in CMSSW?

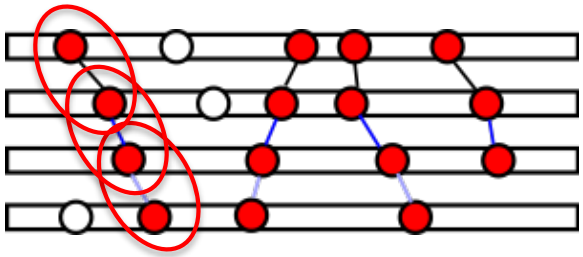
Tracking workflow

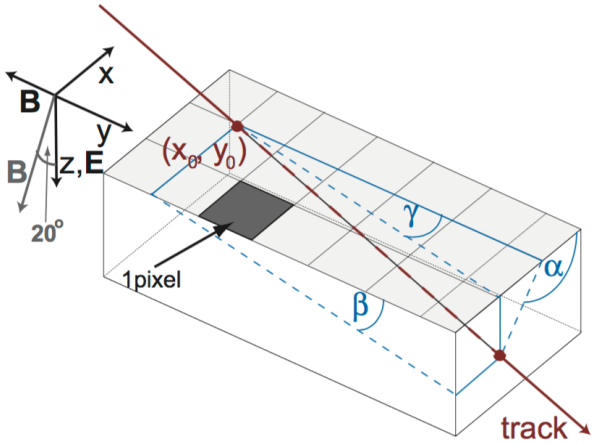
- *Current* Future Tracking Workflow plan

Doublets construction based mainly on geometry

***BUT** we can get some further information form the RECO Hits*







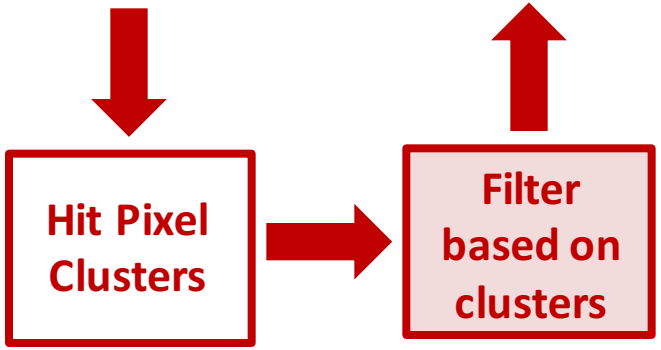
x is azimuthal direction in the *barrel detector* and radial direction in the *forward detectors*

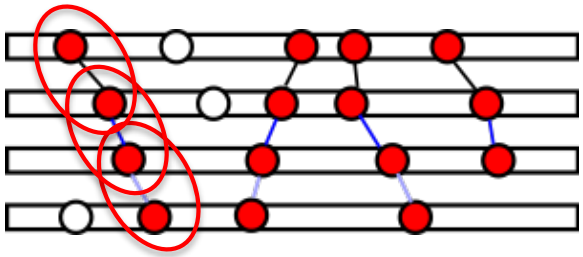
Tracking workflow

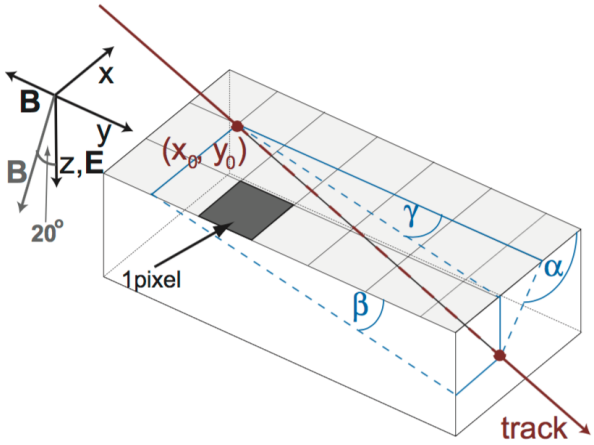
- *Current* Future Tracking Workflow plan

Doublets construction based mainly on geometry

***BUT** we can get some further information form the RECO Hits*







x is azimuthal direction in the *barrel detector* and radial direction in the *forward detectors*

How is a Pixel Cluster represented in the CMSSW?

Pixels Clusters

class `SiPixelCluster`

"collection" of `Pixel`

- **`uint16_t`** `x`
- **`uint16_t`** `y`
- **`uint16_t`** `adc`

Pixels Clusters

`class` `SiPixelCluster` → Two vectors of `uint16_t` with *position* and *adc* values **only** for pixels turned on

"collection" of `Pixel`

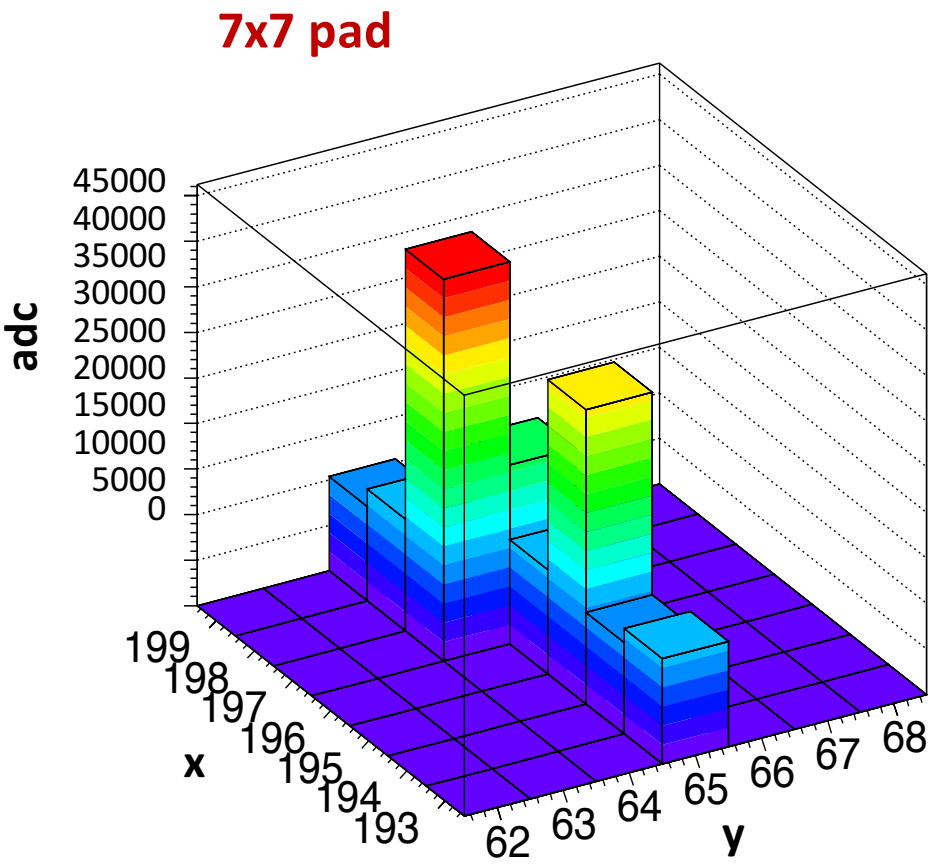
- **`uint16_t`** `x`
- **`uint16_t`** `y`
- **`uint16_t`** `adc`

Pixels Clusters

`class SiPixelCluster` ⟶ Two vectors of `uint16_t` with *position* and *adc* values **only** for pixels turned on

"collection" of `Pixel`

- **`uint16_t`** `x`
- **`uint16_t`** `y`
- **`uint16_t`** `adc`



Pixels Clusters

`class SiPixelCluster` → Two vectors of `uint16_t` with *position* and *adc* values **only** for pixels turned on

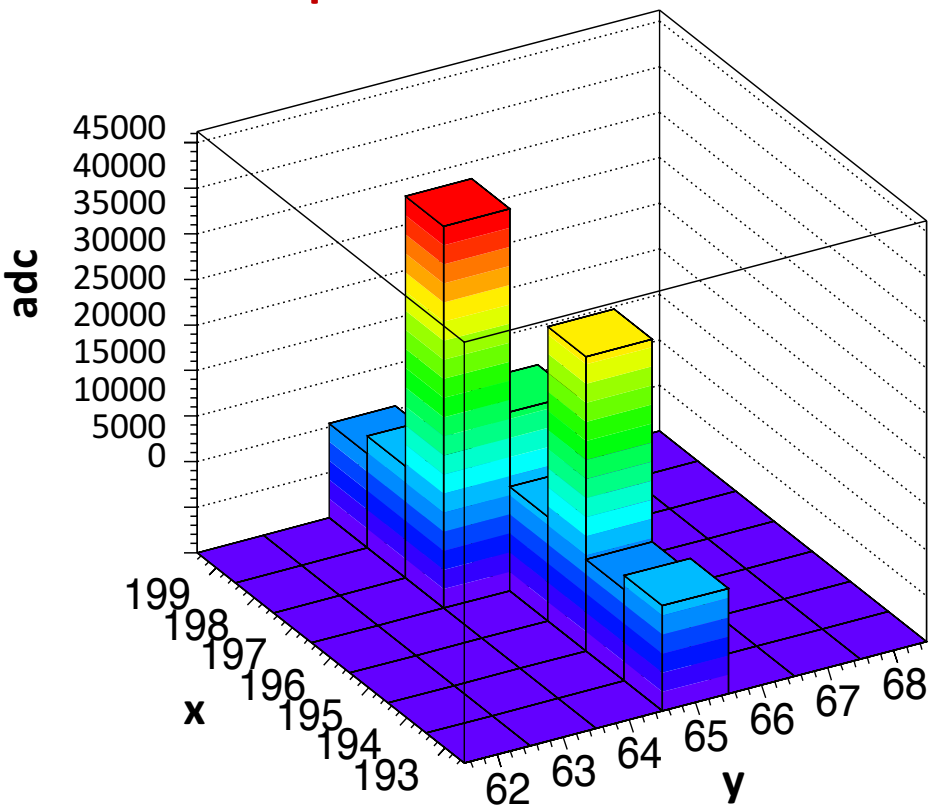
"collection" of `Pixel`

- **`uint16_t`** `x`
- **`uint16_t`** `y`
- **`uint16_t`** `adc`

THE IDEA

Use Machine Learning & Image Recognition techniques to add an additional filtering for doublets based on clusters shapes.

7x7 pad



Pixels Clusters

`class SiPixelCluster` → Two vectors of `uint16_t` with *position* and *adc* values **only** for pixels turned on

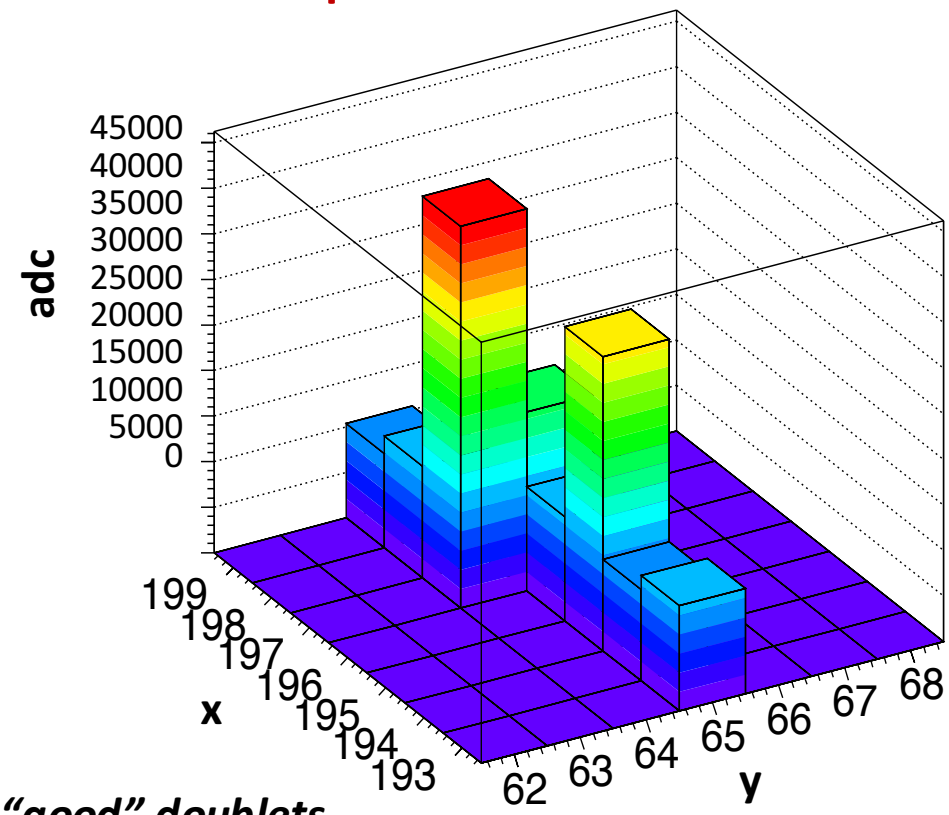
"collection" of `Pixel`

- **`uint16_t`** `x`
- **`uint16_t`** `y`
- **`uint16_t`** `adc`

THE IDEA

Use Machine Learning & Image Recognition techniques to add an additional filtering for doublets based on clusters shapes.

7x7 pad

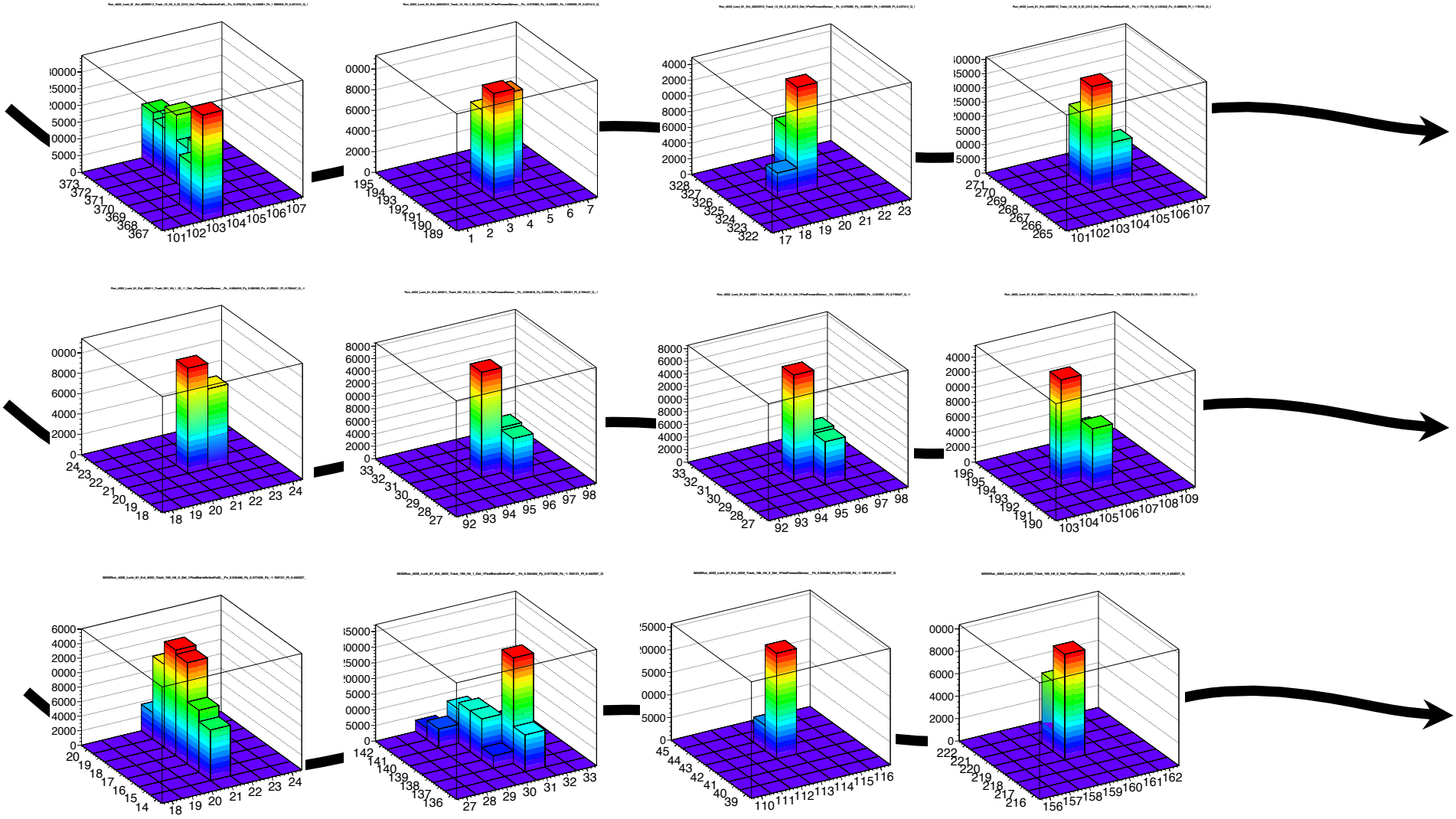


One of the first steps : *link clusters and "good" doublets*.

Matching Tracks

Example – RECO Tracks

TTbar_13+TTbar_13TeV (via runTheMatrix.py)



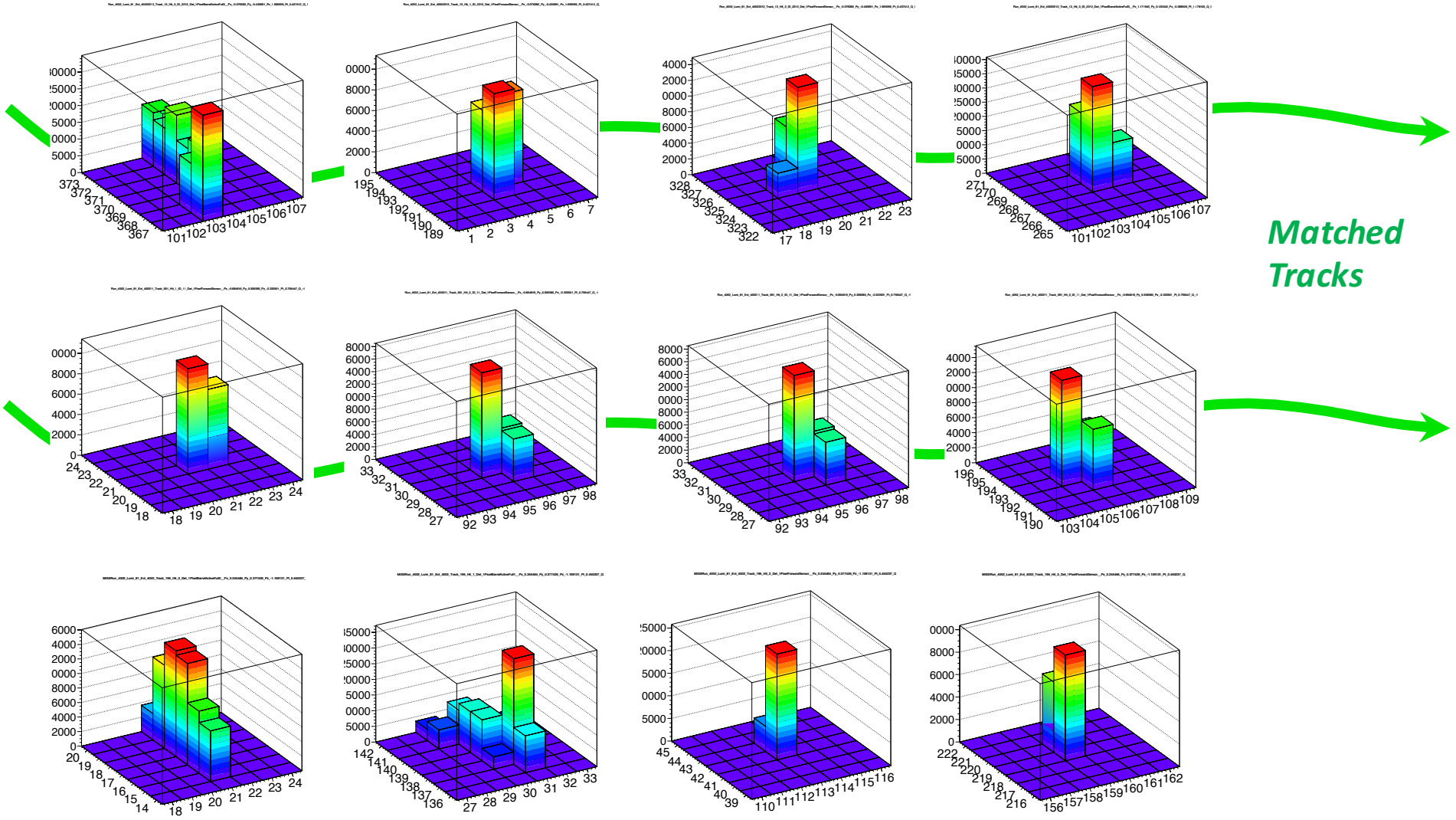
Matching Tracks

Example – RECO Tracks

TTbar_13+TTbar_13TeV (via runTheMatrix.py)

RECO – SIM matching

(MultiTrackValidator.cc)



Matched Tracks

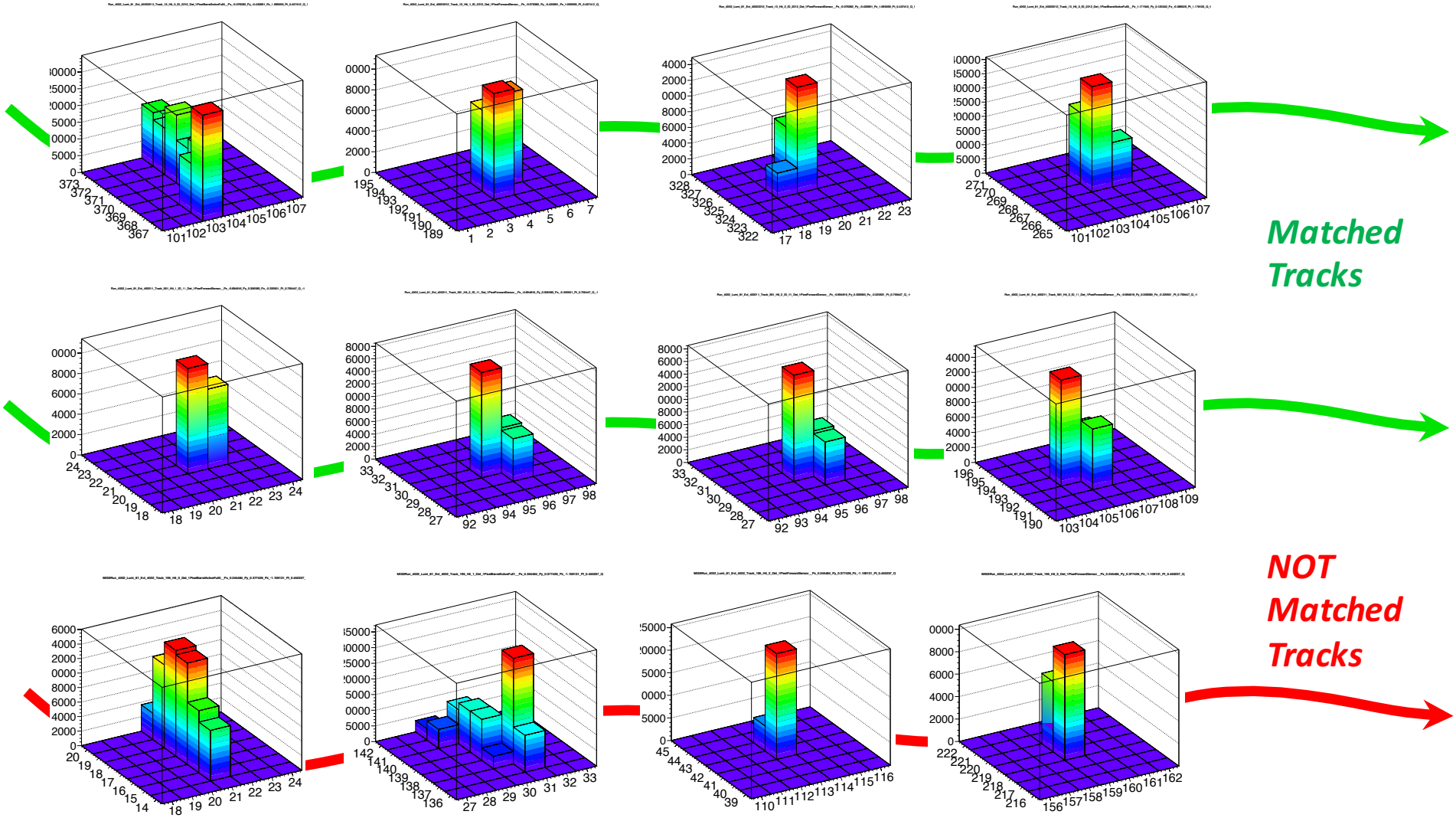
Matching Tracks

Example – RECO Tracks

TTbar_13+TTbar_13TeV (via runTheMatrix.py)

RECO – SIM matching

(MultiTrackValidator.cc)

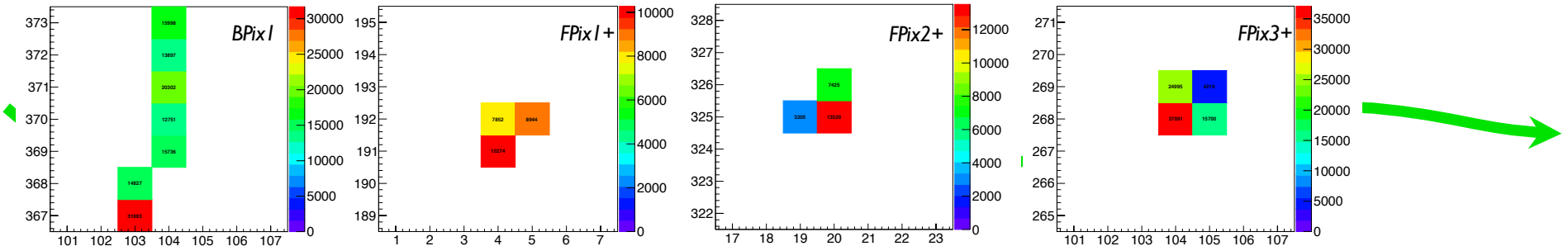


Matched Tracks

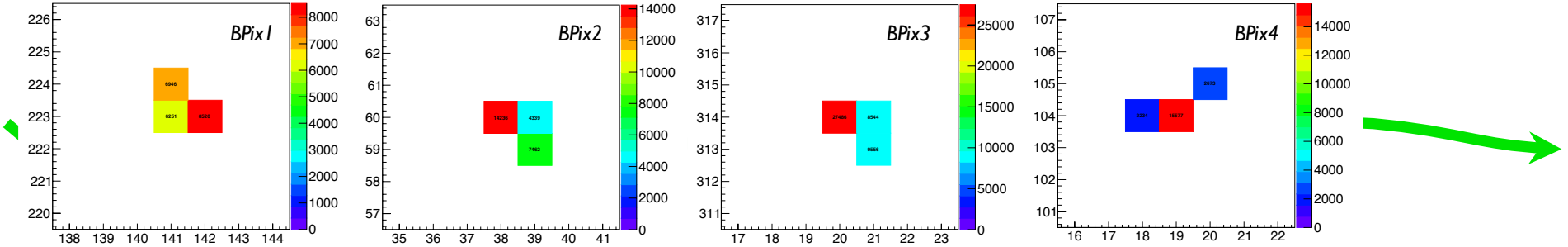
NOT Matched Tracks

Matching Tracks

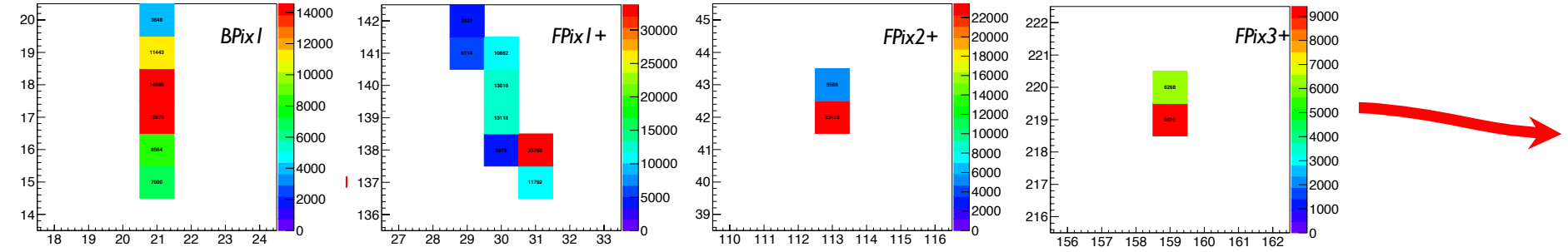
proton (id: 2212)



electron (id: 11)

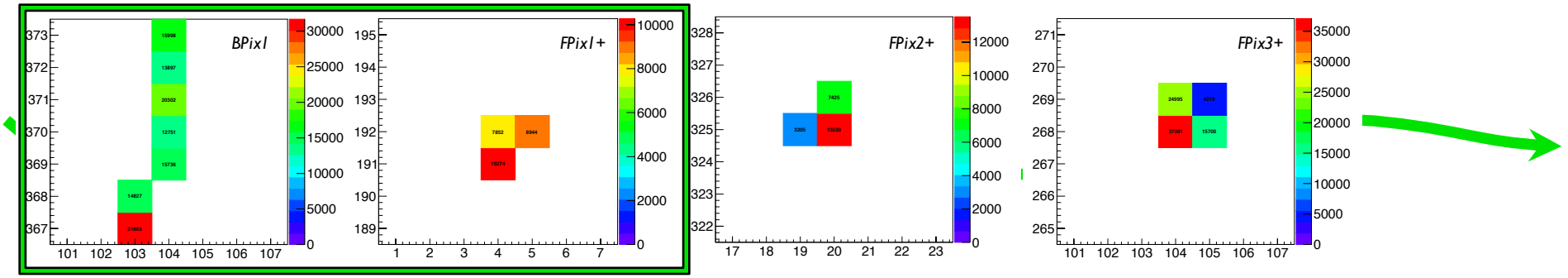


fake track

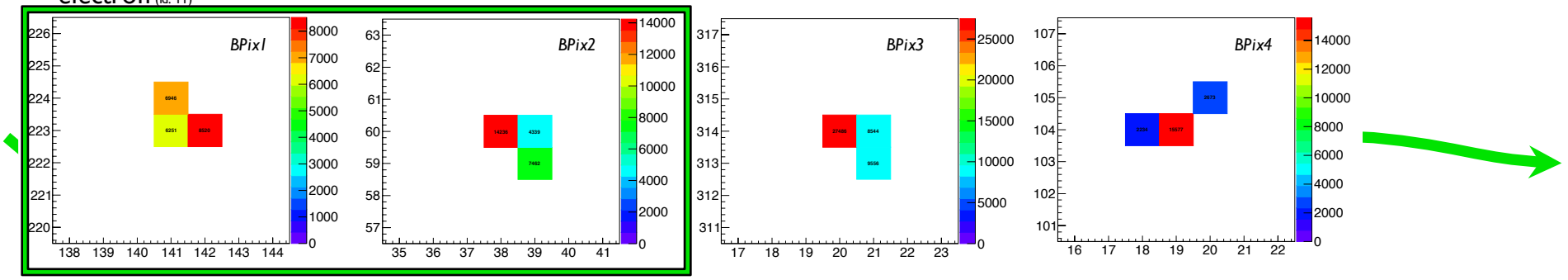


Matching Tracks

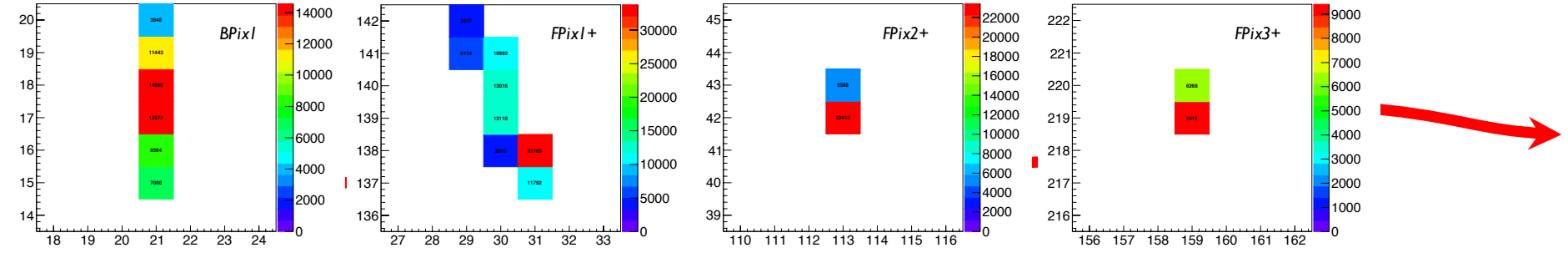
proton (id: 2212)



electron (id: 11)

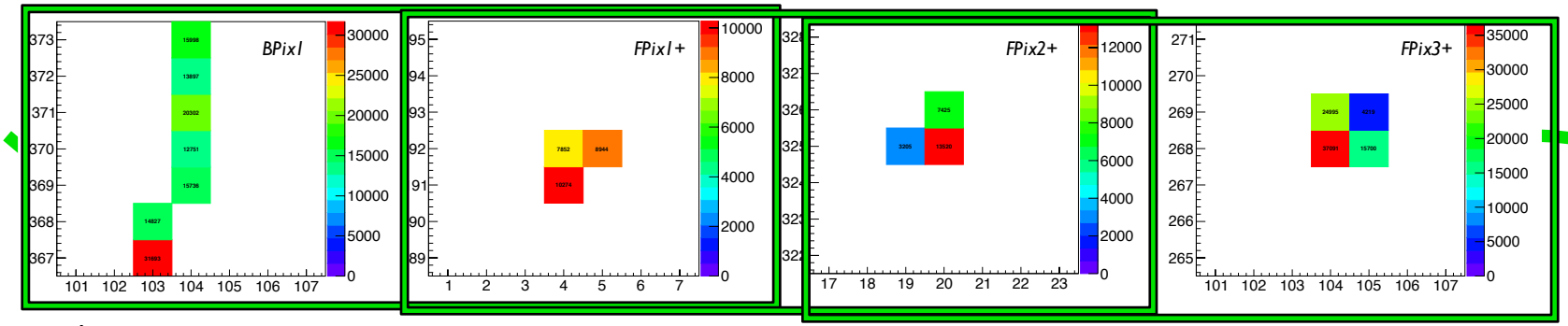


fake track

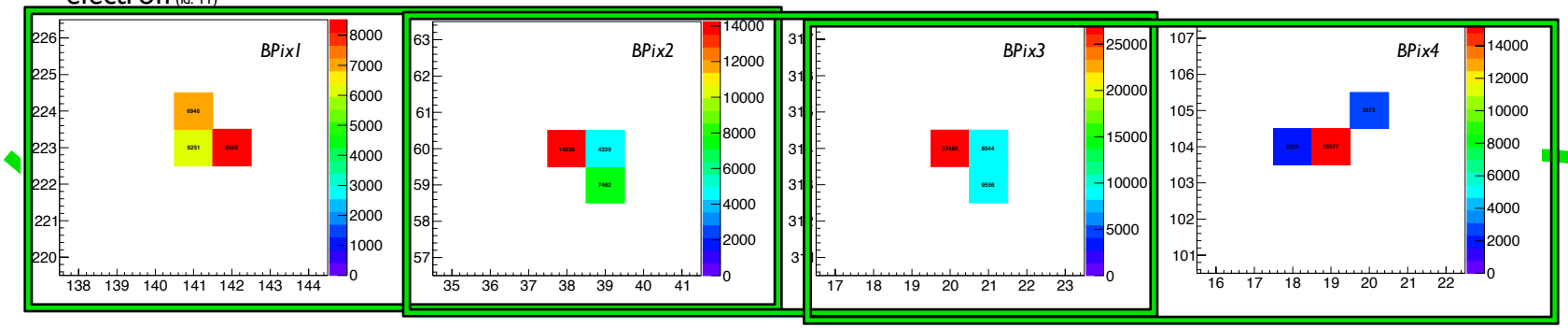


Matching Tracks

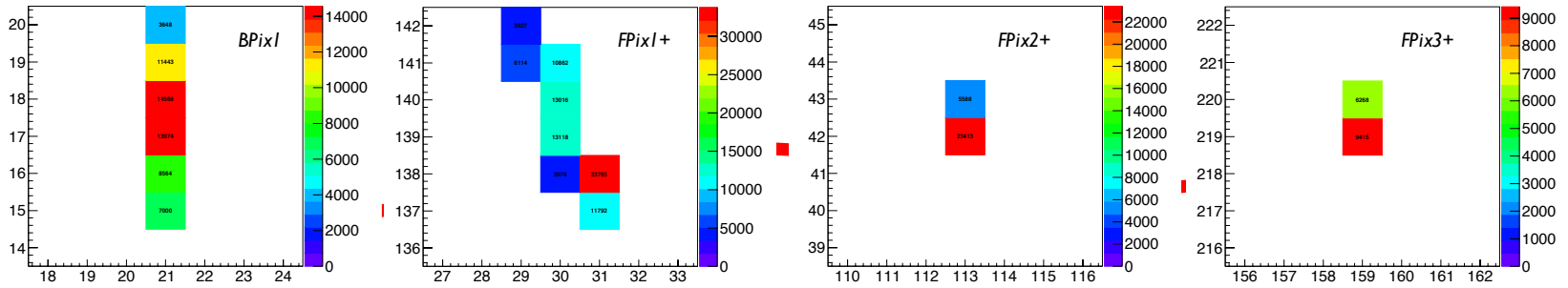
proton (id: 2212)



electron (id: 11)

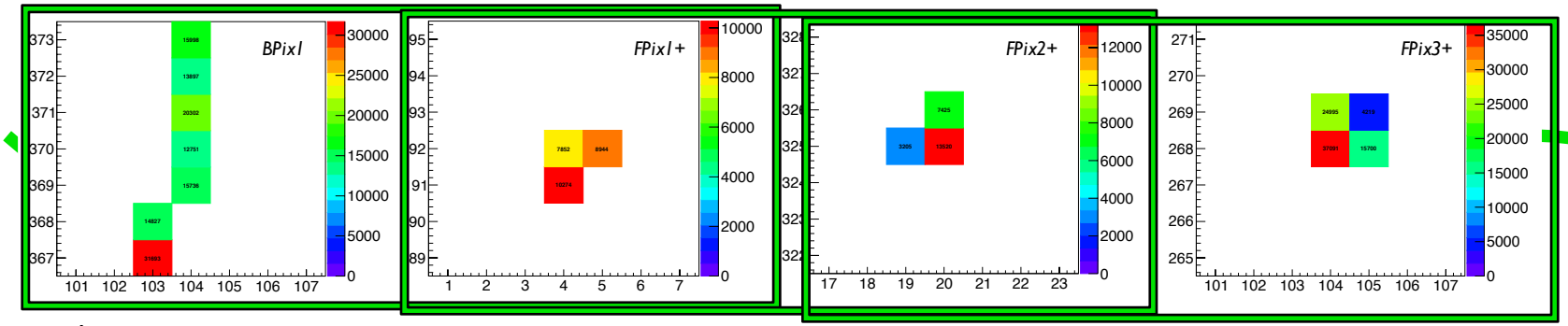


fake track

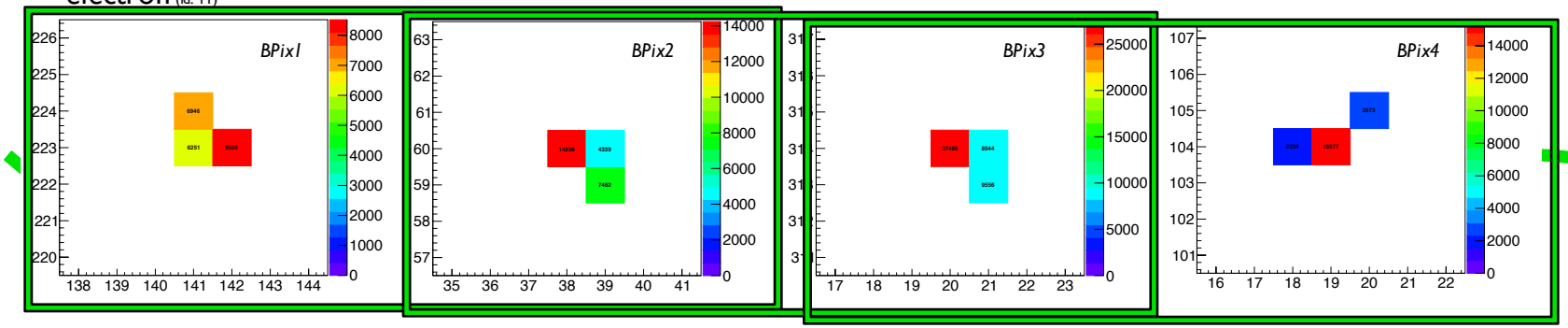


Matching Tracks

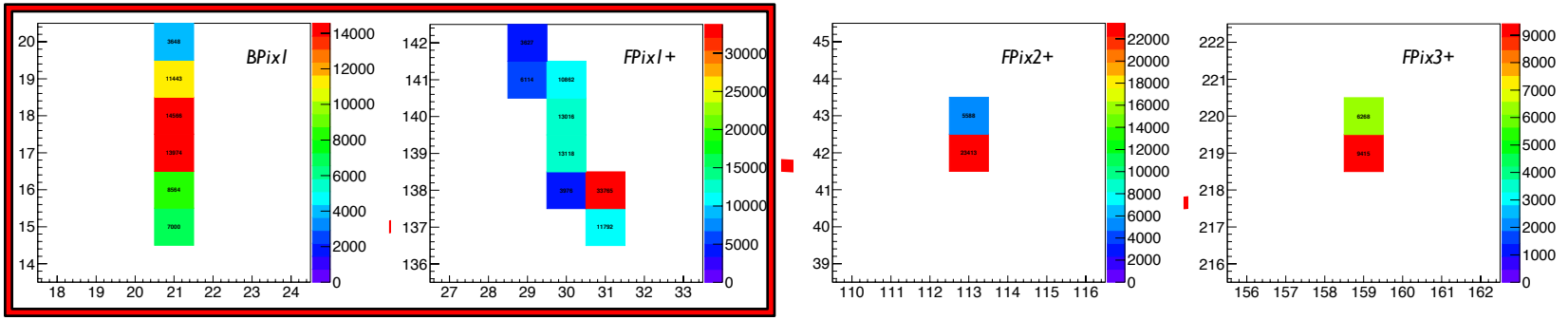
proton (id: 2212)



electron (id: 11)

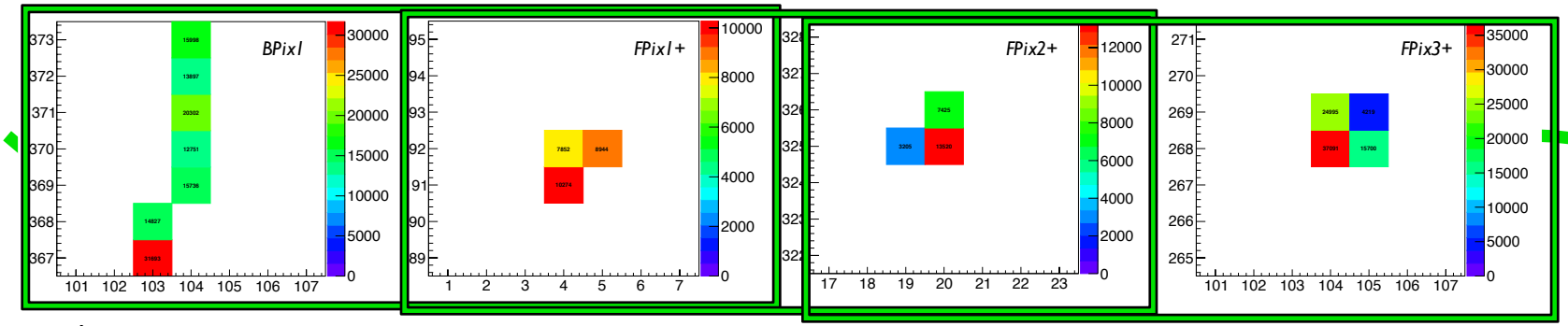


fake track

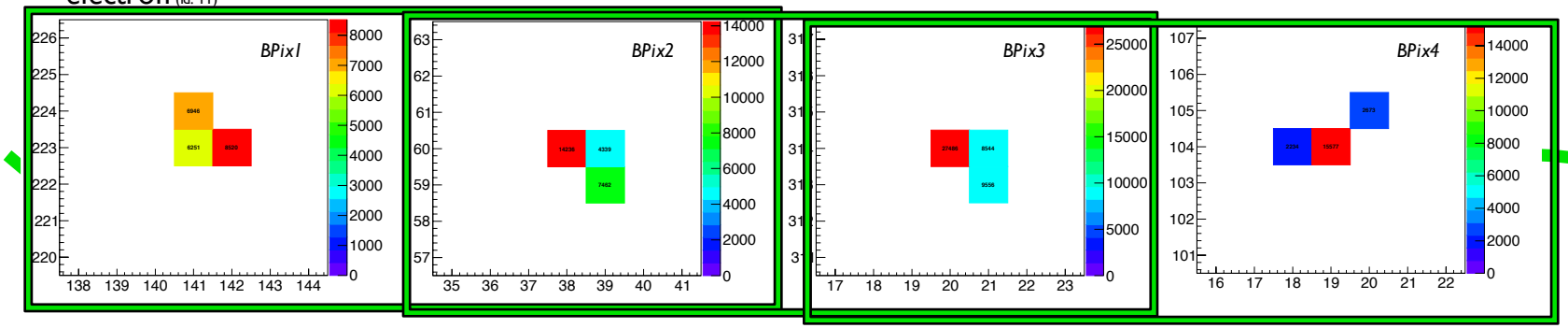


Matching Tracks

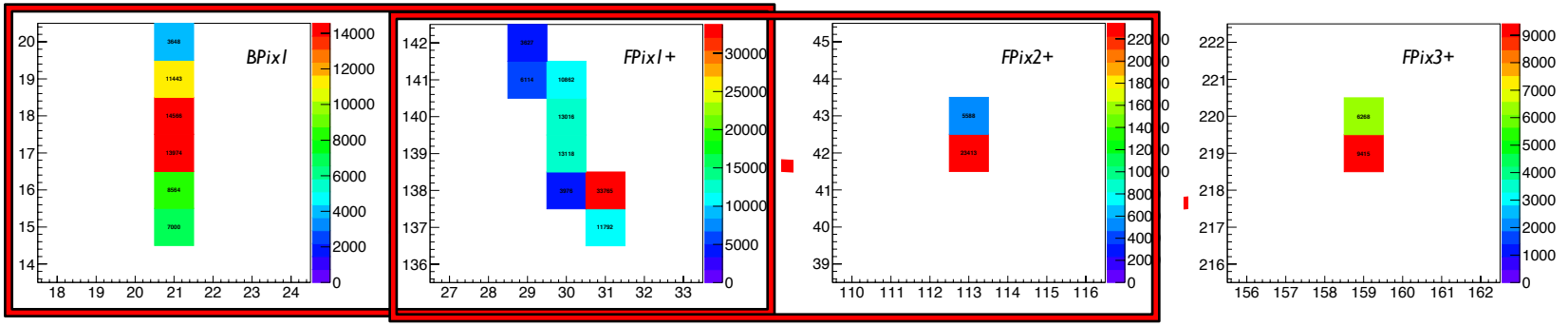
proton (id: 2212)



electron (id: 11)

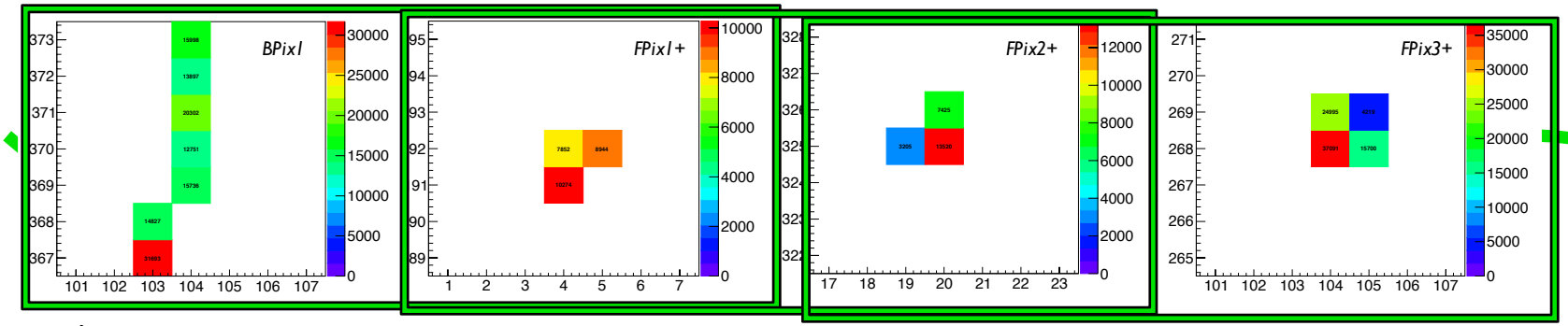


fake track

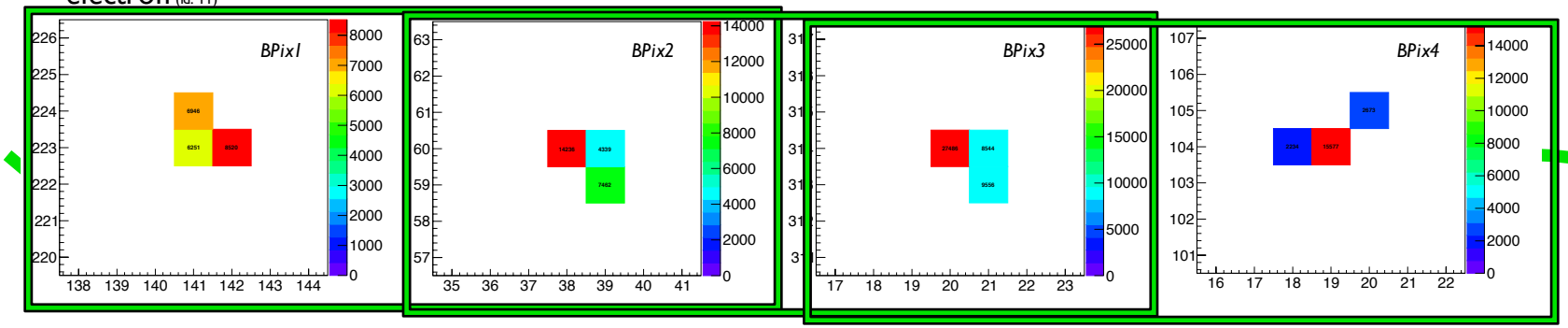


Matching Tracks

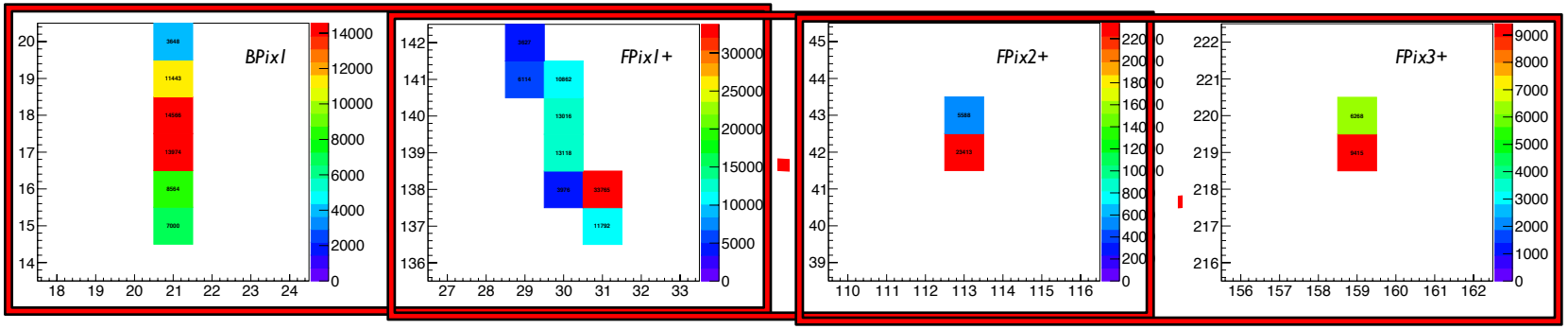
proton (id: 2212)



electron (id: 11)



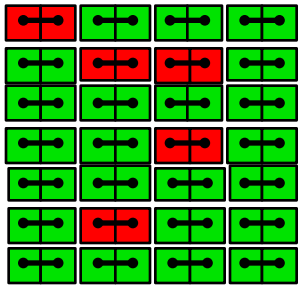
fake track



The basic idea & next plans

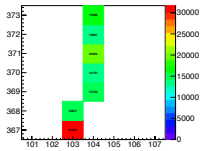
TRAINING SET

- good way?
- data format?
- . . .



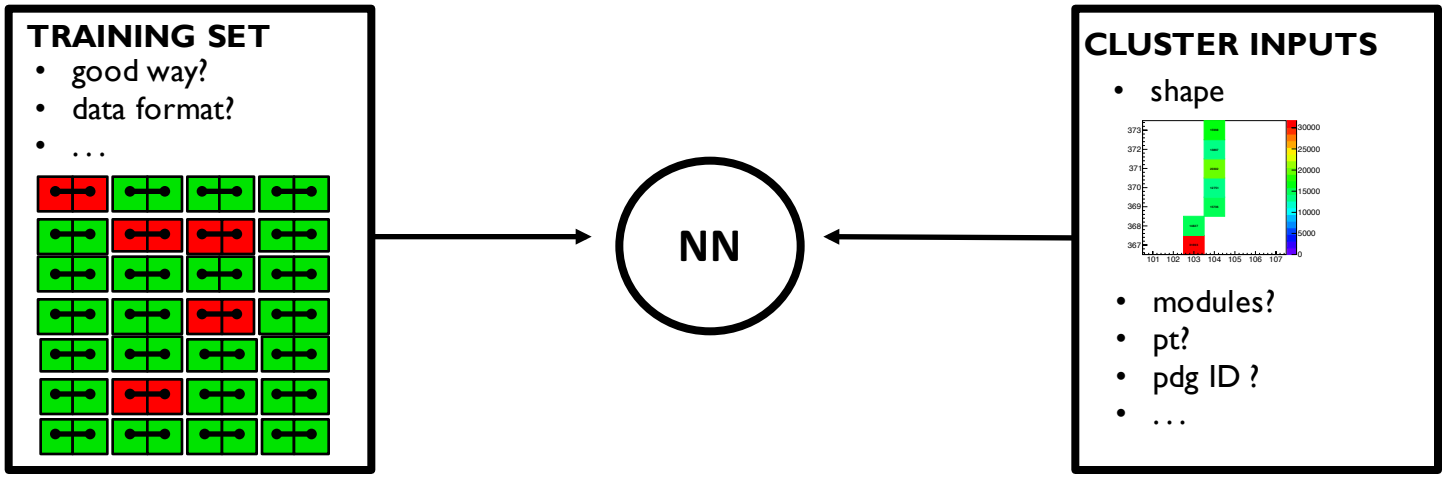
CLUSTER INPUTS

- shape

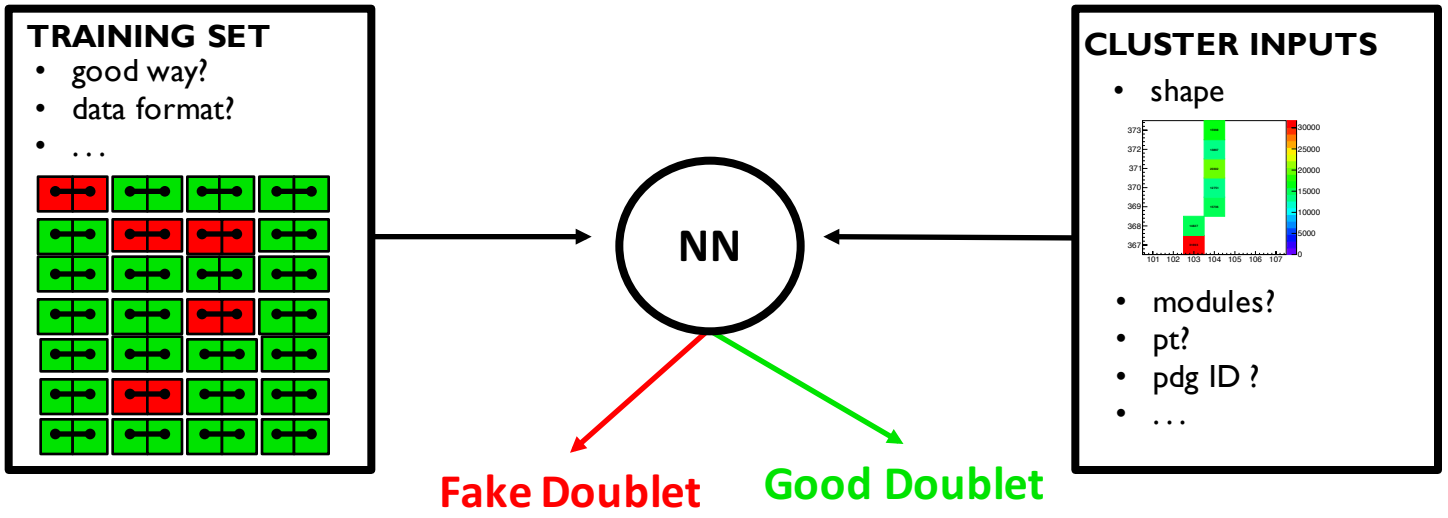


- modules?
- pt?
- pdg ID ?
- . . .

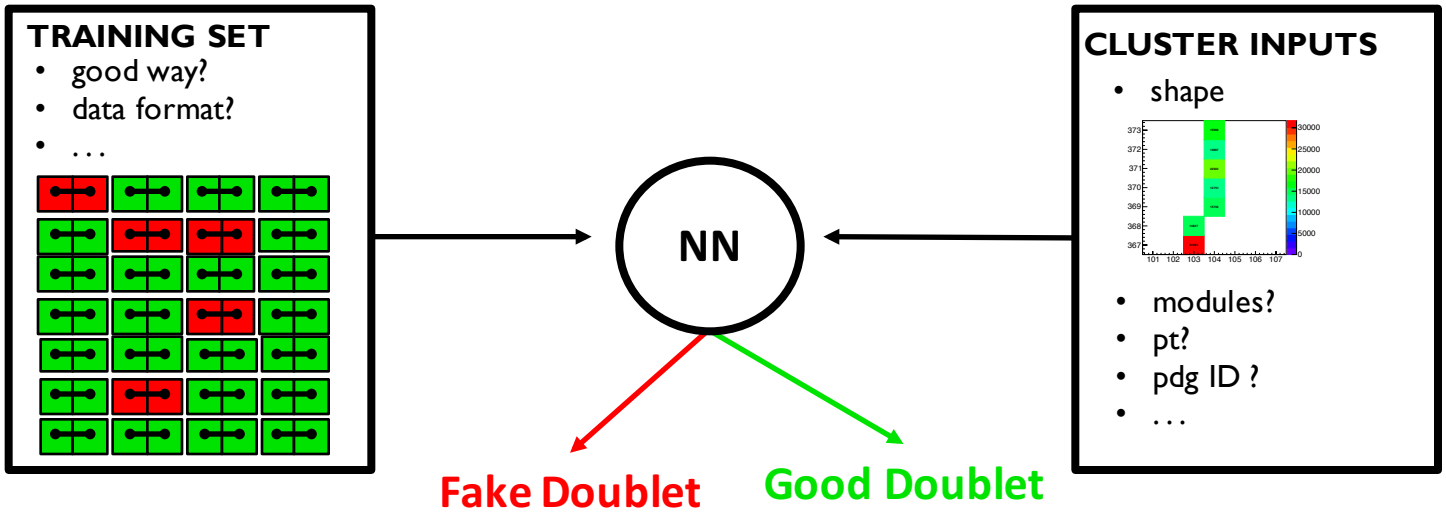
The basic idea & next plans



The basic idea & next plans



The basic idea & next plans



Just a glimpse . . .

- supervised learning tuning
- build up an appropriate training dataset
- undestand which inputs are usefull
- how to deal with data
- any suggestion,criticism **well accepted!**

Thank you

"I am putting myself to the fullest possible use, which is all I think that any conscious entity can ever hope to do"

HAL9000