Automatic Colorization for Jet Clustering (This is test)

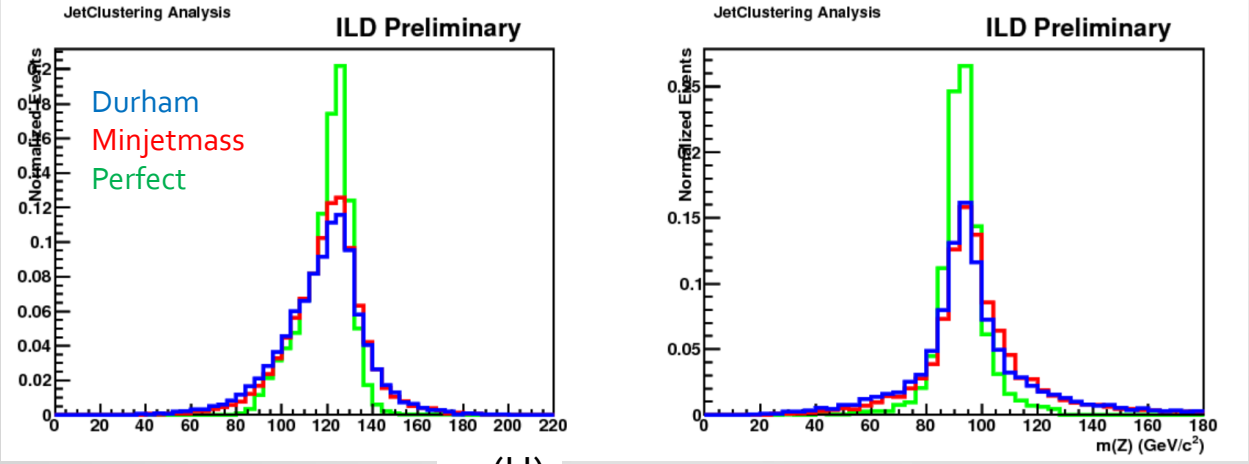
Masakazu Kurata

KEK

02/03/2018

Introduction

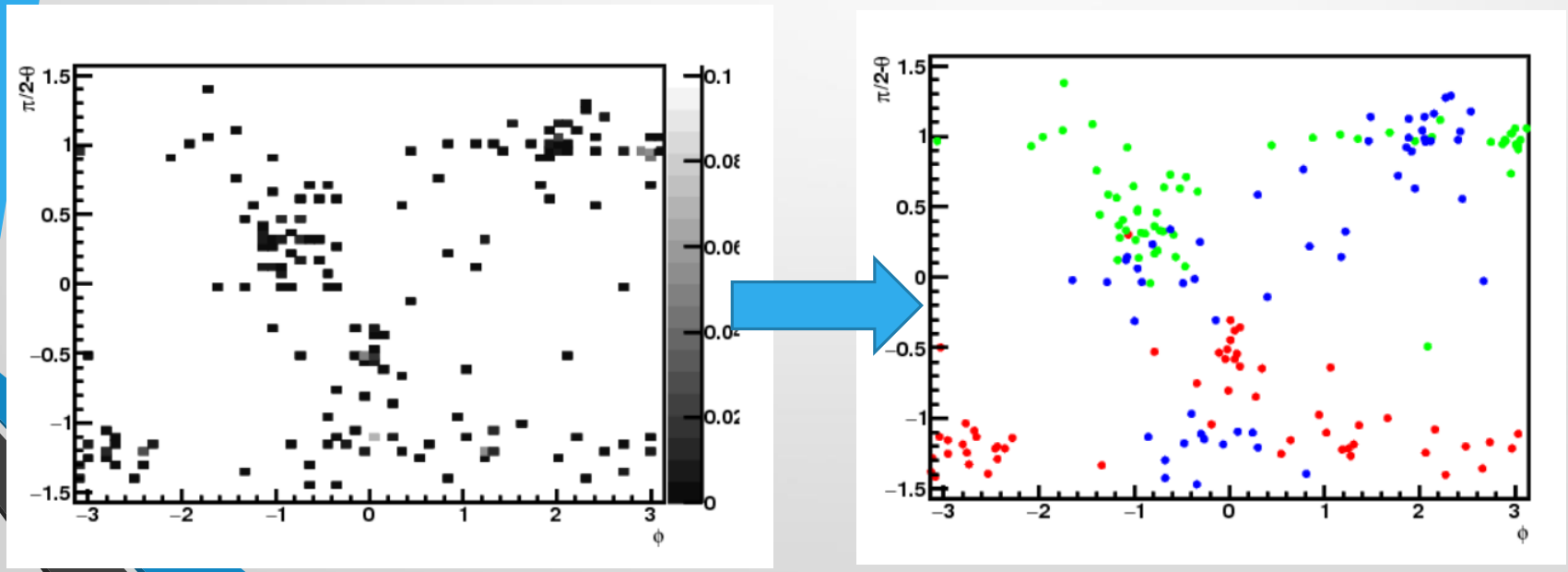
- Jet clustering is one of the main key to obtain better physics results
  - Physics results are strongly limited by mis-clustering
  - To obtain correct jets leads to improve the mass resolution of the resonances
- Present jet clustering is far from good tool for reconstructing jets
  - e.g. Higgs self-coupling: ~40% improvement if perfect!



- Staging: even at 250GeV, clustering is very important
  - Separation of ZH/ZZ/WW in hadronic events

Last time

- One of the problem is how we can absorb the difference between events
  - For very high efficiency, from NN view, all the events look "exception"…→infinite number of nodes & infinite number of events is necessary?
- CNN can relatively absorb position shift & distortion of shape
  - So, CNN meets this?
- One idea: "Automatic Colorization" using CNN
  - Gray scale → Color

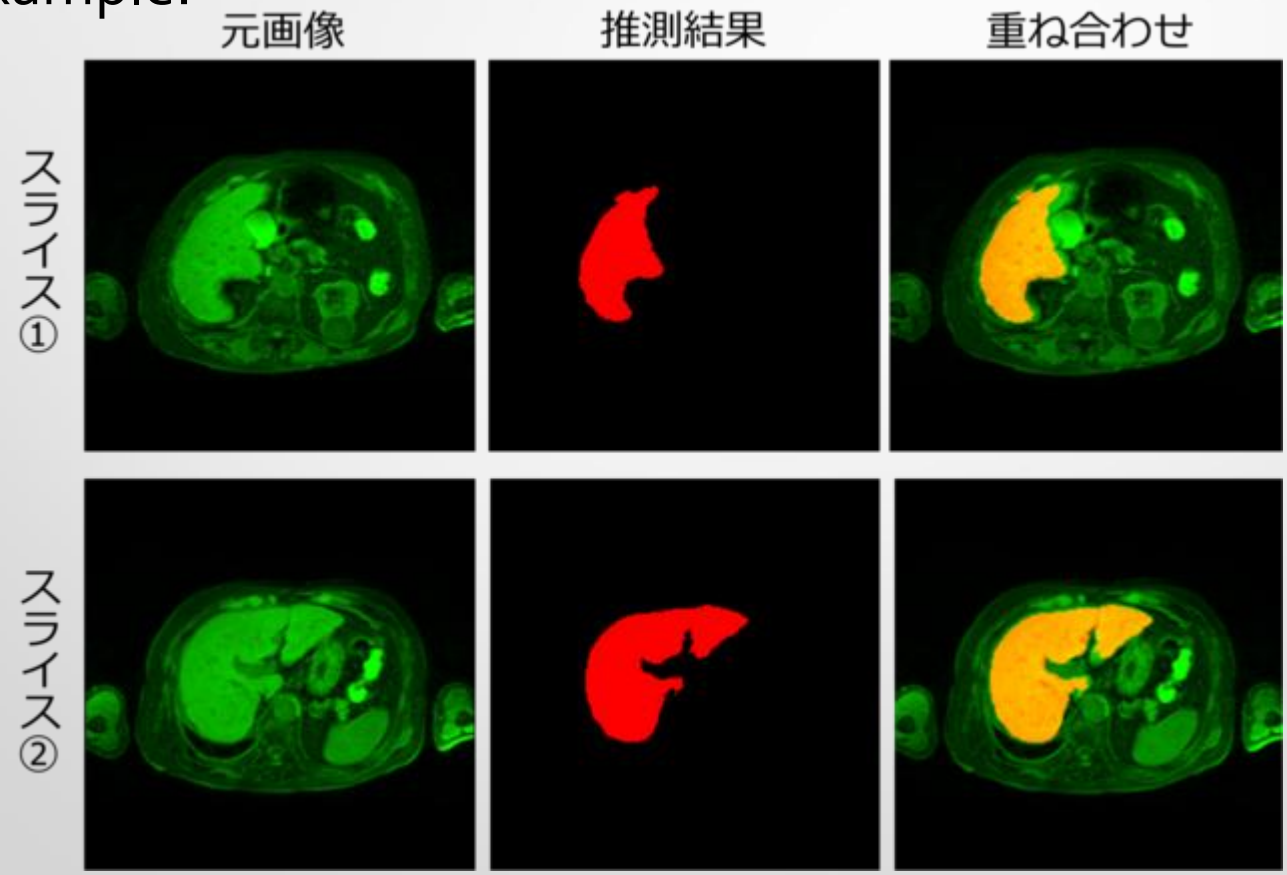


- Now, trying to do that

Example

or

- We can estimate the region of what we want to know
- This calls "semantic segmentation"
- Example:

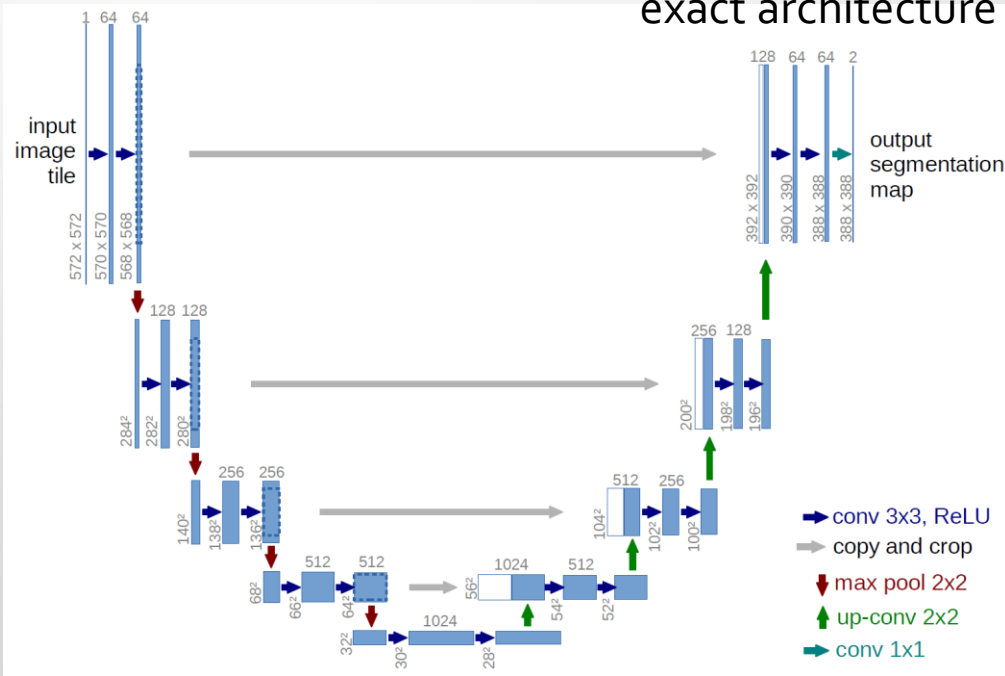


- Can we apply these kinds of techniques for jet clustering?

Use CNN for automatic colorization

- Try to make the most of CNN
- For jet clustering, we need the overall and local information for each event
  - Where is the large energy located?
  - Correlation between neighbors or large energy area?
- Using CNN, we will extract both of the features
  - I don't want to mention about the details of CNN
  - Using u-network
  - Already ~30 layers in CNN!

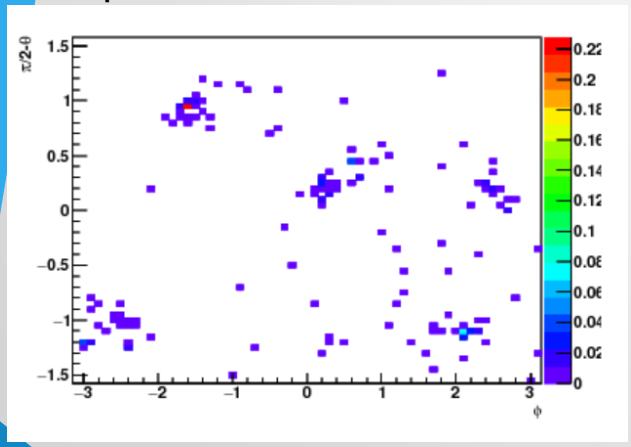
This is not the exact architecture



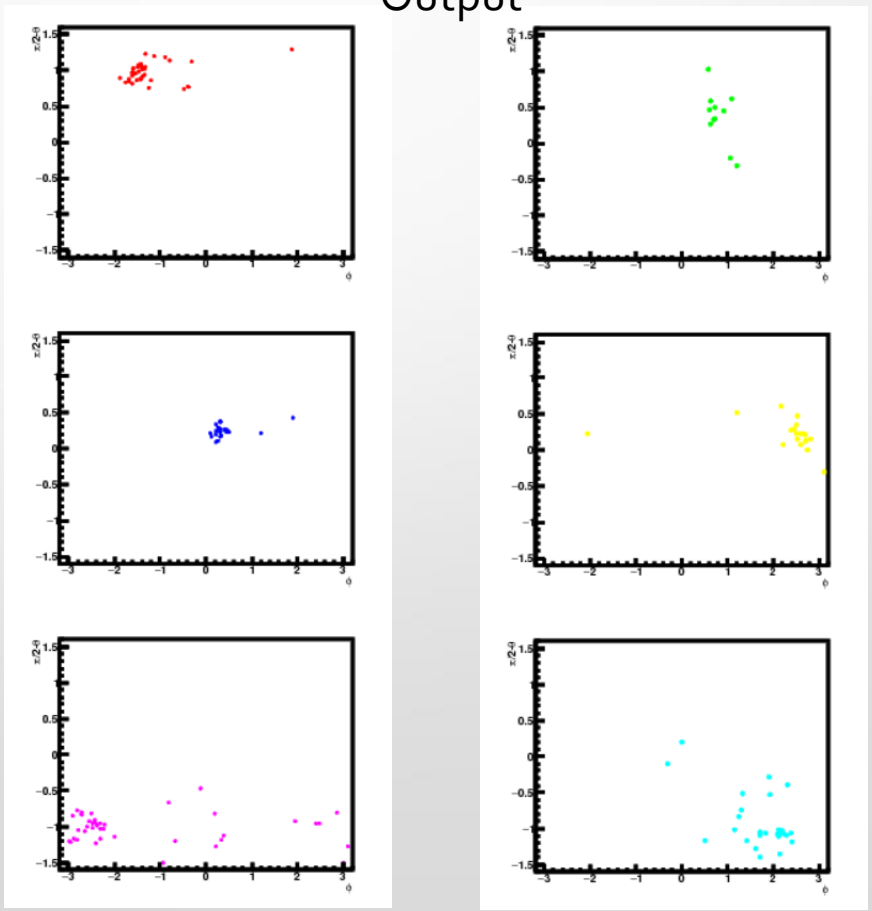
trial

- Using energy map of each event, estimate color of each track
  - ZHH→(qq)(bb)(bb)→6jets

Input



Output



Over fit check

- This is still test stage, so cannot check overfitting well
- Just estimate using loss function(small is better):

$$L = -\frac{1}{N}\sum_{jet}\sum_{track}\frac{E_{track}}{E_{jet}}Logy_{track}$$

- If no overfitting, L is almost same between test and train

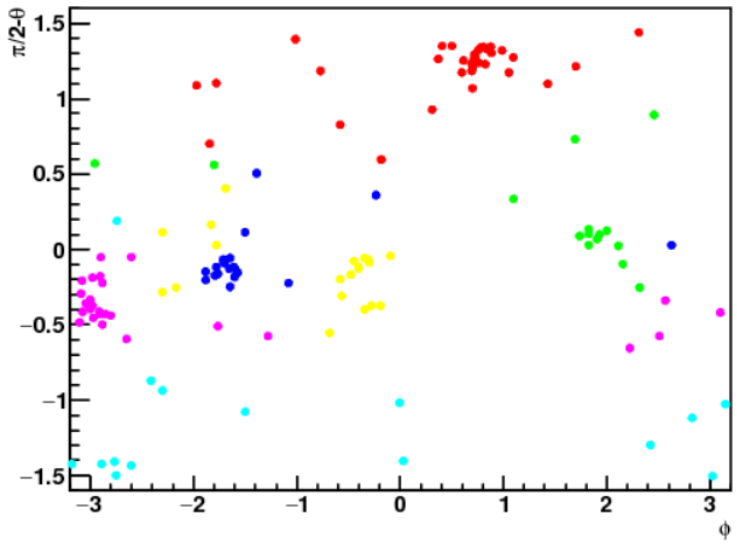
Num. of training events	Loss Train	Loss Test
140	0.185	1.78
4000	0.464	0.725
9000	0.571	0.654

- Over fitting will vanish if num. of training events is O(10000)
  - But, performance gradually degrades
  - So need to optimize the network size to recover the performance

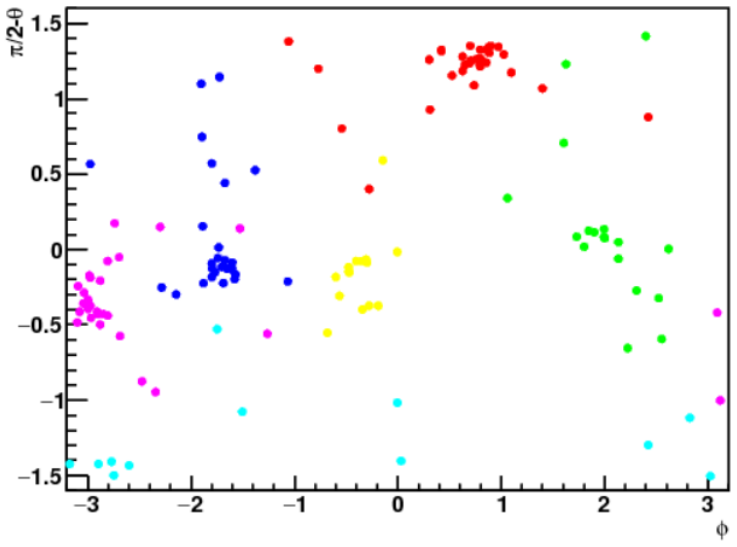
Examples(good?)

- Event 9511:

Estimation

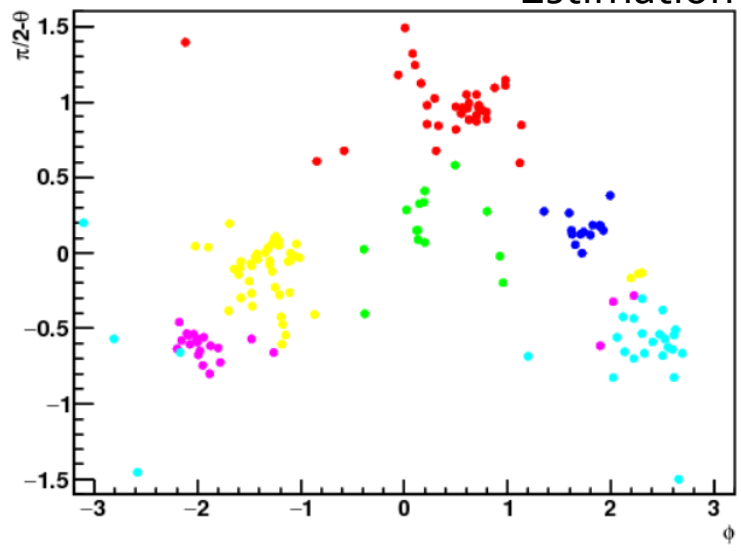


Answer

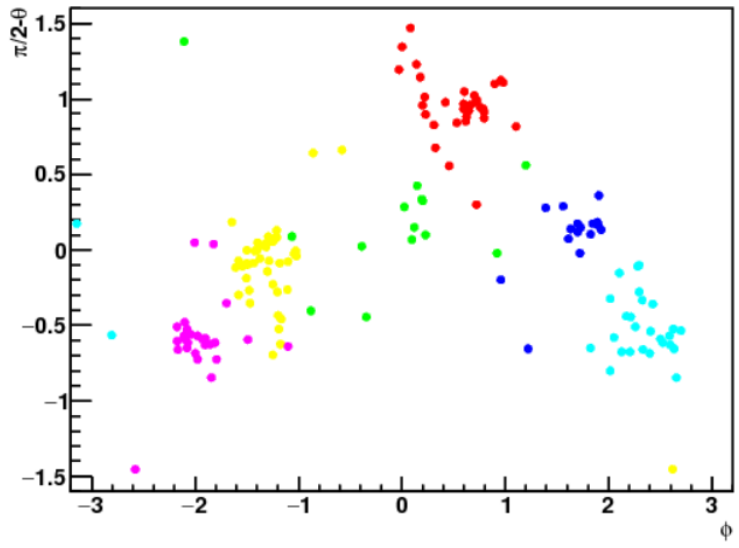


- Event 9526:

Estimation



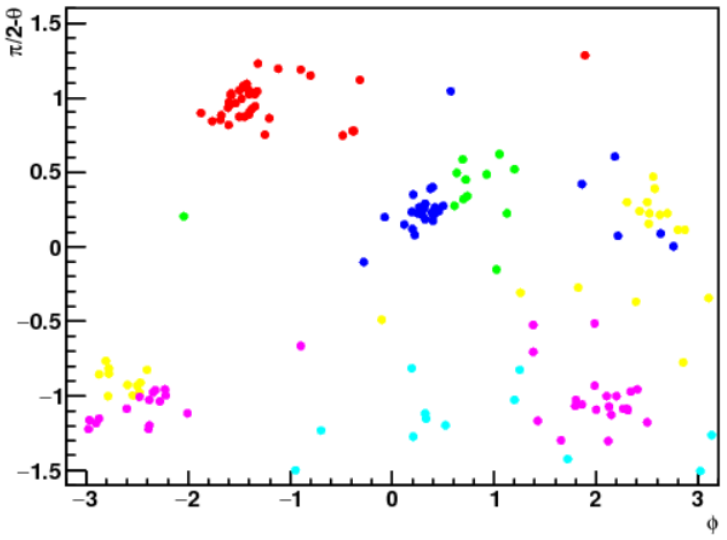
Answer



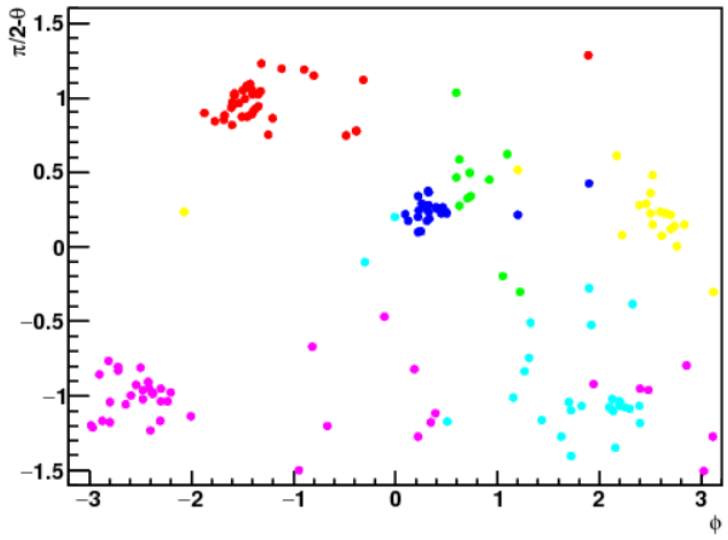
Examples(bad?)

- Event 1:

Estimation

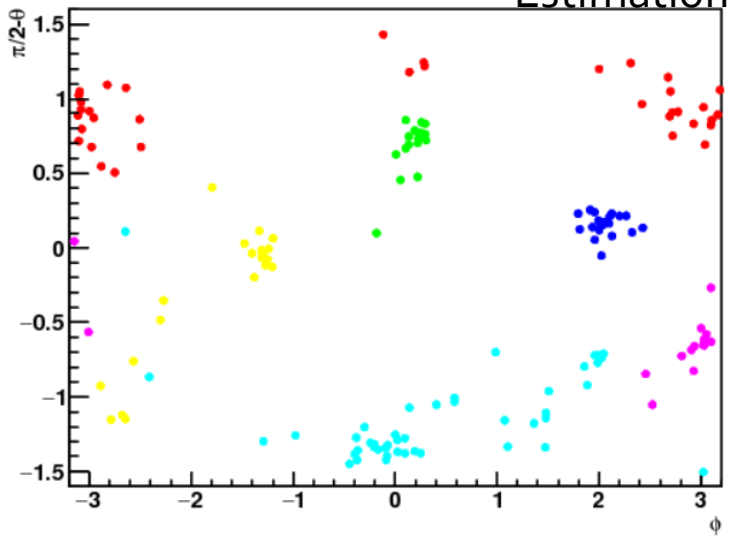


Answer

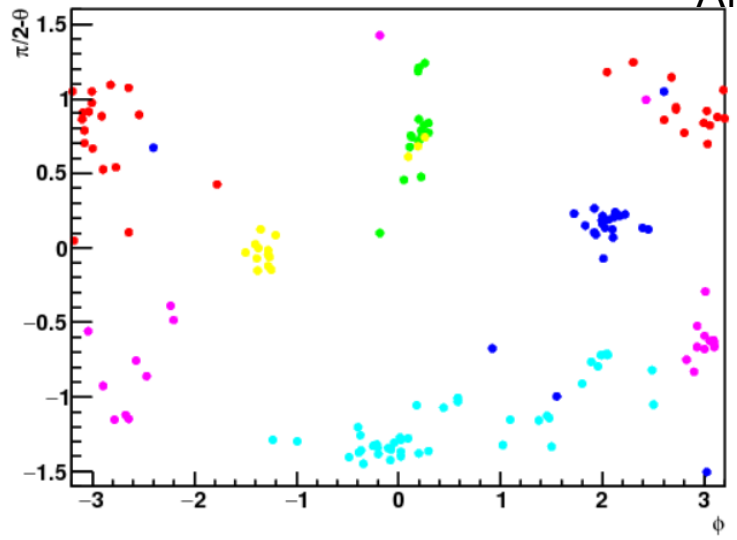


- Event 2:

Estimation

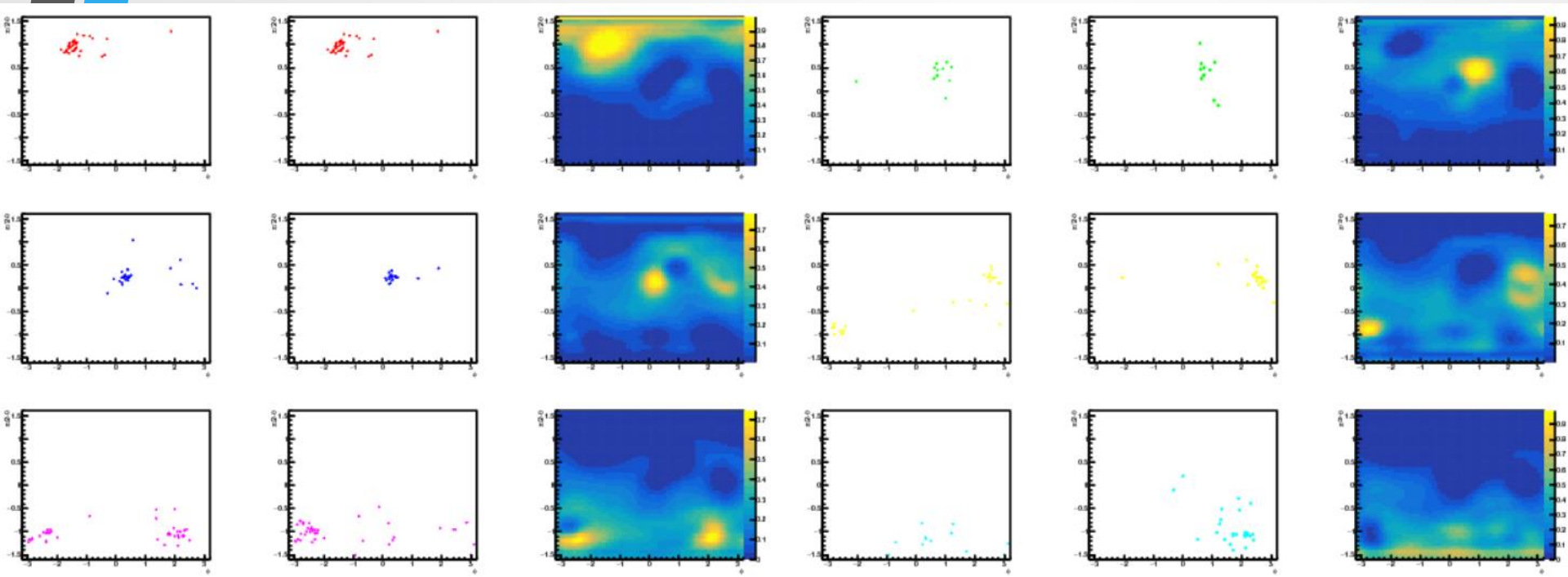


Answer



Prospects

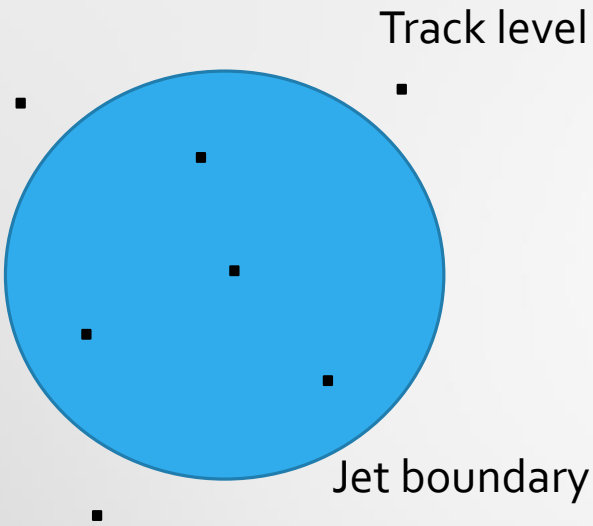
- From semantic segmentation point of view, we can estimate probability of each point
  - Can create heat map for each jet

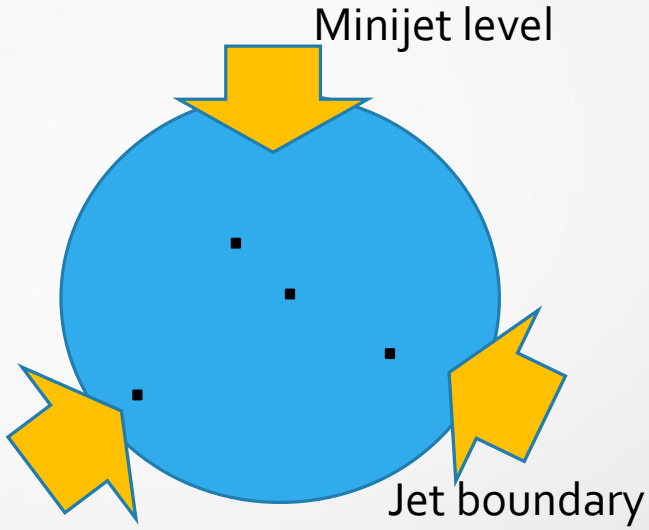


- So we can estimate the probability of minijet using these heat map
  - I think(of course?), some physics effect is necessary to obtain excellent efficiency

Prospects

- It is expected that each minijet move inside the jet boundary which is estimated from CNN





- From Deep learning side, there are some points for improvement
  - Optimize network and hyper parameters
  - Using(adding?) Conditional Random Field
    - 1-2% improvement will be seen in other tasks
    - But now I'm trying to understand
- Can find color flow? → future plan