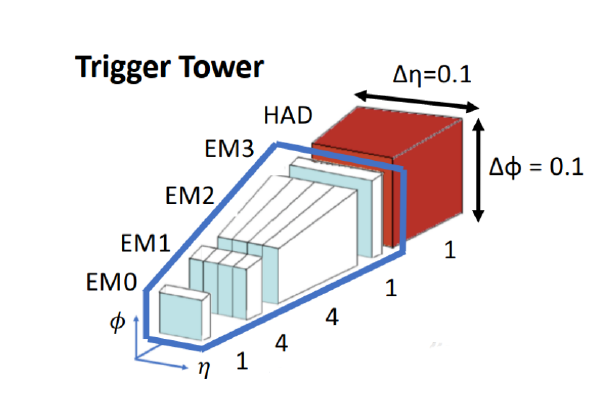
The data we are working on is coming from the sensors of the ATLAS trigger system.

There are 5 sensors and the sensors’ tower looks like this:

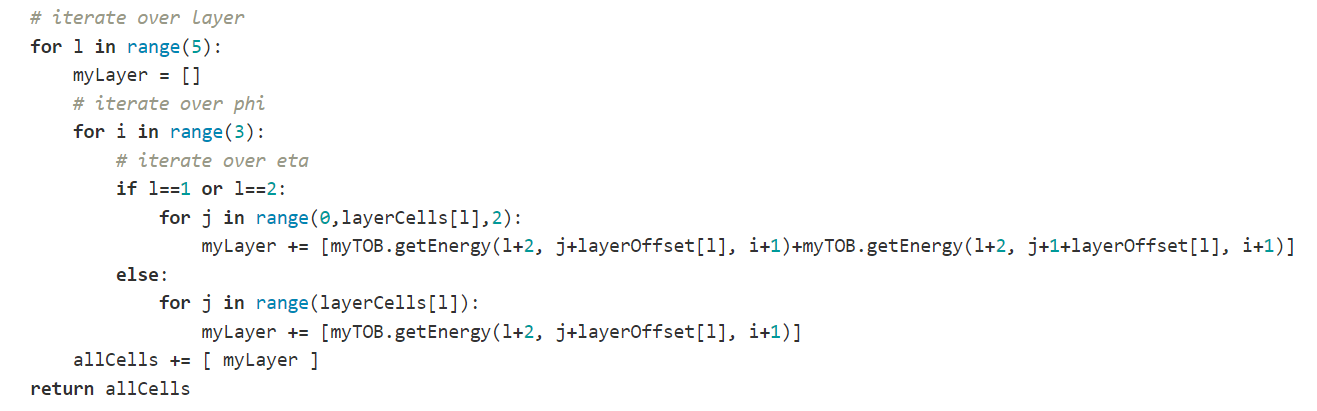


EM0, EM3 and HAD layers are coarse and have 3x3 “pixels”.   
EM1, EM2 are finer granularity and have 12x3 “pixels” (energy deposits).

So in total we have 3x3x3 + 12x3x2 energy deposits. The concatenation of all the cells gives us a vector of 99 values - this is what we have in the CSV files.

In addition we have a label for every row, which signals if the current “snapshot” of an event is an actual tau-tau decay (1) or just a background (0).

The flattening of the tower is done using this algorithm:



Starting from the EM0 to the HAD layer, iterating over eta and phi with “row by row” order.

This means that the backward reconstruction should be according to the following mapping:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C0 | C1 | C2 | C3 | ... |  |  |  |  |  |  |  |  |  |
| EM0  (0,0) | EM0  (0,1) | EM0 (0,2) | EM0 (1,0) | ... |  |  |  |  |  |  |  |  |  |

EM0 Layer:

|  |  |  |
| --- | --- | --- |
| C0 | C1 | C2 |
| C3 | C4 | C5 |
| C6 | C7 | C8 |

EM1 Layer:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C9 | C10 | C11 | C12 | ... | ... | ... | ... | ... | ... | ... | ... |
| C21 | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| C33 | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

We can manipulate this data in multiple ways. For example we can create a fully connected net, so we can just take the vector of 99 cells and “feed” it to the NN.

We can try to use it in CNN and create a 5 channel “picture” similar to what we have with 3 channel RGB pictures.

But here we will have the problem of different channels having different dimensions. In this case we can pad with zeros the EM0, EM3 and HAD layers:

**EM0**:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | C0 | C1 | C2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | C3 | C4 | C5 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | C6 | C7 | C8 | 0 | 0 | 0 | 0 |

Or we can oversample these layers and preserve the spatial relations between the layers.

**EM0**:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C0 | C0 | C0 | C0 | C1 | C1 | C1 | C1 | C2 | C2 | C2 | C2 |
| C3 | C3 | C3 | C3 | C4 | C4 | C4 | C4 | C5 | C5 | C5 | C5 |
| C6 | C6 | C6 | C6 | C7 | C7 | C7 | C7 | C8 | C8 | C8 | C8 |

Both can work and we can apply a kernel of dimension HxWx5