digitizationpp: status

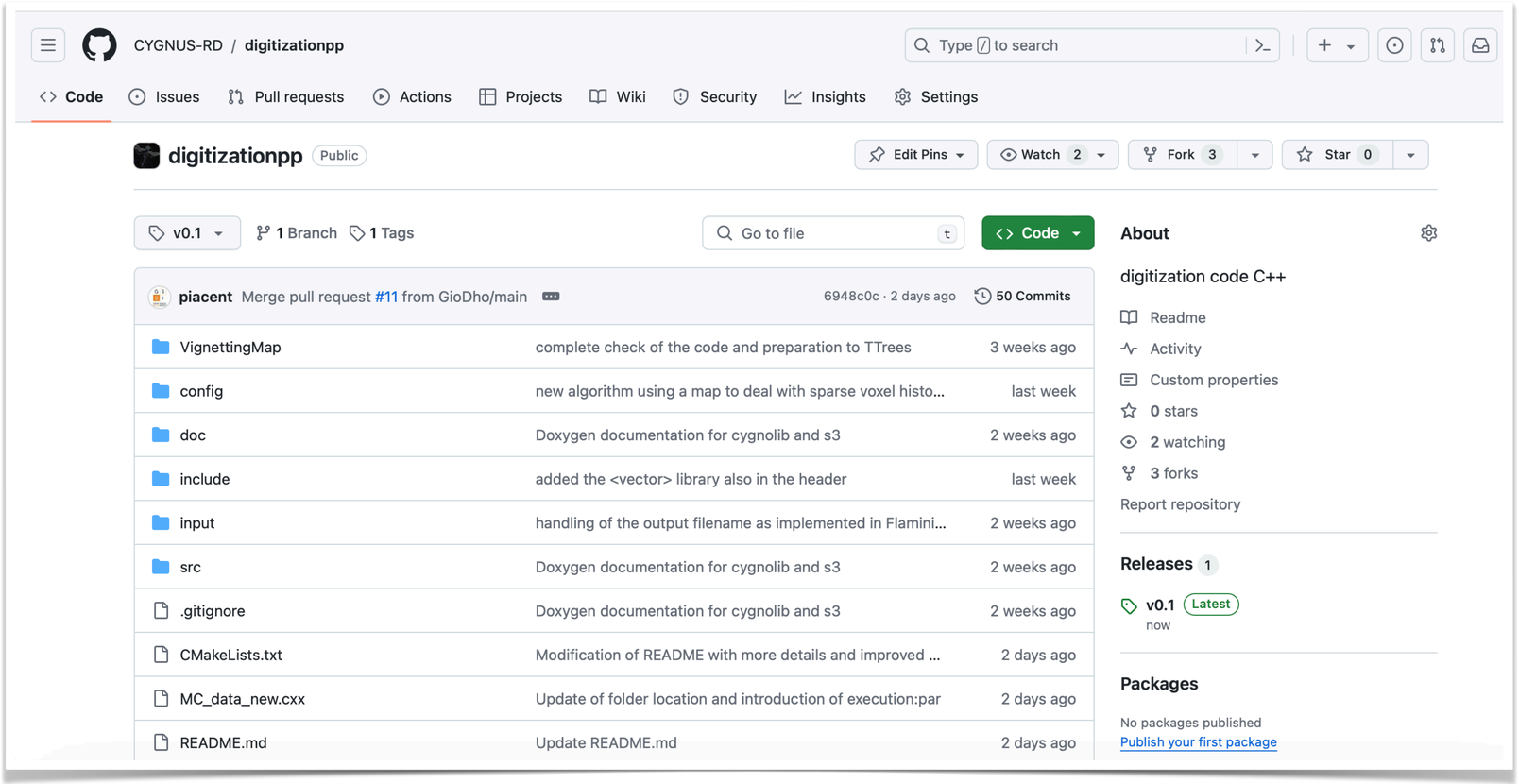
S. Piacentini, G. Dho

Simulation Meeting - 03/06/2024

The code is complete

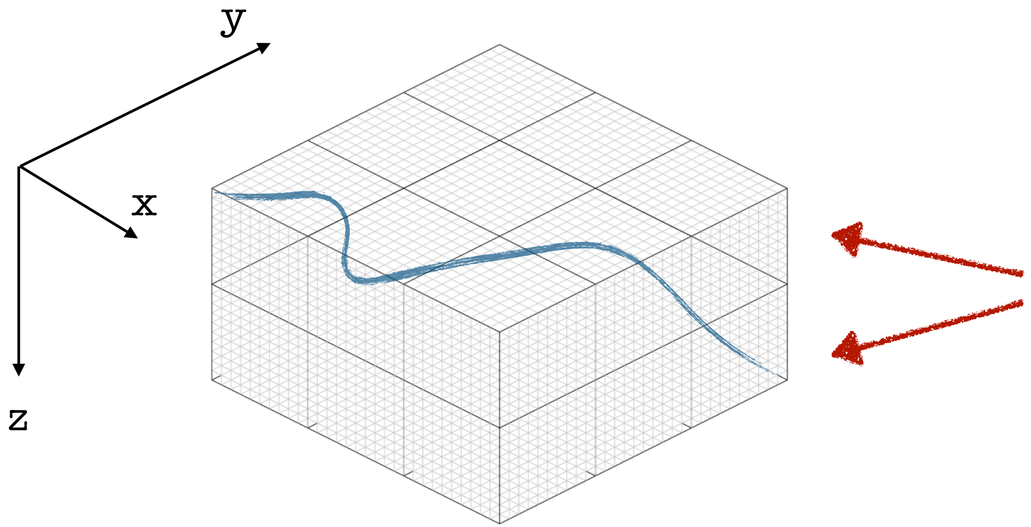
- The code is complete and available new repository: CYGNUS-RD/digitizationpp

- The current version of the code is tagged as “v0.1” and it’s the almost exact transcription of the digitization code (https://github.com/flaminiadigiamba/digitization), with the addition of the **new map algorithm** in case of long tracks.



Traditional “voxelization” algorithm

- Geant4 hits → GEM1 gain → GEM2 gain → diffusion smearing
  - ➡ After the smearing we have (x, y, z) position of **each electron after GEM2**
- Diffusion smearing → **Voxelization** → Voxel-by-Voxel GEM3 gain (with saturation)
  - ➡ Voxelization is the most expensive part:
    - ➡ CPU consumption (unavoidable)
    - ➡ memory usage: 1 integer per voxel, typical voxel sizes are
      0.150 mm × 0.150 mm × 0.1 mm ← if track is too long, voxels divided in sub-regions, at the price of CPU usage

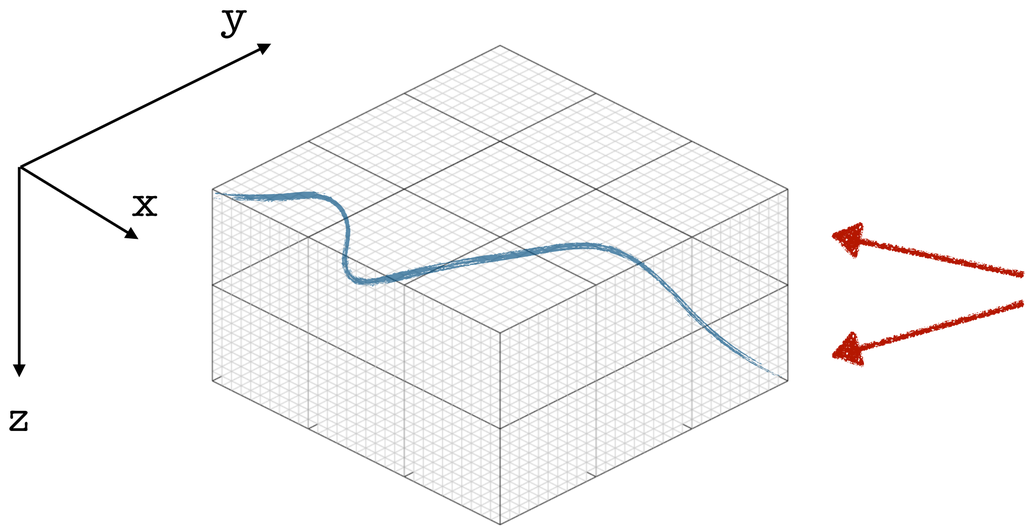


Voxels sub-regions

In other words voxels are a **3D histogram**!

When track is long enough, more than 99% of the bins (aka voxels) are equal to 0!

The new map algorithm



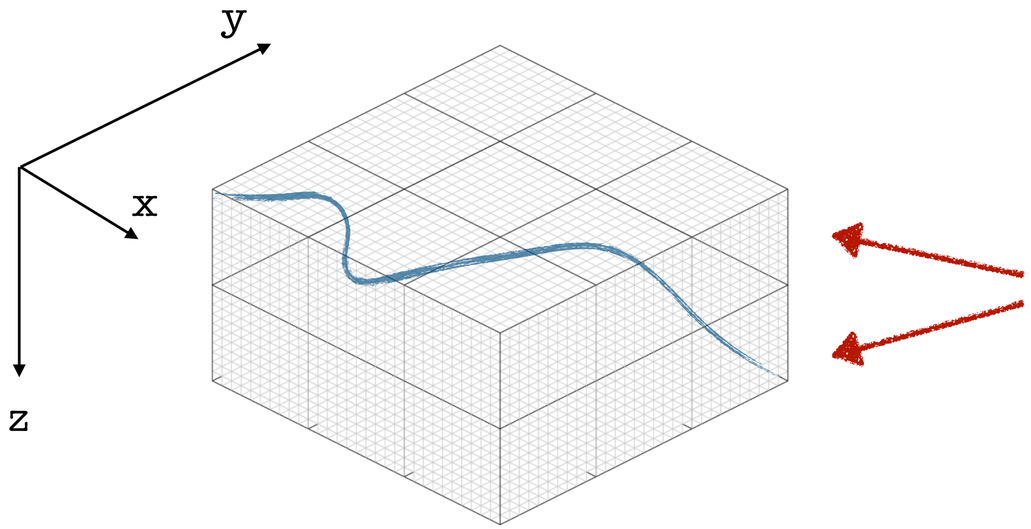
Voxels sub-regions

In other words voxels are a **3D histogram**!

When track is long enough, more than 99% of the bins (aka voxels) are equal to 0!

- Voxels: 3D tensor with almost all bins equal to 0. This is what is defined as a **sparse** tensor.
- A lot of literature about sparse structures:
  - Usually implemented as a **map** where the **key is the coordinate** and the **value is the bin content**
  - Most appropriate structure if you have to do other operations on the tensor (e.g. tensor-by-tensor multiplication etc.) is the boost::unordered_map

The new map algorithm



Voxels sub-regions

In other words voxels are a **3D histogram**!

When track is long enough, more than 99% of the bins (aka voxels) are equal to 0!

- Voxels: 3D tensor with almost all bins equal to 0. This is what is defined as a **sparse** tensor.
- A lot of literature about sparse structures:
  - Usually implemented as a **map** where the **key is the coordinate** and the **value is the bin content**
  - Most appropriate structure if you have to do other operations on the tensor (e.g. tensor-by-tensor multiplication etc.) is the boost::unordered_map

Sparse-ness of traditional voxel tensor

- For the tests I used **https://github.com/CYGNUS-RD/digitizationpp/blob/main/input/LIME_CADshield_6Cu_210Bi_part0.root** that contains 9 long and short tracks.

- Sparse-ness levels for these tracks:

```
Amplifying voxel region z=[-60.482,-13.4057] 0/1
Sparse-ness of voxel region: 0.620984 %
Amplifying voxel region z=[-13.4057,9.91016] 1/1
Sparse-ness of voxel region: 0.12549 %
```

```
Amplifying voxel region z=[156.17,257.345] 0/0
Sparse-ness of voxel region: 0.131361 %
```

```
Amplifying voxel region z=[-232.945,-84.5695] 0/0
Sparse-ness of voxel region: 1.5666 %
```

```
Amplifying voxel region z=[107.85,899.422] 0/0
Sparse-ness of voxel region: 0.137454 %
```

```
Amplifying voxel region z=[-215.064,-30.8756] 0/0
Sparse-ness of voxel region: 1.1785 %
```

```
Amplifying voxel region z=[-51.3594,-19.7318] 0/0
Sparse-ness of voxel region: 0.559226 %
```

```
Amplifying voxel region z=[-195.349,-148.496] 0/3
Sparse-ness of voxel region: 0.246952 %
Amplifying voxel region z=[-148.496,-101.643] 1/3
Sparse-ness of voxel region: 0.174277 %
Amplifying voxel region z=[-101.643,-54.7903] 2/3
Sparse-ness of voxel region: 0.347762 %
Amplifying voxel region z=[-54.7903,-42.2128] 3/3
Sparse-ness of voxel region: 0.600658 %
```

```
Amplifying voxel region z=[52.7753,59.9897] 0/26
Sparse-ness of voxel region: 0.0349279 %
Amplifying voxel region z=[59.9897,67.2041] 1/26
Sparse-ness of voxel region: 0.00139291 %
Amplifying voxel region z=[103.276,110.49] 7/26
Sparse-ness of voxel region: 0.00395583 %
Amplifying voxel region z=[110.49,117.705] 8/26
Sparse-ness of voxel region: 0.0621038 %
Amplifying voxel region z=[117.705,124.919] 9/26
Sparse-ness of voxel region: 0.028816 %
Amplifying voxel region z=[124.919,132.134] 10/26
Sparse-ness of voxel region: 0.00821146 %
Amplifying voxel region z=[146.562,153.777] 13/26
Sparse-ness of voxel region: 0.0108552 %
Amplifying voxel region z=[153.777,160.991] 14/26
Sparse-ness of voxel region: 0.00806448 %
Amplifying voxel region z=[160.991,168.206] 15/26
Sparse-ness of voxel region: 0.011482 %
Amplifying voxel region z=[197.063,204.277] 20/26
Sparse-ness of voxel region: 0.00844118 %
Amplifying voxel region z=[211.492,218.706] 22/26
Sparse-ness of voxel region: 0.00296492 %
Amplifying voxel region z=[218.706,225.921] 23/26
Sparse-ness of voxel region: 0.00273326 %
Amplifying voxel region z=[225.921,233.135] 24/26
Sparse-ness of voxel region: 0.00498469 %
Amplifying voxel region z=[233.135,240.349] 25/26
Sparse-ness of voxel region: 0.0128613 %
Amplifying voxel region z=[240.349,241.487] 26/26
Sparse-ness of voxel region: 0.0004742 %
```

```
size = 22925188
Amplifying voxel region z=[-172.097,-157.618] 0/28
Sparse-ness of voxel region: 0.107559 %
Amplifying voxel region z=[-157.618,-143.139] 1/28
Sparse-ness of voxel region: 0.040935 %
Amplifying voxel region z=[-143.139,-128.66] 2/28
Sparse-ness of voxel region: 0.0330321 %
Amplifying voxel region z=[-128.66,-114.181] 3/28
Sparse-ness of voxel region: 0.033211 %
Amplifying voxel region z=[-114.181,-99.7022] 4/28
Sparse-ness of voxel region: 0.0275708 %
Amplifying voxel region z=[-99.7022,-85.2231] 5/28
Sparse-ness of voxel region: 0.00221025 %
Amplifying voxel region z=[-85.2231,-70.7441] 6/28
Sparse-ness of voxel region: 0.0337033 %
Amplifying voxel region z=[-70.7441,-56.2651] 7/28
Sparse-ness of voxel region: 0.031156 %
Amplifying voxel region z=[-56.2651,-41.7861] 8/28
Sparse-ness of voxel region: 0.0429288 %
Amplifying voxel region z=[-41.7861,-27.3071] 9/28
Sparse-ness of voxel region: 1.97697e-06 %
Amplifying voxel region z=[-12.8281,1.6509] 11/28
Sparse-ness of voxel region: 0.00331636 %
Amplifying voxel region z=[1.6509,16.1299] 12/28
Sparse-ness of voxel region: 0.0267266 %
Amplifying voxel region z=[16.1299,30.6089] 13/28
Sparse-ness of voxel region: 0.0368961 %
Amplifying voxel region z=[45.0879,59.5669] 15/28
Sparse-ness of voxel region: 0.0329461 %
Amplifying voxel region z=[59.5669,74.0459] 16/28
Sparse-ness of voxel region: 0.0118311 %
Amplifying voxel region z=[74.0459,88.525] 17/28
Sparse-ness of voxel region: 1.68042e-05 %
Amplifying voxel region z=[88.525,103.004] 18/28
Sparse-ness of voxel region: 0.0290733 %
Amplifying voxel region z=[117.483,131.962] 20/28
Sparse-ness of voxel region: 0.0162368 %
Amplifying voxel region z=[131.962,146.441] 21/28
Sparse-ness of voxel region: 0.000310384 %
Amplifying voxel region z=[146.441,160.92] 22/28
Sparse-ness of voxel region: 0.0200573 %
Amplifying voxel region z=[175.399,189.878] 24/28
Sparse-ness of voxel region: 0.009956 %
Amplifying voxel region z=[189.878,204.357] 25/28
Sparse-ness of voxel region: 0.01579 %
Amplifying voxel region z=[204.357,218.836] 26/28
Sparse-ness of voxel region: 2.96545e-06 %
Amplifying voxel region z=[218.836,233.315] 27/28
Sparse-ness of voxel region: 0.0062136 %
Amplifying voxel region z=[233.315,242.729] 28/28
Sparse-ness of voxel region: 0.0143958 %
```

Sparse-ness of traditional voxel tensor

- For the tests I used **http**[illegible]**itizationpp/blob/main/input/LIME_CADshield**[illegible]ains 9 long and short tracks.

Longer tracks are, as expected, sparser!

- Sparse-ness levels for these tracks:

```
Amplifying voxel region z=[-60.482,-13.4057] 0/1
Sparse-ness of voxel region: 0.620984 %
Amplifying voxel region z=[-13.4057,9.91016] 1/1
Sparse-ness of voxel region: 0.12549 %
```

```
Amplifying voxel region z=[156.17,257.345] 0/0
Sparse-ness of voxel region: 0.131361 %
```

```
Amplifying voxel region z=[-232.945,-84.5695] 0/0
Sparse-ness of voxel region: 1.5666 %
```

```
Amplifying voxel region z=[107.85,899.422] 0/0
Sparse-ness of voxel region: 0.137454 %
```

```
Amplifying voxel region z=[-215.064,-30.8756] 0/0
Sparse-ness of voxel region: 1.1785 %
```

```
Amplifying voxel region z=[-51.3594,-19.7318] 0/0
Sparse-ness of voxel region: 0.559226 %
```

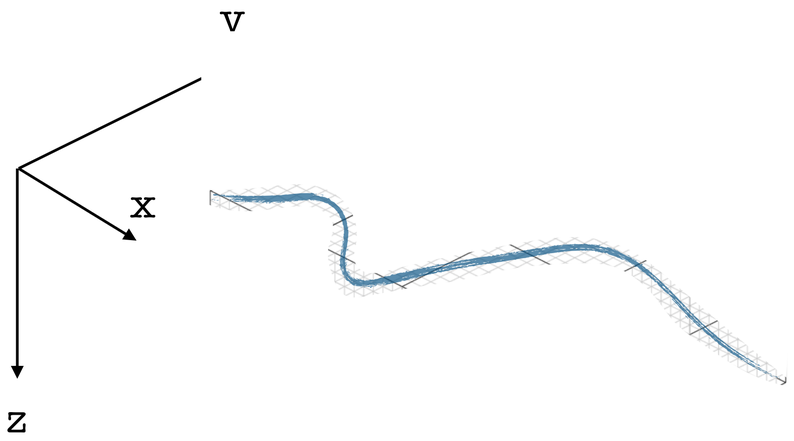
```
Amplifying voxel region z=[-195.349,-148.496] 0/3
Sparse-ness of voxel region: 0.246952 %
Amplifying voxel region z=[-148.496,-101.643] 1/3
Sparse-ness of voxel region: 0.174277 %
Amplifying voxel region z=[-101.643,-54.7903] 2/3
Sparse-ness of voxel region: 0.347762 %
Amplifying voxel region z=[-54.7903,-42.2128] 3/3
Sparse-ness of voxel region: 0.600658 %
```

```
Amplifying voxel region z=[52.7753,59.9897] 0/26
Sparse-ness of voxel region: 0.0349279 %
Amplifying voxel region z=[59.9897,67.2041] 1/26
Sparse-ness of voxel region: 0.00139291 %
Amplifying voxel region z=[103.276,110.49] 7/26
Sparse-ness of voxel region: 0.00395583 %
Amplifying voxel region z=[110.49,117.705] 8/26
Sparse-ness of voxel region: 0.0621038 %
Amplifying voxel region z=[117.705,124.919] 9/26
Sparse-ness of voxel region: 0.028816 %
Amplifying voxel region z=[124.919,132.134] 10/26
Sparse-ness of voxel region: 0.00821146 %
Amplifying voxel region z=[146.562,153.777] 13/26
Sparse-ness of voxel region: 0.0108552 %
Amplifying voxel region z=[153.777,160.991] 14/26
Sparse-ness of voxel region: 0.00806448 %
Amplifying voxel region z=[160.991,168.206] 15/26
Sparse-ness of voxel region: 0.011482 %
Amplifying voxel region z=[197.063,204.277] 20/26
Sparse-ness of voxel region: 0.00844118 %
Amplifying voxel region z=[211.492,218.706] 22/26
Sparse-ness of voxel region: 0.00296492 %
Amplifying voxel region z=[218.706,225.921] 23/26
Sparse-ness of voxel region: 0.00273326 %
Amplifying voxel region z=[225.921,233.135] 24/26
Sparse-ness of voxel region: 0.00498469 %
Amplifying voxel region z=[233.135,240.349] 25/26
Sparse-ness of voxel region: 0.0128613 %
Amplifying voxel region z=[240.349,241.487] 26/26
Sparse-ness of voxel region: 0.0004742 %
```

```
size = 22925188
Amplifying voxel region z=[-172.097,-157.618] 0/28
Sparse-ness of voxel region: 0.107559 %
Amplifying voxel region z=[-157.618,-143.139] 1/28
Sparse-ness of voxel region: 0.040935 %
Amplifying voxel region z=[-143.139,-128.66] 2/28
Sparse-ness of voxel region: 0.0330321 %
Amplifying voxel region z=[-128.66,-114.181] 3/28
Sparse-ness of voxel region: 0.033211 %
Amplifying voxel region z=[-114.181,-99.7022] 4/28
Sparse-ness of voxel region: 0.0275708 %
Amplifying voxel region z=[-99.7022,-85.2231] 5/28
Sparse-ness of voxel region: 0.00221025 %
Amplifying voxel region z=[-85.2231,-70.7441] 6/28
Sparse-ness of voxel region: 0.0337033 %
Amplifying voxel region z=[-70.7441,-56.2651] 7/28
Sparse-ness of voxel region: 0.031156 %
Amplifying voxel region z=[-56.2651,-41.7861] 8/28
Sparse-ness of voxel region: 0.0429288 %
Amplifying voxel region z=[-41.7861,-27.3071] 9/28
Sparse-ness of voxel region: 1.97697e-06 %
Amplifying voxel region z=[-12.8281,1.6509] 11/28
Sparse-ness of voxel region: 0.00331636 %
Amplifying voxel region z=[1.6509,16.1299] 12/28
Sparse-ness of voxel region: 0.0267266 %
Amplifying voxel region z=[16.1299,30.6089] 13/28
Sparse-ness of voxel region: 0.0368961 %
Amplifying voxel region z=[45.0879,59.5669] 15/28
Sparse-ness of voxel region: 0.0329461 %
Amplifying voxel region z=[59.5669,74.0459] 16/28
Sparse-ness of voxel region: 0.0118311 %
Amplifying voxel region z=[74.0459,88.525] 17/28
Sparse-ness of voxel region: 1.68042e-05 %
Amplifying voxel region z=[88.525,103.004] 18/28
Sparse-ness of voxel region: 0.0290733 %
Amplifying voxel region z=[117.483,131.962] 20/28
Sparse-ness of voxel region: 0.0162368 %
Amplifying voxel region z=[131.962,146.441] 21/28
Sparse-ness of voxel region: 0.000310384 %
Amplifying voxel region z=[146.441,160.92] 22/28
Sparse-ness of voxel region: 0.0200573 %
Amplifying voxel region z=[175.399,189.878] 24/28
Sparse-ness of voxel region: 0.009956 %
Amplifying voxel region z=[189.878,204.357] 25/28
Sparse-ness of voxel region: 0.01579 %
Amplifying voxel region z=[204.357,218.836] 26/28
Sparse-ness of voxel region: 2.96545e-06 %
Amplifying voxel region z=[218.836,233.315] 27/28
Sparse-ness of voxel region: 0.0062136 %
Amplifying voxel region z=[233.315,242.729] 28/28
Sparse-ness of voxel region: 0.0143958 %
```

The new map algorithm

- Usually implemented as a **map** where the **key is the coordinate** and the **value is the bin content**



key	value
$\text{index}(\overrightarrow{bin_0})$	$\text{content}(\overrightarrow{bin_0})$
$\text{index}(\overrightarrow{bin_1})$	$\text{content}(\overrightarrow{bin_1})$
$\text{index}(\overrightarrow{bin_2})$	$\text{content}(\overrightarrow{bin_2})$
$\text{index}(\overrightarrow{bin_3})$	$\text{content}(\overrightarrow{bin_3})$

- There's a gain in performance (CPU and memory) only if the track is sufficiently long

Optimization still to be found:

```
1367          // If voxel regions <= 2 -> use the standard algorithm, otherwise use the map algorithm
1368          bool map_algorithm = true;
1369          if( x_n_bin * y_n_bin * z_n_bin_MAX < 2*max_3Dhisto_volume) map_algorithm = false;
```

digitizationpp: performance

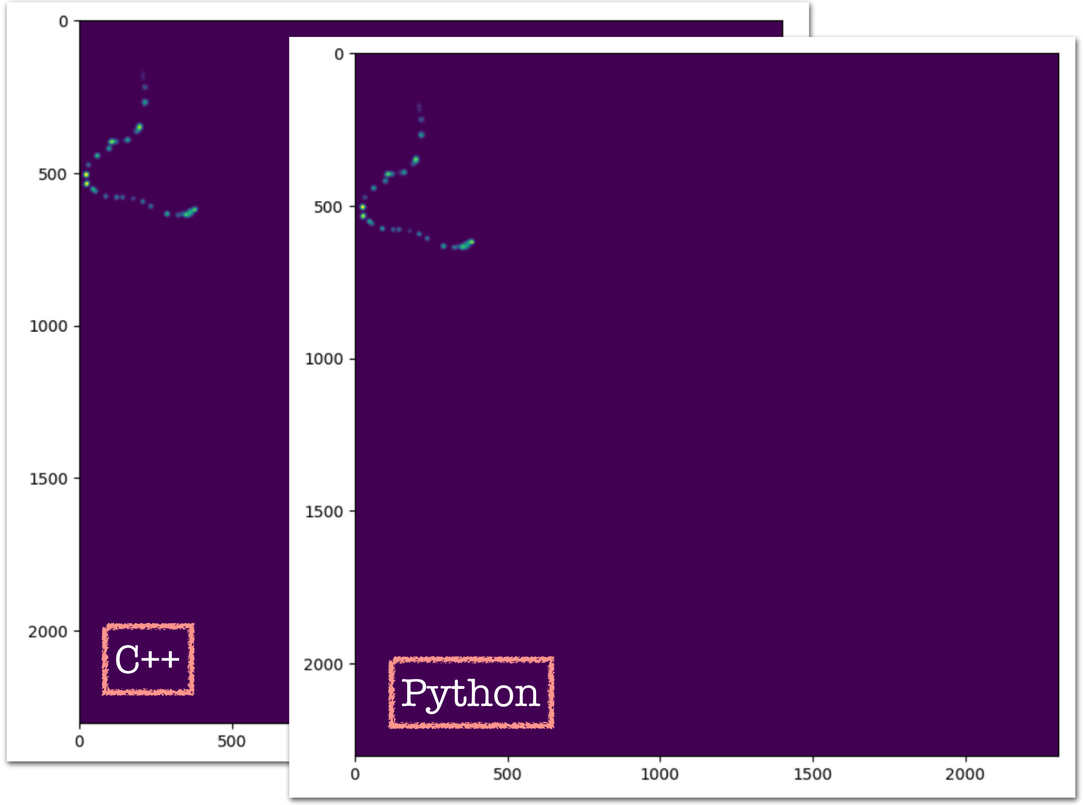
- For the tests I used **https://github.com/CYGNUS-RD/digitizationpp/blob/main/input/LIME_CADshield_6Cu_210Bi_part0.root** that contains 9 long and short tracks.

- Total time to digitize 9 tracks:

  ➡ Traditional voxels, Python: 178 s

  ➡ Traditional voxels, C++: 25 s

  ➡ New map algorithm on long tracks, C++: 19 s



Conclusions

- The code is completed and it's ready to be intensively tested on simulations, also of NR tracks
- There's still room for optimization, any volunteer is welcome
- When you'll use it, please, **check the configuration** file values for diffusion, saturation, etc. etc.

Fully digitized track