MAX-PLANCK-INSTITUT FÜR KERNPHYSIK

Convolutional neural networks for S2 position reconstruction

- Applying CNNs to HEX arrays

Andrii Terliuk
(MPI for Nuclear Physics/Heidelberg University)

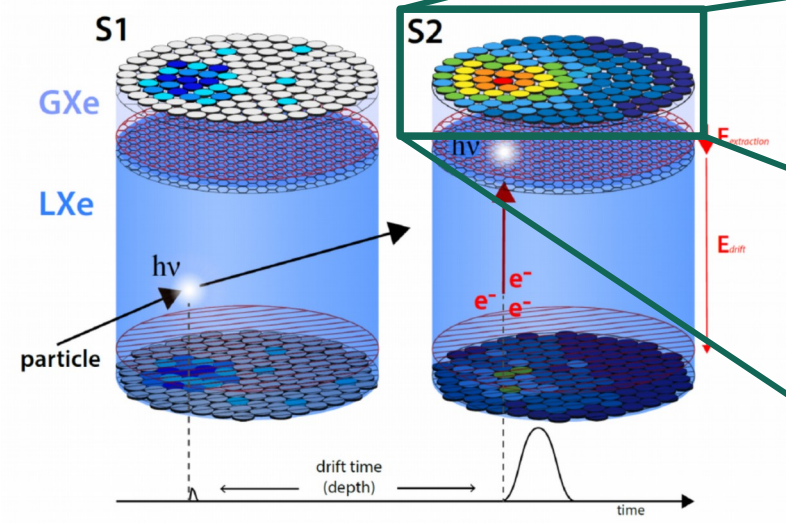
DANCE ML workshop
05 August 2020

MAX-PLANCK-INSTITUT FÜR KERNPHYSIK HEIDELBERG

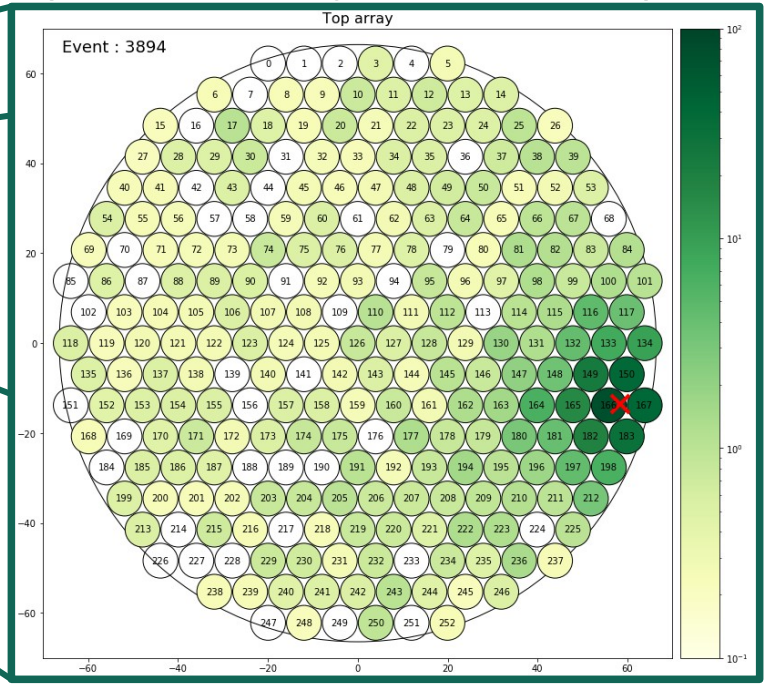
UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

What is S2 position reconstruction?

- S2 reconstruction – critical task for every dual-phase TPC:
  - 3D position reconstruction
  - Background rejection
  - Possibility of larger fiducial volume
  - Event quality assurance

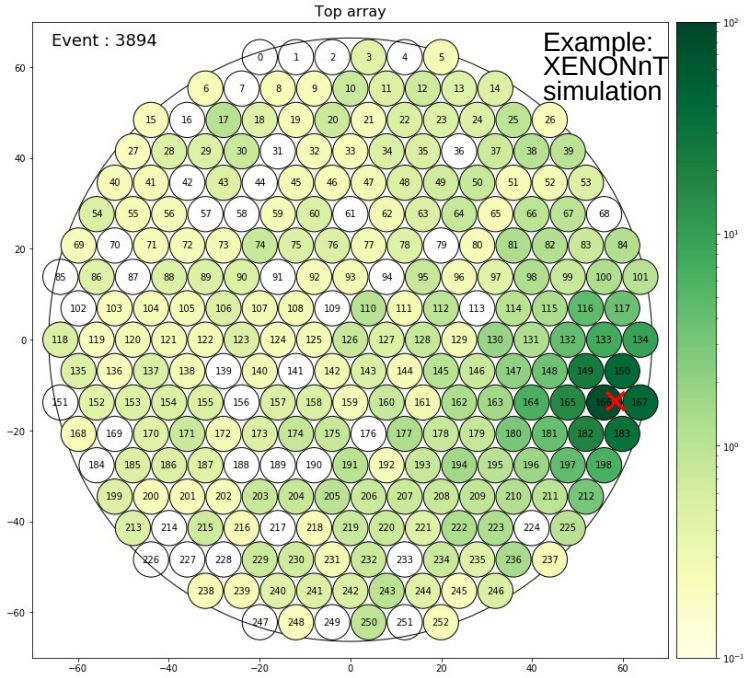


Typical top array
(simulated S2 signal in XENONnT)



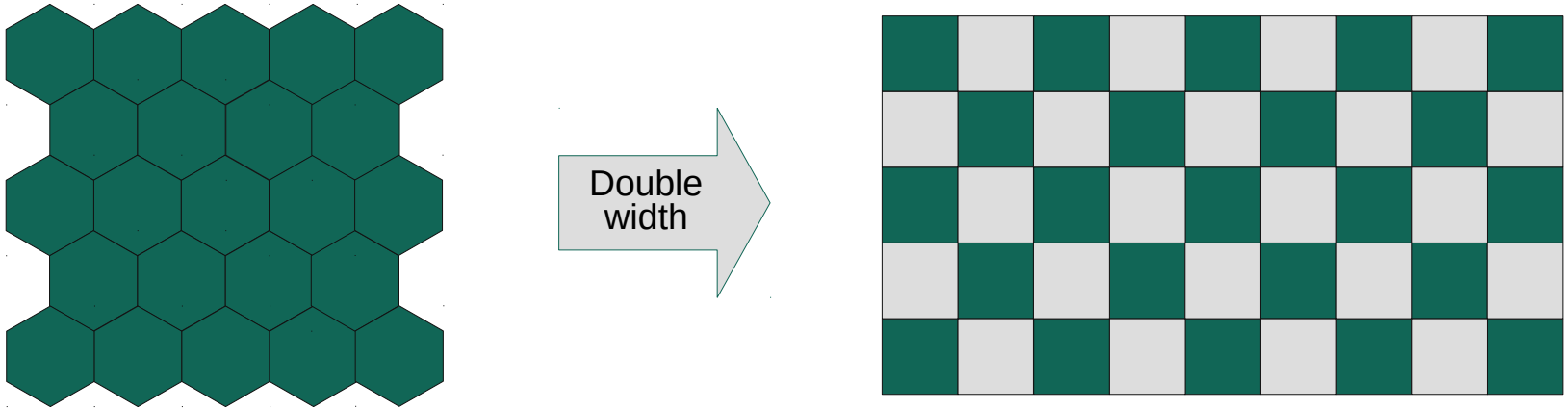
S2 position reconstruction

- Reconstruction with neural networks is very fast and precise
- Complementarity to other reconstruction algorithms for cross-checks
- PMT arrays provide essentially a 2D image
  - Can apply convolutional neural networks
  - Position reconstruction – classical regression problem
- Caveat:
  - PMT arrangement often forms hexagonal bins → most efficient PMT coverage
  - Linear algebra → square matrices



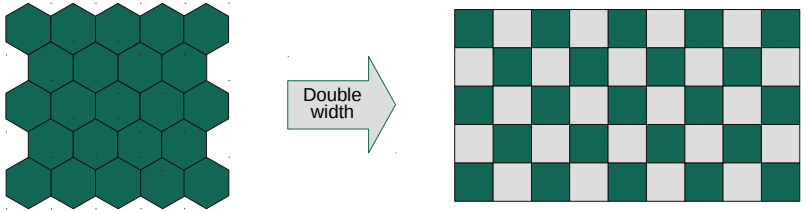
Need a trick to transform PMT arrangement for CNNs

HEX to lattice transformation

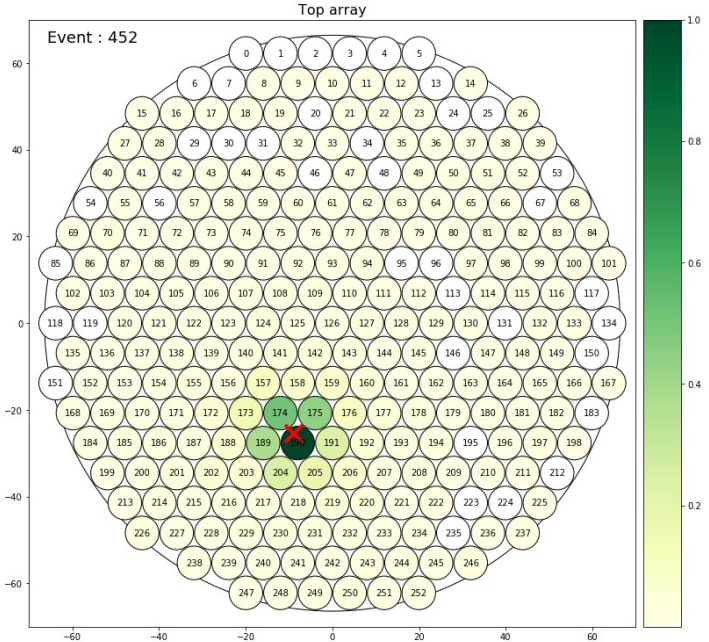


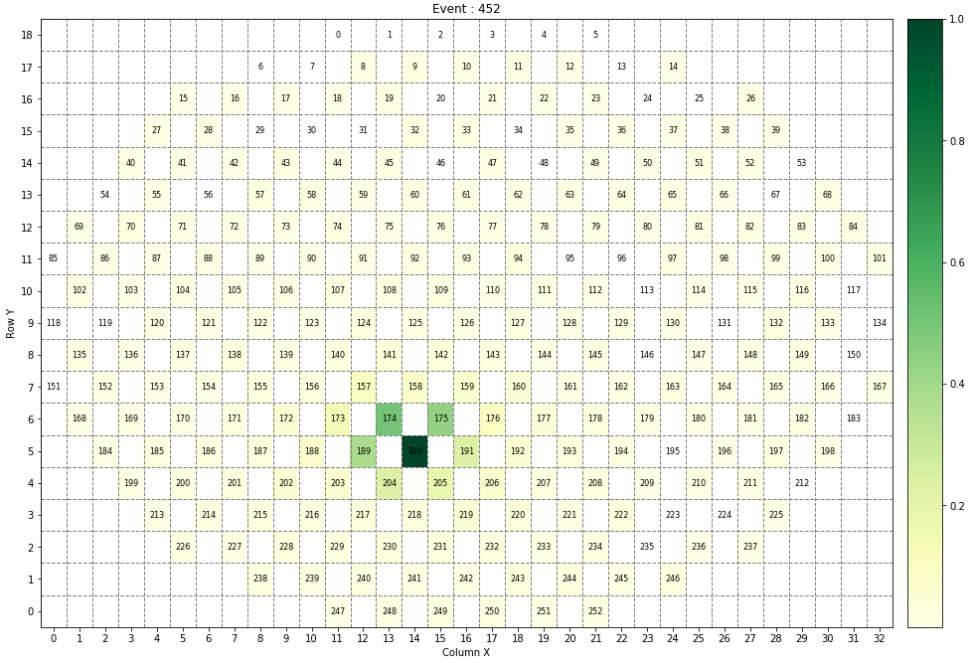
- Possible option for HEX-like PMT geometry using double-width transformation
  - Empty “pixels” are introduced between the PMTs → double the image width
  - Pros: uniformity of kernels across the board
  - Contra: images are larger in memory

Application to XENONnT top array



- Transformation of S2 pattern (simulation) in XENONnT





Example of position reconstruction CNN

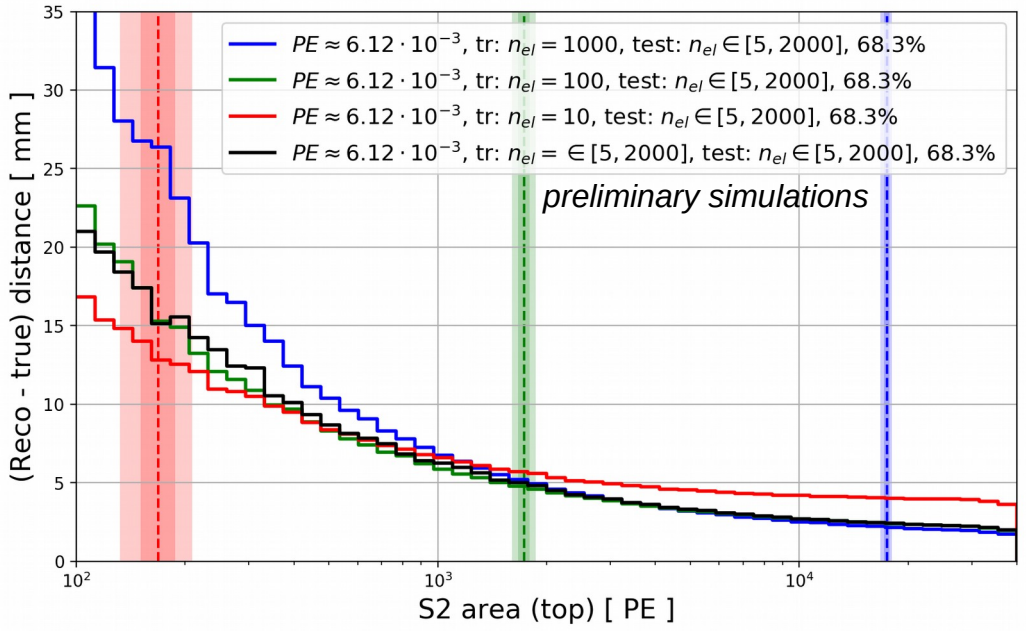
- Using Keras interface of TensorFlow
  - Preliminary network design:
    - Conv2D with size (13,7), 16 kernels (ELU activation)
    - Conv2D with size 5, 32 kernels (ELU activation)
    - Conv2D with size 5, 32 kernels (ELU activation)
    - MaxPool layer with size (3,2)
    - Dense layer with 8 neurons
    - Output layer for (X,Y)
  - Use standard TF layers:
    - Compatible for every machine
    - Fast and easy to use
    - But has extra parameters

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
reshape_1 (Reshape)          (None, 33, 19, 1)         0
_________________________________________________________________
zero_padding2d_3 (ZeroPaddin (None, 45, 25, 1)         0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 33, 19, 16)        1472
_________________________________________________________________
zero_padding2d_4 (ZeroPaddin (None, 37, 23, 16)        0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 33, 19, 32)        12832
_________________________________________________________________
zero_padding2d_5 (ZeroPaddin (None, 37, 23, 32)        0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 33, 19, 32)        25632
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 11, 9, 32)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 3168)              0
_________________________________________________________________
dense_2 (Dense)              (None, 8)                 25352
_________________________________________________________________
dense_3 (Dense)              (None, 2)                 18
=================================================================
Total params: 65,306
Trainable params: 65,306
Non-trainable params: 0
_________________________________________________________________
```

Example of application

- Network are trained on full waveform signal simulations:
  - Includes light propagation and PMT detection effects
  - Will be using the most accurate calibration information
- Preliminary performance:
  - About 15 mm precision for small signals
  - As low as 2 mm for at large signal amplitudes
  - Training can be optimized for a given amplitude range/analysis
  - Will be revised with more calibration data

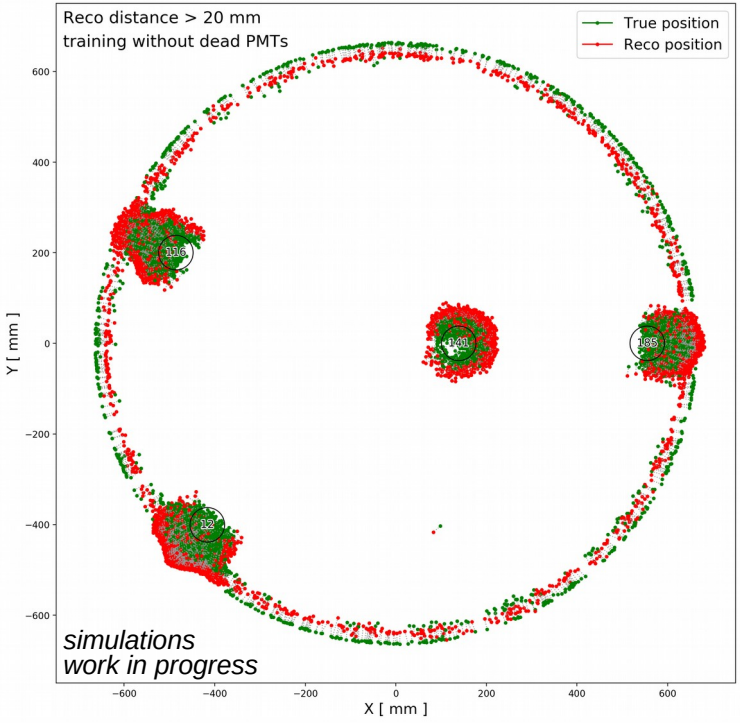
Example of reconstruction performance for simulated point-like S2 patterns

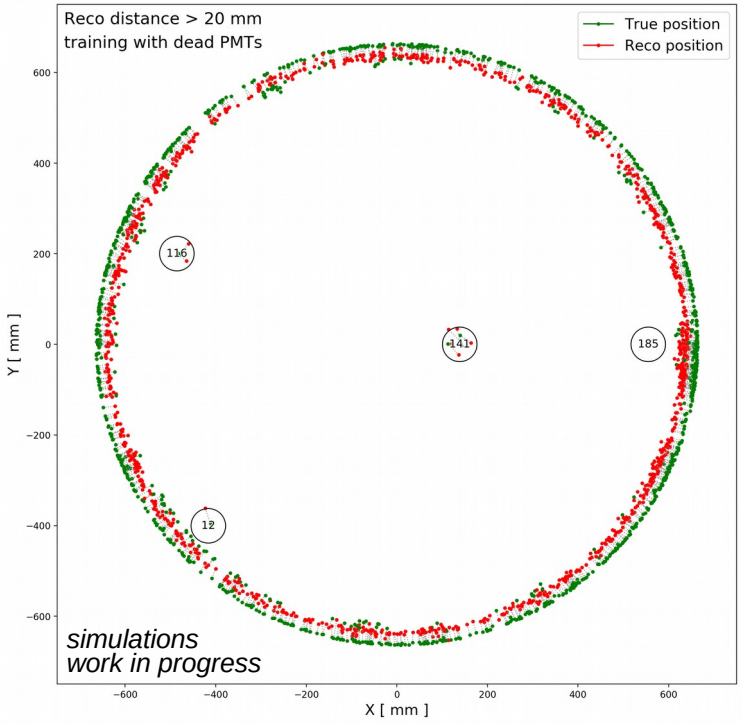


*simulations of electrons at the surface → no diffusion included

Dealing with missing information

- Dealing with missing PMTs
  - Example of toy simulations with 4 “switched off” PMTs
  - Network successfully learns and mitigates negative impact such PMTs



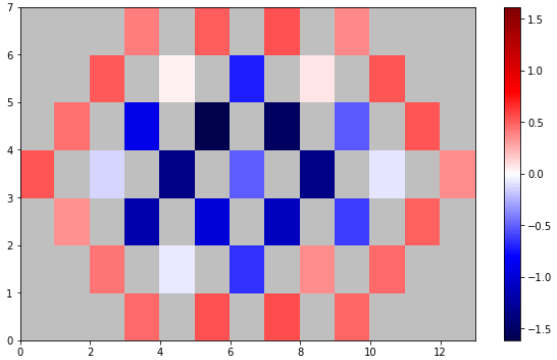


Next steps and summary

- We develop dedicated “HEX” CNN:
  - custom TF HEX-convolution layers
  - reduced number of training parameters
  - “physics” model with potential of P6 rotation symmetry and interpretability

- Summary:
  - Successful application of CNNs to HEX bins using double-width transformation
  - Expected application soon using the data (detector in commissioning phase)
  - Dedicated HEX convolution layers are in development

Example of kernel with R=(3+1)



Example of kernel with R=(2+1)