3^{ieme} Atelier Reprises

ISC-PIF, 27 novembre 2019

G. Grasseau

Retours de CHEP’19, Adelaide, 4-8 novembre

- Les présentations que j’ai appréciées, tendances, non exhaustif
- Deux « tracks » parmi 9 tracks :
  - Track 2 (T2) – Offline Computing
  - Track 6 (T6) - Physics Analysis

Retours de CHEP’19 (1)

Python partout … un très petit aperçu

- Scikit-HEP Project
  https://indico.cern.ch/event/773049/contributions/3476182/attachments/1938227/3213530/EduardoRodrigues_2019-11-05_CHEP2019Adelaide.pdf

- COFFEA - Columnar Object Framework For Effective Analysis
  comment : encore du python, colomnar analisys,, ..., awkward array, parsl, parallélisation en py
  https://indico.cern.ch/event/773049/contributions/3476048/attachments/1937453/3211202/ncsmith-chep2019-coffea.pdf

HPC avec les GPUs vers/au plus près des détecteurs LHCb, ALICE

- Session plénière LHCb (LPNHE) – Superbe !
  https://indico.cern.ch/event/773049/contributions/3474298/attachments/1938619/3213523/vom_Bruch_Allen_chep2019.pdf
  « Buy GPU than expensive network »

- Alice et GPUs
  https://indico.cern.ch/event/773049/contributions/3474317/attachments/1938130/3212553/2019-11-05_CHEP_2019.pdf

- Autres présentations (moins intéressantes ou pas assisté)
  Ray tracing sur GPU (Juno) :

Retours de CHEP’19 (2)

Ordinateurs quantiques

- Session plénière
- *Quantum annealing algorithms for track pattern recognition*
  Impressionnant : description hamiltonienne du pb contruction des doublets, triplets. effet tunnel entre les puits de potentiels ... Whaouu !!!
  Pour les « developpeurs quantique »
  https://indico.cern.ch/event/773049/contributions/3474750/attachments/1931661/3212537/QA_Tracking_CHEP2019_3.pdf

ML/DL : plus de maturité ! critiques intéressantes ! … parfois extrémistes ..

- TRACKML : 3 gagnants du challenges Kaggle sont des algorithmes sans ML/DL
- Graph Neural Network …
  https://indico.cern.ch/event/773049/contributions/3474765/attachments/1937737/3211868/GraphNNDune.pdf
- *Aligning the MATHUSLA Detector Test Stand with Tensor Flow*
- DL for BSM, évènement anormaux, *Agnostic searches for New Physics* ->détection d’anomalies
  https://indico.cern.ch/event/773049/sessions/323861/#20191104
  https://indico.cern.ch/event/773049/contributions/3476055/attachments/1936870/3211995/CHEP_Presentation.pdf

Retours de CHEP'19 (3)

FPGA

- Session parallèle la plus intéressante (GPU/FPGAs)

ExaScale HPC, prace, ExaScale sciences, ExaScale system (Europe légèrement en retard)

- CPU for HPC (Xeon, Epyc, power9, upcoming processor initiative)
  Vectorisation : ISA, Arm Scalable Vector Extension)
- GPUs avenir, compétition AMD et Intel GPUs, mémoire pas de révolution, réseau Dragonfly topo, stockage)
  Passe en revue les Systeme ExaSacle
- Emerging techno FPGA,
- programing models MPI + X (X= OpenACC, OpenMP ???), DSL !

Retours de CHEP’19 (4)

Autres exposés

- Review of High-Quality Pseudo Random Number Generators
  https://indico.cern.ch/event/773049/contributions/3474761/attachments/1937908/3212113/RNG_chep19.pdf
- Sur la vectorisation, notamment sur le KF CMS
- *Reperforming a Nobel Prize discovery on Kubernetes* – tres belle demo deployment sur le cloud Google ou Amazone
- Cyber criminalité : Session plénière, sensibilisation, très bien
- Extreme Compression for Large Scale Data Store
  https://indico.cern.ch/event/773049/contributions/3476150/attachments/1937404/3211119/CHEP2019_DataCompression.pdf
  Using Declarative Languages for Analysis at the LHC
  https://indico.cern.ch/event/773049/contributions/3476174/attachments/1938123/3212535/Declarative.pdf

A deep neural network method for analyzing the CMS High Granularity Calorimeter (HGCal) events

G. Grasseau[1], A. Kumar[2], A. Sartirana[1],
A. Lobanov[1] and F. Beaudette[1],

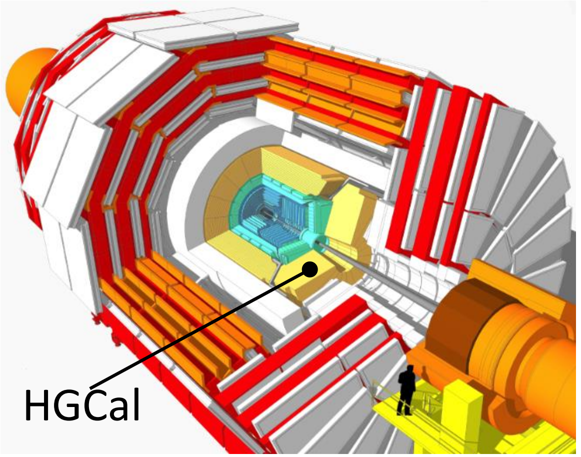
CHEP 2019 Conference, 4-8 November, Adelaide, Australia

[1] Leprince-Ringuet Laboratory (LLR), Ecole Polytechnique, Palaiseau, France
[2] Birla Institute of Technology and Science (BITS), Pilani, India

Motivations: HL@LHC

The High-Luminosity at LHC (HL-LHC) is a major evolution of the accelerator and the CMS detector (2024)

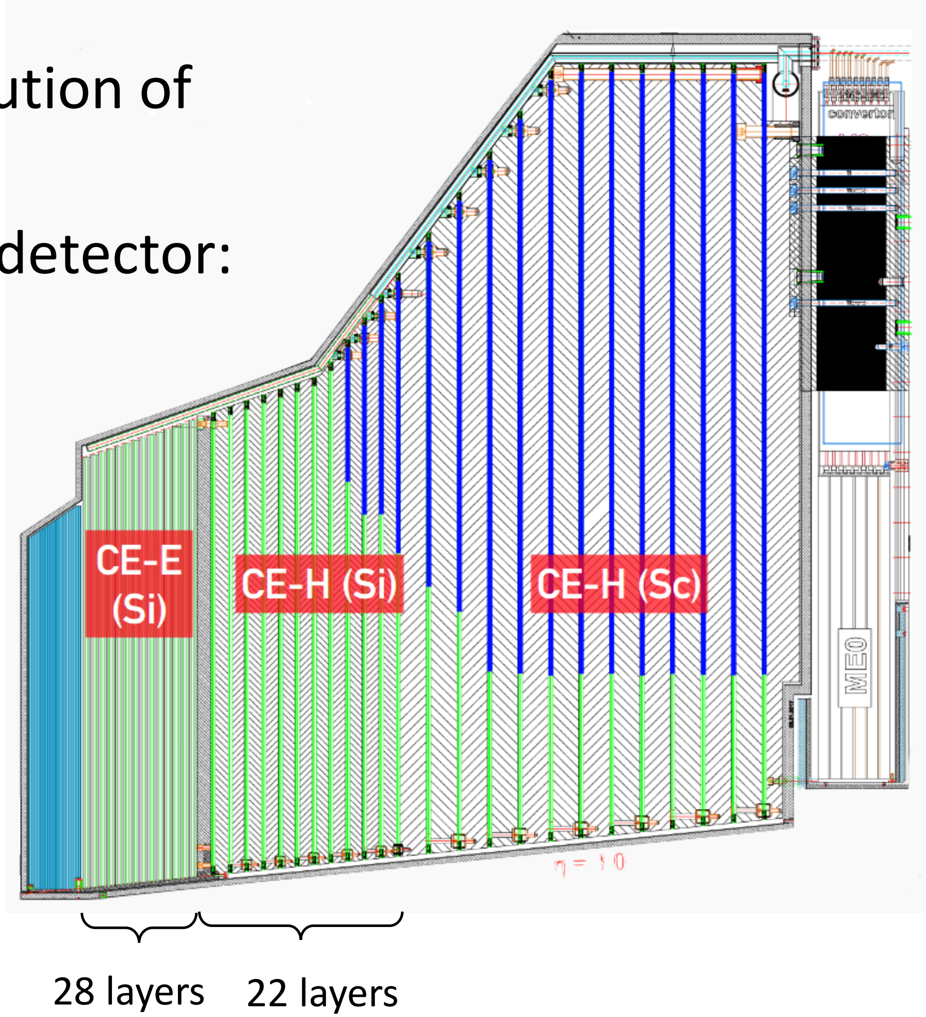
Our team is involved in the *endcaps of the* CMS sub-detector: *High Granularity Calorimeter* (HGCal)



HGCal Challenges

- Increasing pile-up (~200)
- The high granularity (> 6M channels)
- High occupancy
- High trigger rate at 40 MHz
- Time resolution : vertices spread in position and time (towards 4D analysis)

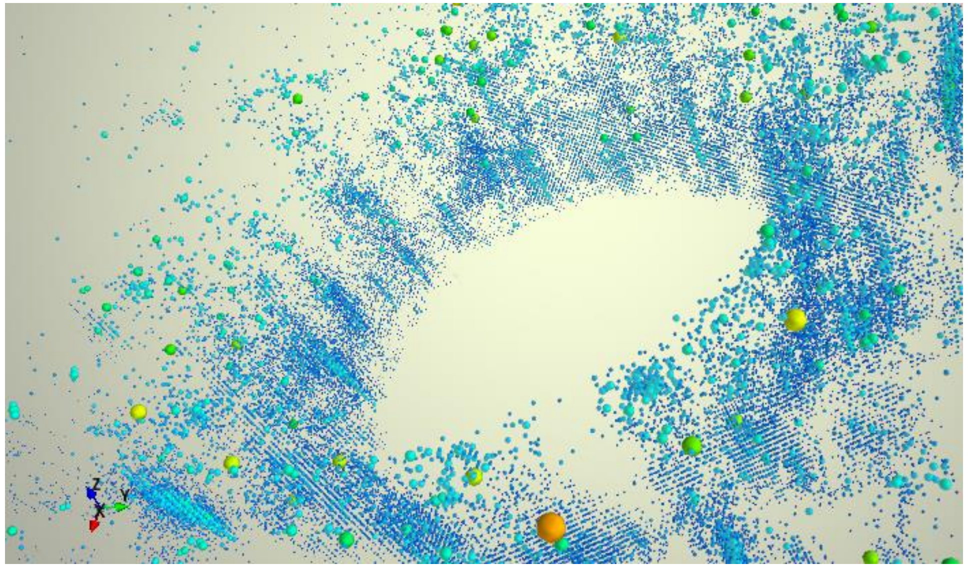
Involve drastic changes in the event reconstruction



Motivations : HGCal event reconstruction

Current flexible approach:

- The Iterative CLustering (TICL)
- Combining clustering and pattern recognition iteratively



Event simulation in HGCAL sub-detector: energy deposits (log scale)

We propose to carry out the two steps simultaneously based on recent DL in image processing technics:

What we want:

- *Classify* in cluster categories : EM clusters (dense) or Pions showers (sparse)
- *Localize* all the clusters and their footprint

In DL field our problem falls in the "Object detection" realm

The model : Mask RCNN

Benefit from the applied research, motivated by industrial challenges:

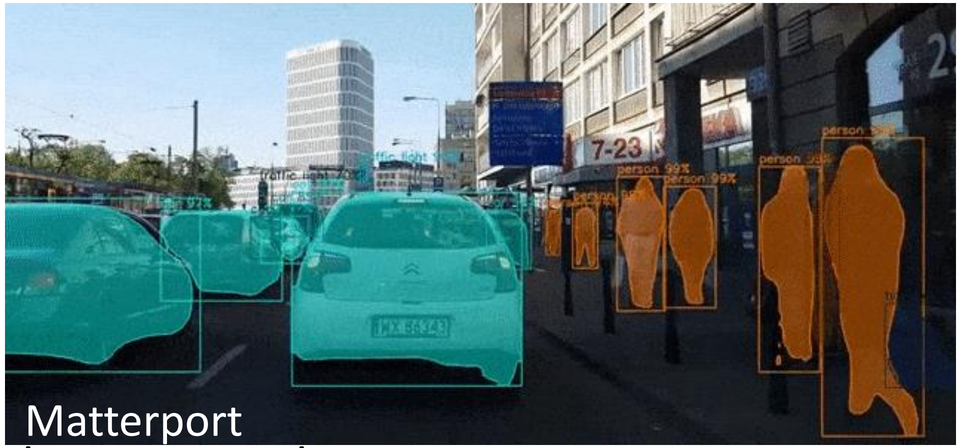
- automotive, face recognition, satellite imagery, medical, …

Object detection evolution:

- CNN with Sliding Windows
- R-CNN (2013),
- Fast RCNN (2015),
- Faster RCNN (2015) ,
- Mask-RCNN (2017-18)

Model Competition (speed & accuracy)

- Yolo - You Only Look Once (*SxS* grid)
- SSD - Single Shot Detection

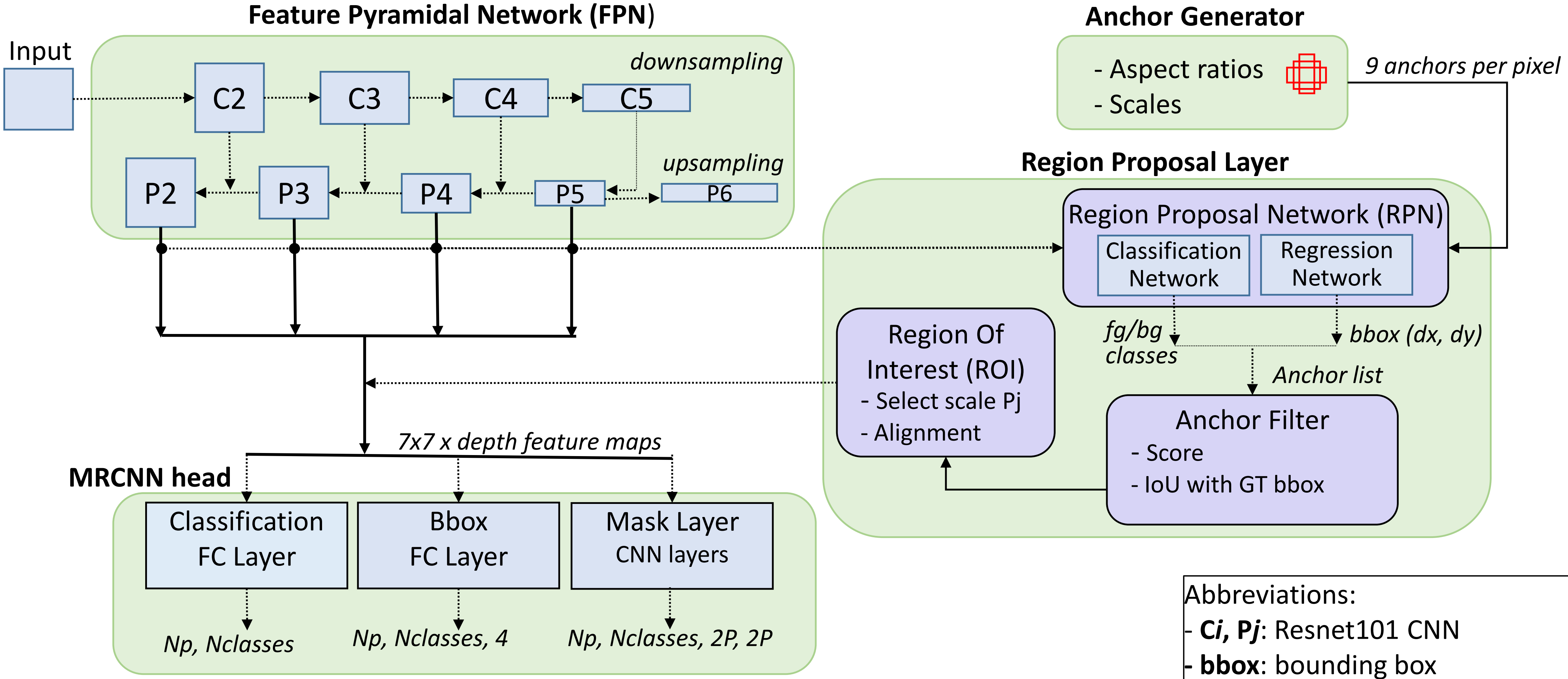


Mask R-CNN implementation

- Original Facebook Research https://github.com/facebookresearch/Detectron
- Matterport (TF, Keras) https://github.com/matterport/Mask_RCNN
- Medical Detection Toolkit (3D, PyTorch) https://github.com/pfjaeger/medicaldetectiontoolkit/tree/master/models
- Tensorflow implementation https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/instance_segmentation.md

We decided to start with, as a test bench, the 2D Matterport implementation before tackling the 3D problem

Mask R-CNN Model



Abbreviations:
- ***Ci*, *Pj***: Resnet101 CNN
- **bbox**: bounding box
- **GT**: ground truth
- ***fg/bg***: *foreground / background*

Mask R-CNN Model

Loss computation

$$L = L^{rpn} + L^{mrcnn}$$

with

$$L^{rpn} = L^{rpn}_{class} + L^{rpn}_{bbox}$$

$$L^{mrcnn} = L^{mrcnn}_{class} + L^{mrcnn}_{bbox} + L^{mrcnn}_{mask}$$

and

- *Cross-entropy* for class

$$L^{class} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} P_{truth}(C_{y_i} = c).\log(P_{model}(C_{y_i} = c))$$

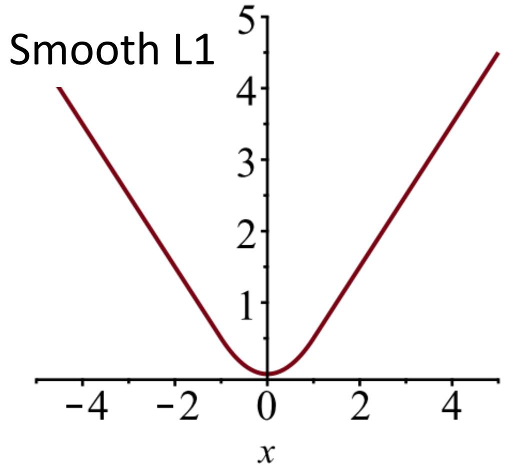
- For bounding boxes : *smooth L1 (see fig.)*

$$L^{bbox} = -\frac{1}{N}\sum_{i=1}^{N} L_1^{smooth}(y_i)$$

- Mask Loss
  Binary cross-entropy (similar to L_{class})

Main model characteristics

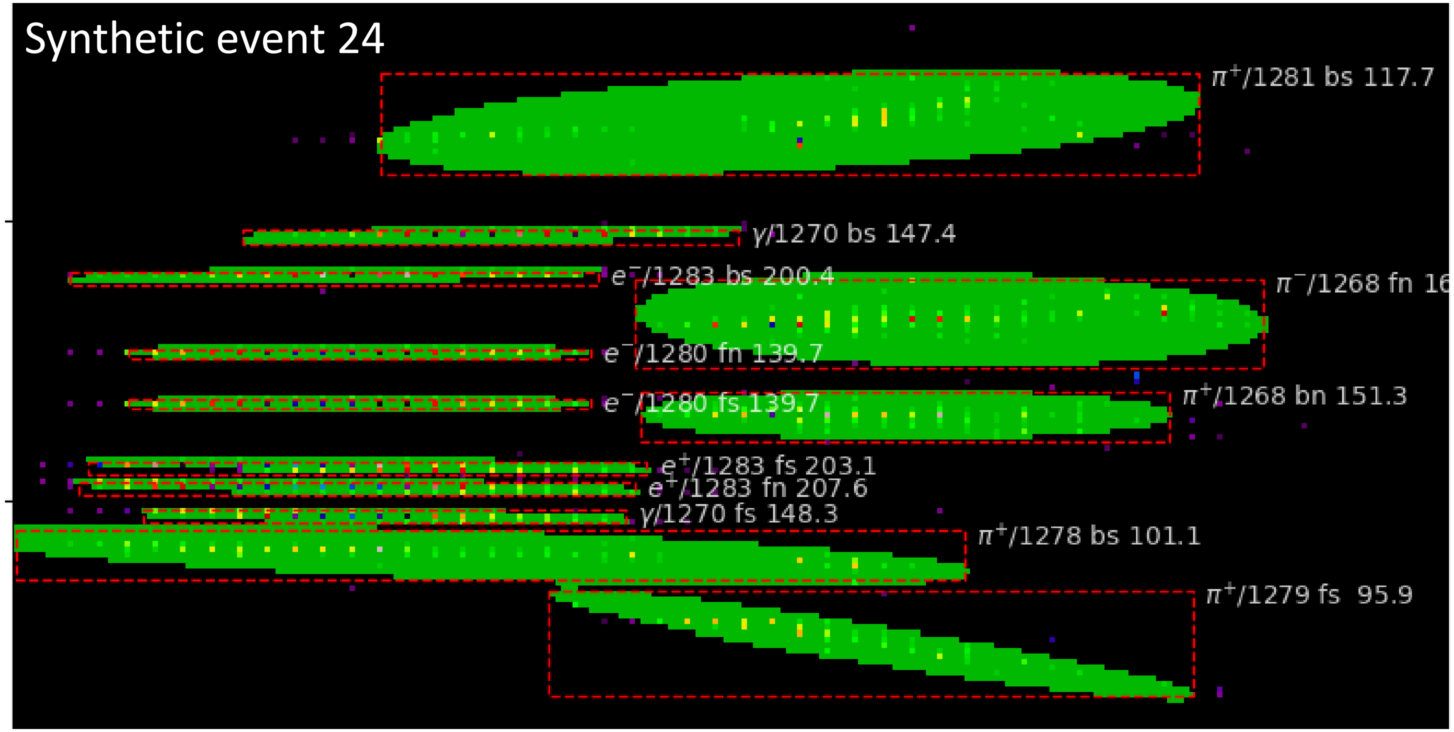
- Batch size: 8
- Optimizer:
  - Stochastic gradient descent (SGD),
  - learning rate 0.001,
  - momentum 0.9
- Penalization: L2 regularization



Building the Data-Set

Need samples with their “object” *location* (bounding box or bbox) and their *classification*.
Difficult to extract all the details of each object from a simulation with pile-up.
Choose to simulate single particles ($e^{+/-}$, γ, $\pi^{+/-}$) that are overlaid on-the-fly:
small approximation and a lot of flexibility.

Y



X

Build a primary data-set with a unique object

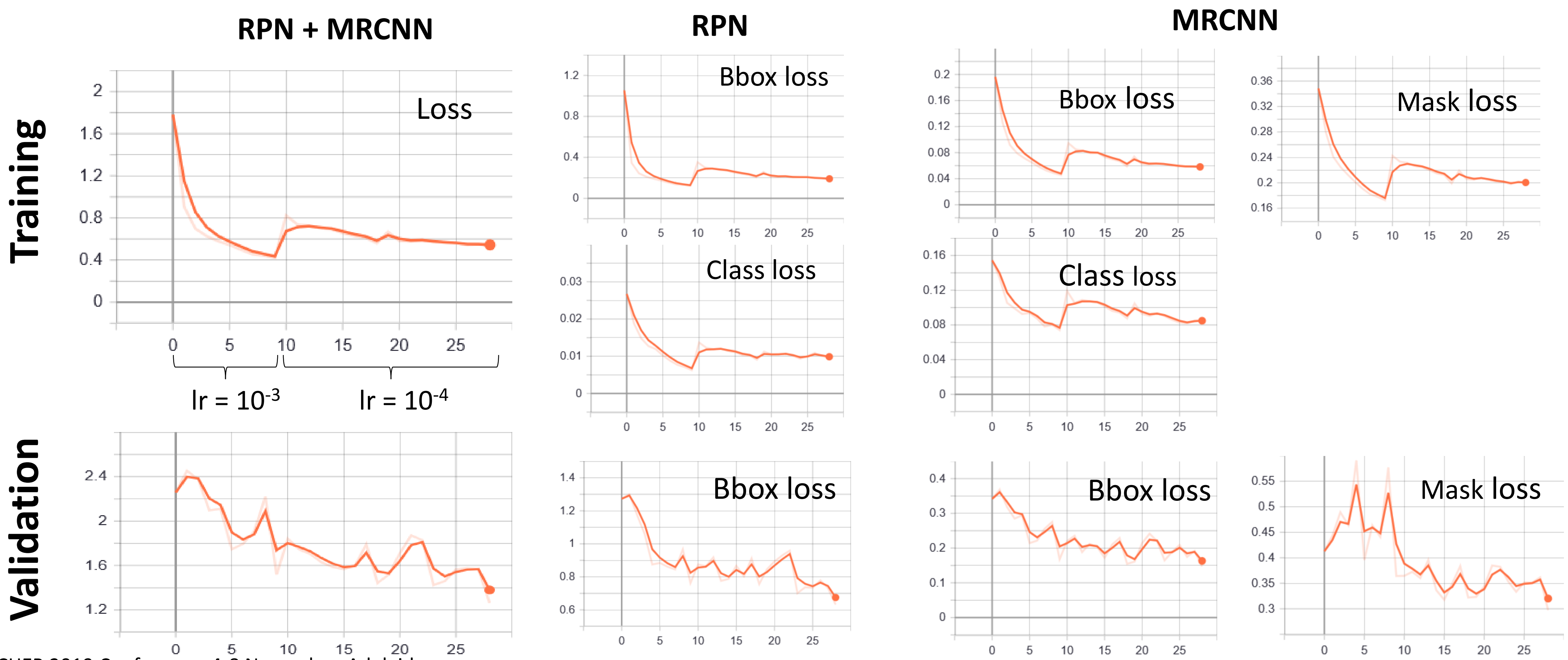
- Get events with **unique** particle with E > 20 Gev (in fwd or in bwd detector) to get the bbox
- Build a **2D histogram** (3D -> 2D image)
- The mask will be an ellipse (PCA of the cluster/shower)

Compose a training/validation data-set with “objects” in the primary data-set

- Set a number of “objects” in the image (a range)
- Select randomly them among the primary data-set
- Random operation (data augmentation): object symmetry, random shift dy = +/- 1 pixel

Results: Loss

12-20 objects, Training data-set 5000 ev., Evaluation data-set 50 ev., epoch ~30

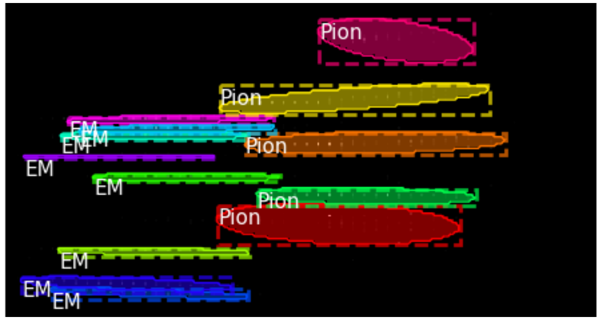


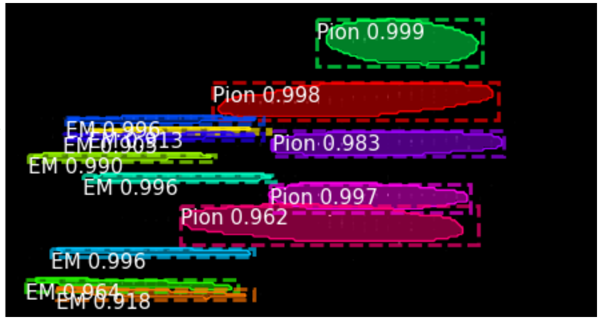
Results: nice predictions

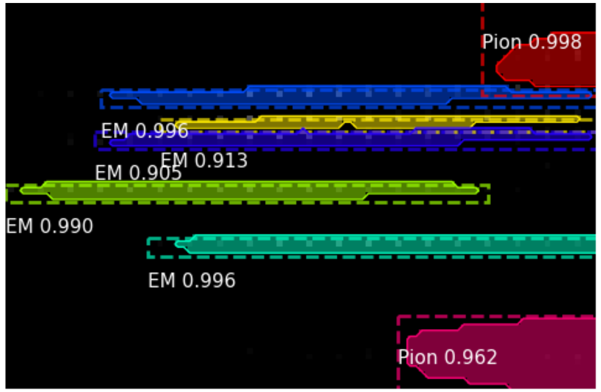
Zoom of predicted ev.

Ground truth ev. | **Predicted ev.**

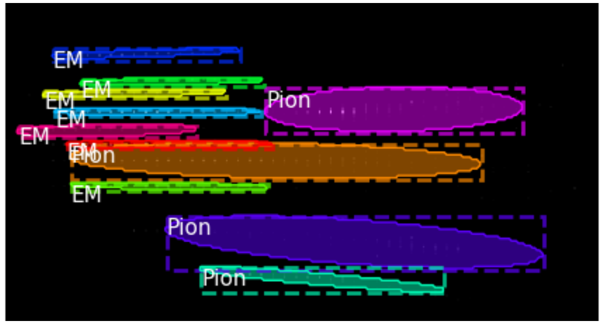
Ev 14

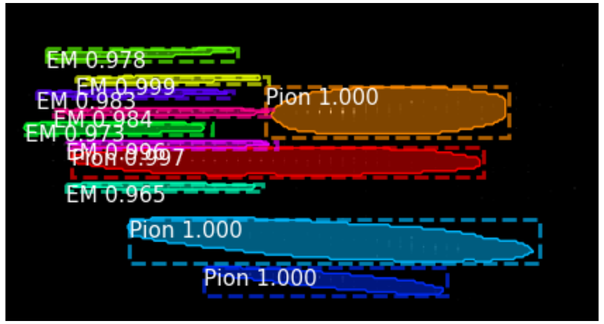


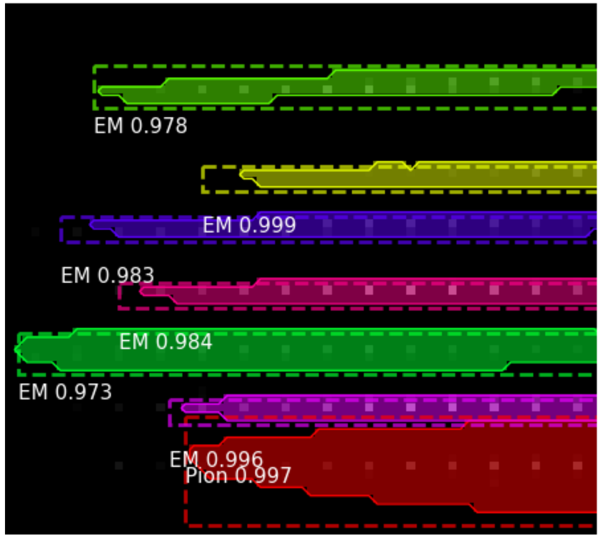




Ev 17







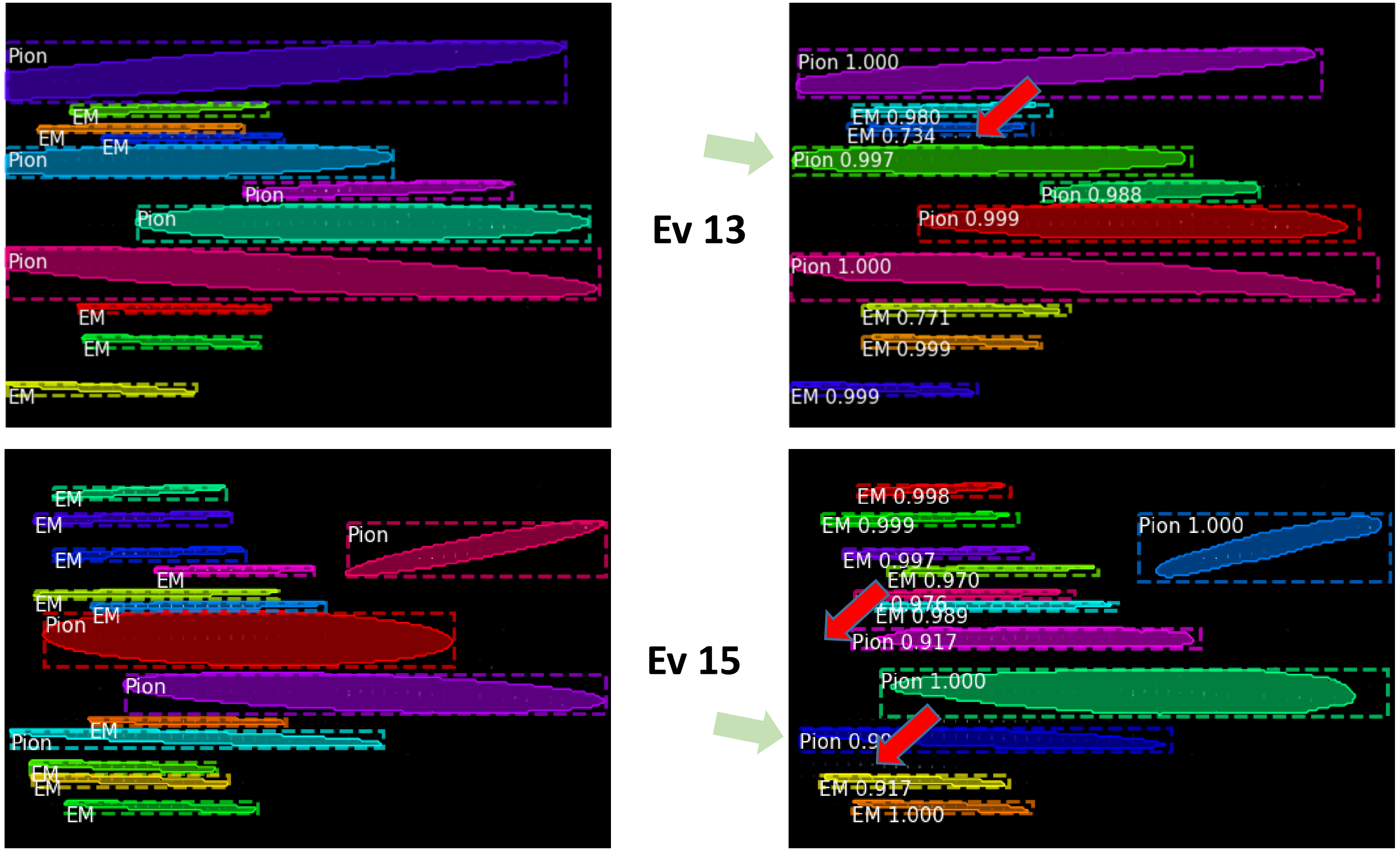
Good predictions:

- Classification, localization (bbox), mask
- Dense region of objects (green arrows)
- mAP (mean Average Precision) = 0.73

But …

Results: … to improve

Ground truth ev. **Predicted ev.**



Good

- Pion showers start in EM region (green arrows)

To improve

- Missing object (red arrows)
- Small mask for pion shower (red arrows)

mAP = 0.73, ~ 15 % objects missing

Conclusion / Perspectives

HGCal 2D test bench

- Challenging conditions: small data-set, rough histograms, the layers are far from each others, int8 as input, …
- However, gives pretty good results
- Mask R-CNN captures the scattered hits coming from Pion showers

Next Steps

HGCal 2D

- Getting better conditions to train
- Modify the model in MRCNN

HGCal 3D

- Apply the lessons of HGCal 2D
- Medical Detection Toolkit (3D, PyTorch)

Acknowledgments

- Funding project P2IO (GPU platforms)
  Accelerated Computing for Physics
- Google Summer of Code 2019
  HAhRD project : DL & HGCAL
- IN2P3 project: DECALOG/Reprises
- S. de Guzman – Ecole polytechnique internship - testing 2D/3D implementations

- CHEP 2019 organizers

After CHEP 2019

mAP@IoU (mean Average Precision)

- Parameter t = IoU *on masks*

$$IoU(A,B) = \frac{A \cap B}{A \cup B}.$$

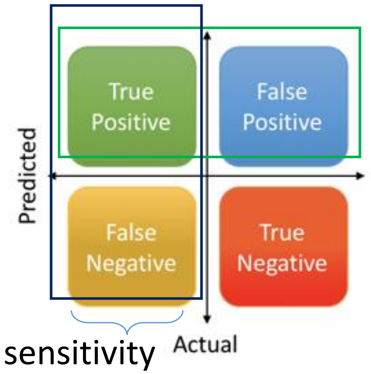
- Precision value for t values true comparing the predicted object to all ground truth objects:

$$\frac{TP(t)}{TP(t) + FP(t) + FN(t)}$$

TP(t) = good prediction as a cluster/shower
FP(t) = bad predicted area as a cluster/shower ??? Multiple proposals
FN(t) = not predicted cluster/shower
TN(t) = background

- Our training/validation:
  - Missing clusters (FN): mAP decrease
  - Multiple solutions for the same cluster : mAP decrease ?
  - Recall or sensitivity should be more sensible to the missing clusters

- Precision, recall & accuracy



Recalls, sensitivity

Precision (positive pop.)

$$Precision = \frac{TF}{Actual\ Results} = \frac{TP}{TP + FP}$$

$$Recall = \frac{TF}{Predicted\ Results} = \frac{TF}{TP + FN}$$

$$Accuracy = \frac{TF + TN}{Total}$$

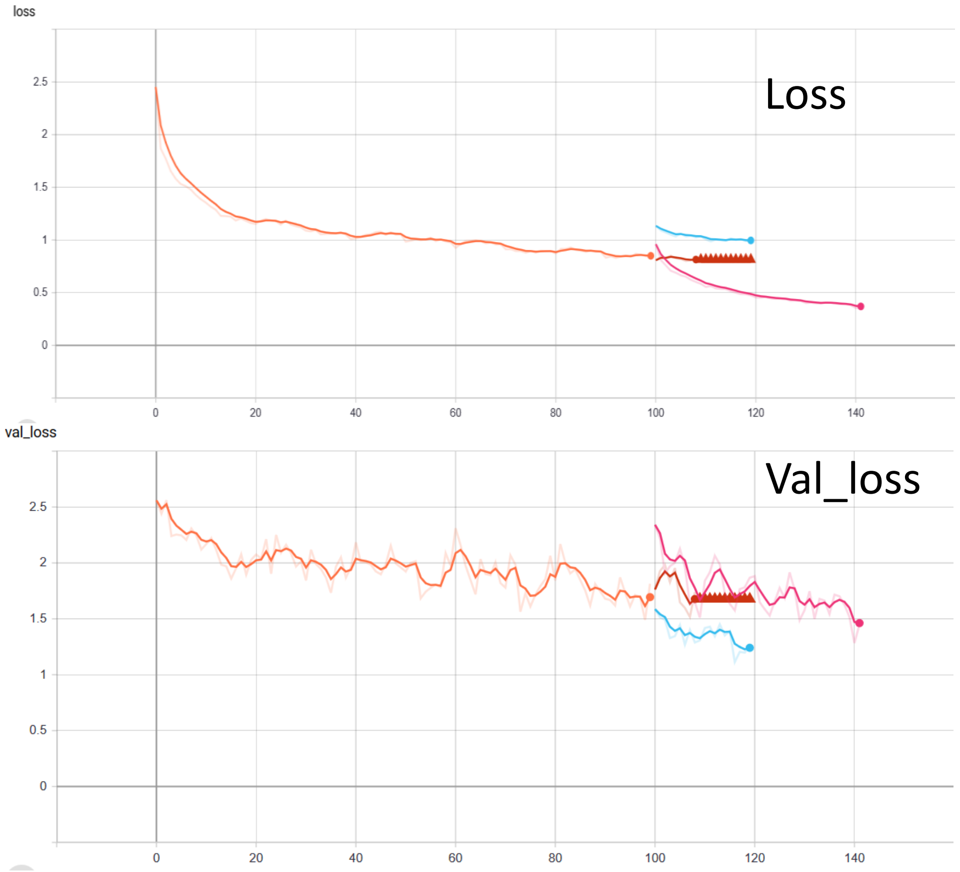
- Kaggle (Data Science Bowl), *t in { .5, .55, …, .95}*

$$\frac{1}{|thresholds|} \sum_t \frac{TP(t)}{TP(t) + FP(t) + FN(t)}.$$

A lot of work … (1)

- Learn about training process (loss nan, how to improve the model, RPN, multiple ways to train)
- Purify the data-set
- Adjust the architecture
  - RGB input -> real4 (normalization)
  - 256x64 images : should improve the training (demonstrated with "layer" mode)
  - Aspect ratio (again !)

- Training "at best"
  Validation/test data-set ⊂ training data-set



A lot of work ... (2)

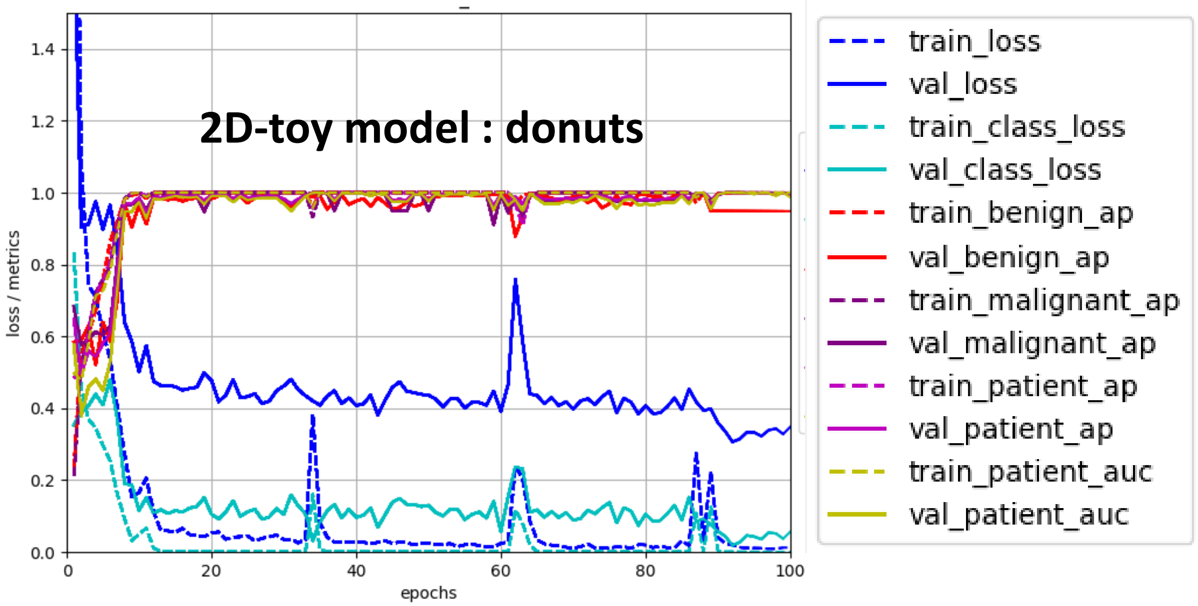
PyTorch implementation
Medical Detection Toolkit (MDT) developed for images analysis

Silver planning
Test with “shapes” data-set

- OK — Installation on ACP/GridCL ... tricky
- In Progress — Adjust to the 2D-shapes data-set
- TO DO — Extend to 3D-shapes data-set

HGCAL planning TO DO

- Adjust to 2D-HGCal data-set
- Compare with Mask RCNN
- Extend to 3D-HGCal data-set



Conclusion

Decalog/Reprises ne peut ignorer la techno du DL avec notre contexte HPC / portabilité / reproductibilité numérique

- Placement des données, …
- Analyses de graphes
- Pb d'optimisation
- Technologie TF, Torch, Keras, …

Et le chemin est long avant de maitriser de tels systèmes …

Réunion Projets IN2P3

- Budget en baisse 11 keuros
- Projets qui publient ou gros impact avec moins de moyens

Conférences

- ACAT 2020, sep., Corée du Sud
- CHEP 2021, may -Norfolk, Vitginie, USA