Laboratoire d'Annecy-le-Vieux de Physique des Particules

# Multi-task learning for astroparticle physics

SOS 2021, 29/01/2021

Thomas Vuillaume, Mikael Jacquemont

UNIVERSITÉ SAVOIE MONT BLANC

# Plan and objective for today

- Objective: see an example of real-life machine learning project
  - How the problem is adressed
  - What is the analysis chain
  - Technological choices

- Plan
  - Introduction to multitask learning
  - The Cherenkov Telescope Array and the event reconstruction problem
  - A standard approach
  - Deep multi-task learning
  - Application to CTA

# Run it yourself

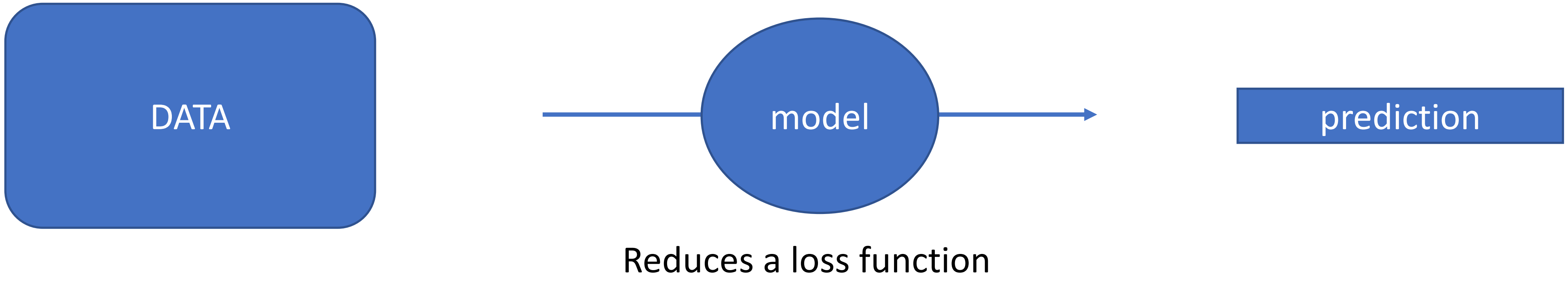
For the demo part, you may run the code yourself.

All the content is openly available at:

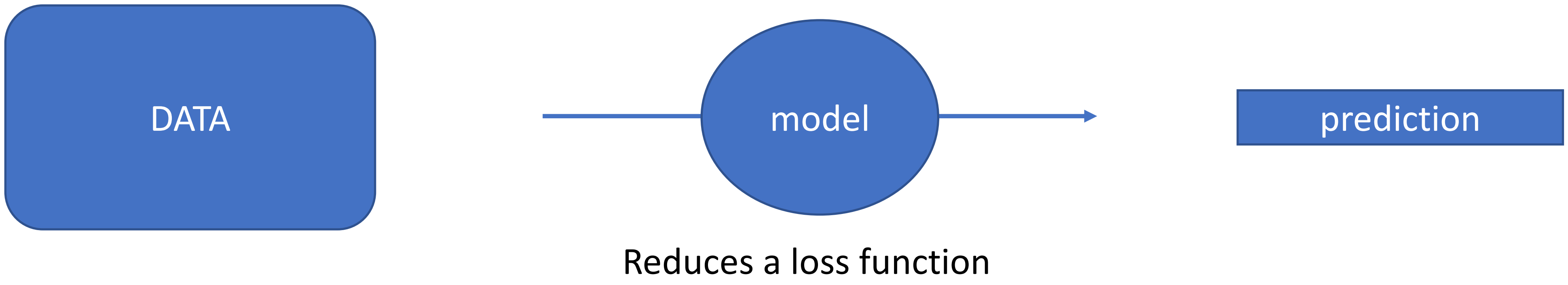
https://github.com/vuillaut/cta_mtl_course

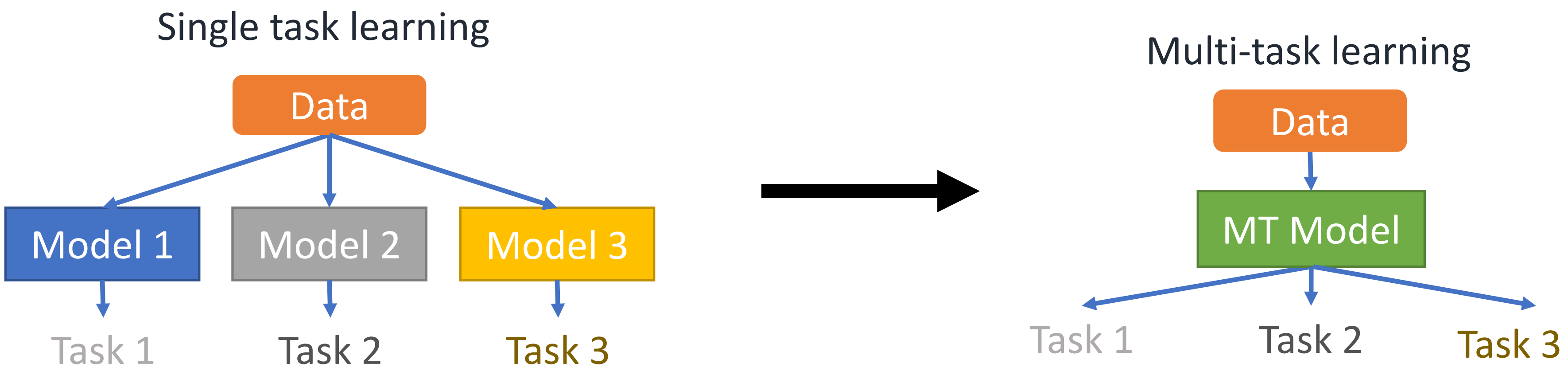
# Multitask learning
# An intro

# Multitask learning



# Multitask learning



# Multitask learning

- What : learning to predict multiple outputs that have some sort of relationship between them

- Learning a specific task might help learning a related but different task

- Happens in Human learning too
  - Learning to dissociate between sounds will help recognize people from their voice
  - Or waxing a car will help to learn karate…

- Caruana, R. Multitask Learning. *Machine Learning* **28,** 41–75 (1997). https://doi.org/10.1023/A:1007379606734

Why does it work?

- Implicit data augmentation
- Attention focusing: having more tasks help focusing on relevant features
- Regularization: the model generalizes better (avoids overfitting)
- Improves cross-tasks coherence
    - Exemple: predicting animal type and its color, you (probably) want to avoid the possibility to predict green cats

When does multitask make sense

- Training on a set of tasks that could benefit from having shared low-level features

- You lack data for a specific task but have a lot of data for another related task

- Not limited in the size of the model you build (otherwise can face negative transfer)

It is different from transfert learning but can help in similar ways

# Auxilliary tasks

Auxilliary tasks are tasks that are not the main goal you are trying to solve but are added because they can help you reaching that goal

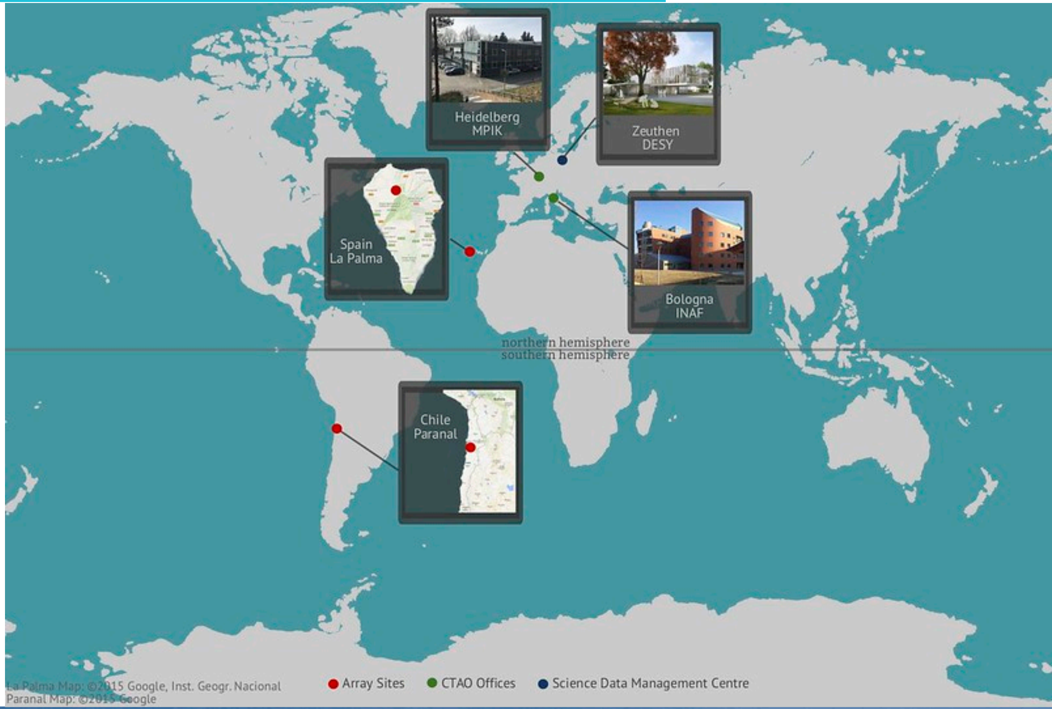
- Assumes that the tasks are related
- Enables the model to learn representations that are shared or helpful for the main task
- Examples:
  - Predict road characteristics to predict the steering direction of a self-driving car
  - Estimate head pose for facial landmark detection
  - Learning depth perception will help catching an object

cta

cherenkov
telescope
array

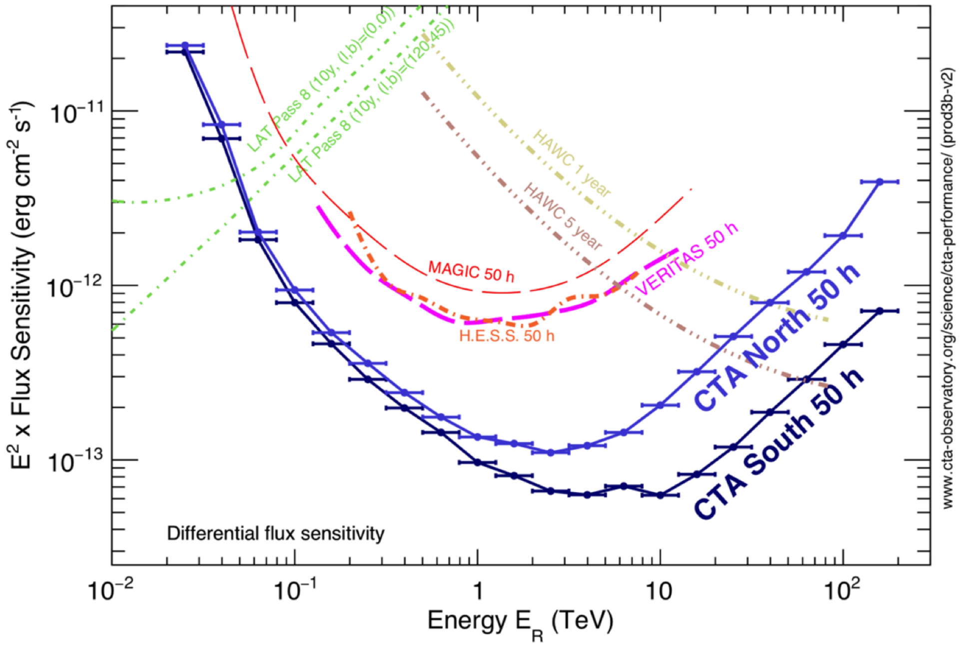
# The Cherenkov Telescope Array

- 2 sites, > 100 telescopes



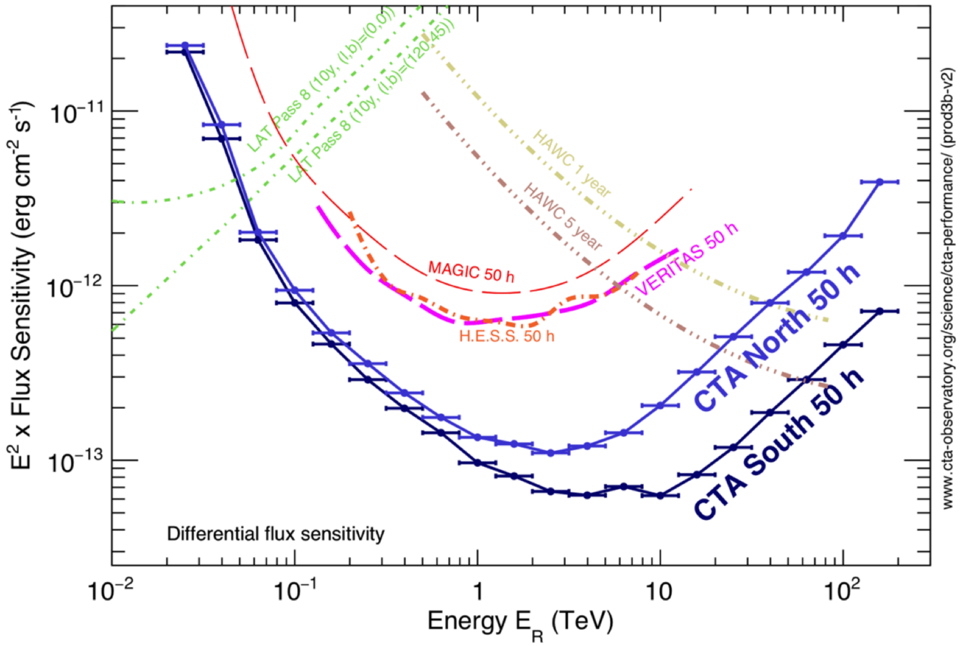

# The Cherenkov Telescope Array

- 2 sites, > 100 telescopes
- Observe the sky from 10GeV to 200 TeV
- Sensitivity x10 compared to current generation of instruments




# The Cherenkov Telescope Array

- 2 sites, > 100 telescopes
- Observe the sky from 10GeV to 200 TeV
- Sensitivity x10 compared to current generation of instruments
- Observatory – sending and receving alerts from other infrastructures

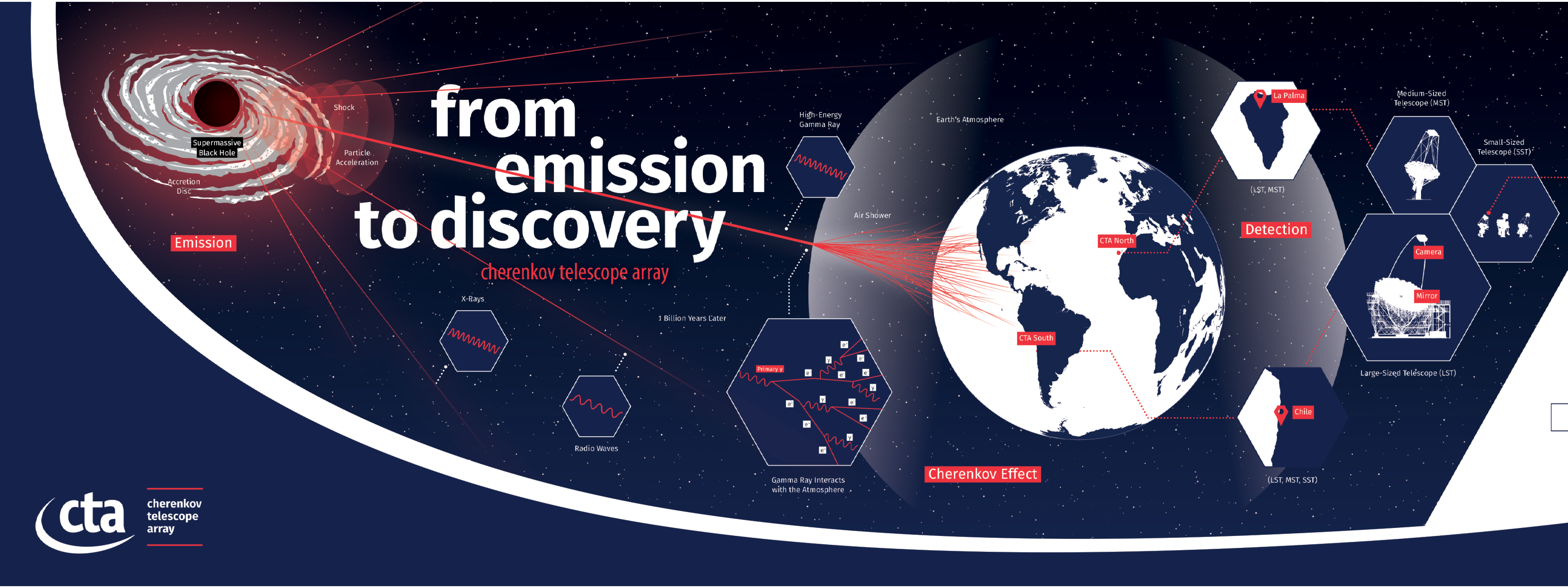



# The Cherenkov Telescope Array

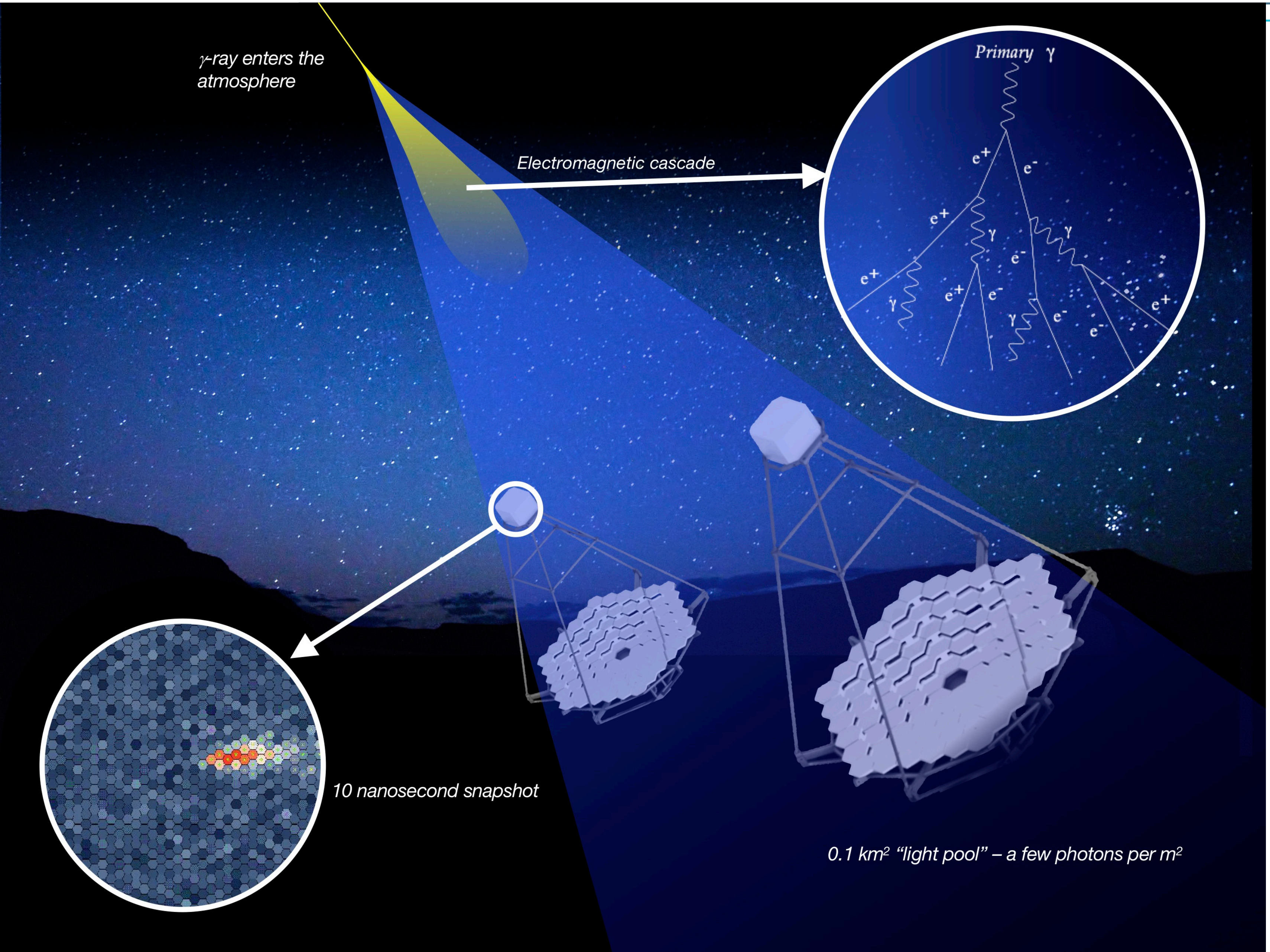
Data analysis challenges:

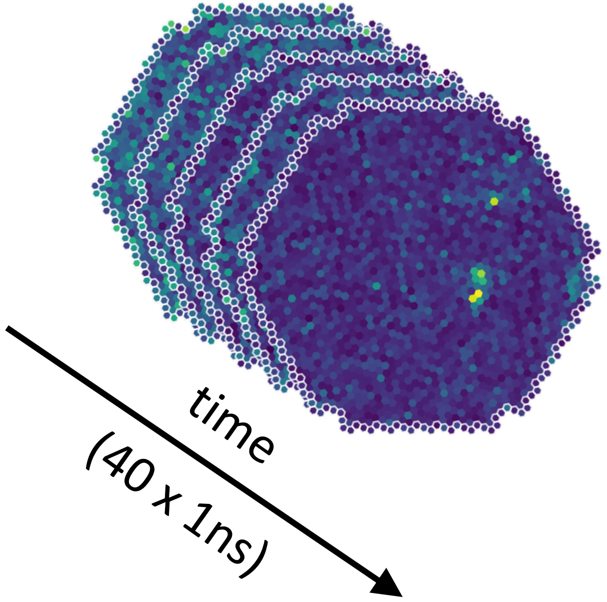
- Several types of telescopes and cameras
- Data flux for real time analysis: ~5GB/s/camera
- Data volume: 3PB/year (after volume reduction)

# From emission to discovery



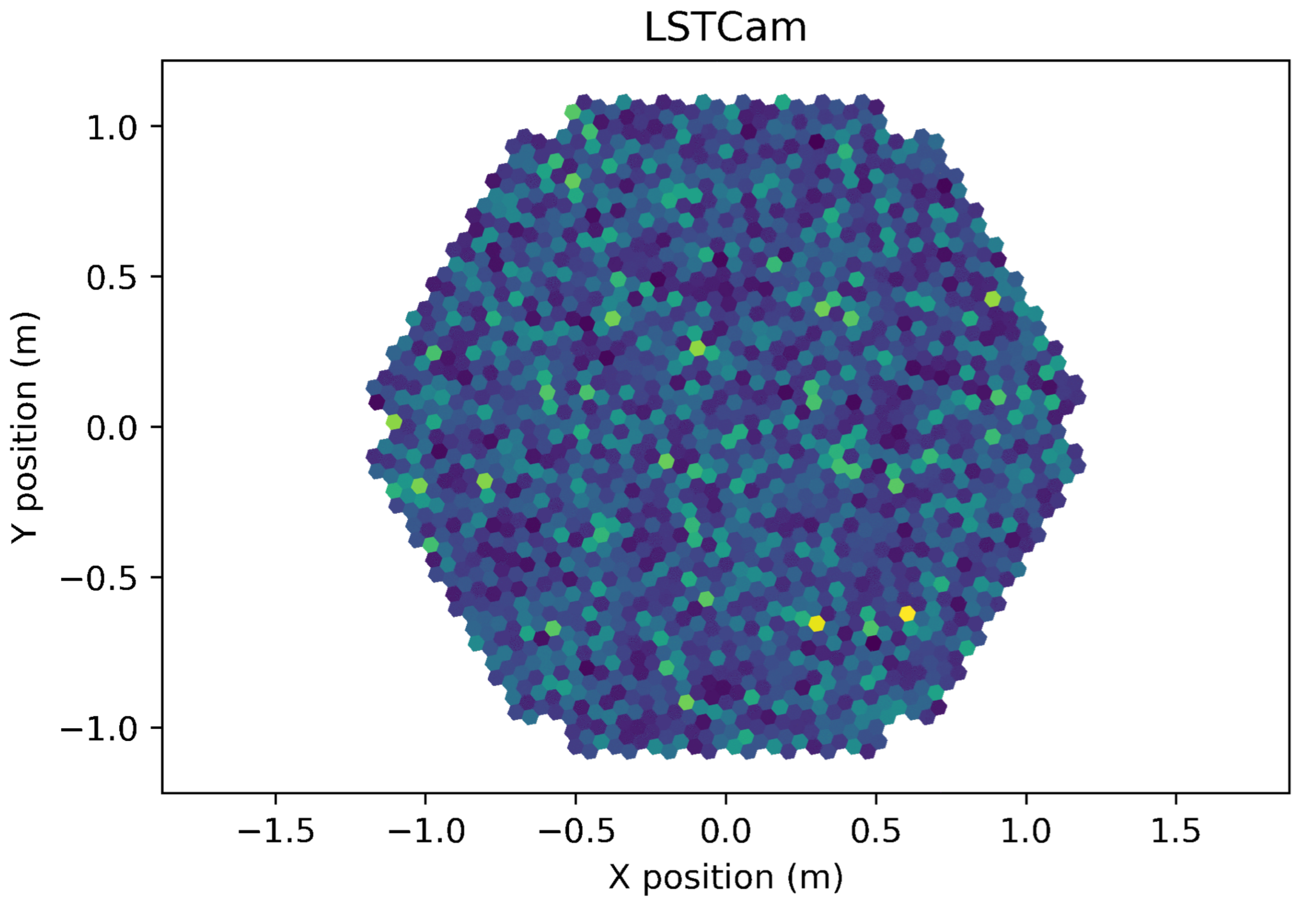
# How does it work?




- Évènement gamma
- E = 0,189 TeV
- Direction = (1,23; 6,22)rad

# The problem



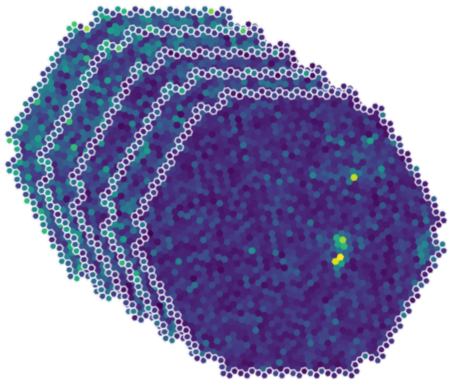

?

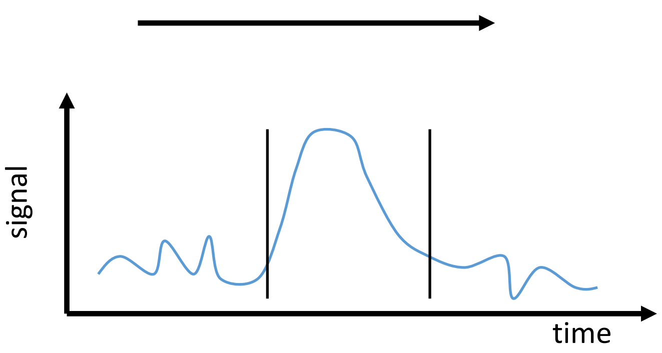
- Évènement gamma
- E = 0,189 TeV
- Direction = (1,23; 6,22)rad

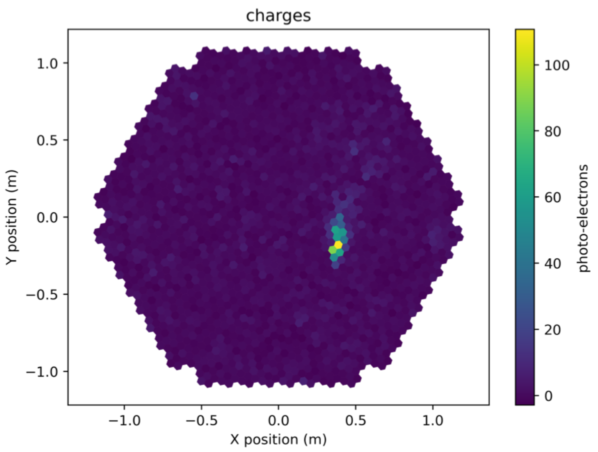
# The standard approach

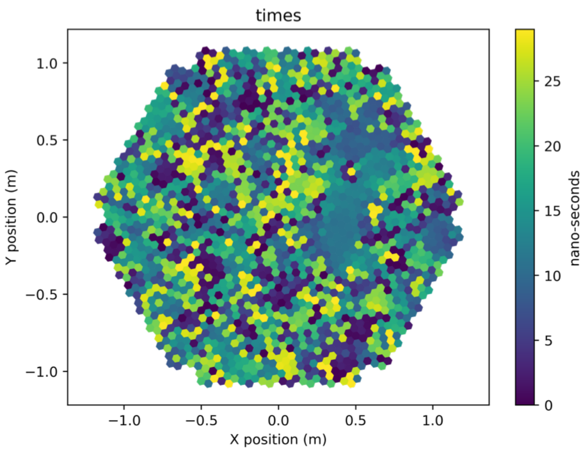
# 1. Signal extraction: calibration and integration

The signal in each pixel is integrated to max signal/noise ratio





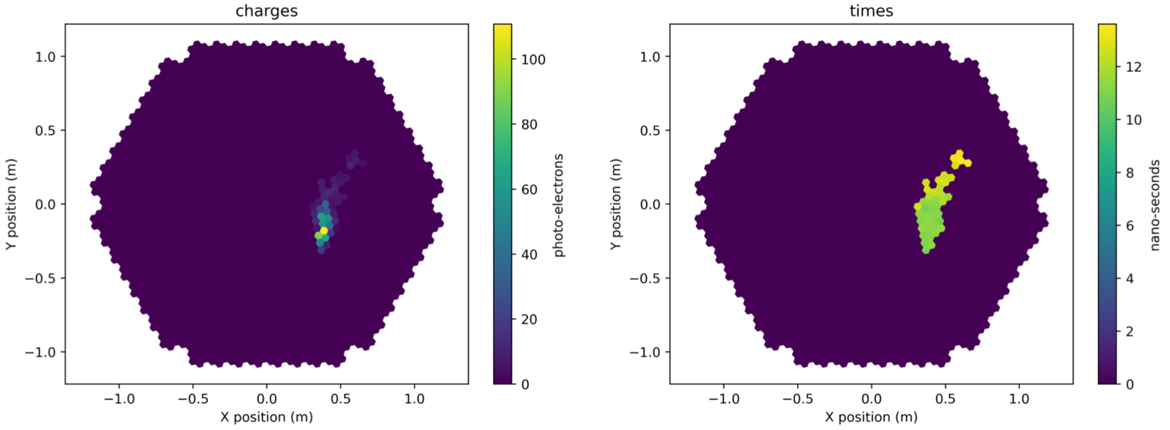



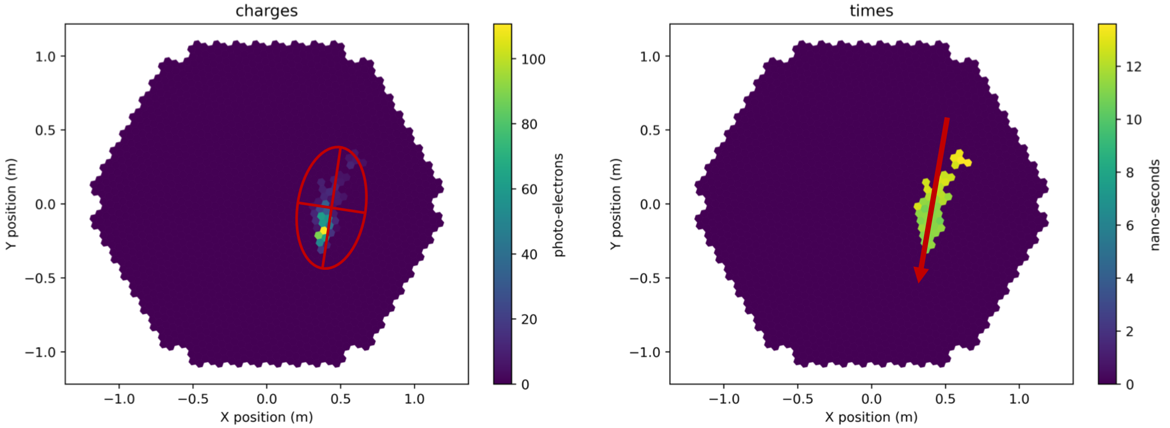

We get 2 images: the integrated charges and the photons mean arrival time

# 2. Parameters extraction

## Clean images

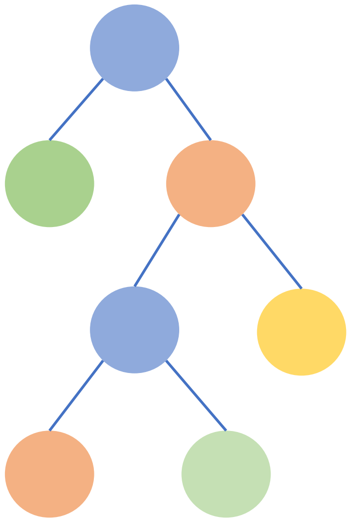



Compute some charactersitic parameters of the signal:
size of the ellipsoid, position in the camera, signal intensity, axis orientation, time gradient….



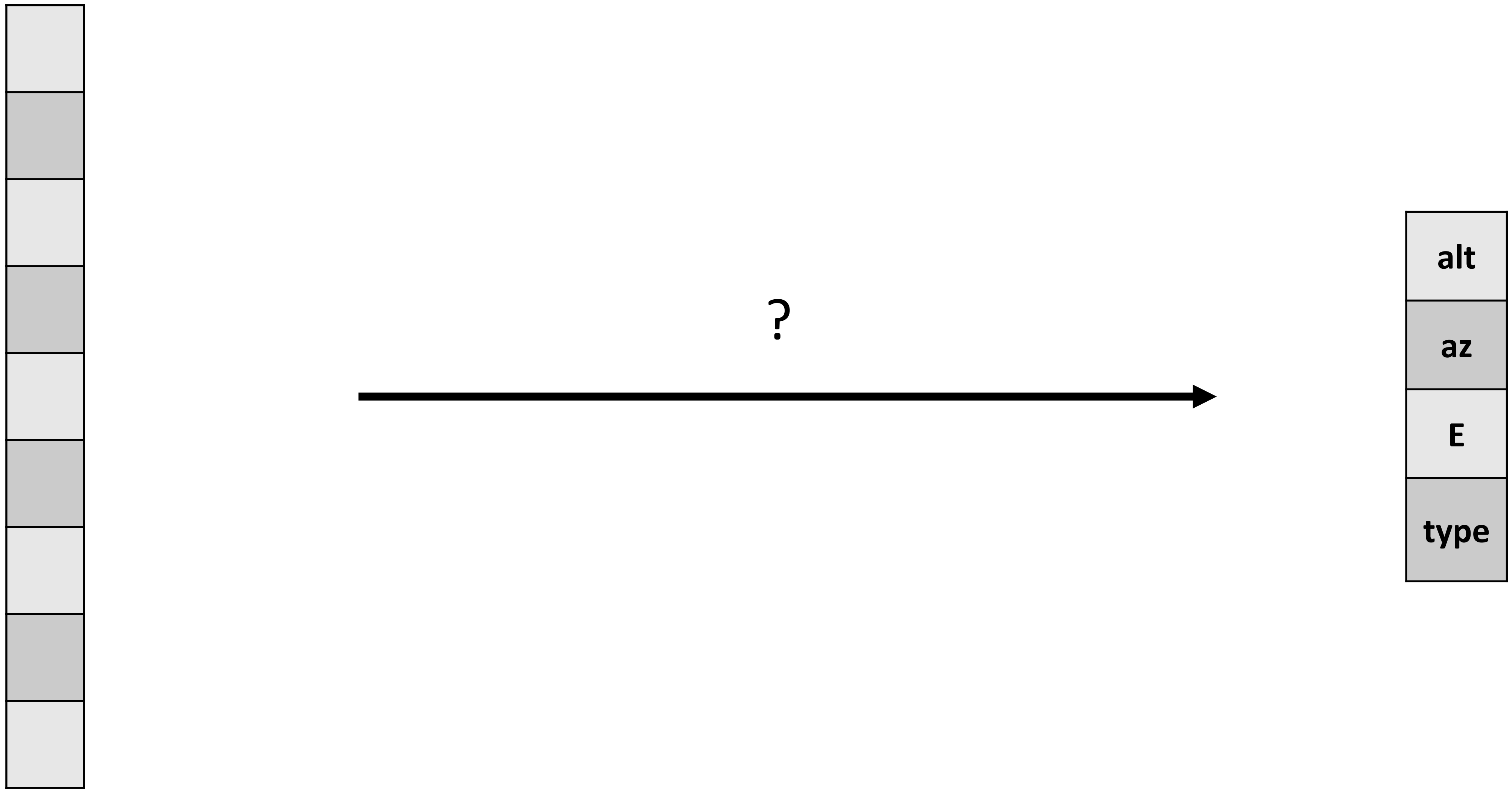
Train random forests or BDT
on Monte-Carlo simulations

Image parameters

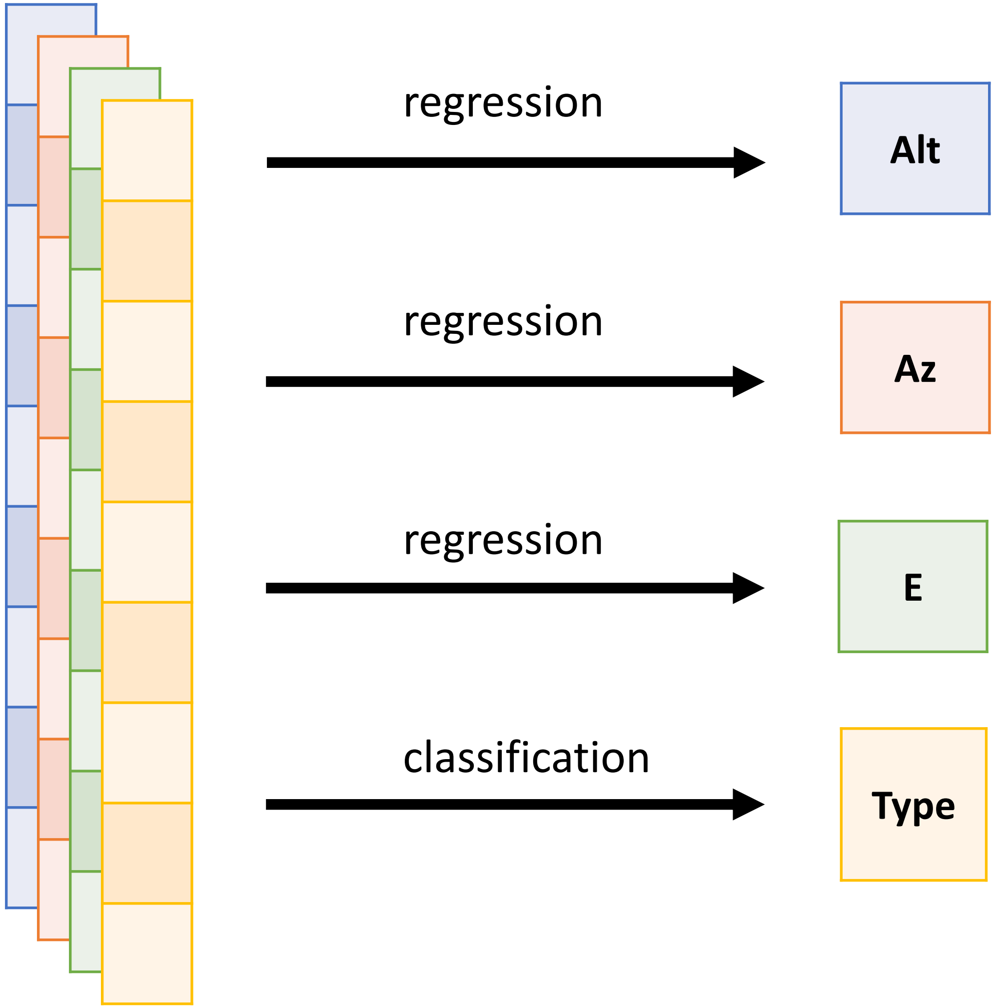


E = 0,189 TeV

And then infer on new data


?
alt
az
E
type

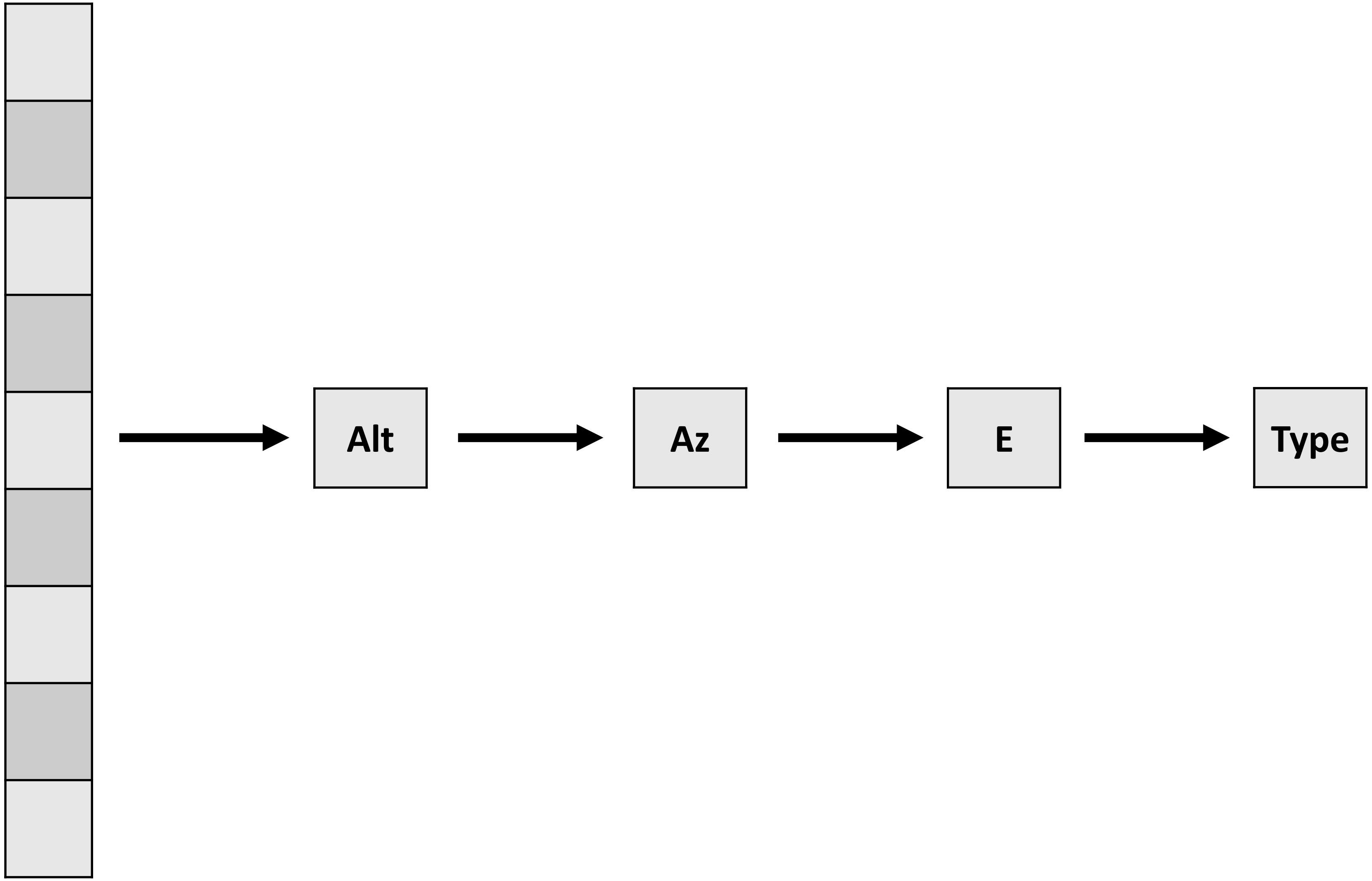
# Parallel approach




Advantages:
- simple
- can run in parallel

Drawbacks:
- 4 models to train/use
- Reconstruction degeneracy

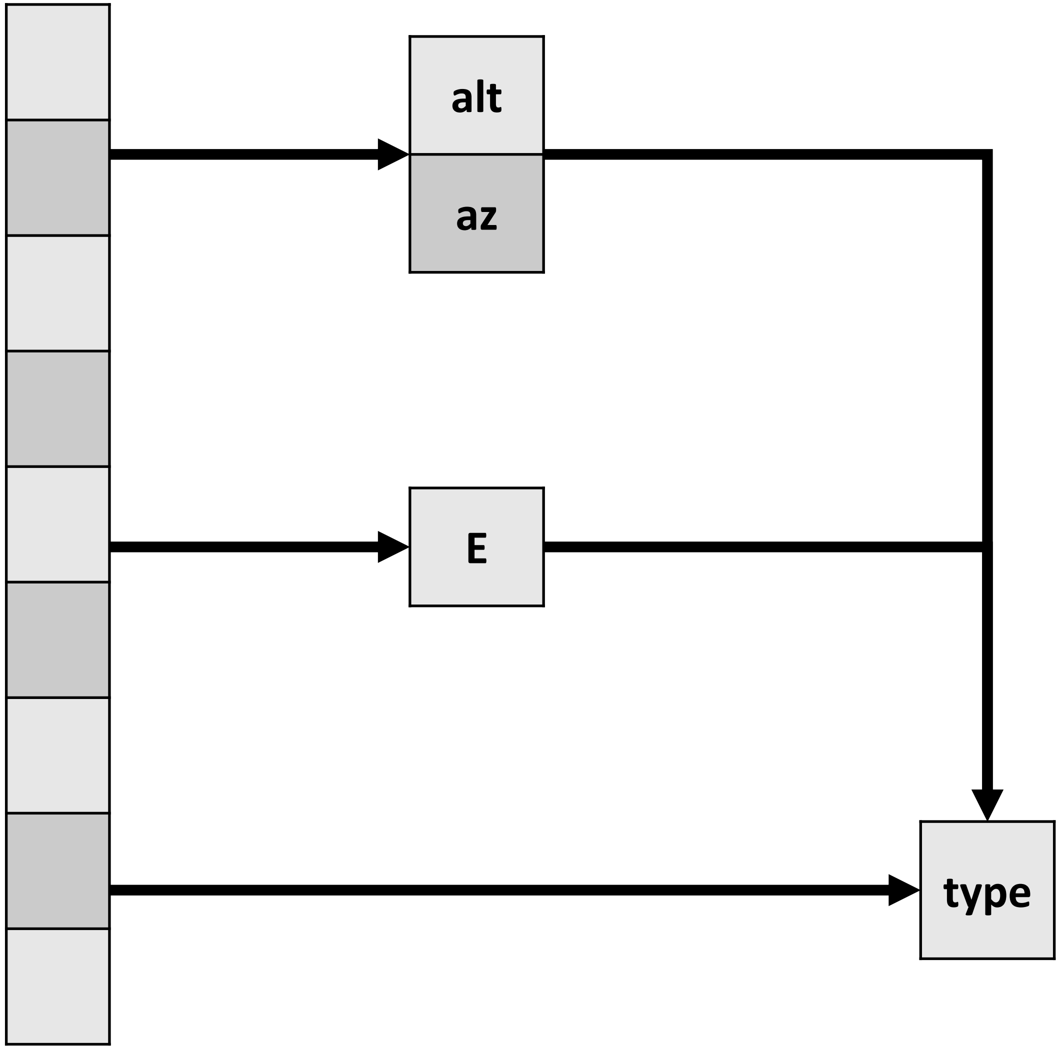
# A serial approach




Advantage:
- less degeneracy

Drawback:
- slow

# A mixed approach



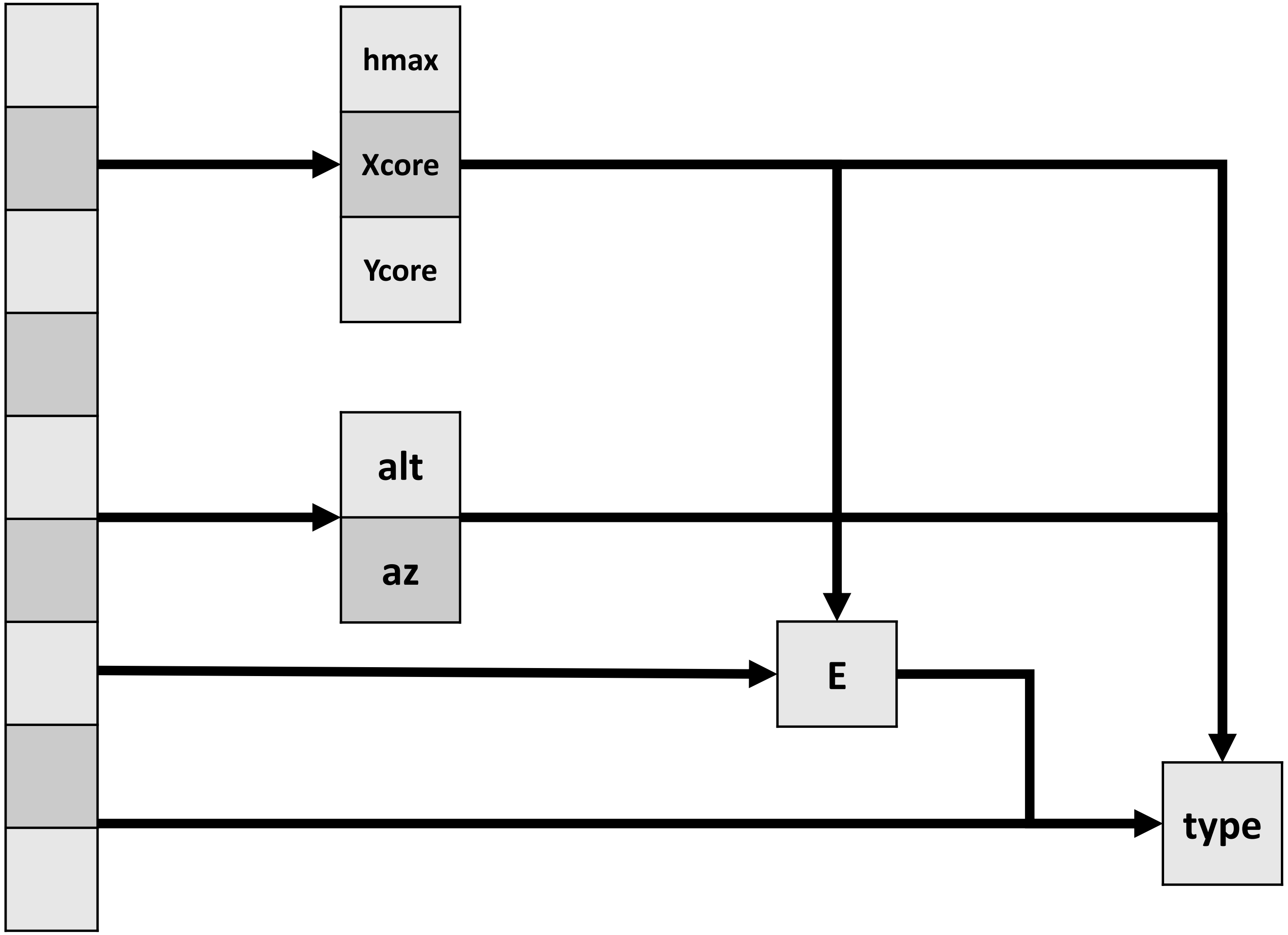

Advantage:

- Take the physics into consideration

Drawback:

- More complicated?

# Adding intermediate parameters




Advantage:
- More powerful

Drawback:
- More complicated
- Heavier

DEMO #1

See notebook random forests

# The deeper approach

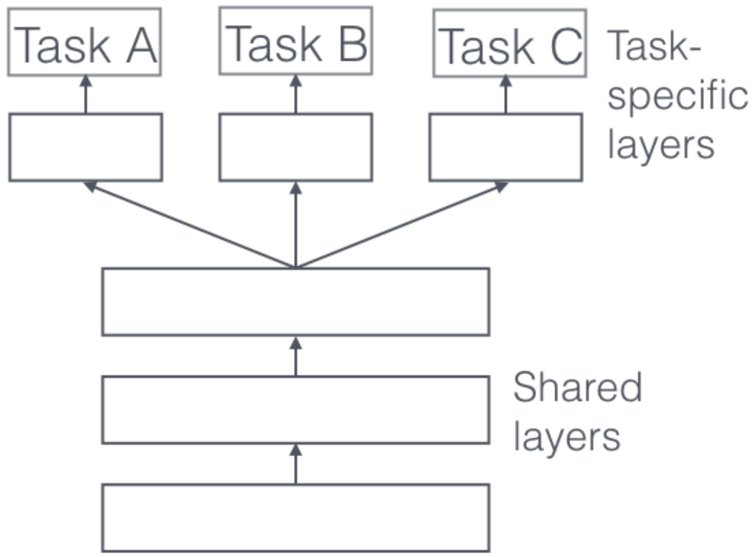
- More abstraction
- Starting from raw data

# Deep multi-task learning

# Hard parameter sharing

- First idea: Caruana, R. "Multitask learning: A knowledge-based source of inductive bias." Proceedings of the Tenth International Conference on Machine Learning. 1993.
- Several hidden layers common to all task
- Task-specific layers

$$\mathcal{L} = \sum_{i=1}^{N} w_i \mathcal{L}_i$$




Advantages:

- Common representation of the data
- Reduces risk of overfitting (of an order ~ number of tasks*)
  - The more tasks, the more general the model has to become

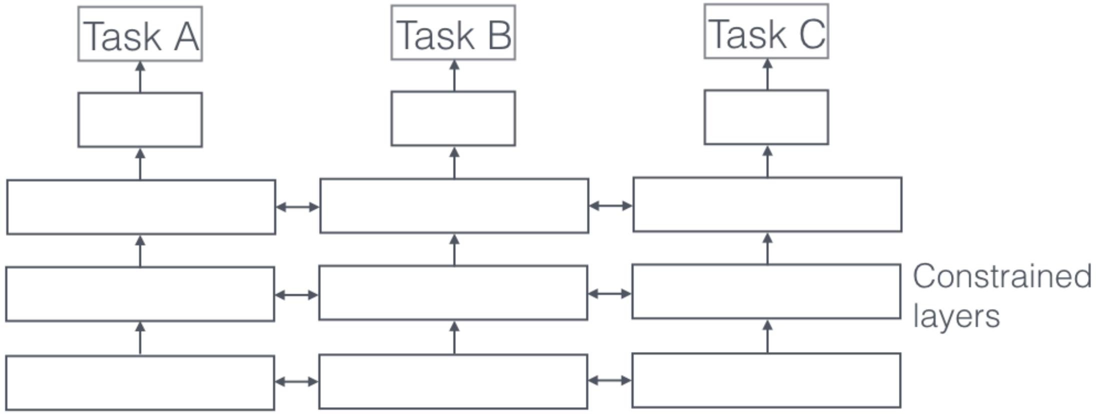
Drawbacks:

- The tasks **must** be physically related (negative transfert)
  - Requires some level of knowledge/understanding of the problem complexity

*Baxter, J. (1997). A Bayesian/information theoretic model of learning to learn via multiple task sampling. Machine Learning, 28, 7–39. http://link.springer.com/article/10.1023/A:1007327622663

# Soft parameter sharing

- One model per task
- The models parameters are regularized to encourage similar parameter distributions




- Used in language processing (learning one language with a lot of data helps learning another language with less data) to learn common tasks but not exact word translation*

$$\mathcal{L} = \sum_{i=1}^{N} w_i \mathcal{L}_i - \lambda_i ||W_i - W_j||$$

W = model weights

*Duong, L., Cohn, T., Bird, S., & Cook, P. (2015). Low Resource Dependency Parsing: Cross-lingual Parameter Sharing in a Neural Network Parser. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), 845–850

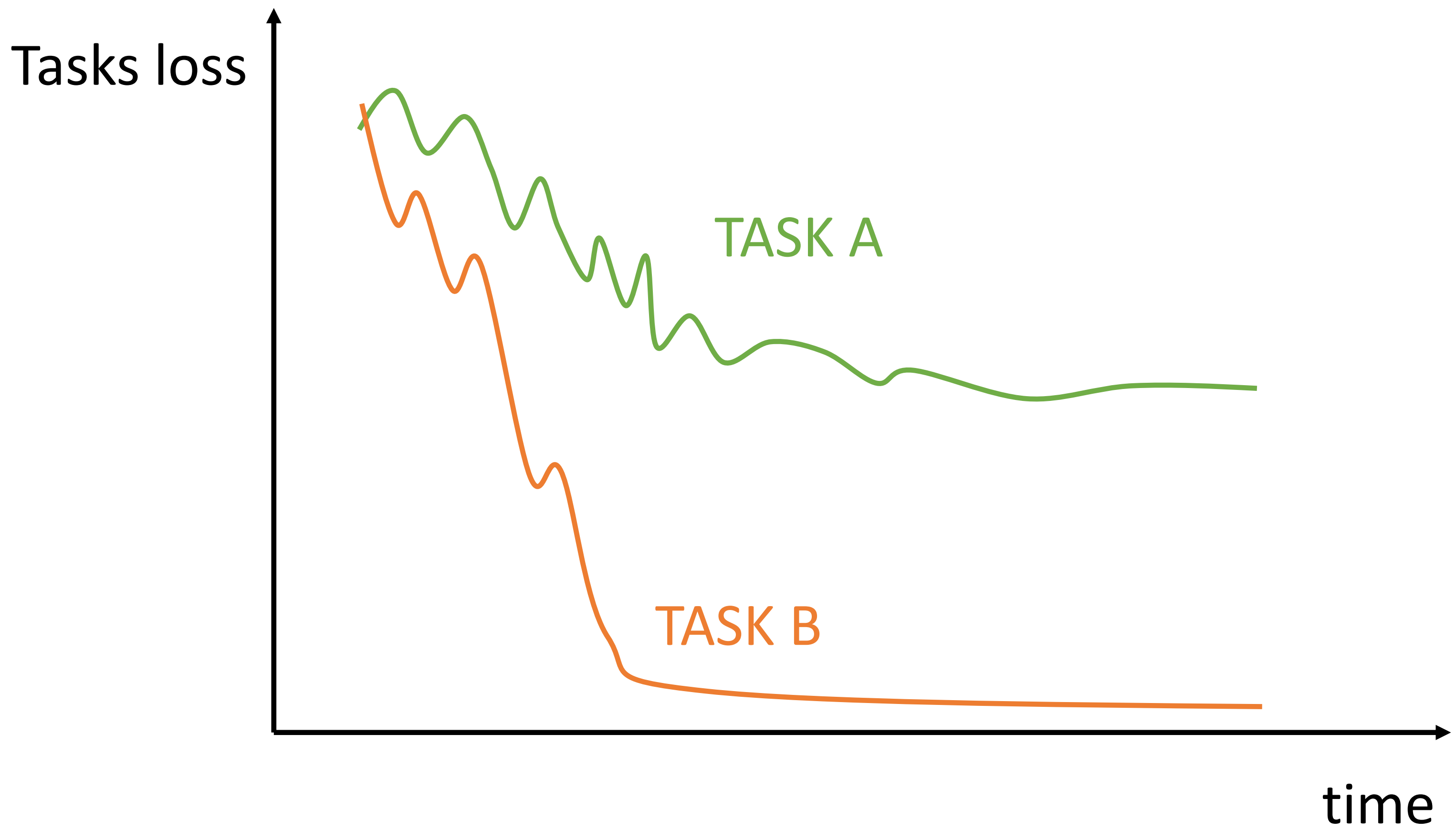
# Balancing task importance

$$\mathcal{L} = \sum_{i=1}^{N} w_i \mathcal{L}_i$$

# Why we need task balancing

Some tasks are easier to learn than others

$$\mathcal{L} = \sum_{i=1}^{N} w_i \mathcal{L}_i$$

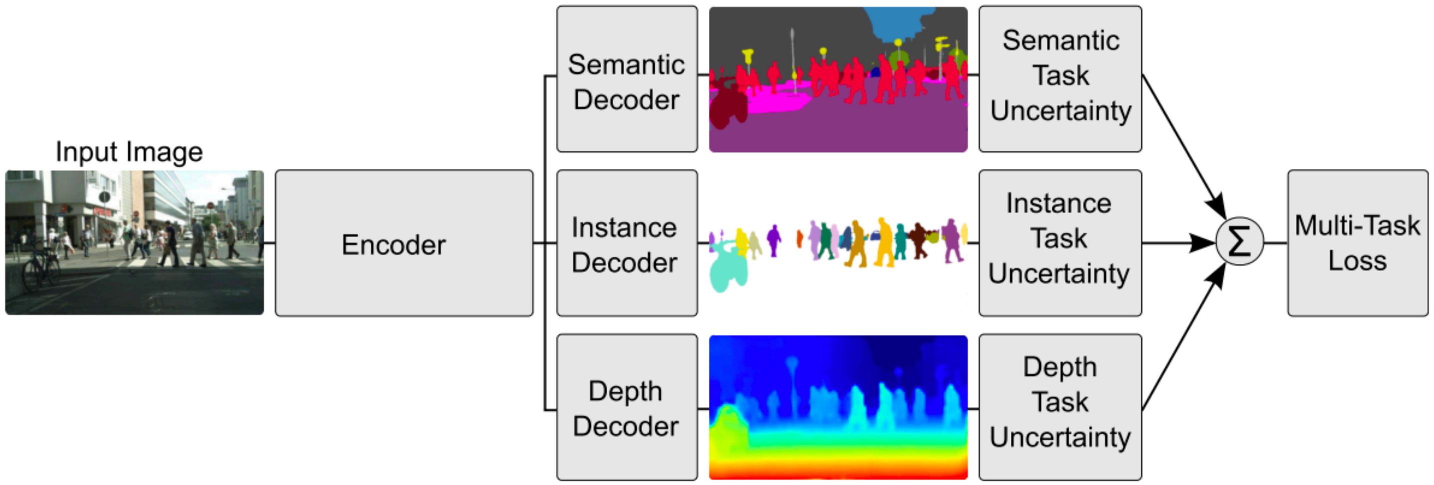



They end up dominating the learning process

# loss balancing with uncertainty

- Each task relative weight is adjusted as the uncertainty of this task

$$\mathcal{L} = \sum_{i=1}^{N} \frac{1}{2\sigma_i^2} \mathcal{L}_i(W_i) + \log\left(\prod_{i=1}^{N} \sigma_i\right)$$

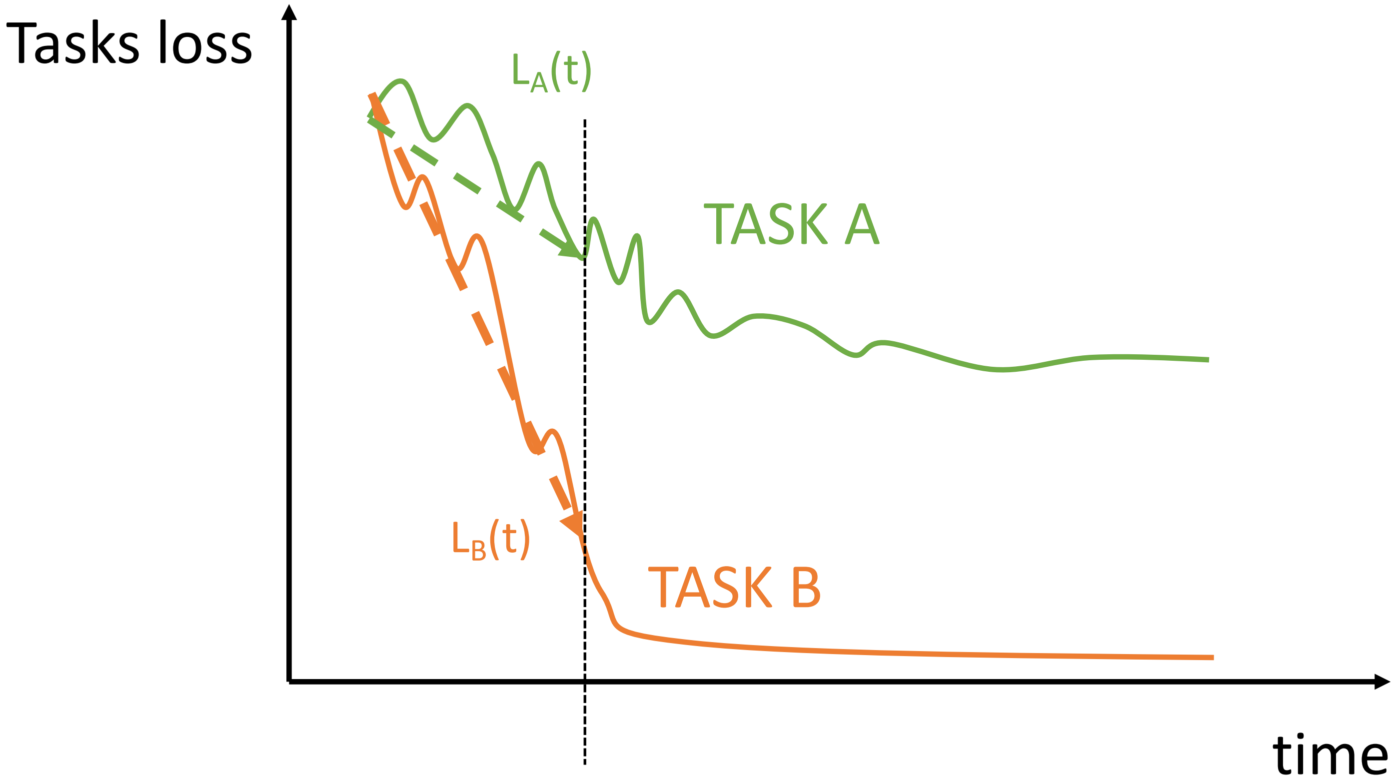



- $\sigma_i \nearrow \quad \rightarrow \quad W_i \searrow$
- $\log\left(\prod_{i=1}^{N} \sigma_i\right)$ Acts as noise regularizer

Kendall, A., Gal, Y., & Cipolla, R. (2017). Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. Retrieved from http://arxiv.org/abs/1705.07115

# loss balancing using GradNorm

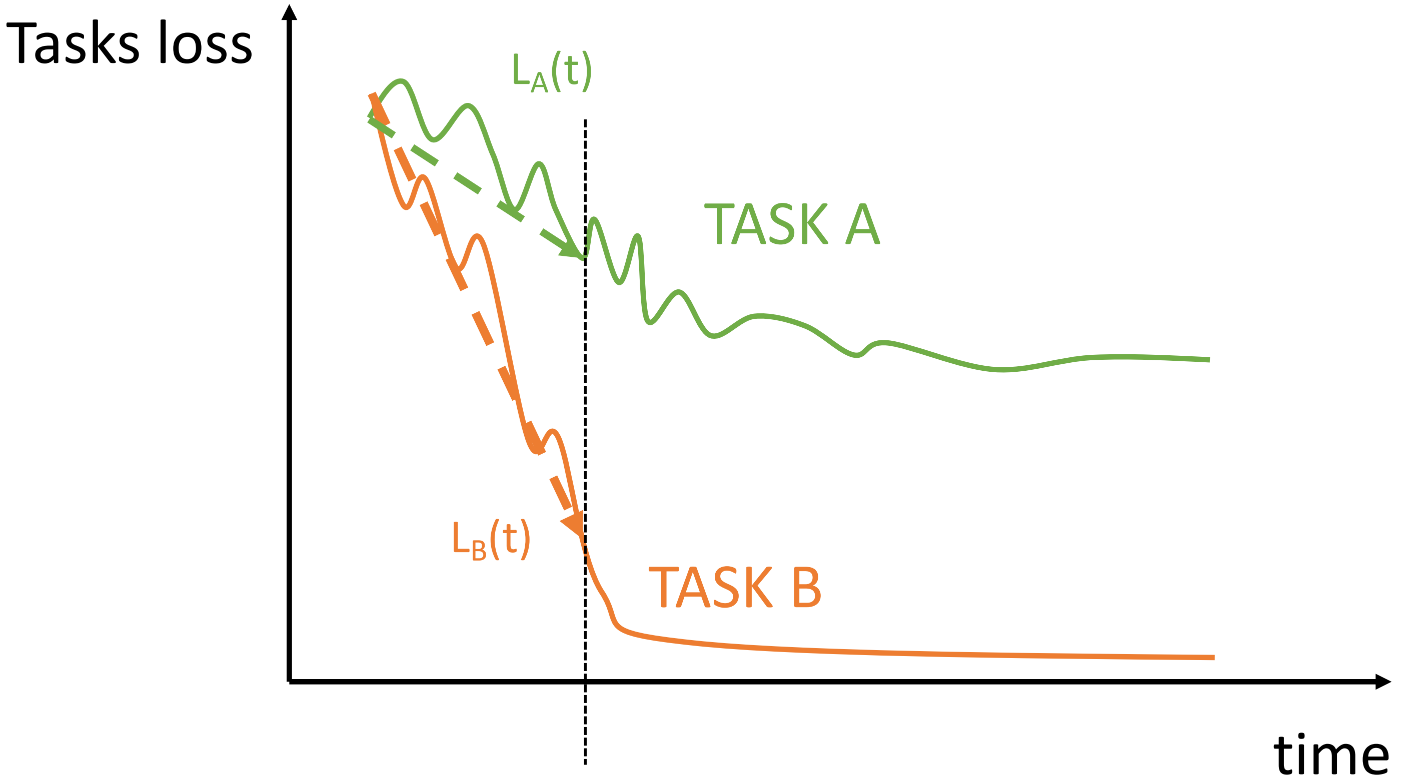
- Gradient normalisation




Chen, Z., Badrinarayanan, V., Lee, C.-Y., & Rabinovich, A. (2017), GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks, arXiv e-prints, arXiv:1711.02257.

# loss balancing using GradNorm

- Gradient normalisation



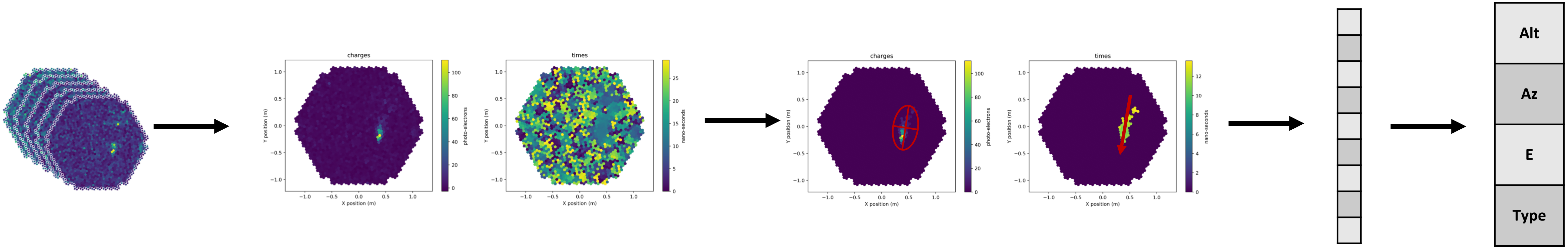

Tasks losses are weigthed by the differences of gradients between tasks
→ A faster learning task will become hard to learn so others have time to adjust

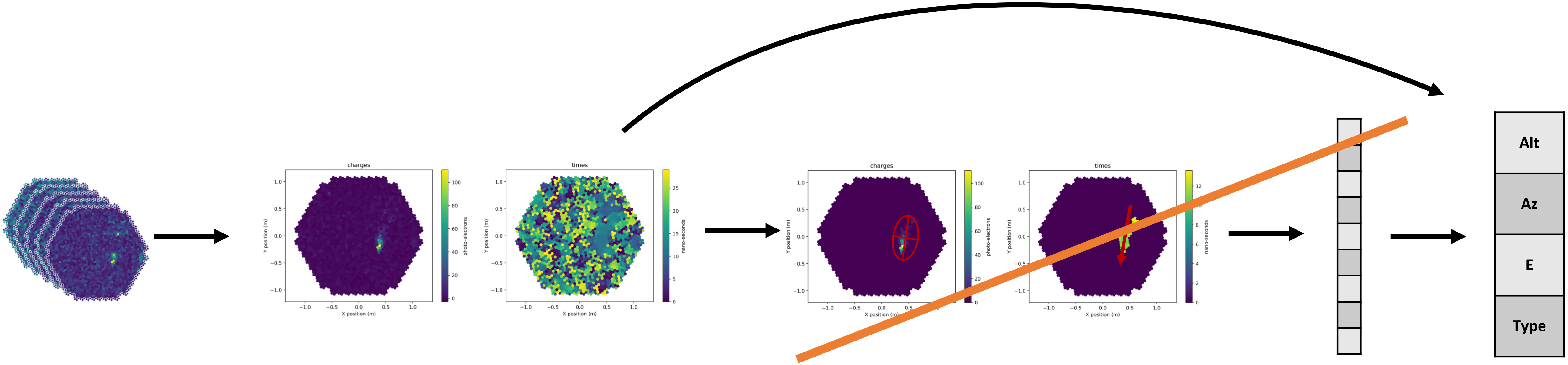
$$\mathcal{L}_{grad} = \sum_i \left| G_W^i(t) - G_W(t) r_i(t)^\alpha \right|$$

$G^i_W$ = gradient norm of task i
$G_W$ = average gradient norm
$r_i$ = relative training rate
$\alpha$ = restoring force strength

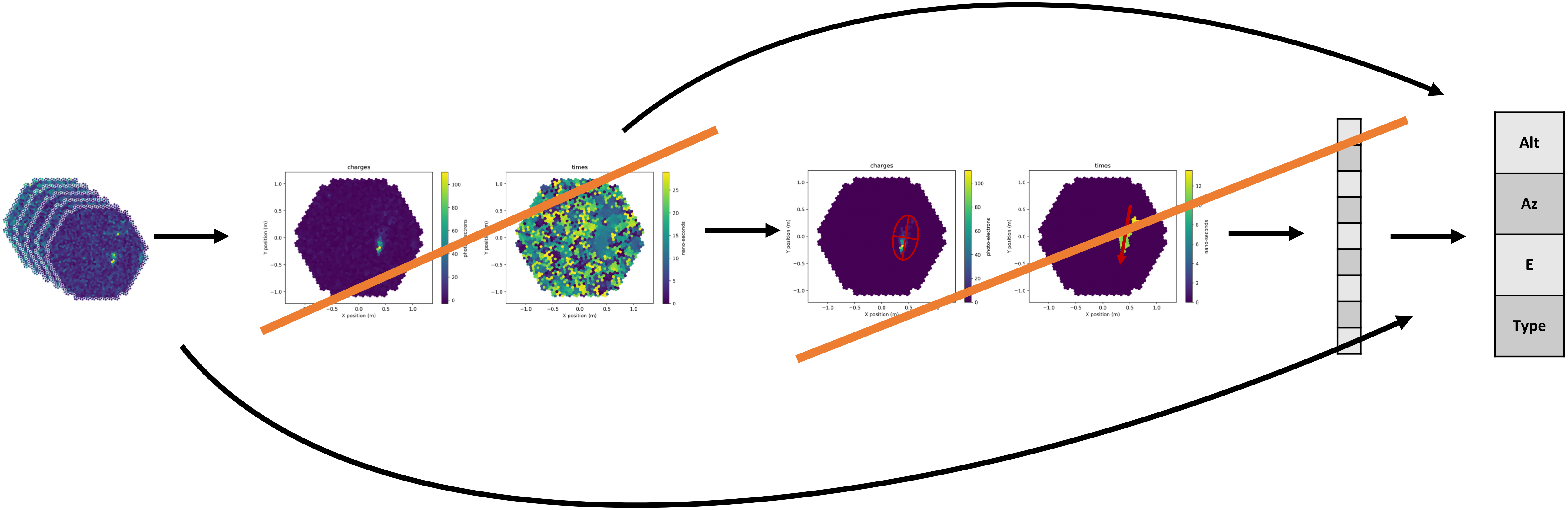
Chen, Z., Badrinarayanan, V., Lee, C.-Y., & Rabinovich, A. (2017), GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks, arXiv e-prints, arXiv:1711.02257.

Back to CTA


charges
times
X position (m)
Y position (m)
photo-electrons
nano-seconds
charges
times
Alt
Az
E
Type



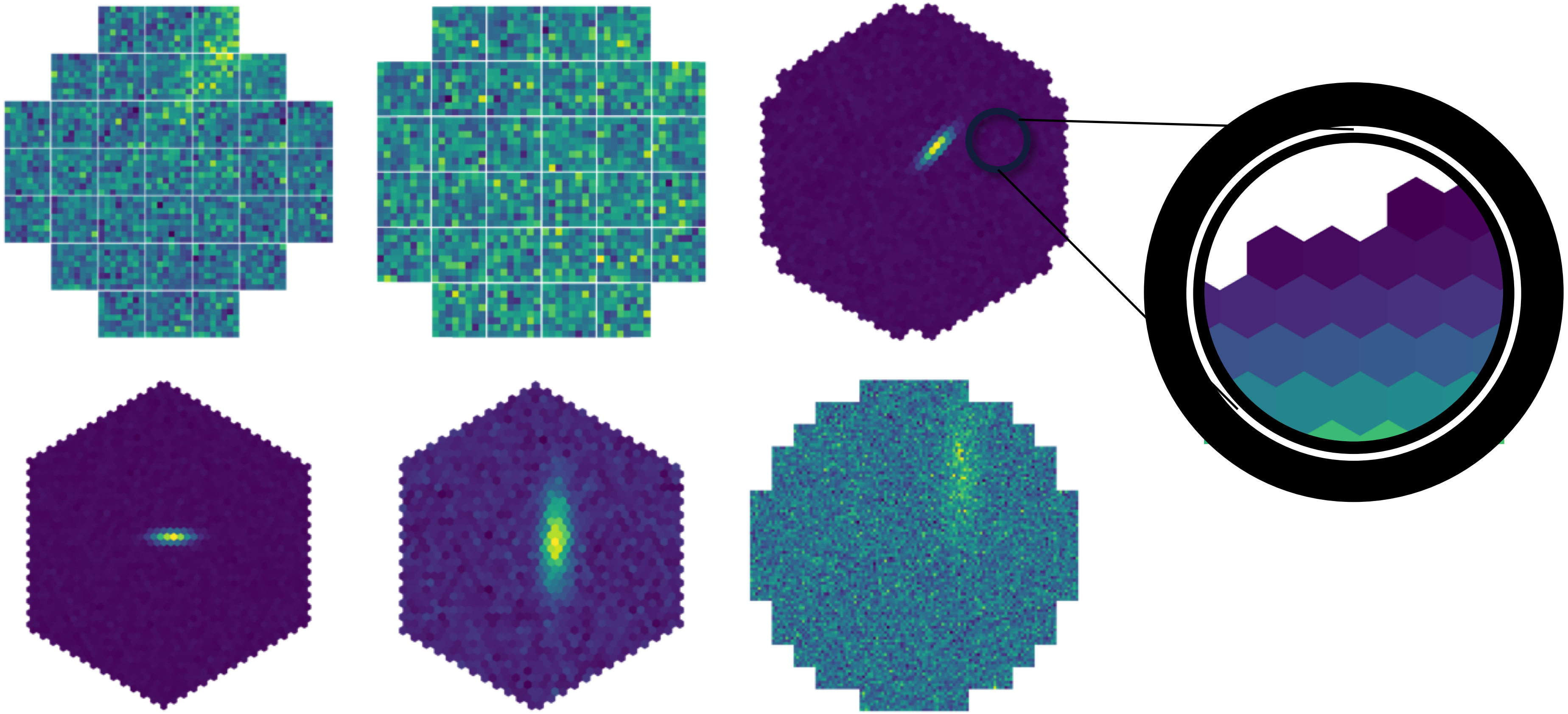

- Starting from images: getting rid of the expert engineered feature extraction




- Starting from waveforms: end-to-end system (computationaly much heavier)

- Well-suited problem for deep learning
- Well-suited problem for multi-task learning

# The hexagonal images and pixels issue

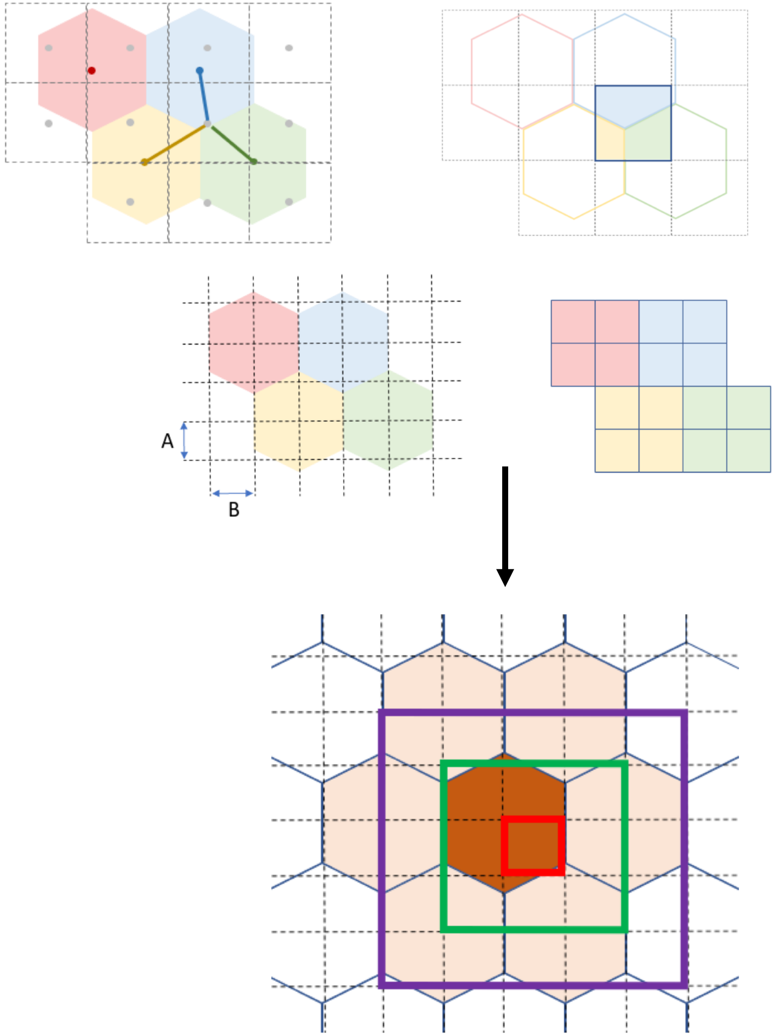


Question for the audience : How to apply convolution on that ?

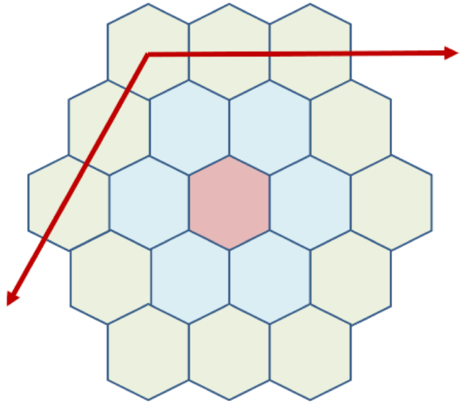
# Going deeper

Hexagonal pixel processing with deep learning

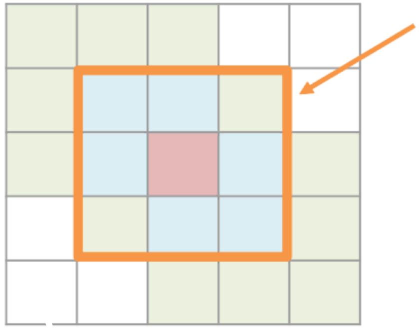
- Resampling: oversampling, rebinning, interpolation
- Image shifting + masked convolution




**Axial addressing system**



**Shifted image**



**3 x 3 Convolution Kernel**

**Kernel mask**

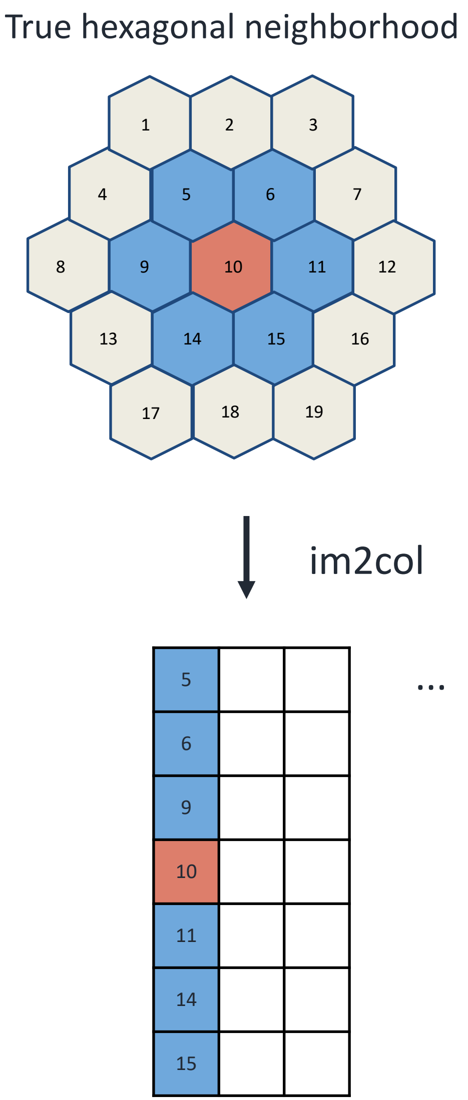
| | | |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 1 |

1170 « fake » pixels (LST)

# Indexed convolutions

- Convolution for arbitrary lattices
  - Respects the true neighborhood
  - ➞ Avoids image distortion
  - ➞ Less parameters to train

- Based on GEMM implementation of convolution
  - Convolution → matrix multiplication
  - im2col operation to select pixel based on the list of neighbors



# Indexed convolutions

Validation on CIFAR-10 (reference data set)

Hexagonal pixel vs square pixels

No significant difference

CIFAR-10 performance

| | Hexagonal kernels (i.c.) | Square kernels |
|---|---|---|
| Classification accuracy | 88.51 ± 0.21 | 88.39 ± 0.48 |

Original image

Interpolated image

Implementation optimization needed

No preprocessing

Find the open-source code at:

DOI 10.5281/zenodo.4419866

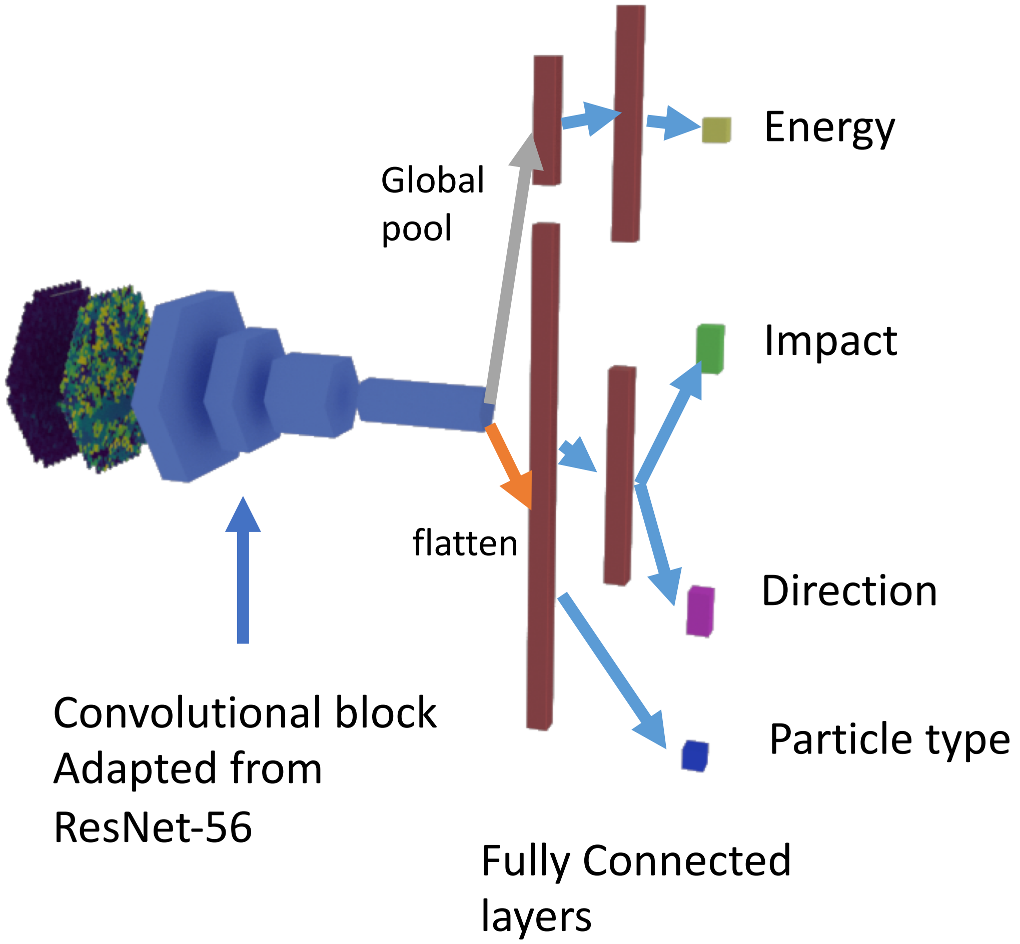
Jacquemont, M.; Antiga, L.; Vuillaume, T.; Silvestri, G.; Benoit, A.; Lambert, P. and Maurin, G. (2019). **Indexed Operations for Non-rectangular Lattices Applied to Convolutional Neural Networks**. In Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, ISBN 978-989-758-354-4, pages 362-371. DOI: 10.5220/0007364303620371

DEMO #2

See notebook deep_multi

Supported by the European Commission Framework Programme Horizon 2020 Research and Innovation action under grant agreement n. 653477 and n. 824064.

# GammaLearn - γPhysNet




- Full event reconstruction for LST1 data
- multitask learning (hard parameter sharing)
- Input = 2 channels = charges + temporal map
- Indexed Convolution
- Physically guided (global vs local features)
- Masked loss
- Uncertainty task balancing

Multi-Task Architecture with Attention for Imaging Atmospheric Cherenkov Telescope Data Analysis, VISAPP 2021, M. Jacquemont et al 2021

*M. Jacquemont's thesis

# GammaLearn - classification

Multitask advantage: $\gamma$-PhysNet vs single task classifier

| | AUC | Precision | Recall |
|---|---|---|---|
| ResNet-56 | 0.954±0.001 | 0.956±0.001 | 0.942±0.001 |
| γ-PhysNet | **0.960±0.002** | **0.957±0.003** | **0.956±0.006** |

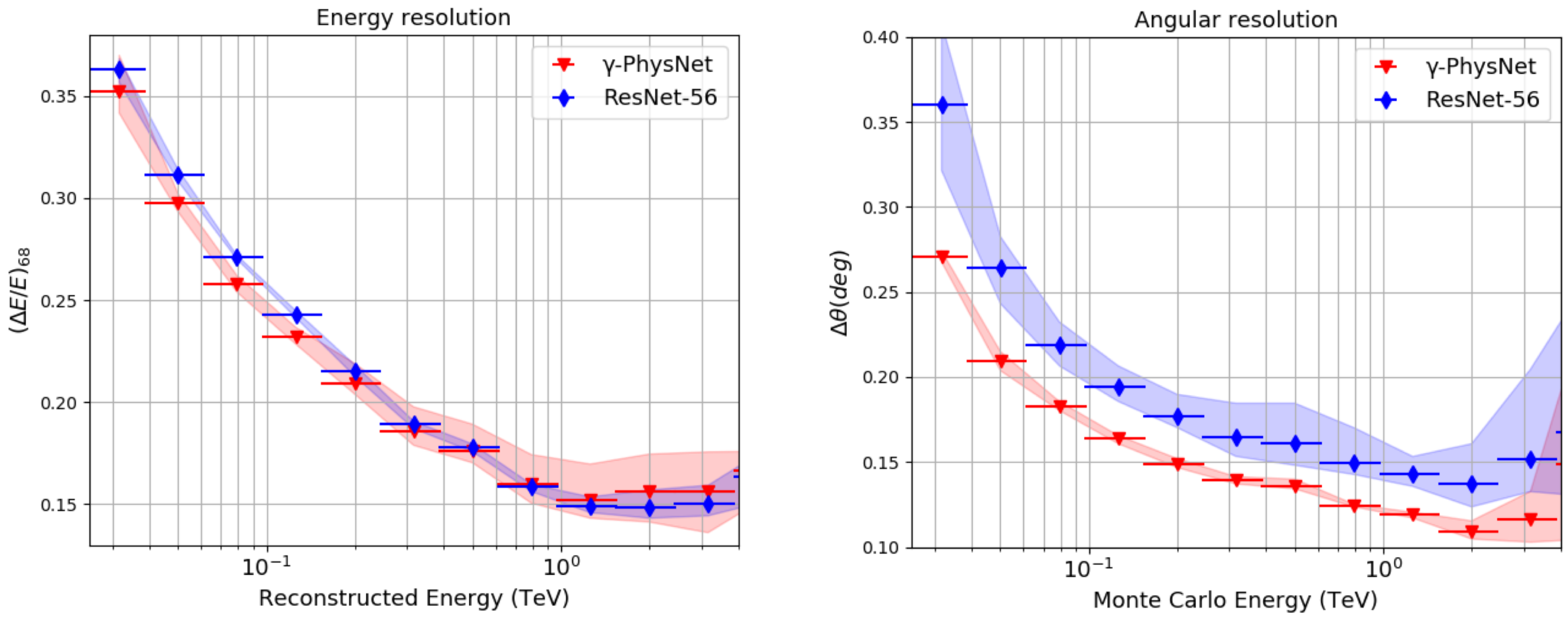
Better recall → Keep more gammas

*M. Jacquemont's thesis

# GammaLearn - regression

## Multitask advantage: γ-PhysNet vs single task regressor

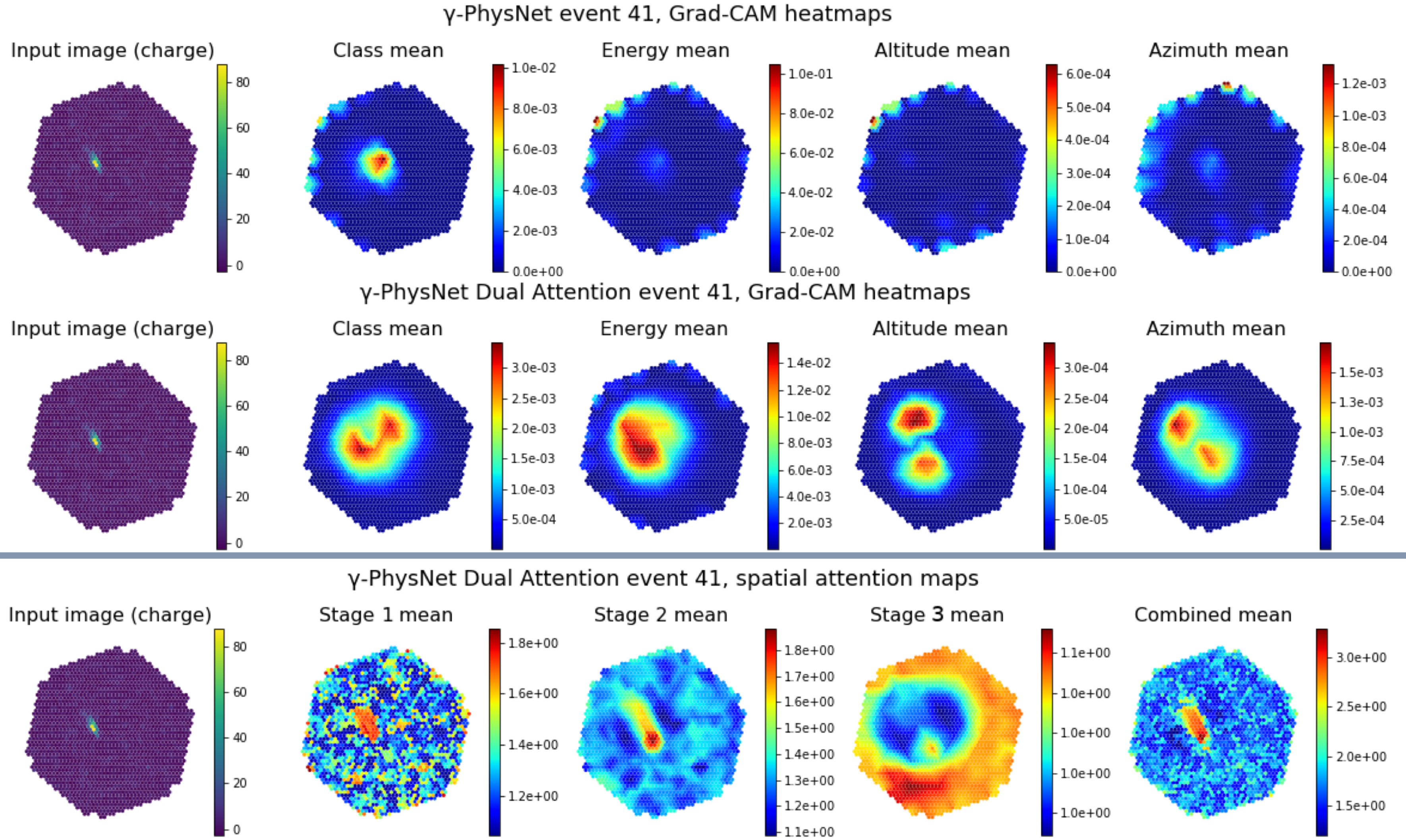
Lower
Is
better




Significant improvement on the direction reconstruction

*M. Jacquemont's thesis

# Attention, please!




**Without attention**

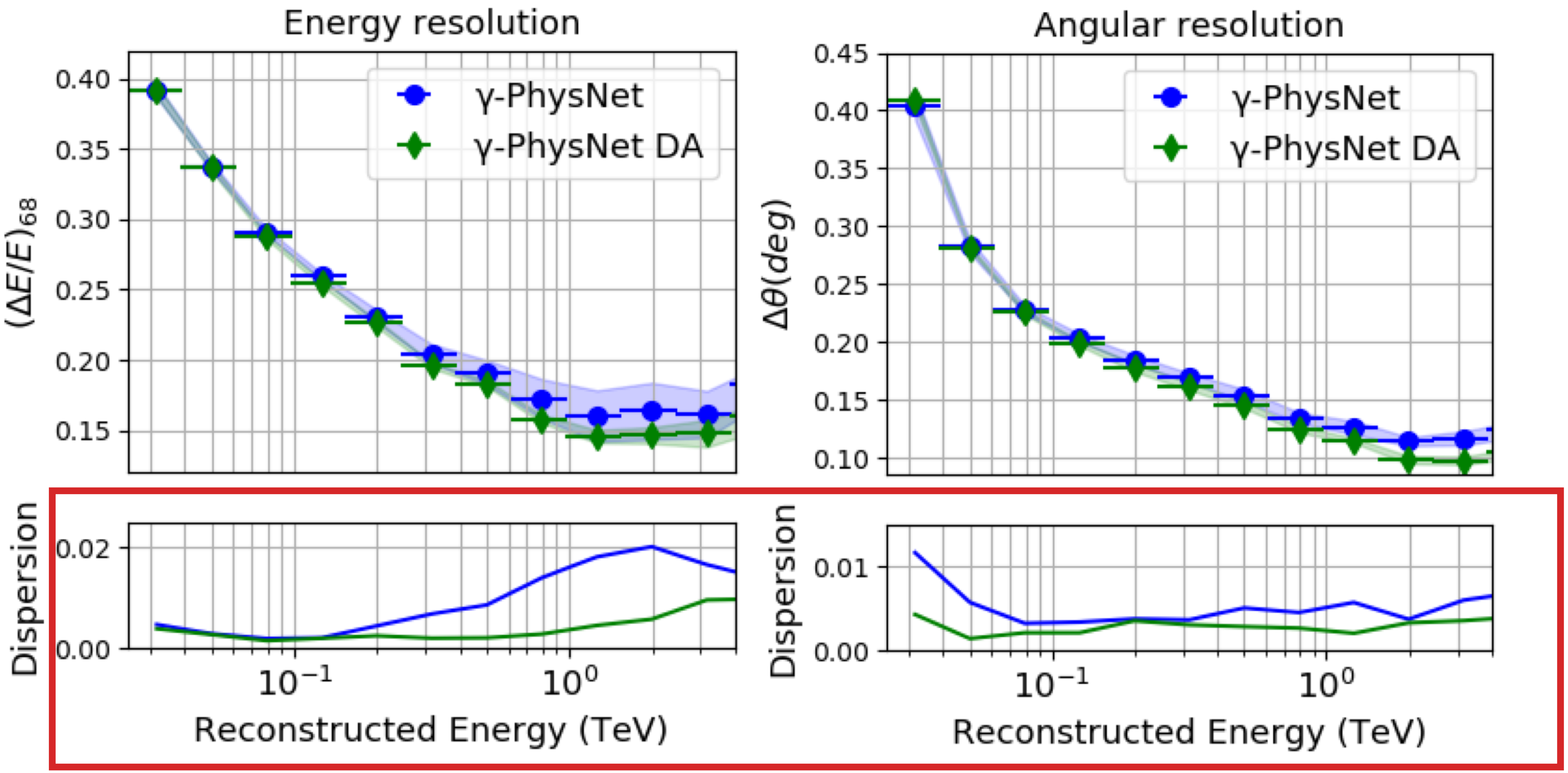
**With attention**

Deep Learning for Astrophysics, Understanding the Impact of Attention on Variability Induced by Parameter Initialization, EDL-AI ICPR 2020 workshop, M. Jaquemont et al 2020

# GammaLearn - regression

## Attention advantage on γPhysNet

Lower
Is
better

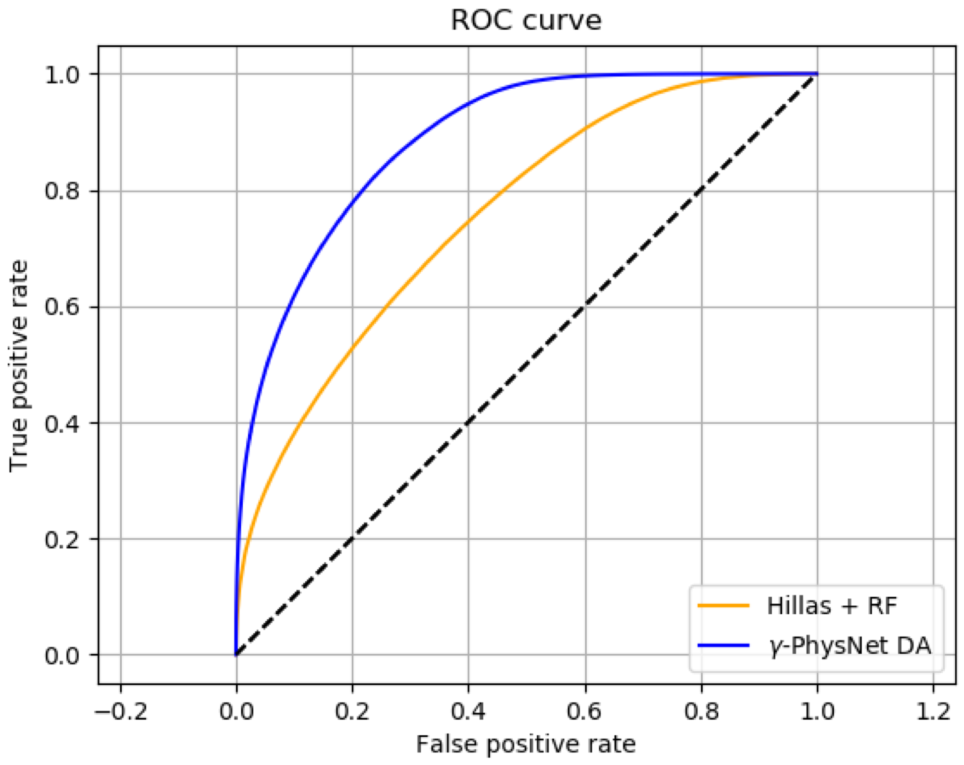



↘ dispersion

↗ robustness

*M. Jacquemont's thesis

# Comparison with standard analysis




| | AUC | Precision | Recall | Gammaness cut |
|---|---|---|---|---|
| Hillas + RF | 0.756 | 0.977 | 0.099 | 0.770 |
| γ-PhysNet DA | **0.887** | **0.981** | **0.277** | 0.948 |

γ-PhysNet DA

➡ Slightly better precision

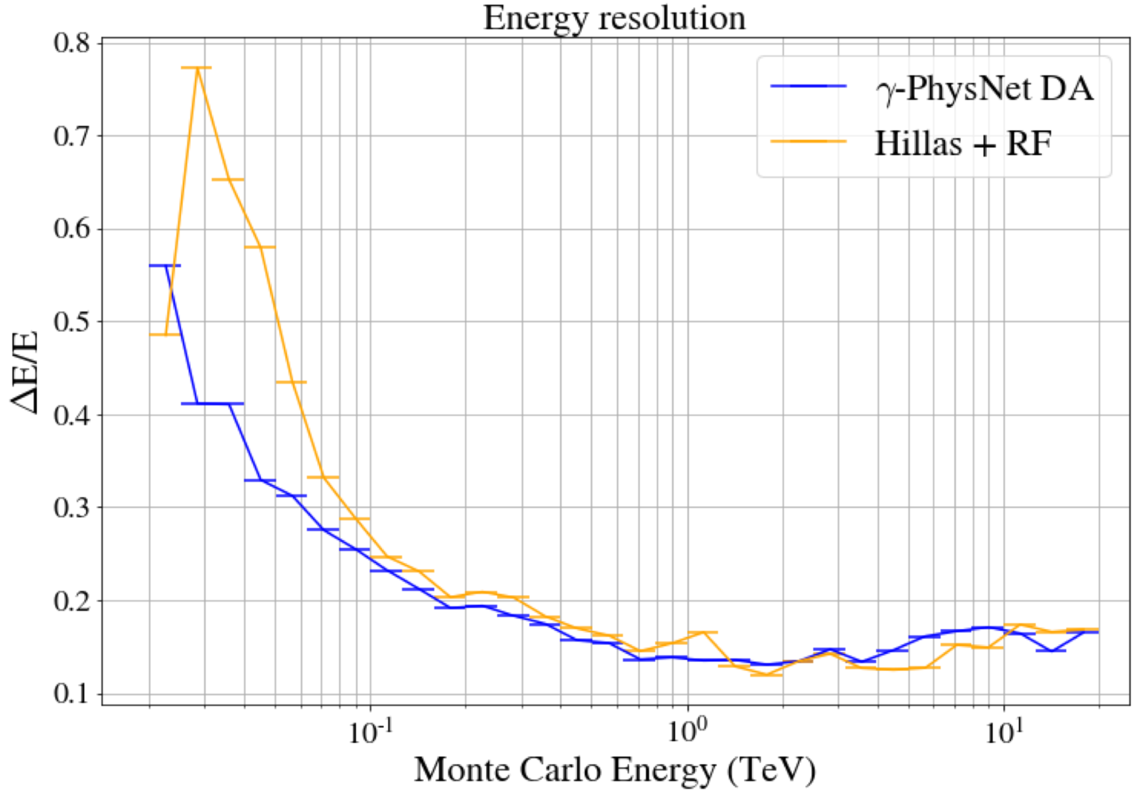
➡ Better recall

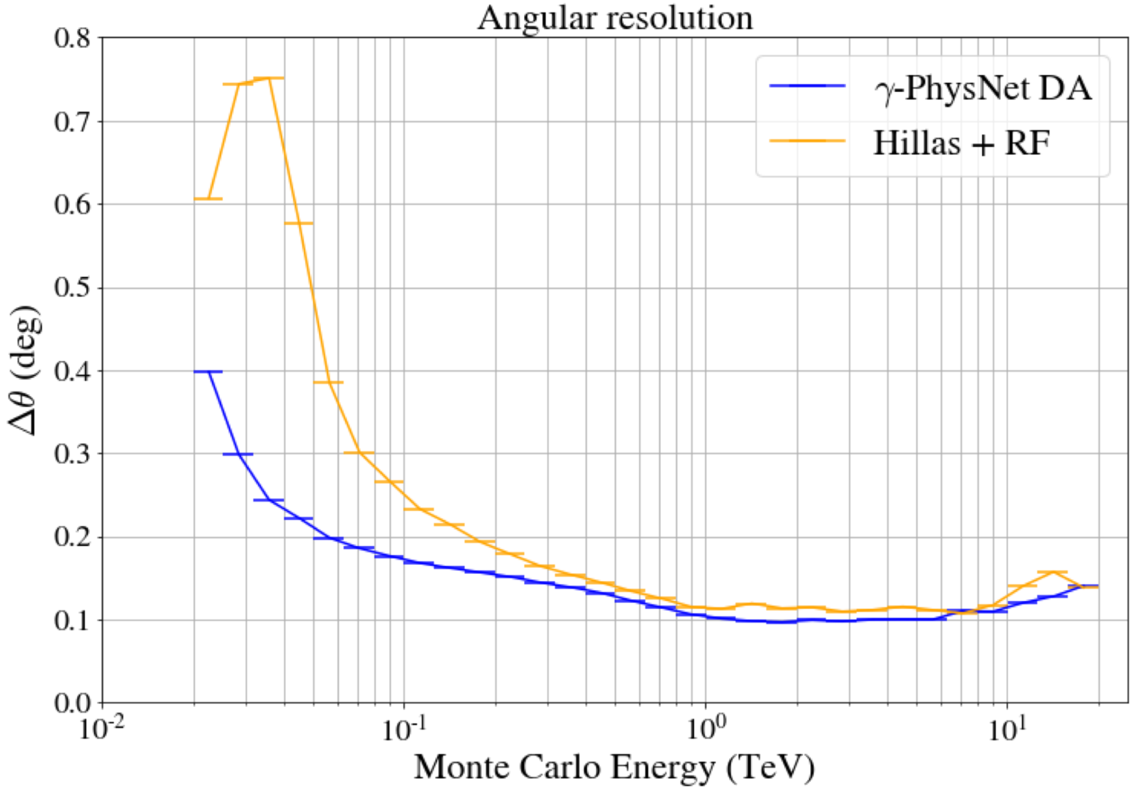
➡ Better overall performance

Multi-Task Architecture with Attention for Imaging Atmospheric Cherenkov Telescope Data Analysis, VISAPP 2021, M. Jacquemont et al 2021

*M. Jacquemont's thesis

# Comparison with standard analysis




➞ γ-PhysNet DA is significantly better at low energies

Multi-Task Architecture with Attention for Imaging Atmospheric Cherenkov Telescope Data Analysis, VISAPP 2021, M. Jacquemont et al 2021

*M. Jacquemont's thesis

More to come !

Currently working on assessing the sensitivity gain on real data…

# Some important points not covered here…

What we covered today:

- What is multitask learning
- How it is implemented with a real use-case: Imaging Atmospheric Cherenkov telescopes event reconstruction

Important points not covered today:

- Stereoscopy : merging prediction from each telescope in CTA
- More complex deep multitask learning are being invented these days where the structure itself of the network and thus the relation between the tasks is learnt

# Some useful references


- *Sebastian Ruder (2017). An Overview of Multi-Task Learning in Deep Neural Networks. arXiv preprint arXiv:1706.05098.*
- https://www.coursera.org/lecture/machine-learning-projects/multi-task-learning-l9zia
- Baxter, J. (1997). A Bayesian/information theoretic model of learning to learn via multiple task sampling. Machine Learning, 28, 7–39. Retrieved from http://link.springer.com/article/10.1023/A:1007327622663
- Zhanpeng Zhang, Ping Luo, Chen Change Loy, et al. “Facial Landmark Detection by Deep Multi-task Learning”. In: Computer Vision — ECCV 2014: 13th European Conference, Zurich, Switzerland, Cham: Springer International Publishing, 2014, pp. 94–108.