Search for Meteors in the Mini-EUSO Orbital Telescope Data with Neural Networks

Mikhail Zotov[1] **for the JEM-EUSO collaboration**

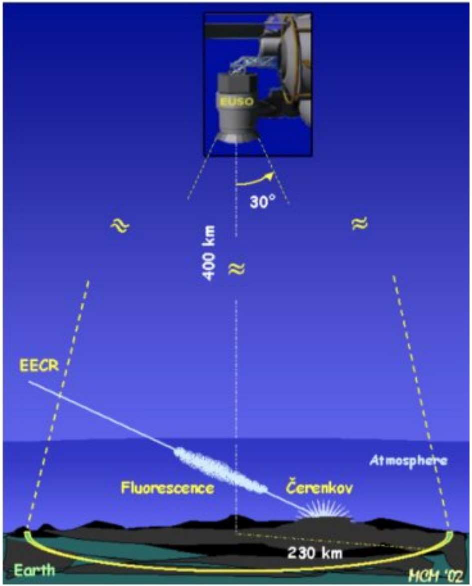
D. V. Skobeltsyn Institute of Nuclear Physics, M. V. Lomonosov Moscow State University

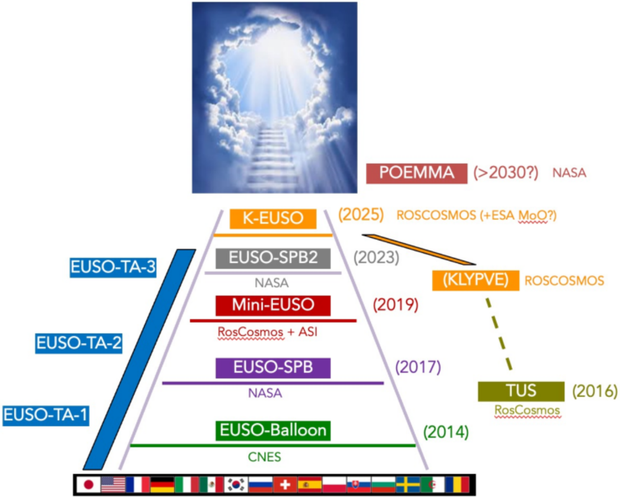
The 7th International Conference on Deep Learning in Computational Physics, Peterhof, 2023

[1]Supported by the Russian Science Foundation grant number 22-22-00367

JEM-EUSO: the goal of observing UHECRs from space (E $\gtrsim$ 50 EeV)

1980: Linsley, Benson:

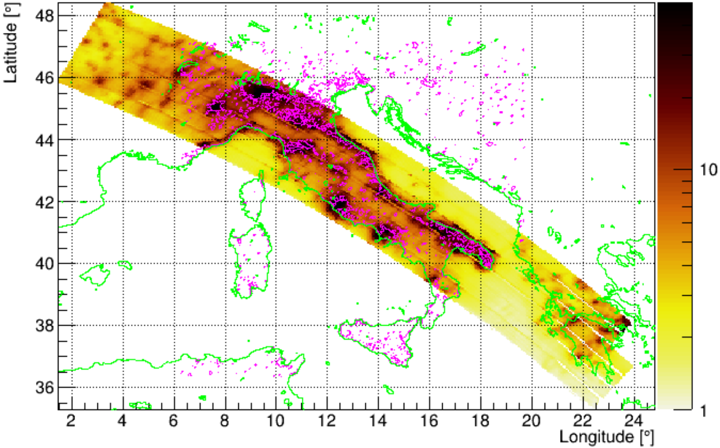




(Image: E. Parizot)

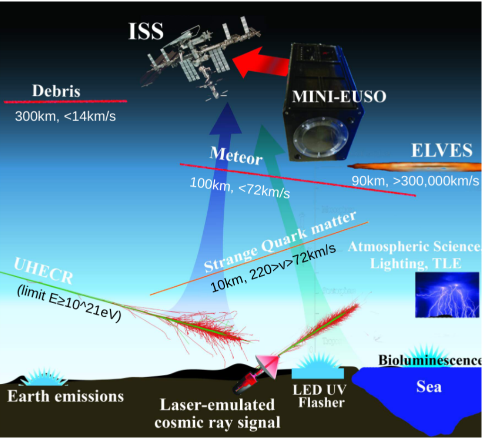
JEM-EUSO collaboration: 16 countries, including Russia

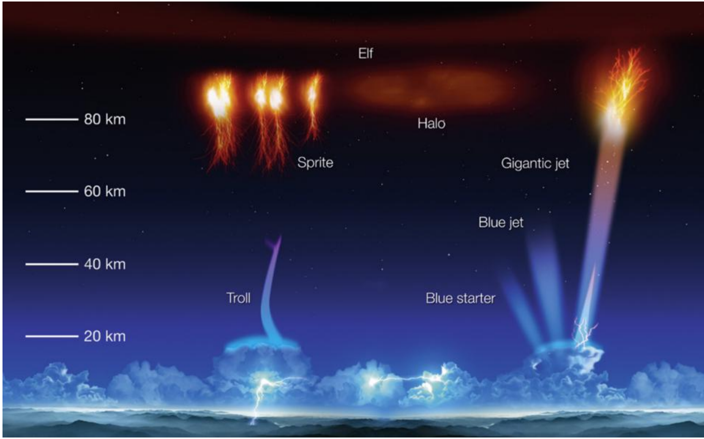
Mini-EUSO ("UF Atmosfera") telescope onboard the ISS since 2019



Two Fresnel lenses $\varnothing = 25$ cm, focal surface 48×48 pixels. **Time resolution:** $T_1 = 2.5\ \mu$s, $128\,T_1$, $128^2\,T_1 = 40.96$ ms. **Field of view:** $\sim 300 \times 300$ km^2; one pixel: $\sim 6.3 \times 6.3$ km^2. **UV range!**

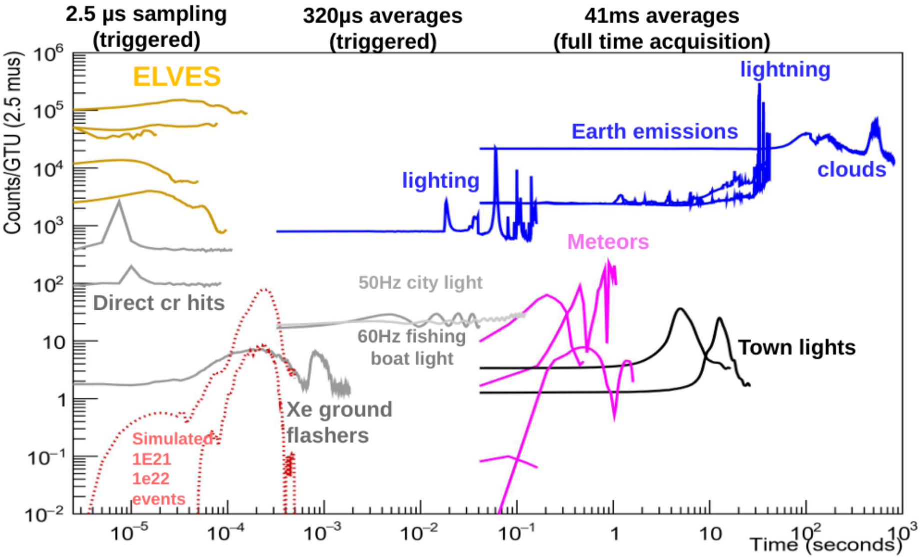
Variety of UV illumination in the Earth atmosphere





The variety of UV illumination is enormous! Figures: Mini-EUSO collaboration, arXiv:2201.01213, ESO.

Zoo of signal shapes and durations

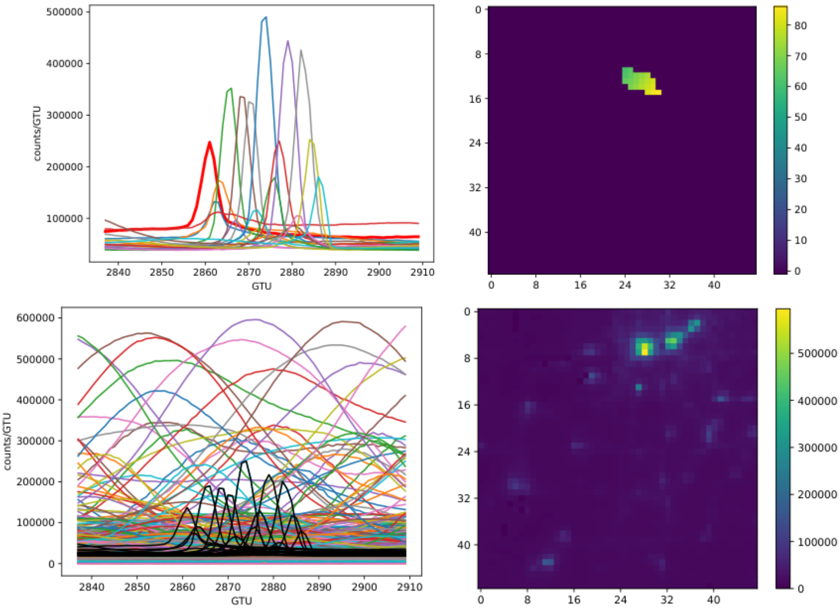


Typical signal shapes at different time scales [arXiv:2201.01213]

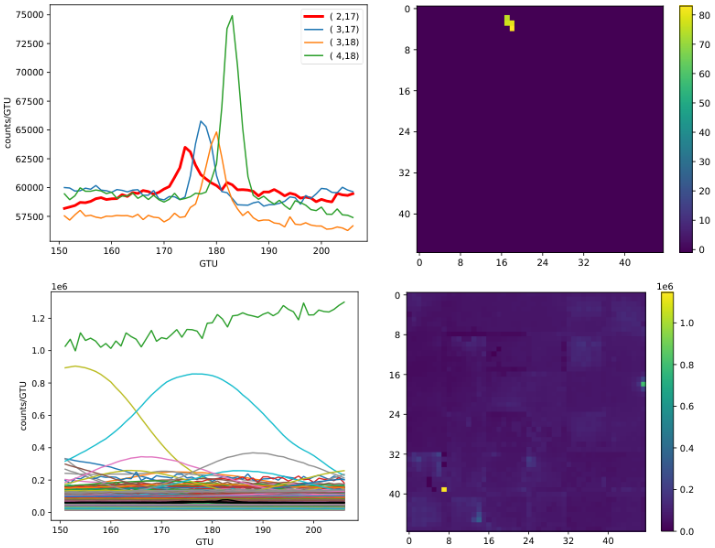
Meteors

Everybody has seen them! (Pic source: redorbit.com)

A clearly pronounced meteor (a long bright track, many hit pixels)



A typical meteor



Meteor signal features:

- Gaussian-like shape of signals
- Quasi-linear tracks on the FS
- Signal movement (except small zenith angles)
- Low amplitude comparing to the "background"
- Small footprint on the FS

Besides this:

- Different observation conditions (Moon phase, BG illumination)
- The FS is moving
- Many simultaneous flashes with similar shapes, mostly over large areas

Meteor signal recognition as binary classification

Dataset: 8 sessions of observations from 2019/11/27 till 2020/04/01 (approximately 12 hours each)

Original data representation: "continuos" data flow with time resolution 40.96 ms

No data cleaning, no pixel calibration (flat-fielding)

Meteor dataset: 1068 meteors selected with a conventional algorithm

Important: there are no ground-truth labels since the origin of signals is sometimes unclear

Two-step procedure for recognizing meteor signals:

1. Find 3D data chunks of size $M \times M \times N$ that contain meteor signals:
   $M \times M$ pixels on the FS, N time steps (a task of binary classification)
2. Recognize hit pixels in meteor data chunks selected during Step 1.
   (another binary classification)

Any 7 sessions for training, 1 for testing, $M \times M = 48 \times 48$**, different** N**. Tried CNN, LSTM.**

Meteor signal recognition as binary classification

Dataset: 8 sessions of observations from 2019/11/27 till 2020/04/01 (approximately 12 hours each)

Original data representation: "continuos" data flow with time resolution 40.96 ms

No data cleaning, no pixel calibration (flat-fielding)

Meteor dataset: 1068 meteors selected with a conventional algorithm

Important: there are no ground-truth labels since the origin of signals is sometimes unclear

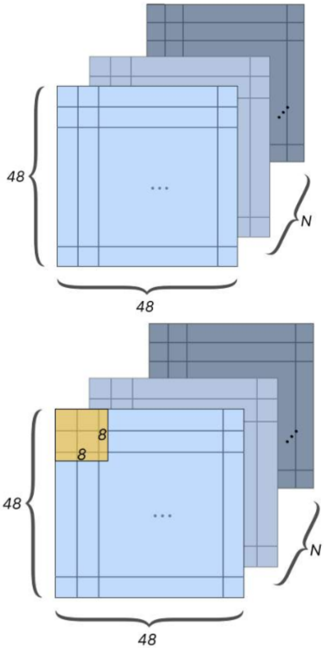
Two-step procedure for recognizing meteor signals:

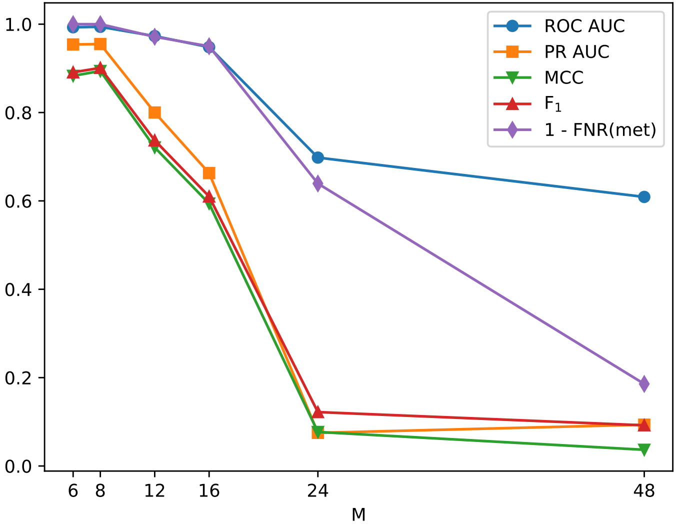
1. Find 3D data chunks of size $M \times M \times N$ that contain meteor signals:
   $M \times M$ pixels on the FS, N time steps (a task of binary classification)
2. Recognize hit pixels in meteor data chunks selected during Step 1.
   (another binary classification)

Any 7 sessions for training, 1 for testing, $M \times M = 48 \times 48$**, different** N**. Tried CNN, LSTM.**

No-go regardless of architecture/hyperparameters! ROC AUC $\lesssim$ **0.75**

Input data rearrangement: split the focal surface into pieces





Dependence of performance metrics on M for $N = 48$ (session 6)

Step 1 results (CNN): recognition of 8×8×48 "meteor" chunks

Table 1. Performance of the CNN on different sessions of observations.

Test session	5	6	7	8	11	12	13	14
ROC AUC	0.992	0.994	0.999	0.993	0.994	0.998	0.993	0.996
PR AUC	0.937	0.955	0.876	0.933	0.946	0.973	0.931	0.956
MCC	0.872	0.894	0.732	0.857	0.888	0.921	0.782	0.901
F1	0.873	0.901	0.718	0.863	0.892	0.922	0.776	0.904
FNR(met)	0/65	0/280	0/18	0/186	0/193	0/106	0/90	0/130

Columns: test sessions. The CNN was trained on data of all other sessions.
Bottom line: lost meteors/total meteors in the test session
Test data: all meteor chunks + 100,000 non-meteor chunks

Zero of 1068 meteors are lost!

NB: standard metrics vary strongly from one session to another ⇒ none is perfect in our case

Step 2 results (2-layer MLP): recognition of “meteor” pixels

Table 2. Performance of the MLP on different sessions of observations.

Test session	5	6	7	8	11	12	13	14
ROC AUC	0.992	0.995	0.993	0.994	0.996	0.993	0.995	0.995
PR AUC	0.899	0.932	0.877	0.916	0.932	0.887	0.936	0.928
MCC	0.790	0.841	0.744	0.826	0.847	0.812	0.809	0.835
F1	0.794	0.845	0.737	0.832	0.852	0.814	0.810	0.840
FNR(pxl)	2/422	2/1428	0/80	7/928	5/958	0/492	0/457	2/630

Bottom line: lost meteor pixels/total meteor pixels in the test session

16 out of 5395 hit pixels were not recognized: $\langle \mathrm{FNR(pxl)} \rangle \approx 0.3\%$

The worst result: $\mathrm{FNR(pxl)} \approx 0.75\%$ for session 8

Accuracy better than 99%

A pipeline of a simple CNN and a 2-layer MLP recognizes meteor tracks in Mini-EUSO data with accuracy $>$99%

Remarks:

- Is this a silver bullet? No, it is not: 3D chunks used for the CNN are overlapping to avoid signal loss $\Rightarrow$ duplicate entries in the output $\Rightarrow$ extra work is needed to clean the output
- There is a considerable number of false positives (though some of them are not false)
- It can be helpful to complement the training set with simulations
- It is seemingly possible to do the work in one step by another input data arrangement — work in progress

Discussion & conclusions

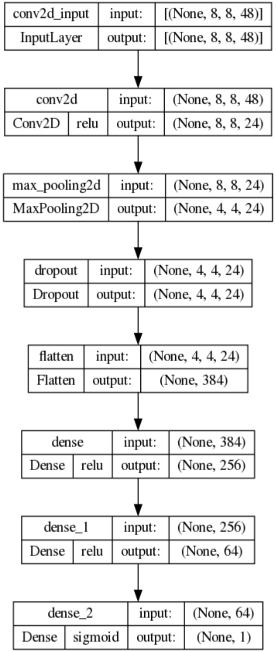
A pipeline of a simple CNN and a 2-layer MLP recognizes meteor tracks in Mini-EUSO data with accuracy $>$99%

Remarks:

- Is this a silver bullet? No, it is not: 3D chunks used for the CNN are overlapping to avoid signal loss $\Rightarrow$ duplicate entries in the output $\Rightarrow$ extra work is needed to clean the output
- There is a considerable number of false positives (though some of them are not false)
- It can be helpful to complement the training set with simulations
- It is seemingly possible to do the work in one step by another input data arrangement — work in progress

Thank you for your attention!

Backup: CNN architecture



Convolutional layer with 24 filters and a kernel of size 3
Activation function: ReLU
L2 kernel regularizer with factor 0.1
Maxpooling and dropout layers
Two FCL 256, 64 neurons
Optimization: Adam
Loss: Binary crossentropy
Output activation function: sigmoid

Data transformation: StandardScaler (scikit-learn)
Data augmentation: rotation

Performance metrics (intrinsically unbalanced data!)

ROC AUC: area under the Receiver Operating Characteristic curve (TPR vs. FPR)

PR AUC: area under the precision-recall curve

$$\text{Precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \qquad \text{Recall} = \text{TPR} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$$

Matthews correlation coefficient:

$$\text{MCC} = \frac{\mathrm{TP}\cdot\mathrm{TN}-\mathrm{FP}\cdot\mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}$$

F_1 **score:** the harmonic mean of precision and recall

$$F_1 = \frac{2\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}$$

Final goal: minimize **FNR**$(\mathrm{met},\mathrm{pxl})$ (the false negative rate in terms of meteor signals)