Refined STACK-CNN for Meteor and Space Debris Detection in Highly Variable Backgrounds

Leonardo Olivi*[1], Antonio Montanaro‡[22], Mario Edoardo Bertaina†[1,2], Antonio Giulio Coretti [1,2], Dario Barghini [1,2,3], Matteo Battisti [2,9], Alexander Belov [4,5], Marta Bianciotto [1], Francesca Bisconti [6], Carl Blaksley [8], Sylvie Blin [9], Karl Bolmgren [10], Giorgio Cambiè [6,7], Francesca Capel [11], Marco Casolino [6,7,8], Igor Churilo [12], Marino Crisconio [13], Christophe De La Taille [14], Toshikazu Ebisuzaki [8], Johannes Eser [15], Francesco Fenu [16], George Filippatos [15], Massimo Alberto Franceschi [18], Christer Fuglesang [10], Alessio Golzio [1,2], Philippe Gorodetzky [9], Fumiyoshi Kajino [19], Hiroshi Kasuga [8], Pavel Klimov [5], Viktoria Kungel [17], Vladimir Kuznetsov [12], Massimiliano Manfrin [1,2], Laura Marcelli [6], Gabriele Mascetti [13], Włodzimierz Marszał [20], Marco Mignone [2], Hiroko Miyamoto [1,2], Alexey Murashov [5], Tommaso Napolitano [18], Hitoshi Ohmori [8], Angela Olinto [15], Etienne Parizot [9], Piergiorgio Picozza [6,7], Lech Wiktor Piotrowski [21], Zbigniew Plebaniak [1,2,7], Guillaume Prévôt [9], Enzo Reali [6,7], Marco Ricci [18], Giulia Romoli [6,7], Naoto Sakaki [8], Sergei Sharakin [5], Kenji Shinozaki [20], Jacek Szabelski [23], Yoshiyuki Takizawa [8], Valerio Vagelli [13], Giovanni Valentini [13], Michal Vrabel [20], Lawrence Wiencke [17], Mikhail Zotov [5]

[1] Department of Physics, University of Turin, V. P. Giuria 1, 10125 Turin, Italy
[2] INFN Section of Turin, Via P. Giuria 1, 10125 Turin, Italy
[3] INAF Astrophysics Observatory of Turin, Via Osservatorio 20, 10025 Pino Torinese, Italy
[4] Faculty of Physics, M.V. Lomonosov Moscow State University, ul. Kolmogorova 1(2), 119234 Moscow, Russia
[5] Skobeltsyn Institute of Nuclear Physics, Lomonosov Moscow State Univ., ul. Kolmogorova 1(2), 119991 Moscow, Russia
[6] INFN Section of Rome Tor Vergata, Via della Ricerca Scientifica 1, 00133 Rome, Italy
[7] Department of Physics, University of Rome Tor Vergata, Via della Ricerca Scientifica 1, 00133 Rome, Italy
[8] RIKEN, 2-1 Hirosawa, Wako, Saitama 351-0198, Japan
[9] Université Paris Cité, CNRS, Astroparticule et Cosmologie, 10 Rue Alice Domon et Léonie Duquet, 75013 Paris, France
[10] KTH Royal Institute of Technology, Brinellvgen 8, 114 28 Stockholm, Sweden
[11] Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany
[12] S.P. Korolev Rocket and Space Corporation Energia, Lenin str., 4a Korolev, 141070 Moscow area, Russia
[13] ASI, Italian Space Agency, Via del Politecnico, 00133 Rome, Italy
[14] Omega, Ecole Polytechnique, CNRS/IN2P3, Rte de Saclay, 91120 Palaiseau, France
[15] Department of Astronomy and Astrophysics, The University of Chicago, 5640 S. Ellis Avenue, Chicago IL 60637, US
[16] Karlsruhe Institute of Technology, Institute for Astroparticle Physics, 76021 Karlsruhe, Germany
[17] Department of Physics, Colorado School of Mines, 1523 Illinois St., Golden CO 80401, US
[18] INFN National Laboratories of Frascati, Via Enrico Fermi 54, 00044 Frascati, Italy
[19] Department of Physics, Konan University, 8-9-1 Okamoto, Higashinada, Kobe, Hyogo 658-8501, Japan
[20] National Centre for Nuclear Research, Andrzeja Sołtana 7, 05-400 Otwock, Poland
[21] Faculty of Physics, University of Warsaw, Ludwika Pasteura 5, 02-093 Warsaw, Poland
[22] Polytechnic University of Turin, Corso Duca degli Abruzzi 24, 10129 Turin, Italy
[23] Stefan Batory Academy of Applied Sciences, Batorego 64, 96-100 Skierniewice, Poland

Corresponding Authors: *leonardo.olivi.98@gmail.com, †bertaina@to.infn.it, ‡antonio.montanaro@polito.it

***Abstract*—In this work we present cutting-edge machine learning based techniques for the detection and reconstruction of meteors and space debris in the Mini-EUSO experiment, a detector installed on board of the International Space Station (ISS), and pointing towards the Earth. We base our approach on a recent technique, the Stack-CNN, originally developed as an online trigger in a orbiting remediation system to detect space debris. Our proposed method, the Refined Stack-CNN (R-Stack-CNN), makes the STACK-CNN more robust thanks to a Random Forest (RF) that learns the temporal development of these events in the camera. We prove the flexibility of our method by showing that it is sensitive to any space object that moves linearly in the field of view. First, we search small space debris, never observed by Mini-EUSO. Due to the limiting statistics, also in this case no debris were found. However, since meteors produce signals similar to space debris but they are much more frequent, the R-Stack-CNN is adapted to identify such events while avoiding the numerous false positives of the Stack-CNN. Results from real data show that the R-Stack-CNN is able to find more meteors than a classical thresholding method and a new method of two neural networks. We also show that the method is also able to accurately reconstruct speed and direction of meteors with simulated data.**

***Index Terms*—Neural network applications, Space technology**

I. Introduction

How safe is the space environment around the Earth? This is an important question that is worrying many space agencies and scientists in recent years. The exploration and utilization of Earth's orbit are no longer confined to the realms of governmental space agencies. The rapid growth of the commercial space sector has led to a new era of innovation and opportunity, with private companies launching satellites for telecommunications, Earth and space observation, and navigation, among other purposes.

Considering this evolving landscape, numerous questions and challenges emerge, demanding careful consideration and collaborative action. At the forefront is the pressing need for effective space traffic management. With an ever-increasing number of satellites, spacecraft, and space debris sharing the

same orbital pathways, the risk of collisions and congestion poses a significant threat to the sustainability and safety of space activities.

In this paper we consider the problem of the detection of small Space Debris (SD) [1], i.e. parts of defunct satellites and rockets in Earth orbit or re-entering the atmosphere. Debris are generated by events of fragmentation, including collisions, explosive break-ups, wear and tear, which generate entire populations that stagnate around the Earth. Because of their high speed, they pose a threat to functioning satellites in orbit, requiring them to perform dodging maneuvers.

According to ESA's 2023 space environment report [2], Space Surveillance Networks are tracking and maintaining in their catalogue about 34810 debris but the vast majority of objects still remain unidentified. Statistical models by ESA estimate 36500 space debris objects greater than 10 cm, 1 million space debris objects between 1 cm to 10 cm and 130 million space debris objects between 1 mm to 1 cm. In order to avoid collisions with spacecrafts, unidentified debris should be detected, tracked to estimate their trajectory, and possibly removed from their orbit.

In this paper, we propose an new strategy for the detection and tracking of SD around 10 cm-size, named *Refined Stack-CNN* (R-Stack-CNN), that stands for Refined Stacking Method and Convolutional Neural Network. This technique is based on the recent technique Stack-CNN [3], developed to trigger SD on board of space telescopes. Although the original method was effective and higher performing in simulated data, there were some challenges to be addressed. For example the method was not applied to real data, but only to simple simulated data. In this work we apply the Stack-CNN to real data, and we notice that there are many false positives events coming from highly variable background, causing a lack of reliability. To address this issue, we apply the R-Stack-CNN to the offline analysis of simulated and real data, showing its improvements in both cases.

Then we also demonstrate how this method can be adapted for the offline data analysis of meteors as they share similar properties as SD (similar magnitude and speed), leading to the R-Stack-CNN outperforming standard techniques and discovering new meteors and new events never found before.

The data come from the experiment Mini-EUSO, a telescope on board of the International Space Station (ISS) since August 27, 2019. The instrument observes Earth in the UV range (290 - 430 nm) from a UV-transparent window in the Russian Zvezda module, aiming at the same scientific objectives of JEM-EUSO, among which are meteors and space debris. Moreover, given that Mini-EUSO is co-moving with the ISS, the observed background is not static and extremely variable, with light emissions coming from cities, clouds and moon reflections, making the detection of SD and meteors very challenging.

This is the first work that analyzes long sessions of Mini-EUSO data to find SD and meteors with the specific methodology proposed by the R-Stack-CNN.

The problem of detecting, tracking and possibly even removing space debris of size 1 cm to 10 cm has already been studied in the context of JEM-EUSO collaboration [4], a future space-based detector flying attached to the ISS at an altitude of $\sim$ 400 km or on a free-flyer in low orbit ($\sim$ 500 km) looking downwards at Earth with a wide Field of View (FoV, $\pm$ 20-30° in the near-UV spectrum, 300 - 400 nm). The main operational procedure consists of online detection and tracking by the telescope, followed by the debris removal with laser ablation (see [5] for further information).

While the Stack-CNN was proposed to the online detection of SD in a future space detector, as a method that should be fast, accurate and with low memory, the R-Stack-CNN is an offline version of it, aiming to search SD already in Mini-EUSO data, making the method more robust to false positives and false negatives. The main difference of the R-Stack-CNN is the development of a Random Forest (RF) to distinguish the light-curves of the interested objects, e.g. space debris or meteors, from other light sources that could be triggered by the Stack-CNN, such as cities or aircraft. The light-curve of an object refers to the variation in its brightness over time as observed from the detector. Moreover, the shape of a light-curve can provide valuable information about the object's properties, such as its rotation rate, variability, and physical characteristics. Since different objects, like debris and meteors, emit light in different ways, they will have different light-curves. The Stack-CNN method does not consider the development of the light over the time, but only the the object in a single frame (or more frames in a stacked image as described in the following sections), hence it loses an important feature to identify SD. To this aim, we propose a random forest able to distinguish light-curves of SD or meteors from that ones of other events. It turns out that the Stack-CNN assembled with the RF makes the method more robust, excluding many events that are triggered by the Stack-CNN but that are not debris-like events. We present all the details of the random forest, from the training strategy to the evaluations, providing also an ablation study to select the best hyper-parameters of the R-Stack-CNN. Finally we show results in terms of performances and computational time and compare the R-Stack-CNN with the baseline Stack-CNN and a standard threshold-based algorithm.

In summary, here we list our contributions:

- we propose the *Refined Stack-CNN*, an improved version of the Stack-CNN, aimed to work as an offline data analysis to detect moving space objects like space debris and meteor.
- we apply both the *Stack-CNN* and the *R-Stack-CNN* to search new events of space debris and meteors in simulated data and Mini-EUSO data.
- we demonstrate that the *R-Stack-CNN* is more robust against false positives, preserving high performances especially for the detection of faint events and finding new meteors events not found before.

The content of this paper is structured as follows. Chapter II gives details about similar works from which we took inspiration and highlights the advantages of our approach. Chapter III and IV are respectively about the Mini-EUSO detector characteristics and the dataset we used to validate our method. In Chapter V we explain our method and in

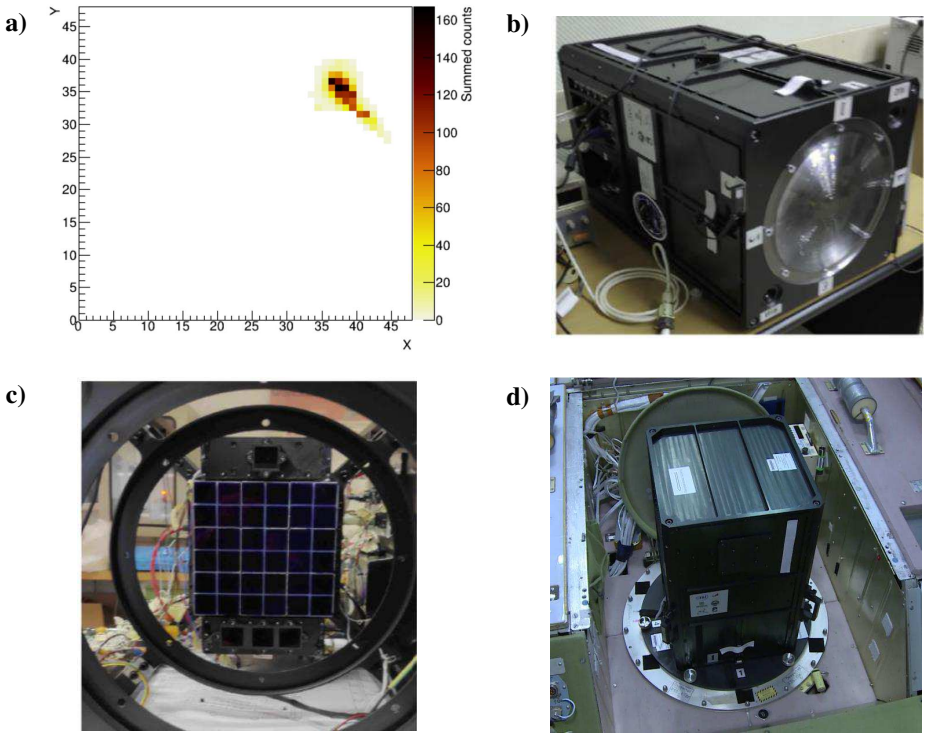

Fig. 1. Panel a) displays an example of meteor track detected on Mini-EUSO focal surface. X and Y indicate the pixel coordinates and the color scale indicates the photon counts detected by each pixel in D3 mode rescaled to D1 mode. Panel b) displays the Mini-EUSO detector facing its front lens during pre-launch tests. Panel c) displays the Focal Surface of Mini-EUSO which is composed of 36 MAPMTS, each of them equipped with 64 photo-detecting pixels. Panel d) displays Mini-EUSO detector mounted on the UV transparent window of the ISS Zvezda module (see text for details, Figure adapted from [6], [7]).

Chapter VI we evaluate the performance on both real data and simulated data.
Finally, we conclude our paper with a discussion section to highlight possible and future improvements of our work and a summary of the results.

II. Related Work

In recent years many space agencies have been addressing the problem of space debris removal by means of new techniques. For instance, in the work of A. Ruggiero et al. [8], a platform using electric propulsion is proposed. Another method for larger space debris involves the use of adhesion properties to capture debris [9]. To the authors' knowledge, there is no official technique for small debris removal. This is due to their small reflective surface and low albedo, reaching SNR of $\sim$ 1, that makes the signal related to these objects very faint and difficult to track.
The JEM-EUSO project [4] aims to detect, track and remove these objects by using an online detection and tracking by the telescope, followed by the debris removal with laser ablation [5]. For online detection the *Stack-CNN* is proposed in [3] as a trigger system to detect faint debris. In this paper we propose the *R-Stack-CNN* as an offline version to analyze data of the Mini-EUSO detector to search for debris and meteors.
The detection of debris and more generally of space objects has been studied for a long time through standard and advanced techniques. For example, in [10], the authors presented an adaptive algorithm based on maximum likelihood ratio to reconstruct paths and positions of space objects. Another solution is to use three-dimensional filter theory [11] to match the possible trajectories of debris with known velocity and direction. A more feasible algorithm was proposed by Barniv and Yair [12] with a dynamic programming approach. Space debris are usually detected using ground-based telescopes pointed at the sky. Depending on exposure mode and times, debris can be seen as streak-like objects superimposed on a static background consinsting of stars, or as point-like objects on a moving background. Given this non-trivial setup, traditional algorithms like the ones presented before might not be complex and powerful enough, although it is worth noting that they have the advantage of not requiring many computational resources.
In order to increase the performance, traditional machine learning techniques and more recent deep learning algorithms have also been investigated. Many recent techniques are based on this new paradigm with many applications on both meteors and space debris. Regarding debris detection, in [13], space debris are detected in a low SNR configuration and with high probability, using feature learning to extract the candidate regions and then classify the space debris. Another work [14] shows how machine learning can also be used to model the orbital prediction errors of space debris, thus correcting orbital prediction results. In Hui Li's paper [15], noisy labels in space

debris detection are mitigated using a new label-noise learning paradigm comprised of the mutual rectification of the two networks. This approach is shown to surpass previous state-of-the-art methods. Considering machine learning applications for meteors, an example is [16], where a feed forward neural network denoising method is applied to near-earth-asteroids data obtained from the Goldstone Solar System Radar. A similar work is [17], where a deep learning method of object detection, YOLOv5, is improved via an attention mechanism able to detect small boulders from planetary images.

These previously cited algorithms have the advantage of being extremely powerful, but this comes also with a steep increase of the computational resources required, both during training and testing. On the contrary, given that our algorithm should be implemented as an online trigger in a Field Programmable Gate Arrays (FPGA), only shallow architectures (low parameters required) can be used. Thus, in [3], a stacking procedure similar to [18] and [19] is enhanced by a shallow Convolutional Neural Network (CNN) classifying right and wrong combinations of speed and direction of the moving object. CNNs are a specific type of neural networks, mostly used in computer vision tasks, such as image classification. The advantage with respect to classical methods is that image features are learnt implicitly during training instead of being hard-engineered by an human, thus increasing the overall performance (more details will be given in the Method section). One of the main challenges that these new methods have to address is the application to real data, since most of the works focus on simulated data, not considering many problems that could arrive from real data, such as pixels with outliers, weird light sources and variable background. In this paper we are the first presenting a technique that has worked with real data for a total of $\sim$ 160 min of acquisition time. The data come from Mini-EUSO experiment on board of the ISS. Other experiments share a similar configuration with extremely variable background. An example is the orbital detector TUS (Tracking Ultraviolet Setup), onboard the Lomonosov Satellite [20], which showed promising results in meteor detection from space images. In parallel to this work, a new approach using a CNN and a fully-connected network [21] is being investigated to find new meteors in some sessions of the Mini-EUSO data. Their approach still implements a CNN to select meteor images, and then a fully connected layer to classify pixels of such image containing meteor events. While they use real data to train the network we base our method only on simulated data and then show the effectiveness on real data. Besides this, another difference is that the R-Stack-CNN finds automatically meteor pixels through the Stack-CNN classification (image classification) and then through a Random Forest (light-curve classification). We show a comparison in terms of new meteors found by both the methods in the appendix.

III. The Mini-EUSO detector and its acquisition modes

The Mini-EUSO focal surface consists of 36 Multi-Anode Photomultiplier Tubes (MAPMTs) where each MAPMT has 8×8 pixels resulting in a total of 2304 channels which can detect individual photon (see panel c) of Fig 1). Given that the optical system is made of two Fresnel lenses of 25 cm each (see panel b) of Fig 1) with a FoV of $44° \times 44°$, each one of these pixels corresponds to a projected spatial resolution on Earth of $\sim$ 6.3 km, and $\sim$ 4.7 km at 100 km height where typically the meteor tracks develop in atmosphere. Mini-EUSO operates on three different data acquisition time scales (D1, D2 and D3), with different exposure times (2.5 μs, 320 μs, 40.96 ms) making it capable of addressing events of varying duration. The D3 time scale is the one sensitive to meteor and space debris events. Along the paper we will call Gate Time Units (GTUs) the acquisition time scales. As Mini-EUSO detects typically $\sim$1 photon count per GTU in D1 mode, thanks to its extremely high photon sensitivity, very often we will renormalize the photon counts detected in D3 mode to the D1 time scale by dividing them by 128×128 time which corresponds to the ratio between the two time frames. If not differently mentioned later on, the D3 GTU will be referred to as the nominal GTU within this paper. Panel a) of Fig. 1 shows an example of a Mini-EUSO meteor observation. Other events and details regarding the instrument can be found in [6]. In this framework the need to have a fast trigger system to find debris and possibly infer its direction and speed is crucial to activate the further operations in order to track and then deorbit the fragment.

IV. The physics of space debris and similar events detectable in Mini-EUSO

Space debris do not emit light by themselves which makes them more challenging to detect. The phenomenon through which a sensor can detect them is known as albedo: the light coming from the Sun (or Moon) hits the debris and is reflected, making the object illuminated. Events that look very similar to SD are the meteors that are visible in the Mini-EUSO data as luminous tracks crossing the field of view. Here we give a brief description of these events.

A. Twilight Configuration

Since the telescope is taking data only during night sessions (period of the ISS orbit spent behind the earth's shadow), the optimal configuration is at twilight, when the space debris could still reflect sunlight and Mini-EUSO is still taking data (before sunrise or after sunset), see Fig. 2(a).

In over 37 Mini-EUSO sessions between October 2019 and August 2021 ($\sim$ 141h 12m 15s) only $\sim$ 1h 6m 22s of data (0.78 %) correspond to this configuration. This is due to the fact that, since the telescope doesn't have a baffle to avoid sunlight and it is pointing nadir, the ISS is directly illuminated during the twilight situations, increasing background and compromising the possibility to test this approach. The observation of space debris would be possible for Mini-EUSO if it would measure in the rare conditions in which the ISS has a roll angle of 90° or 180° opposite to the Sun. In those situations the ISS itself would shield Mini-EUSO from direct light. In addition to the above considerations and sticking to the nominal ISS orbiting condition, Mini-EUSO has a protection mechanism which reduces the gain of MAPMTs or turn them

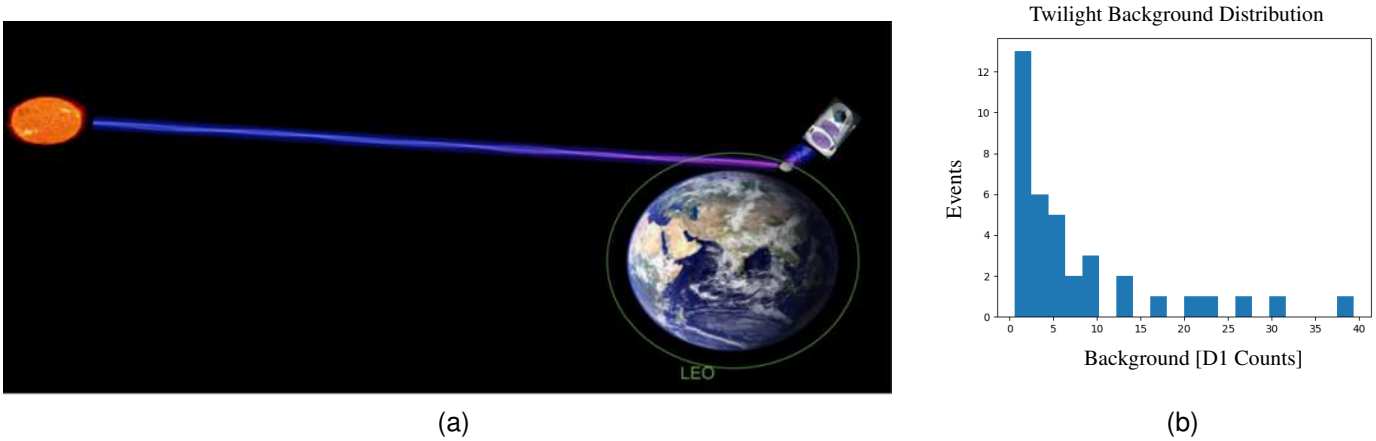

Fig. 2. Figure on the left (a) shows the observation principle of space debris using albedo reflection from the Sun: reflected UV light is shown as blue-violet wave. As can be seen, the detector itself is illuminated, causing an high background, visible on the right distribution (b) where the background is estimated as the median of the data file

off to prevent damage to them from the intense sun light. In this condition the pixels are not sensitive to the standard light levels. This procedure further decreases the real available dataset at twilight, which becomes just the 0.1 % of the 37 Mini-EUSO sessions. In addition, the background distribution (median of data files) as visible in Fig. 2b that roughly only half of data files have background $<$ 5 photon counts / D1 GTU, thus significantly reducing the SNR and consequently the possibility of identifying a debris.

B. Full Moon Albedo

An alternative albedo configuration could be Full Moon reflection. This setup would have the advantage of an increased statistics since it concerns several entire Mini-EUSO sessions. On the other hand, the moon light intensity increases significantly the atmospheric reflection and light diffusion as well as the reflection from objects at ground, resulting in a higher background.

Besides, the apparent magnitude of the Full Moon is larger (fainter) than the Sun, respectively $\mathcal{M}_{app}^{moon}$ = -12.74 and $\mathcal{M}_{app}^{sun}$ = -26.74. Apparent magnitudes (indicated as $\mathcal{M}$) can be used to calculate the ratio of light intensities of Sun and Full Moon using logarithmic properties:

$$\mathcal{M}_{app}^{sun} - \mathcal{M}_{app}^{moon} = -2.5 \cdot log_{10}(\frac{I_{sun}}{I_{moon}}) \quad (1)$$

As a consequence, the sun light intensity is extremely higher than the moon light because of the logarithmic scaling.

$$I_{sun} \sim 4 \cdot 10^5 I_{moon} \quad (2)$$

The reflected light depends on the size (square of the debris radius r^2), the light intensity of the source I_{source} (I_{sun} or I_{moon}) and the distance d from the detector to the debris by the inverse square law ($\frac{1}{d^2}$):

$$I_{albedo} \propto \frac{r^2}{d^2} I_{source} \quad (3)$$

Thus, an algorithm can detect fewer debris as the distance grows, until a certain threshold is crossed and no objects can be detected. In A. Montanaro's paper [3], the performance of the Stack-CNN was tested using several distances and radius, using the Sun as the light source for the albedo in simulated data. It was shown that debris objects of radius $\sim$ 4 cm reflecting sunlight can be detected by the Stack-CNN up to a maximum distance of d $\sim$ 100 km with a 100 % efficiency. A similar threshold can be adapted for moon albedo:

$$(\frac{r^2 I}{d^2})_{moon} = (\frac{r^2 I}{d^2})_{sun} \rightarrow (\frac{r^2}{d^2})_{moon} \sim 6.4 \cdot 10^{-8} \quad (4)$$

In other words this means that a 10 cm-sized space debris would be detectable up to a maximum distance of $\sim$ 200 m. At this altitude, the projected field of view is limited to $\pm$80 m in both x and y directions, which means that the debris trajectory would need to be extremely close to the ISS.

Therefore, the probability of observing a 10 cm-sized debris within this distance is roughly of the same order of magnitude of the probability that the ISS is hit by the debris. ESA's models [2] estimate that the corresponding mean time between impact is $\sim$ 15,000 years, making this approach not suited for Mini-EUSO observations.

Thus, the current Mini-EUSO has not shown any observational usefulness for the detection of space debris because it observes toward the Earth with high background, but if it were to observe darker directions in the sky, it would have a much higher probability of observing smaller and more distant debris.

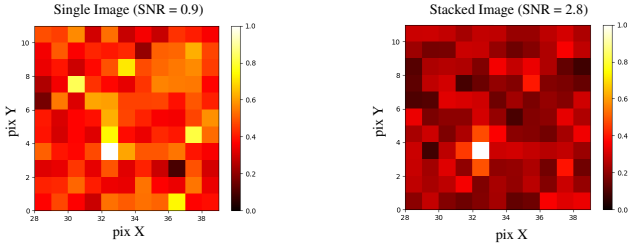

Fig. 3. SNR comparison between Single Image and Stacked Image using a simulated meteor of absolute magnitude $\mathcal{M}_{abs}$ = +6

C. Meteors

Despite the difficulty to test the method to detect space debris, thanks to the flexibility of our approach, the Stack-CNN and the R-Stack-CNN can be applied to any object moving linearly in the field of view of a telescope, such as space debris, meteors and cosmic rays. Therefore, the method was applied for meteors detection, as they share similar properties as space debris (similar magnitude and speed) but they don't suffer from low statistics since they do not require albedo conditions. However, the highly variable background of Mini-EUSO adversely impacts its performance generating a lot of false positives. Since the telescope is pointed downwards the Earth, the observations contain the apparent motion of cities and clouds, hence distinguishing the meteors become challenging. Consequently, the R-Stack-CNN method is introduced to address such difficulties in order to be more efficient and more robust to noise with respect to the standard stack-CNN and a classical thresholding method, using both real data and meteor simulations.

V. The Refined Stack-CNN Method

The proposed method for meteor detection and tracking is an offline version of the Stack-CNN, which is improved by means of a RF. The R-Stack-CNN is based on the Stack-CNN, with the additional implementation of a random forest to make the method more robust, and to be used in offline data analysis of Mini-EUSO experiment. The R-Stack-CNN is comprised of three main techniques, a stacking procedure and a convolutional neural network (already presented in the Stack-CNN) and a final random forest. Here we describe the main components.

A. Stacking Method

The stacking method is applied to objects, e.g. space debris or meteor, moving linearly in the field of view of the telescope with fixed apparent speed $\tilde{v}$ and direction θ. The speed and direction are apparent as the telescope is affected by the speed and azimuth of the platform on which is mounted (in case of Mini-EUSO, the ISS). The method can be described by two main operations, the shifting and the adding procedures. Considering n frames, each of them named I, of raw data depending on pixel position (x, y) and time t, $I(x, y; t)$, t = $\{0, ..., n-1\}$, the shifting is used to shift pixels in the opposite direction of the moving object's trajectory to match the further positions of the object in the initial position of the detector. The movement (dx,dy) depends on the time, speed and direction and it's used to roll the image back in the starting position (x_0, y_0).
In other words, the shifting operation is equivalent to assuming a constant optical flow with fixed speed and direction (opposite to the motion of the signal) over all the pixels of the image, and moving the whole image according to such flow to bring back the signal to its original starting position. This operation is repeated n times, where n is the number of frames. In our case, the n time frames correspond to the D3 GTUs of Mini-EUSO, the nominal GTUs in this paper. The shift along x-axis and y-axis, corresponding to the intensity of the flow applied over each frame, is defined as follow:

$$\begin{cases} dx = \tilde{v} \cdot cos(\theta) \cdot t \\ dy = \tilde{v} \cdot sin(\theta) \cdot t \end{cases} \tag{5}$$

Considering the object starts at the center of the pixel (x_0, y_0) and the xy grid is discrete, dx and dy are transformed into their closest integer value through the *int()* function (e.g. dx = 0.4, dy = 1.7 $\rightarrow$ dx = 0, dy = 2):

$$I^{shift}(x, y; t) = I(int(x - dx), int(y - dy); t) \tag{6}$$

At this point the adding method is used to sum all the shifted images, to recover the moving signal in its starting position:

$$I^{stack}(x, y) = \sum_{t=0}^{n-1} I^{shift}(x, y; t) \tag{7}$$

The main advantage of using $I^{stack}(x, y)$ is that it enhances the signal with respect to a single image. The Signal over Noise Ratio (SNR) is defined as:

$$\text{SNR} = \frac{\text{Signal}}{\sigma_{bkg}} = \frac{\text{Signal}}{\sqrt{\mu_{bkg}}} \tag{8}$$

where the signal is meant to be the difference between the number of counts in a pixel and the average background level μ_{bkg}. If we consider the background to be Poissonian, its fluctuation (in terms of standard deviation) σ_{bkg} is equal to the square root of the background mean $\sqrt{\mu_{bkg}}$. The ideal average background value in the Mini-EUSO dataset is considered to be 1 photon count / D1 GTU, which corresponds to 128x128 counts / D3 GTU. The D3 counts are then rescaled by a factor 128x128, to avoid dealing with large numbers. However, observations also include dynamic background sources such as cities and cloud reflections, which cause an increase in the average background value as well as its complexity, because it no longer can be modelled as Poissonian.
Nevertheless, we chose to consider a Poissonian background for the SNR estimation for the sake of simplicity. The stacked image exhibits an enhanced SNR due to the background that fluctuates between both positive and negative values, while the signal always remains positive which makes the stacked signal $\sqrt{n}$ times brighter than the one in the single image:

$$\text{SNR}^{stack} = \frac{\text{Signal} \cdot n}{\sqrt{\mu_{bkg} \cdot n}} = \sqrt{n} \cdot \text{SNR} \tag{9}$$

where n is the number of frames corresponding to the duration of the event that is assumed to emit constant light. According to the physics of the object this factor could variate. The denominator scales by a factor of $\sqrt{n}$ regardless of the object's presence. Therefore, it's crucial that the number of frames is as close as possible to the track's duration, otherwise the numerator will not scale with a factor of n and there would be only partial improvement on the SNR. Typical numbers for n are 20 (0.8192 s, the average meteor duration) and 40 (1.6384 s) for space debris (a longer track since it is assumed that the debris crosses the entire FoV).

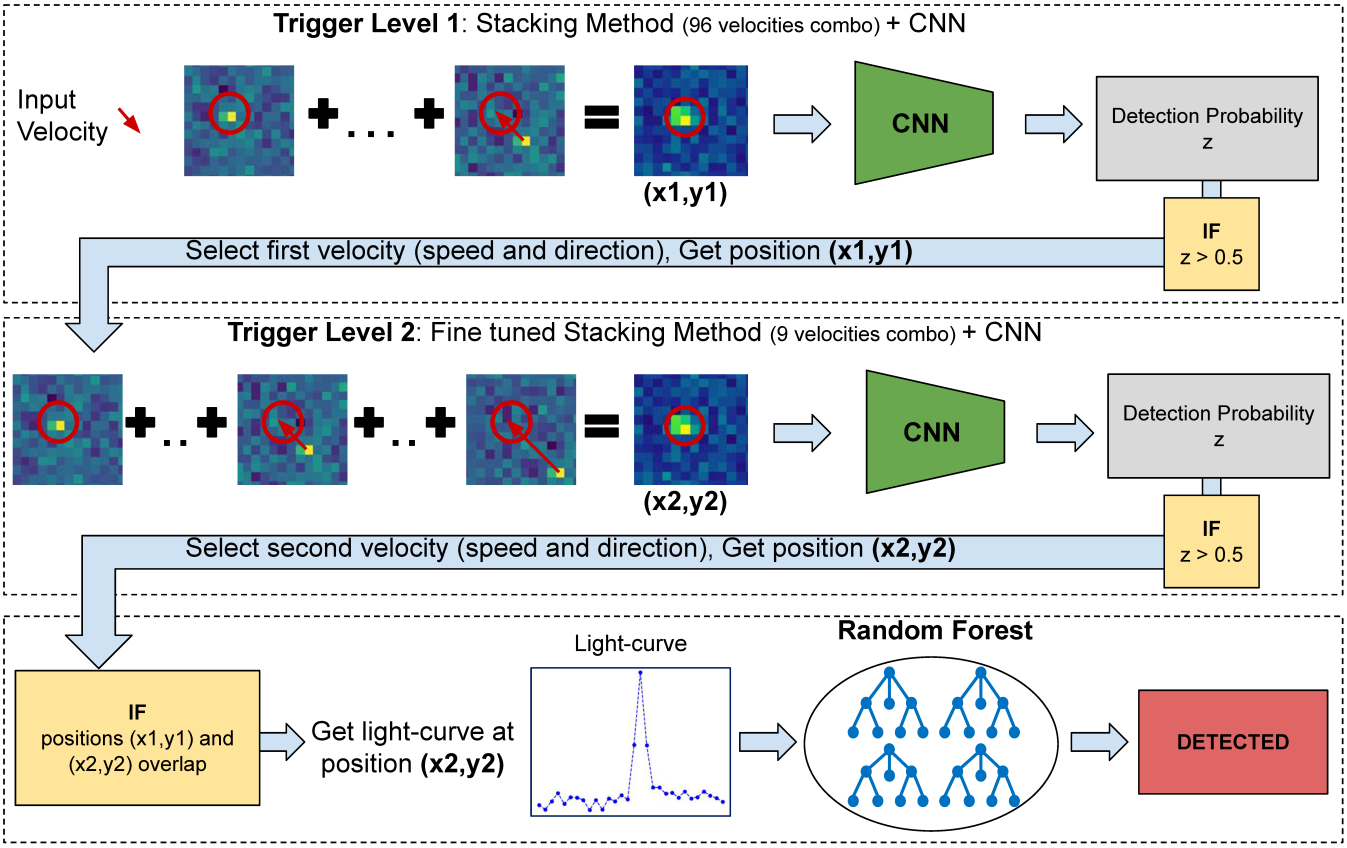

Fig. 4. R-Stack-CNN algorithm for space debris and meteor detection (Source: [3])

Fig. 3 shows the difference between the stacked image and the single image for a simulated meteor of $\mathcal{M}_{abs}$ = +6, for which the intensity of each pixel has been normalized from 0 to 1 for demonstration purpose. The single image is the one with the maximum signal in the meteor track, whereas the stacked one has been stacked with the true simulated speed, direction and duration of N_{GTU} = 13 GTUs. In the single image the meteor is barely visible and the SNR = 0.9, while in the stacked image the SNR is increased by a factor of 3.1 (SNR = 2.8) which is close to $\sqrt{13\ GTUs} \sim 3.6$. Since the ($\tilde{v}$,θ) combination for a triggered object is not known, a significant sample of possible combinations are computed and a classifier is needed in order to distinguish the right compbinations from the wrong ones. Here is where the CNN comes into the game. The network is trained as a binary classifier for Right (1) and Wrong Combination (0).

B. Convolutional Neural Network

CNNs are a class of artificial neural networks most commonly used in Computer Vision (image classification, video analysis, ...). Their advantage with respect to other algorithms is that the network is computationally efficient (due to the convolutional operations involving shared weighted sum with small kernel size of the filter) and they extract image features most relevant to the relative objective, in our case the classification. The CNN implements filters (or kernels) that are optimized through automated learning and captures the spatial dependencies in an image. In the pre-deep learning era these filters were hard-engineered with human intervention and difficult to build. The name convolutional neural network is originated with the design of LeNet-5 [22] by Yann LeCun in 1998, built for handwritten digit classification and it is the first architecture to implement backpropagation for automated learning in a CNN.

Newer architectures developed in the 2000s thanks to the *ImageNet large scale visual recognition challenge (ILSVRC)* and to the usage of GPU during training, which strongly decreased computing time. ImageNet is a common dataset on which researchers tested new algorithms and the first GPU-based CNN to win the competition is AlexNet (2012) [23], which introduced ReLU activation functions and dropout layers for regularization. Similar but deeper architectures are called VGGNets [24], which prove that increasing the number of layers and parameters can yield an higher performance.

In 2015, Google presented GoogLeNet [25], winning ILSVRC by introducing the inception module, whose key idea is to parallelize pooling and convolutional layers.

Then, skip connections between layers were introduced by the ResNet architecture [26], addressing the gradient vanishing problem and achieving even higher performance. The concept was extended by DenseNets [27] with skip connections also between non-consecutive layers.

In our work, the architecture also needs to be suitable for an on-board implementation in a space debris remediation

system. Hence, the CNN must be shallow because an higher number of parameters would require higher computational time and expensive resources. The number of total parameters is 16825, divided across convolutional and fully connected layers, with ReLU activations in the hidden layers and a sigmoid function in the output layer. Regarding the training dataset, a total of 80 space debris were simulated with ESAF [28], a software that generates point-like moving sources in the Mini-EUSO framework. The simulated background is Poissonian with mean value equal to 1 count / D1 GTU. See [3] for details about the event simulation, training, validation and the architecture.

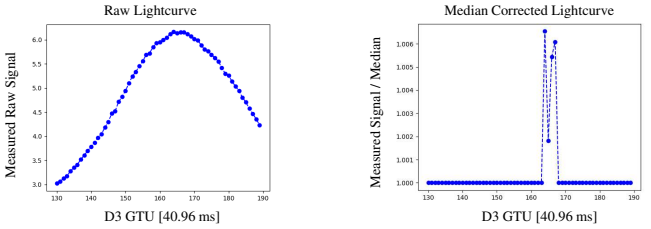

Fig. 5. Example of pre-processed lightcurve on cities observed by Mini-EUSO: raw lightcurve (on the left) is flattened to values close to 1 by a mobile median correction (on the right)

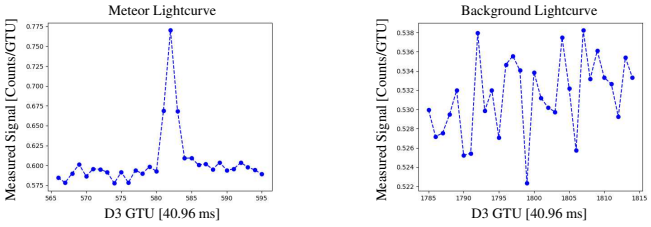

Fig. 6. RF Training Dataset: binary classification of meteor lightcurves with output 1 (left) and background lightcurves with output 0 (right). The time range of each lightcurve is a portion of a complete data acquisition file, which lasts 3200 GTUs.

C. Stack-CNN

The Stack-CNN (see Fig. 4) combines the stacking procedure and the CNN in a detailed framework. The stacked combinations and the number of frames are chosen depending on the physics of the object (space debris and meteor in this work), making this approach extremely versatile as it allows to use the neural network trained on space debris even for meteors, and in principle to anypoint-like object moving linearly in the detector, e.g. cosmic rays. In this last example the method could be applied only offline due to the light-speed of such particles and the requirement to use D1 data. The Stack-CNN is divided in two processing levels. The former performs a rough track reconstruction, by generating 96 combinations of speed and direction. If the object is detected, the second level is implemented to fine-tune the reconstruction and suppresses false positives. For space debris, the first level stacks 12 GTUs, where the considered directions go from 0° to 345° with steps of 15° while speed ranges from 5 to 11 km/s with steps of 2 km/s. It is assumed that the reference height is 370 km, which is below the International Space Station ($\sim$ 420 km), but still in Low Earth Orbit. The range of the speed has been chosen depending on the typical order of the SD orbital speed in Low Earth Orbit, which is about $\sim$ 7-9 km/s, and to account for the relative motion of the ISS, which travels at $\sim$ 7.66 km/s. Then, the trained CNN is applied to each combination, which is positively classified if the output, indicated by y, is greater than 50 %. The second level is used to fine-tune the triggered combination with $\pm$ 0.5 km/s and $\pm$ 5° steps. The number of stacked GTUs is increased to 40 to exploit the longer movement of a space debris with respect to a background event. Finally, if the event is still positively classified by the neural network and the two triggered pixels overlap in a neighborhood of 2 pixels the space debris event is triggered.

In our work, the Stack-CNN was adapted to meteors by modifying stacked frames and speed combinations, according to the meteor physics. The number of stacked GTUs in the first and second levels was changed to 8 GTUs ($\sim$ 0.33s) and 20 GTUs ($\sim$ 0.82s), respectively, because the duration of the event is shorter compared to space debris. Meteor entry speed in the atmosphere is usually estimated using a reference height of 100 km, which is where the light emission starts [29]. Besides, the speed range is bounded by 11 km/s and 72 km/s [30]. Thus, speed combinations have been chosen within a range from 10 km/s to 70 km/s with step of 20 km/s. The fine-tuning steps have also been changed to 5 km/s. The advantage of using meteor events to test the Stack-CNN is that, unlike space debris, the light emission phenomenon does not require any reflection of other light sources, making their observation more frequent in Mini-EUSO. Besides, the study of meteors in Mini-EUSO [31] could be useful because meteor observations are usually performed at visible wavelengths, while Mini-EUSO, operating in the UV range, could detect meteors in UV up to $\mathcal{M}_{abs} \geq +5$. Another advantage is that space observations are not affected by weather conditions like the ground observations and high statistics can be collected in a short time.

However, the CNN was trained using a simulated Poissonian background, which is a strong simplification with respect to real background from Mini-EUSO data. Hence, preprocessing is needed in order to recreate a configuration as similar as possible to the one used during the training procedure.

The background map is affected by the passage of the ISS over cities and clouds. The algorithm suppresses these contributions thanks to the normalization of each pixel by means of a moving median. The median is computed for each pixel and GTU, starting from GTU-4 to GTU+4. Then, the count of each pixel is divided by this value and the resulting background is normalized between 0 and 1. This is useful for discarding slow events (i.e. events that move at the apparent speed of the detector, which is the same of the ISS, i.e. $\sim$ 7.7 km/s), but it doesn't affect fast objects like meteors or space debris. An example of a pre-processed background sample is given in Fig. 5. The threshold of the neural network was also increased to 90 %, instead of the default value of 50 %, in order to suppress as many false positives as possible. To improve the performance and lower the false positive rate coming from

challenging background conditions and noisy pixels, the stack-CNN is improved by adding a RF classifier, analyzing meteor light-curves. We prove in the next chapter how the proposed method reaches better performance than the baseline Stack-CNN.

D. Random Forest

The Random Forest is essential for the offline analysis of Mini-EUSO data to avoid many false positives coming from moving light sources (e.g. cities, ships, lightning), while keeping a true positive rate as good as the original Stack-CNN. An illustration of the random forest used in this paper is in Fig. 7.

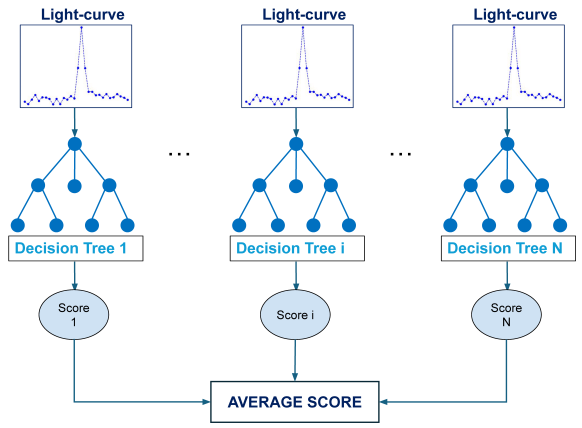

Fig. 7. The random forest, used in the R-Stack-CNN, processing meteor light-curves. Each decision tree outputs a probability score for the recognition or not of a meteor, then the scores are randomly averaged across trees.

The classification is binary, i.e. an output of 0 for background lightcurves and 1 for meteor lightcurves. In astronomy, a light-curve is a curve describing light intensity of celestial objects, in a particular frequency band, over a period of time. In the case of Mini-EUSO, the frequency band is UV, and light intensity is expressed as photon counts and time as GTUs. In this case the light-curve can be considered as a time-series over the pixels illuminated by the signal. The fast movement of meteors (or space debris) in the field of view generates an excess of signal counts in pixels hit by the track (see Fig. 6). In our framework, lightcurves are represented as univariate time series, i.e. series of time-ordered data of length T. With time series, features correspond to data values at each time frame, which means photon counts at every GTU for lightcurves.
We chose the RF for its robustness to limited train data and for its fast convergence during the train. RFs are an extension of decision trees, i.e. tree-like structures, where each internal node represents a decision on a feature, based on which the tree splits in branches.

RFs [32] minimize the tendency of decision trees to overfit, by averaging the output on a forest of bagged decision trees, with low correlation between each other, as they are trained on randomly extracted subsamples of the original dataset. The training and validation procedures were done using real data from an analysis performed on session 11 of Mini-EUSO.

TABLE I
RANDOM FOREST PERFORMANCE ON TEST SET

TP	TN	FP	FN	Accuracy	F1
140	129	3	5	97.1 %	97.2 %

This session was chosen for the large statistics of detected meteors. First, a list of 553 events was obtained through the application of the standard trigger, which will be used as a baseline for all the results in this paper. The standard trigger [33] does not implement machine learning techniques, it scans 25 virtual elementary cells, defined as 16 x 16 pixels, searching for an excess in neighboring pixels lasting 5 consecutive GTUs. The threshold for each pixel is 3σ above the mean background μ_{bkg} computed at every *GTU* ($\mu_{bkg} + 3\sigma$). Then, each event was visually inspected by an expert who verified if meteors were indeed moving objects hitting many pixels. Finally, the results consisted of 416 meteors, divided in two categories, 309 M and 107 $M^?$. The former class is used to classify objects with bright and long-duration movements. On the contrary, short-duration and fainter tracks are usually grouped in the latter class, labeled as $M^?$ with '?' indicating uncertain meteors. The reason for this distinction is that even if shorter tracks could come from real meteors, the physics reconstruction of speed and azimuth would be challenging and affected by high uncertainty. As for the positive events of the training dataset (output = 1), only certain meteors M were chosen, along with the two of the closest and most significant pixels in the meteor track. The meteor track is affected by blurring in the neighbouring pixels, as described by the Point Spread Function (PSF). Hence, the inclusion of lightcurves from closest pixels allows the correct classification of fainter events affected by the PSF, while also having the positive effect of increasing the training set size. Background events (see Fig. 6) were randomly chosen in the S11 dataset. In this way, events include both Poissonian fluctuations and cities, which would have been difficult to simulate. Finally, after visually inspecting the complete set of light-curves, the size of the training dataset became 1384, equally distributed between positives and negatives (692 each). The pre-processing was limited to a normalization between 0 to 1 of the time series. Then, the dataset was split between training set S_{train} (60 %), validation set S_{dev} for best model evaluation (20 %) and test set S_{test} to quantify the performance in unseen data (20 %). Firstly, a validation set was used to tune the RF main hyperparameters, including the length of the time series, the number of decision trees and the maximum depth. Then, the model was trained with S_{train} + S_{dev} and tested in S_{test} using the F1 metric, defined as the harmonic mean of precision and recall. The advantage compared to a symmetric metric like Accuracy is that F1 is used when True Positives (TPs) are more important than True Negatives (TNs), which means that an higher F1 tends to minimize False Negatives (FNs). That makes it more suitable for this application, as it's crucial that the meteors found by the Stack-CNN must not be lost by the RF. The performance on the test set is summarized in Table I (FPs indicate False Positives). The results are extremely

TABLE II
R-STACK-CNN ABLATION PERFORMANCE WITH MINI-EUSO DATASET FROM SESSION 6.

Model	TP	TN	FP	FN	Precision	CPU Time
Standard Algorithm	101 (69 M + 32 $M^?$)	-	17	-	85.6 %	39 min
Stack-CNN	177 (99 M + 78 $M^?$)	-	878	-	16.8 %	65 min
R-Stack-CNN	165 (96 M + 69 $M^?$)	856	22	12 (3 M + 9 $M^?$)	88.2 %	247 min

promising as only an extremely low percentage of meteors are lost (3.4 %) generating an high F1 score. Besides, most of the background events are correctly classified (97.7 %) making the method robust to noise and background fluctuations.
More details about the training and ablation of the Random Forest along with definitions of used metrics are given in Appendix A.

TABLE III
R-STACK-CNN PERFORMANCE WITH MINI-EUSO DATASET FROM SESSION 14

Model	TP	FP	Precision	CPU Time
Standard Algorithm	196	100	66.2 %	180 min
R-Stack-CNN	**276**	**119**	**69.9 %**	1138 min

VI. APPLICATIONS OF STACK-CNN AND R-STACK-CNN TO REAL AND SIMULATED DATA

The R-Stack-CNN method was tested on both real data acquired by Mini-EUSO and simulated ones to study its performance in comparison with the original Stack-CNN and the standard trigger algorithm.

A. Real data: search for space debris and meteors

First, a dataset of 13 files from the Mini-EUSO session 6, which corresponds to roughly $\sim$ 28 min, has been used to quantify the improvement of the R-Stack-CNN with respect to the Stack-CNN.
Then, the entirety of Mini-EUSO session 14 has been used to compare the R-Stack-CNN results to the standard trigger. It is worth noting that because of the protection mechanism discussed in Chapter IV-A, the real available dataset is roughly $\sim$ 129 min, corresponding to $\sim 2\times10^5$ frames.
The Stack-CNN (see Table II) was able to find 89 new meteor candidates than the standard algorithm (32 M + 57 $M^?$), while losing only 13 meteors (2 M + 11 $M^?$) detected by the standard approach. However, the main problem is that 878 False Positives (FPs) were also triggered, making the final Precision (eq: 13) very low, 16.8 %. The standard algorithm had an higher precision (85.6 %) than the Stack-CNN (16.8 %), making it more reliable even if fewer meteors were triggered. On the other hand, the R-Stack-CNN was able to find a total of 79 additional meteor candidates than the standard trigger (30 M + 49 $M^?$), while losing only 15 events (3 M + 12 $M^?$). The final precision was 88.2 %, which is much better than the model without the RF (16.8 %) and also an improvement on the standard algorithm (85.6 %).
We evaluated also the computing time required to process these data files by each algorithm. As Table II shows, introducing the Random Forest in the Stack-CNN increases the CPU time with a factor of 3.8. However, the increase in the precision is much higher, i.e. a factor of 5.25, meaning that using the Random Forest is the optimal solution for trade-off between time and performance.
Then, the R-Stack-CNN was tested using the complete Mini-EUSO session 14 (see Table III): the model R-Stack-CNN found 136 new meteor candidates (Table III) than the standard algorithm (85 M + 51 $M^?$) while losing 56 meteors (26 M + 30 $M^?$). The model also detected 119 false positives, with an overall precision of 69.9 %, which is slightly better than the corresponding 66.2 % of the standard trigger. These results showed that, even for an extended set of data, our method outperformed standard techniques finding a larger number of meteors. It is also worth noting that, although our method proved to be the most powerful and accurate one, the standard algorithm remains a valid faster solution. In addition, we provide a comparison with another neural network based method, and we show that the R-Stack-CNN finds more meteors than this. Further details are provided in the appendix.

B. Simulated data: detection and tracking of meteors

In order to investigate the detection limit and true efficiency of the model, meteor events were also simulated. Considering that the Stack-CNN was originally planned for the online detection and tracking of space debris, it's crucial that the speed and direction combination is as precise as possible. Hence, meteor simulations are also used to quantify the goodness of the speed and azimuth reconstruction.
An important feature of the simulations is that they involve a dynamical model, implementing analytical solutions [34] to the differential equations describing the physical problem of the meteor body deceleration in the atmosphere [35]. The simulated parameters were sampled from their known distributions [36] and the background maps were generated using a Poissonian distribution with an average of μ_{bkg} photon counts per GTU in D1 mode. An example of a simulated event of $\mu_{bkg} = 0.572$ and absolute magnitude $\mathcal{M}_{abs} = +4$ is given in the Appendix C, Fig. 12. From now on the indicated magnitudes are meant to be positive even though the '+' sign is not explicitly indicated. A total of 300 events have been simulated for meteors of $\mathcal{M}_{abs} = \{4, 5, 6\}$ (100 each). Each event has been simulated with a random sampling of μ_{bkg} ranging from 0.5 to 1 photon counts per GTU in D1 mode.
The results using meteor simulations have been summarized in Table V and compared to the standard trigger results. The R-Stack-CNN found 32 additional meteors in 300 simulated

TABLE IV
SUMMARY OF SIMULATED ABSOLUTE MAGNITUDE $\mathcal{M}_{abs}$ AND MEAN BACKGROUND CONFIGURATIONS

Events	$\mathcal{M}_{abs}$	μ_{bkg}
100	4	{0.5:1}
100	5	{0.5:1}
100	6	{0.5:1}

events with respect to the standard trigger. However, only 6 meteors of $\mathcal{M}_{abs}$ = +6 were triggered by the algorithm, defining the detection limit. These events are indeed very faint and their tracks are often difficult to observe because of the background fluctuations. Fig. 9 shows, as an example, a lost event of $\mathcal{M}_{abs}$ = +6, with a horizontal purple line defining the $3\sigma_{bkg}$ range of background fluctuations, where $\sigma_{bkg} = \sqrt{\mu_{bkg}}$.

TABLE V
SUMMARY OF THE R-STACK-CNN RESULTS WITH METEOR SIMULATED EVENTS

Events	$\mathcal{M}_{abs}$	R-Stack-CNN	Standard Trigger
100	4	88	77
100	5	70	50
100	6	6	5
300	{4,5,6}	164	132

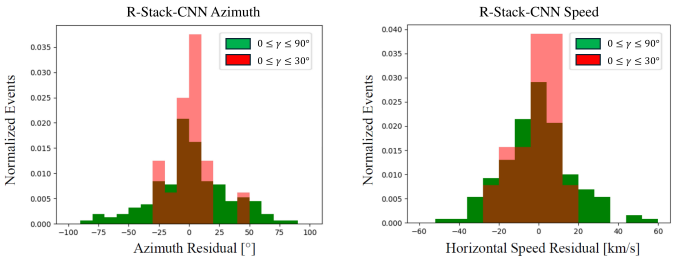

Fig. 8. Residual distribution of Stack-CNN reconstructed variables: meteor azimuth is shown on the left and horizontal speed on the right. The green distribution refers to meteors with inclination $0° \leq \gamma \leq 90°$, while the red distribution refers to events with $0° \leq \gamma \leq 30°$.

Then, the performance of the speed v and azimuth ϕ reconstruction has been estimated by using the standard deviations of the residual distributions of the triggered events.
The azimuth is defined as the direction from the True North, whereas the speed refers only to the horizontal component of the meteor true speed. Since Mini-EUSO doesn't have a stereoscopic view, it would be impossible to estimate the transversal direction of speed.
Besides, it's important to note that since Mini-EUSO is mounted on the ISS, its variables (speed and direction) are affected by the position and speed of the ISS, that travels at $\sim$ 7.66 km/s with an azimuth of $\sim 51.6°$. Therefore, both meteor horizontal speed and azimuth, i.e. the clockwise direction from the true North, were corrected. Moreover, the R-Stack-CNN often triggers the same meteor event more than once with different speed, direction and even starting GTU. Considering that images are processed sequentially, sometimes long tracks can surpass the 20 frames used in the stacking method, causing the event to be triggered more than once. Therefore, the best combination was chosen as the one with the highest number of counts in the maximum pixel of the stacked image. The residual distribution of the azimuth showed $\mu_\phi = (-3 \pm 4)°$ and $\sigma_\phi = (46 \pm 3)°$ (Fig. 8), whereas the longitudinal speed residual distribution had $\mu_v = (0 \pm 1)$ km/s and $\sigma_v = (17 \pm 1)$ km/s (Fig. 8). However, these results refer to meteors with any value of inclination γ, which also means having vertical trajectories with few hit pixels. Hence, given that the goal is to investigate the precision of the Stack-CNN reconstruction of space debris through meteors, the residual distributions were evaluated using mostly horizontal tracks which resemble space debris more accurately. Each meteor event has been simulated with a different inclination γ, with γ = 0° indicating an horizontal track and γ = 90° a completely vertical trajectory with respect to the Mini-EUSO focal surface. Thus, $0° \leq \gamma \leq 30°$ has been set as the range used to define mostly horizontal tracks.
The results showed a great improvement, with $\mu_\phi = (1 \pm 4)°$, $\sigma_\phi = (15 \pm 3)°$ for the azimuth and $\mu_v = (0 \pm 3)$ km/s, $\sigma_v = (10 \pm 2)$ km/s for the horizontal speed. These results show that the Stack-CNN could indeed be implemented in a space debris remediation system with a reliable estimation of speed and direction. Besides, there is no bias in the reconstruction since both average values are compatible with 0.

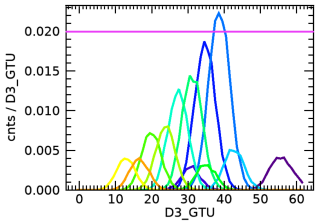

Fig. 9. Example of a meteor not found by R-Stack-CNN due to the low magnitude of $\mathcal{M}_{abs}$ = +6, $\mu_{bkg} = 0.716$. The plot shows a collection of light-curves associated to the same meteor event, with signals hitting different pixels. The horizontal purple line shows the 3σ value of background fluctuations.

VII. DISCUSSION

Although the R-Stack-CNN has shown improvements with respect to the Stack-CNN, a standard thresholding method and another machine learning based technique, it has some limitations. For example, our method, even if it suppressed the extreme number of false positives coming from the Stack-CNN, it lost some meteors that were detected by the latter. In future we want to study new strategies to avoid this loss, such as developing a recurrent neural network for the recognition of light-curves. Another way to improve the whole framework could be training also the CNN with real data, but this would require a large amount of balanced and pre-processed data. Finally, even if we presented the R-Stack-CNN as an offline

trigger we do not exclude the possibility to test it online with an FPGA and compare with the Stack-CNN.

VIII. Conclusions

In this paper we presented the R-Stack-CNN, a refined version of the Stack-CNN that serves as an offline data analysis to detect and track space objects that move linearly in the field of view of a telescope. In particular we applied the method to data of the experiment of Mini-EUSO, a UV telescope on board the ISS pointing on the Earth. With this configuration, the space objects that can be detected are meteors or space debris. Unfortunately, the space debris generally does not emit light themselves, therefore finding such events with a small aperture telescope at satellite orbit, such as Mini-EUSO described in this paper, has turned out to be very difficult at the moment. However, similar phenomena to space debris, but more luminous and more frequent, are meteors. We have shown that the R-Stack-CNN is an effective data analysis method for finding these events with higher precision than other methods. Specifically, the R-Stack-CNN found almost as many meteors as the Stack-CNN (which is the method that finds more meteors than the other methods), but with a false positive rate much lower than that, avoiding to manually look at these events and discard them. With the development of a lightweight recurrent architecture, it is expected that the R-Stack-CNN can be improved to have even higher capabilities. The ability of the R-Stack-CNN to find events in real data, even though most of them are trained on simulated data, offers interesting prospects for applying this technique to other data, such as ground-based telescopes that point on the sky and can detect different space objects, e.g. space debris, asteroids and meteors. Such an improved R-Stack-CNN could also be useful for the space debris observations from the satellite orbit in future.

IX. Acknowledgements

This work was supported by the Italian Space Agency through the agreement n. 2020-26-Hh.0, by the French space agency CNES, and by the National Science Centre in Poland grants 2017/27/B/ST9/02162 and 2020/37/B/ST9/01821. This research has been supported by the Interdisciplinary Scientific and Educational School of Moscow University "Fundamental and Applied Space Research" and by Russian State Space Corporation Roscosmos. The article has been prepared based on research materials collected in the space experiment "UV atmosphere". We thank the Altea-Lidal collaboration for providing the orbital data of the ISS. We would like to express our gratitude to all members of the JEM-EUSO programme for making the Mini-EUSO mission possible.

References

[1] M. Garcia. Space debris and human spacecraft. [Online]. Available: https://www.nasa.gov/mission_pages/station/news/orbital_debris.html

[2] Esa's space environment report 2023. [Online]. Available: https://www.sdo.esoc.esa.int/environment_report/Space_Environment_Report_latest.pdf

[3] A. Montanaro and et al., "Stack-cnn algorithm: A new approach for the detection of space objects," *Journal of Space Safety Engineering*, vol. 9, pp. 72–82, 2022.

[4] J. Adams and et al., "The jem-euso instrument," *Experimental Astronomy*, vol. 40, no. 1, pp. 19–44, 2015.

[5] T. Ebisuzaki and et al., "Demonstration designs for the remediation of space debris from the international space station," *Acta Astronautica*, vol. 112, pp. 102–113, 2015.

[6] S. Bacholle and et al, "Mini-euso mission to study earth uv emissions on board the iss," *The Astrophysical Journal Supplement Series*, vol. 253, no. 36, 2021.

[7] F. Bisconti *et al.*, "Pre-flight qualification tests of the mini-euso telescope engineering model," *Experimental Astronomy*, vol. 53, no. 1, pp. 133–158, 2022.

[8] A. Ruggiero, P. Pergola, and M. Andrenucci, "Small electric propulsion platform for active space debris removal," *IEEE Transactions on Plasma Science*, vol. 43, no. 12, pp. 4200–4209, 2015.

[9] Z. Xie, X. Chen, Y. Ren, and Y. Zhao, "Design and analysis of preload control for space debris impact adhesion capture method," *IEEE Access*, vol. 8, pp. 203 845–203 853, 2020.

[10] N. C. Mohanty, "Computer tracking of moving point targets in space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-3, no. 5, pp. 606–611, 1981.

[11] I. Reed, R. Gagliardi, and H. Shao, "Application of three-dimensional filtering to moving target detection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 6, pp. 898–905, 1983.

[12] Y. Barniv, "Dynamic programming solution for detecting dim moving targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-21, no. 1, pp. 144–156, 1985.

[13] J. Xi, Y. Xiang, O. K. Ersoy, M. Cong, X. Wei, and J. Gu, "Space debris detection using feature learning of candidate regions in optical image sequences," *IEEE Access*, vol. 8, pp. 150 864–150 877, 2020.

[14] B. Li, J. Huang, Y. Feng, F. Wang, and J. Sang, "A machine learning-based approach for improved orbit predictions of leo space debris with sparse tracking data from a single station," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 6, pp. 4253–4268, 2020.

[15] H. Li, Z. Niu, Q. Sun, and Y. Li, "Co-correcting: Combat noisy labels in space debris detection," *Remote Sensing*, vol. 14, no. 20, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/20/5261

[16] N. Rodriguez-Alvarez, J. S. Jao, J. F. Munoz-Martin, C. G. Lee, and K. Oudrhiri, "Feed-forward neural network denoising applied to goldstone solar system radar images," *Remote Sensing*, vol. 14, no. 7, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/7/1643

[17] L. Zhu, X. Geng, Z. Li, and C. Liu, "Improving yolov5 with attention mechanism for detecting boulders from planetary images," *Remote Sensing*, vol. 13, no. 18, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/18/3776

[18] T. Yanagisawa and et al., "Detection of small geo debris by use of the stacking method," *Transactions of the Japan Society for Aeronautical and Space Sciences*, vol. 51, no. 589, pp. 61–70, 2003.

[19] M. Bertaina and et al., "The trigger system of the jem-euso project," *Proceedings of the 30th International Cosmic Ray Conference*, vol. 5, pp. 1049–1052, 2008. [Online]. Available: http://refhub.elsevier.com/S2468-8967(22)00001-5/sbref0002

[20] P. Klimov, B. Khrenov, M. Kaznacheeva, G. Garipov, M. Panasyuk, V. Petrov, S. Sharakin, A. Shirokov, I. Yashin, M. Zotov, V. Grebenyuk, A. Grinyuk, M. Lavrova, A. Tkachenko, L. Tkachev, A. Botvinko, O. Saprykin, A. Puchkov, and A. Senkovsky, "Remote sensing of the atmosphere by the ultraviolet detector tus onboard the lomonosov satellite," *Remote Sensing*, vol. 11, no. 20, 2019. [Online]. Available: https://www.mdpi.com/2072-4292/11/20/2449

[21] M. Zotov and et al., "Neural network based approach to recognition of meteor tracks in the mini-euso telescope data," *accepted for publication in Algorithms*, 2023.

[22] Y. LeCun and et al., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou,

and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1409.1556

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: http://arxiv.org/abs/1409.4842

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[27] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269.

[28] F. Fenu, K. Shinozaki, H. Miyamoto, A. Liberatore, N. Sakaki, S. Sharakin, M. Zotov, and G. Chiritoi, "Simulations for the jem-euso program with esaf," *arXiv preprint arXiv:1909.12012*, 2019.

[29] Z. Ceplecha, J. Borovička, W. G. Elford, D. O. Revelle, R. L. Hawkes, V. Porubčan, and M. Šimek, "Meteor Phenomena and Bodies," *Space Science Reviews*, vol. 84, pp. 327–471, 1998.

[30] O. Popova, J. Borovička, and M. D. Campbell-Brown, *Modelling the Entry of Meteoroids*. Cambridge University Press, 2019.

[31] G. Abdellaoui and et al., "Meteor studies in the framework of the jem-euso program," *Planetary and Space Science*, vol. 143, pp. 245–255, 2017.

[32] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[33] H. Miyamoto and et al., "Space debris detection and tracking with the techniques of cosmic ray physics," *Proceedings of Science*, vol. 253, 2019.

[34] M. Gritsevich, "Determination of parameters of meteor bodies based on flight observational data," *Advances in Space Research*, vol. 44, pp. 323–334, 2009.

[35] V. Bronshten, *Physics of Meteoric Phenomena*, ser. Geophysics and Astrophysics Monographs. Springer Dordrecht, 1983, vol. 22.

[36] A. Bouquet and et al., "Simulation of the capabilities of an orbiter for monitoring the entry of interplanetary matter into the terrestrial atmosphere," *Planetary and Space Science*, vol. 103, pp. 238—-249, 2014.

APPENDIX

In this section we provide supplementary materials regarding our implementation of Random Forest, the specific metrics we used, ablation and sensitivity studies.

A. Random Forest Definitions

The Random Forest is a traditional machine learning algorithm consisting of an ensemble of M decision trees, with M defining the forest size. The final output, indicated as $\bar{y}$, is defined as the average of each output y_i of each single decision tree. A sketch is shown in Fig. 7. As a consequence, the variance associated to the output depends on the correlation, indicated as ρ, between decision trees, as described by the following formula:

$$\text{Variance } [\bar{y}] = \rho \cdot \text{Variance } [y] + (1-\rho) \cdot \frac{\text{Variance } [y]}{M} \quad (10)$$

Therefore, the main objective of this algorithm is to decrease the correlation ρ, while also increasing the forest size M, so that the averaged variance is better than the single one. The technique is called bagging, it consists on training each decision tree with bootstrap samples, randomly chosen from the training dataset and replaced so that data can be used more than once. Besides, random forests also decrease correlation by considering only a fraction of randomly selected features in each split node. The bias-variance tradeoff in random forests is reflected by the fact that the chance of underfitting slightly increases because subsamples are smaller than the full dataset. Therefore, it's crucial to train random forests with big enough datasets. Another disadvantage is that forests are less easy to visualize and interprete than single decision trees, but they are much more powerful.

We can also define the probability associated to its output as binomial (p_1 is the probability of having prediction equal to 1), with M being the forest size and N_1 the number of decision trees associated with output equal to 1:

$$p_1 = \frac{N_1}{M} \quad (11)$$

In this supplementary material we will also provide sensitivity analysis on this parameter, showing how it can deeply affect the performance of the R-Stack-CNN.

Given that our task was a classification task, i.e. time series binary classification, we used a set of metrics suitable for our goal. In particular, accuracy is the baseline metric used to measure performance for finding positive and negative labels, precision is used to quantify the tradeoff between finding true positives and false positives, recall measures how many positive instances are detected from all the actual positive samples and finally F1 is defined as harmonic mean of precision and recall, giving a more complete and exhaustive view on the model performance. We provide the explicit definitions (TP are True Positives, FP are False Positives, TN are True Negatives and FN are False Negatives):

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (12)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (13)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (14)$$

$$\text{F1} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{\text{TP}}{\text{TP} + \frac{\text{FP}+\text{FN}}{2}} \quad (15)$$

B. Ablation Study

In addition, an ablation study was performed to determine whether removing decision trees during training could drastically reduce the performance. The number of decision trees in the Random Forest has been reduced iteratively from a maximum of 2000 to a minimum of 1, corresponding to a traditional decision tree. For each iteration, the performance has been estimated both in training dataset and validation dataset using the cross-validated F1 score on 10 folds.

As can be seen from Fig. 11, the performance reaches a plateau with 1000 decision trees. Training with more iterations would be pointless as the model would preserve the same accuracy at the cost of an higher computational time.

Another study has been performed regarding the probability of the Random Forest algorithm. The default value is set to 50 %, meaning that if more than half of the decision trees have outputs 1, the overall output is also 1 (and vice

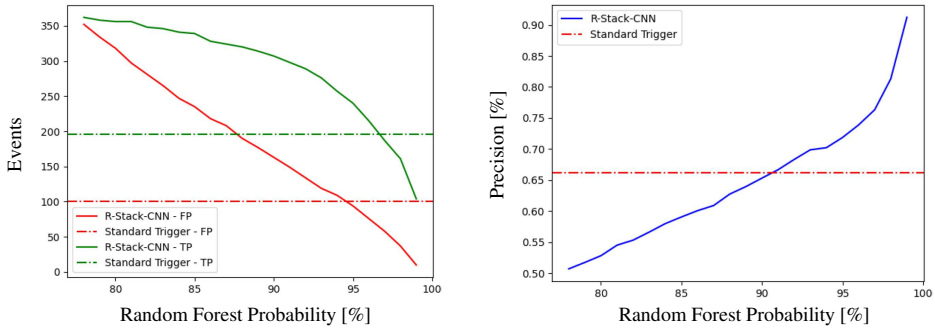

Fig. 10. Number of true positives (TP) and false positives (FP) on the left figure and precision on the right as a function of increasing RF threshold. Horizontal lines indicate the performance of the standard trigger.

versa). We investigated other possible values, by increasing and decreasing its value and estimating the R-Stack-CNN performance on real data.

The optimal performance of the R-Stack-CNN in session 6 (Table II) has been achieved by increasing the RF threshold to 78%, meaning that at least 78% of decision trees have outputs 1. The threshold has been set by maximizing the F1 metric, which is equal at its maximum to 90.7 %. The study was completed using session 14 of Mini-EUSO data: the R-Stack-CNN precision was estimated using different values of the Random Forest probability threshold. Table VI and Fig. 10 show that by increasing the threshold more meteors are found with respect to the standard trigger. Using 78 % threshold, the model found 193 new meteor candidates than the standard algorithm (115 M + 78 $M^?$) and lost only 27 meteors (10 M + 17 $M^?$) detected by the standard approach. Unfortunately, because of higher background configurations, 352 false positives were also triggered. Thus, an higher RF threshold has been set to eliminate as many false positives as possible and a study has been performed to estimate the optimal threshold: the precision, the number of true positives and the number of false positives were evaluated using increasing RF thresholds. The results in the left panel of (Fig. 10) show that there is a strong decrease in the number of the R-Stack-CNN false positives with an increasing RF probability of 95 %. On the other hand, the number of the R-Stack-CNN true positives decreases with a weaker slope, causing an increment in the precision as shown in the right panel. The optimal threshold has been set to 93 % because with higher thresholds too many meteors would be lost. This setup has been used also to compare our method to another one, developed in parallel in the Mini-EUSO collaboration. This method implements a Convolutional Neural Network trained to detect chunks of meteors in the field of view of the detector. Then, the algorithm searches for meteor lightcurves in meteor chunks and implements a multi-layer perceptron (MLP) to classify them. We will refer to it by the acronym CNN + MLP. See [21] for more details. The overall structure is extremely similar to our method, as both algorithms implement CNNs to classify images, and then a light-curve classifier is used to suppress false positives. In the alternative method the algorithm is a MLP while in our case it's a Random Forest. A substantial difference, however, is that our CNN was trained on simulated data while in their case on the real data.

The results (Table VI) show that the R-Stack-CNN was still the most performing method to find new meteor events (95 more meteors M were found). However, the CNN + MLP method was more precise (80 % vs 69.9 % of R-Stack-CNN). This behaviour was probably caused by the different training dataset, which generates fewer false positives as data from Mini-EUSO sessions is more noisy and complex. An improvement of our method would probably fine-tune our CNN using real data, making it more robust to noise.

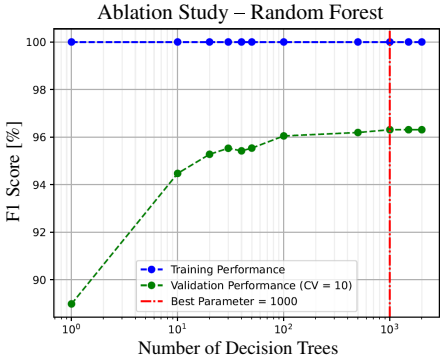

Fig. 11. Visualization of F1 metric computed on training (blue) and validation (green) datasets by iteratively removing decision trees from the Random Forest. The best parameter is shown in red.

TABLE VI
R-STACK-CNN PERFORMANCE WITH MINI-EUSO DATASET FROM SESSION 14

Model	Total meteors	New meteors
Standard Algorithm	196 (133 M + 63 $M^?$)	0
CNN + MLP	264 (97 M + 167 M?)	68
R-Stack-CNN	**276** (192 M + 84 $M^?$)	80

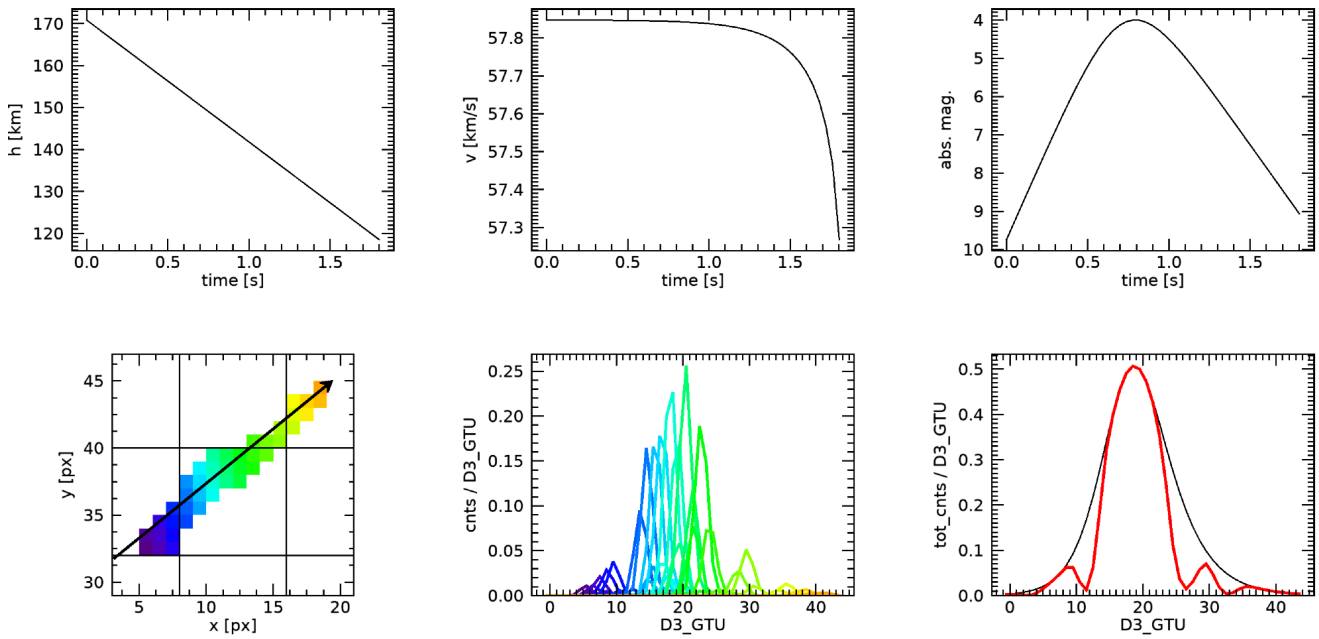

Fig. 12. Example of a simulated meteor event of $\mathcal{M}_{abs}$ = +4. The three plots on the top represent height, meteor speed and absolute magnitude as a function of time. The three plots on the bottom show the pixels hit by the meteor event (left), the respective photon counts per D3 GTUs (center) and the sum of the photon counts per D3 GTU (left), which have been indicated by the red curve. The black curve represents the expected meteor counts on the focal surface in the absence of dead spaces among MAPMTs.

C. Illustration of a simulated meteor

An illustration of a simulated event with a background rate of $\mu_{bkg} = 0.572$ and an absolute magnitude of $\mathcal{M}abs = +4$ can be found in Figure 12. Henceforth, the magnitudes mentioned are assumed to be positive, although the '+' sign is not explicitly stated. A total of 300 events have been simulated for meteors with absolute magnitudes of $\mathcal{M}abs = 4, 5, 6$ (100 events for each magnitude). Each event has been simulated with a random sampling of background rates, ranging from 0.5 to 1 photon counts per GTU in D1 mode.