IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 December 2022, accepted 12 January 2023, date of publication 18 January 2023, date of current version 25 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3237800

RESEARCH ARTICLE

Fast Training Data Generation for Machine Learning Analysis of Cosmic Ray Showers

TOMASZ HACHAJ, ŁUKASZ BIBRZYCKI, AND MARCIN PIEKARCZYK, (Member, IEEE)
Institute of Computer Science, Pedagogical University of Krakow, 30-084 Krakow, Poland

Corresponding author: Tomasz Hachaj (tomekhachaj@o2.pl)

This work was supported by the Pedagogical University of Krakow Statutory Research Grant, through the subsidies for science granted by the Polish Ministry of Science and Higher Education.

ABSTRACT Applying Machine Learning (ML) methods for the analysis of muon lateral distributions in Extensive Air Showers detected by citizen science projects, while taking into account the spatial distribution of detectors requires enormous training data sets. Therefore, generating these data sets with typical Monte Carlo (MC) generators like CORSIKA is computationally prohibitive. Here we present a method which by the application of special augmentation procedures produces the training dataset that is compatible in all essential aspects to the data produced with regular MC computations while avoiding their time overhead. We utilize the Nakamura-Kamata-Greisen (NKG) distribution which was proven to be an attractive alternative to full-fledged simulations. The simulation of 10^4 muons at the ground level takes just a few seconds using our implementation of the NKG approach. For 10^6 muons this figure is still around 1 minute. For comparison, CORSIKA based simulation performed on Prometheus supercomputer at CYFRONET computing center an ensemble of $\sim$ 100 showers initiated by a particle of $10^{16} eV$ resulted in $\sim 10^4$ muons and $\sim 10^5$ electrons required computation time of the order of a few days.

INDEX TERMS Cosmic ray shower, simulation, data generation, detectors, machine learning.

I. INTRODUCTION

The application of smartphone cameras as cosmic ray detectors is a new research opportunity actively pursued for a few years now. Citizen science initiatives like CREDO [1], DECO [2] and CRAYFIS [3] have attracted tens of thousands of users who provided cosmic ray hits registered with their devices for the analysis. Classifying individual hits observed in smartphones turned into detectors is already a relatively well-studied [4], [5], [6], [7] topic. Another extremely important issue is the identification and study of the entire cosmic ray showers observed as correlated hits in many devices in short time interval.

Extensive Air Showers (EAS) are produced in the Earth's atmosphere from the primaries coming from the outer space. The energies of the primaries range from a few GeV up to 10^{11} GeV with the steeply decreasing energy distribution [8]. The primaries are mostly electrons, protons and nuclei but at the sea level the charged component of hadron initiated EASs consists mainly from secondary muons produced in pion and kaon decays. Due to the flux of primaries rapidly decreasing with energy, for energies above 10^5 GeV the only practical method to observe cosmic rays is indirect, through EASs. Thus, for the ultra high energy cosmic rays the detailed information on the distribution and development of EAS is necessary to reconstruct the properties of primary cosmic rays. These properties have been extensively studied in experiments like Kaskade [9], KASKADE-Grande [10], Telescope Array [11] or Pierre Auger [12], to name just a few. These facilities employ measurements based on fluorescence, Cherenkov radiation, radio emission or the combination of them.

Deep Neural Networks (DNN) are nowadays a standard tool in the EAS studies to analyse eg. regularities in arrival directions of primaries [13], determination of the shower maximum [14] or noise rejection [15]. Application of ML methods requires creating large training and evaluation datasets obtained either from simulations or from direct

The associate editor coordinating the review of this manuscript and approving it for publication was Hong-Mei Zhang.

measurments like in [14]. To this end the EAS development is simulated with CORSIKA [16] or AIRES [17] while detectors' response with Geant4 [18], respectively. Even for detectors with characteristic sizes of $\sim$1 m^2 like in the Pierre Auger or Telescope Array experiments, the simulations require simplifying assumptions. Here we are concerned with the detectors typical for dual scientific-educational projects whose characteristic sizes range from $\sim$1 cm^2 like for Cosmic Watches [19] down to $\sim$1 mm^2 like in the case of smartphone based experiments [20]. Given enormous number of detecting smartphones, their small effective surfaces and little control over their spatial distribution, conventional simulations seem prohibitively computationally expensive. Therefore an efficient method to generate training data sets is a must. When creating ML models, it is critical to have access to a sufficiently large amount of data that is representative to the considered problem. In our work, we do not directly use ML methods to generate particle distributions. The software we have prepared is an efficient tool for generating large amounts of data for building, analyzing and verifying Machine Learning methods in projects that gather cosmic rays data by distributed sensor networks. These methods will be used to analyze experimental data and evaluate the acquired signals from the detector networks. The citizen science cosmic ray experiments lift typical constraints of *professional science* facilities, like space constrain (in principle they can be arbitrarily large in terms of the surface covered) and the number of detectors constrain (there are billions of smartphones in the World, with other cheap amateur detectors like Cosmic Watch being able to multiply in large quantities). However, realistically we can expect that one of the two operation modes is realized, either the experiment is performed on the large scale (like continent-wide or even inter-continent-wide) but with rather small detector density or experiment is performed on small area with large detector density, eg. during educational events or detection campaigns with many participants. The first mode may be useful for studying distant but coincident showers like in the Gerasimova-Zatsepin effect [21]. The second mode is relevant for studying small showers and further derivation of the information on the primary particles and possibly their energies and arrival directions [22]. In this study, we focus on the second experiment mode.

In the CORSIKA manual [23] one can find the information that the simulation of the electromagnetic component of the shower is 40 times slower than employing the Nakamura-Kamata-Greisen (NKG) distribution [24], [25]. The application of NKG distribution is thus an attractive alternative to full-fledged dynamics-informed simulations. It was used eg. as a tool to evaluate EAS parameters (shower size or Moliere radius) [26] or global properties of the shower [27]. It is then natural to expect that for the simulation of the complete shower, including the hadronic component, this difference is even larger. Since we are interested in the lateral particle distribution at the ground level, a detailed simulation of the vertical and temporal shower development using CORSIKA or AIRES is not required. We need, however, the (training) data sets abundant enough to make the deep learning approaches feasible.

The rest of the article is organized as follows. We first present the theoretical basis of the simulation (Section II). Afterwards in Section III, it is discussed the physical and technical assumptions and the proposed simulation algorithm. Then, in Section IV, we discuss the technologies used, the implementation details, and the results obtained during the simulation itself under different initial conditions.

Against this background, Section V discusses the simulation results in terms of accuracy, reliability, and suitability for potential applications, particularly in the CREDO project. The work concludes with Section VI, which contains a summary, conclusions and suggestions for future research work.

II. STATISTICAL DESCRIPTION OF THE MUON LATERAL DISTRIBUTION

Generating the training data sets for ML analyses of data obtained in the citizen science experiments as defined in Section I entails several concerns. Here we discuss them, as they define the parameters for simulations described below. Footprints of air showers on the ground can reach sizes of several hundreds of square kilometers. Here we are interested in much smaller showers whose linear dimensions do not exceed several hundred meters but typically the linear dimension of observed region is below 100 meters. Still, observing such a relatively small region on the cm or mm scale results in a problem of very large granularity with 10^8 or 10^{10} simulation cells, respectively. On the other hand the detector distribution is still pretty sparse with the typical number $<$100 in the analyzed area. Moreover, the only observable accessible to smartphone detectors is the number of hits registered by each device in fixed time interval which translates to lateral cosmic ray distribution at ground level. This time interval is constrained by obtainable device synchronization times and the time needed to process individual CMOS camera frames in smartphones. It can be estimated at the level of $\sim$1 s down to at best several tens of ms. Here, quite conservatively, we assume the snapshot duration is 1 s. On the other hand, the typical time span between the fastest and slowest particles of the EAS is 400 ns [9]. Given the snapshot time the EAS arrival can be treated as an instantaneous phenomenon simulated in 1 s lasting intervals. Since we are concerned only with the lateral cosmic ray density distribution sampled by either randomly distributed detectors or detectors arranged in some predefined configuration, we can disregard the full spatiotemporal evolution of the EAS and content ourselves with its lateral component as observed at the ground level. Such effective lateral distribution indeed exists and can be parametrized in terms of the Nishimura-Kamata-Greisen distribution [24], [25]. For small showers but large enough for the statistical description to be justified, this distribution was shown to be compatible with CORSIKA predictions [22]. Since we are considering the EASs whose muon size N_μ is no smaller than 10^4, the statistical description is fully legitimate.

The lateral distribution of particles in the EAS is well described by a formula proposed by Greisen [24] and confirmed by numerical calculations on electro-magnetic showers by Kamata and Nishimura [25] thus known as NKG formula. As shown in [22], by proper adjustment of parameters, this distribution can be used also to describe muon lateral density.

We use a slightly generalized form of this formula, Eq. 1, as shown at the bottom of the next page, which accounts for the fact that the distribution is singular at $r = 0$ for typical values of the age parameter s, thus needs to be truncated for distances smaller than r_{min} [28], [29].

The age parameter s describes the relation between the lateral shape of the distribution and the height of the shower maximum and is typically assumed in the interval $0 < s < 2$ [22]. For our exploratory simulation we put the representative value $s = 1.3$ but generally it should be deemed as a parameter floating in the aforementioned interval. Fig.1 shows that due to small variability of the distribution for distances below 200 m our choice is representative. As already mentioned for the statistical description to be justified, the shower size parameter N_μ has to be sufficiently large. Therefore in our exploratory simulations we consider the shower sizes in the $10^4 - 10^6$ interval which corresponds to a primary particle energy of more than $10^{16} eV$ [30]. Finally the r_0 parameter describes the characteristic size of the shower and in our simulations it is put equal to 100 m. This value is compatible with the Molière radius in the Earth's atmosphere at the ground level [31].

Eq. (1) describes the vertical shower ie. the shower that hits the ground at the angle of 90^o. The general case of inclined shower was considered in [28] and we adopt an approach presented there. We obtain the general shower orientation in two steps. First inclining the shower axis by a θ angle in the xz plane, where $\theta \in \langle 0, \pi/2)$. In the lateral coordinates the first transformation reads

$$\begin{cases} x_a = x/\cos\theta \\ y_a = y \end{cases} \tag{2}$$

where x and y are coordinates in plane perpendicular to shower axis. Then the general orientation is obtained by performing a rotation in lateral plane by a ϕ angle, where $\phi \in \langle 0, 2\pi)$. Note that in the transformation given by Eq. (2) we neglect the vertical dimension thus ignoring the angular dependence of the shower maximum and retaining only geometrical effects.

This way we have obtained a highly parametrizable formulation of the muon lateral distribution that will be a starting point for further simulations.

In Figure 1 we have shown the muon's lateral distribution $\rho_\mu(r)$ as a function of the radial distance from the shower axis r for several values of the age parameter s. As can be seen, the value of s has in practice little effect on the lateral distribution of cosmic ray showers. Distribution decreases rapidly as the distance from the cosmic ray impact center increases.

III. IMPLEMENTATION OF PARTICLE SHOWER DETECTION ON THE GROUND SURFACE

The assumptions we made in preparing our simulation algorithm can be divided into physical assumptions and the technical assumptions. Technical assumptions are determined mostly by implementation method. The physical assumptions are:

1) The cosmic ray shower phenomenon is practically detectable within a radius of a few hundred meters from the center of the burst. In such a relatively small area it is not necessary to take into account the curvature of the Earth.
2) Let us assume that the cosmic ray shower can strike at different angles thus affecting the shape of the lateral distribution on the earth's surface. We will model the angle of impact of the particles on the Earth's surface through a pair of polar coordinates (ϕ, θ).
3) In practice, the frequency of recording and transmission of cosmic radiation by the devices we use in the CREDO project is 1 Hz (data transmission takes place after 1 second of recording).
4) We assume that the phenomenon of impact of the whole shower takes place in a time quantum equal to 1 second. In fact, the particles of the shower move at the speed of light and the phenomenon of shower formation takes much less than a second. Therefore, our assumption should not significantly affect the quality of the simulation.
5) According to [8], the muon flux density at the earth's surface is 1 $\frac{muon}{cm^2 \cdot min}$. Thus, we can assume that over a period of 1 second, the background radiation density at 1 cm^2 of the earth's surface averages $\rho = \frac{1}{60} \frac{muons}{s \cdot cm^2} = 0.01(6) \frac{muons}{s \cdot cm^2}$.
6) We assume that the number of particles in the shower will be at most 10^6.
7) We assume that each detector has 100% efficiency, that is, it detects every particle that passes through it.

Further assumptions result from the way we have chosen to implement the above physical assumptions. The technical assumptions are:

1) The phenomenon will be modeled in a square area of at most 0.5 $km \times 0.5$ km (0.25 km^2). Considering a larger area has no practical application.
2) The area of a cosmic ray particle detector is on the order of a few to tens of square centimeters. Let us assume that the minimum detector dimension that we will consider in the simulation will be 1 $cm \times 1cm$. With this assumption, a square sit of 0.25 km^2 will be represented by $2.5 \cdot 10^9$ measurements.
3) It is improbable that more than $3.2 \cdot 10^4$ particles fall in the center of the shower in one second. For this reason, it is completely sufficient for the number of particles that fall on 1 cm^2 to be stored using the short type. In most programming languages, the short type takes 2 bytes.

4) A grid that meets the assumptions of a maximum size of 0.25 km^2 and the number of particles per 1 cm^2 will occupy about $2 \cdot 2.5^{10} bytes \approx 4.66\ GB$ in computer memory. This is a value that comfortably fits into the RAM of a modern computer. In fact, the simulations we will perform are more likely to cover tens or at most a few hundred meters, so the memory requirements will be much smaller.
5) From the detector's point of view, it is indistinguishable whether it registers background radiation or shower radiation. From the simulation point of view, it is convenient if we can distinguish which radiation particle the detector has registered. This is some additional information that can be used in subsequent studies. In this case, you have to store the shower and the background radiation separately. If we also store it using the short data type, this will double the required memory. However, this amount of memory can still be freely reserved on a modern PC.

We will now present an algorithm (Alg. 1) for generating impact simulations and registering the shower through detectors on the ground. In order to generate a lateral distribution consistent with Eq. 1, we used the rejection method for generating random variables [32].

The computational complexity is affected by two processes. The first is the generation of a random stream of particles, which consists of N elements. The computational complexity of drawing each sample (1) (whether accepted or rejected) is the computational complexity of drawing a sample (1), plus the computational complexity of calculating the value of a uniformly distributed PDF, plus comparing a uniform random variable and a threshold. The second process is to generate the background radiation, the computational time of which is a function of $gridX$, $gridY$, the density of background radiation per unit area equal to the size of the grid sample in 1 second (ρ), plus the computational complexity of calculating the value of the uniformly distributed PDF.

IV. RESULTS

The summation algorithm presented in Section III was implemented in Java 1.8. The interface to the application was developed in the Swing library. Since we used only standard JAVA libraries our program can be run on any operating system that has a JAVA virtual machine. The application data exchange file is a text file. We have also prepared a script in Python 3 language, which allows to generate a random arrangement of non-overlapping particle detectors in an area of a given size. We use it as input. We have also prepared a second script in Python demonstrating how to read the simulation results. The source codes we prepared can be downloaded from https://github.com/browarsoftware/credoshowersimulator.

We counted all performance tests of our method on a PC computer with Intel Core i7 3.00 Ghz; 64 GB RAM, Windows 10 OS. The purpose of the tests was to evaluate the ability of the model described in Section II to generate data that could then be used by machine learning algorithms.

Figure 2 shows the 2D histogram of the number of particles during a simulated shower impact at the center of a 64 $m \times 64\ m$ square area vertically from above ($\phi = 0, \theta = 0$). Each pixel corresponds to a 10 $cm \times 10\ cm$ area. In the first column, the number of bundle particles is $\#p = 10^4$, in the second column $\#p = 10^5$ and in the third column $\#p = 10^6$. The images have been colored using a look-up table. Because particle multiplicities span across several orders of magnitude o make the images in the second row more readable we used the logarithmic scale.

Figure 3 shows a visualization of the histogram of the number of particles during a simulated impact of a shower of 10^6 muons into the center of a 64 $m \times 64\ m$ square area vertically from above ($\phi = 0, \theta = 0$) for different values of the parameter r from Eq. 1. Each pixel corresponds to a 10 $cm \times 10\ cm$ area. For the following columns, the values of r_0 are 100, 200, 300 and 400, respectively. The images were colored using the look-up table. The images in the second row are on a logarithmic scale.

Table 1 shows the execution speed of the algorithm implementation in seconds for a given number of particles in the shower and the simulation grid size (in meters). In order to get samples count the grid size should be multiplied additionally by 10^4, since the single sample size was 1 cm^2.

Figure 4 shows a visualization of the histogram of the number of particles during the simulated impact of a 10^6 particles at the center of a 64 $m \times 64\ m$ square area at different angles (ϕ, θ). Each pixel corresponds to a 10 $cm \times 10\ cm$ area. Since Figure 4 has to be slightly smaller to fit the page we used a different look-up table, that nicely represents the shapes of the areas with various number of particles.

Figure 5 shows the 2D histogram of the number of particles during the simulated impact of a 10^6 muons at the center of a 16 $m \times 16\ m$ square surface at different angles (ϕ, θ). Detection is performed with a different number of randomly distributed detectors, $\#d = 100$, $\#d = 50$ and $\#d = 25$, respectively. Each detector is assumed to have an area of 10 $cm \times 10\ cm$. In our implementation, the smallest simulated detector can be 1 $cm \times 1\ cm$. The first column shows the simulated particle distribution in this space. The next three are histograms of the simulated impact recorded by the detectors. The random distribution of detectors in each column is

$$\rho_\mu(r) = \begin{cases} \rho_\mu(r_{min}), & r \le r_{min} \\ \dfrac{N_\mu}{2\pi r_0^2} \dfrac{\Gamma(4.5-s)}{\Gamma(s)\Gamma(4.5-2s)} \left(\dfrac{r}{r_0}\right)^{s-2} \left(1+\dfrac{r}{r_0}\right)^{s-4.5}, & r > r_{min} \end{cases} \quad (1)$$

Algorithm 1 Algorithm of Cosmic Ray Shower Generation and Its Detection on the Ground

Data: Input: gridX, gridY - simulation array size;
ρ - background radiation density per unit area equal to the size of the grid sample in 1 second;
(ϕ, θ) - polar coordinates of shower direction;
N - particle number in shower;
offsetX, offsetY - The offset of the center of the jet relative to the center point of the simulation grid. (0,0) means to hit the point (gridX/2, gridY/2) of the grid;
D - list of detectors, each detector is an rectangular object that has left-bottom corner (startX, startY), right-top corner (stopX, stopY) and count of background and shower particle count (appropriately background and shower). Those last two values are initialized to 0.
Result: D - list of detectors with updated background and hit values.

```
backgroundN ← floor(gridX * gridY * ρ);
// Arrays initialization with zeros
background ← zeros(gridX, gridY);
shower ← zeros(gridX, gridY);
// Generate background
for a in range(backgroundN) do
    x ← randomInt(0,gridX);
    y ← randomInt(0,gridY);
    background[x,y] ← background[x,y] + 1;
end
// Generate shower
for a in range(N) do
    // Sample radial distance value from distribution given by Eq. 1 using
       rejection method [32]
    rndr ← randomVariate(s, r0);
    // Sample azimutal angle from the uniform distribution and rotate particle
       location
    φ ← randomFloat(0, 2π);
    x_h ← rndr * cos(φ);
    y_h ← rndr * sin(φ);
    x_h ← x_h / cos(θ);
    x ← x_h;
    y ← y_h;
    x_h ← x * cos(φ) − y * sin(φ);
    y_h ← x * sin(φ) + y * cos(φ);
    // Offset particle
    x ← floor((gridX / 2) + x_h + offsetX);
    y ← floor((gridY / 2) + y_h + offsetY);
    if x > 0 and x < gridX and y > 0 and y < gridY then
        shower[x,y] ← shower[x,y] + 1;
    end
end
// Generate detections
for d in D do
    for x in range(d.starX, d.stopX) do
        for y in range(d.starY, d.stopY) do
            d.shower ← d.shower + shower[x,y];
            d.background ← d.background + background[x,y];
        end
    end
end
return D
```

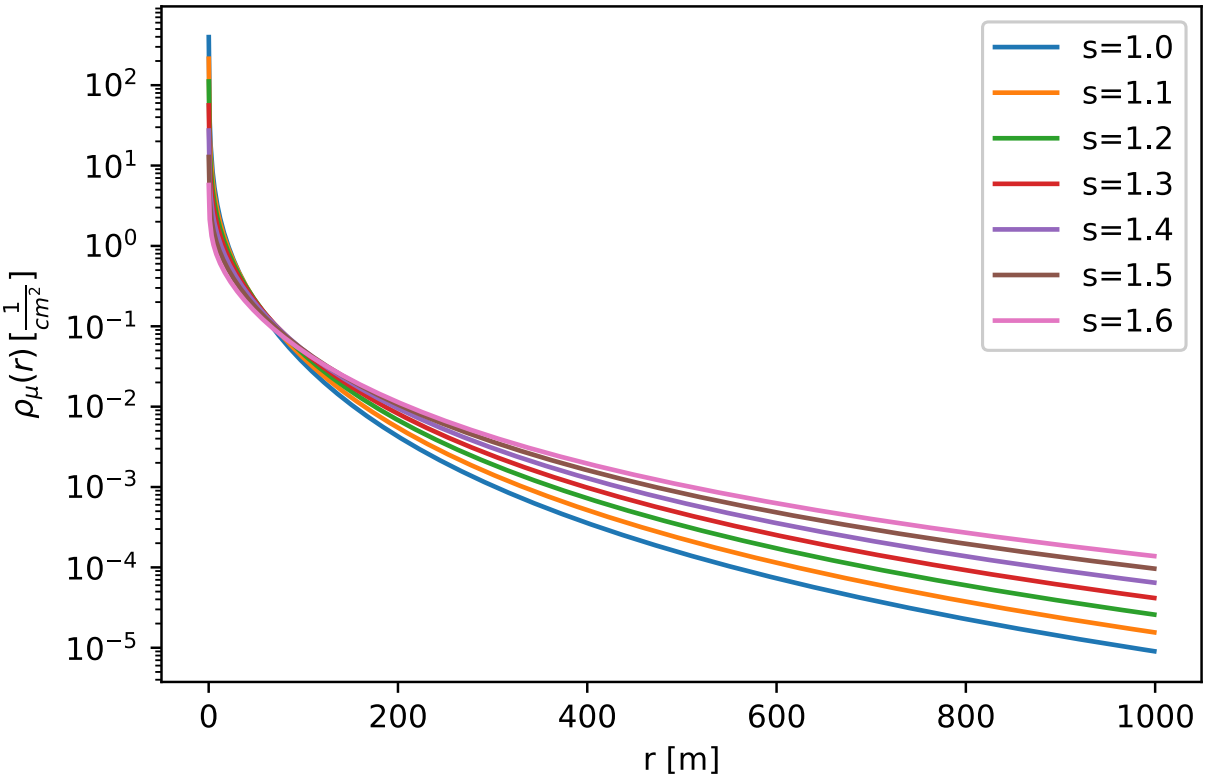


FIGURE 1. Muon lateral distribution ρ_μ as a function of radial distance r and the value of the age parameter *s*.

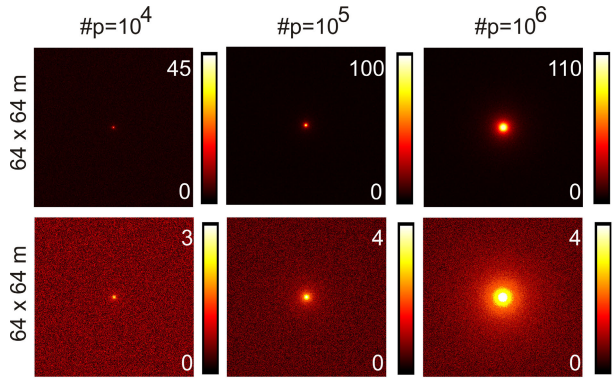


FIGURE 2. The 2D histograms of the number of particles during a simulated burst impact at the center of a 64 *m* × 64 *m* square surface vertically from above ($\phi = 0$, $\theta = 0$) for different numbers of #*p* particles in the burst. The heat density maps are shown on a linear (top row) and logarithmic (bottom row) scale, respectively.

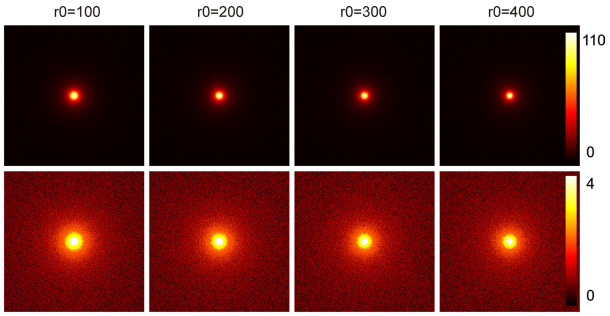


FIGURE 3. The 2D histograms of the number of particles during a simulated impact of a bunch of 10^6 particles into the center of a 64 *m* × 64 *m* square surface vertically from above ($\phi = 0$, $\theta = 0$) for different values of the parameter r_0 from the equation (1).

identical. We have intentionally reduced the simulation area considerably so that the reader can more easily interpret the results in the picture. To emphasize the importance of the number of detectors on the interpretation of the PDF shape and to compensate for the background effect we have applied a convolution Gaussian filter [33] with kernel sized 251×251.

V. DISCUSSION

As can be demonstrated in Section IV proposed algorithm and its implementation is capable to simulate impact and detection of cosmic ray shower on a given earth surface. The obtained results are in good agreement with theoretical models and simulation results from independent molecular-scale simulation software packages such as CORSIKA [22]. Decisive for the execution speed of the algorithm are the grid size and the number of particles N.The larger the grid size the longer it takes to generate the background distribution. The more particles in the shower, the longer it takes to count [32]. As can be seen in Table 1 those two values are calculated independently. The number of particles in the burst affects the area of the region where the number of particles is greater than the background radiation. The more particles in the burst, the larger the diameter of this area (see Figure 2). The

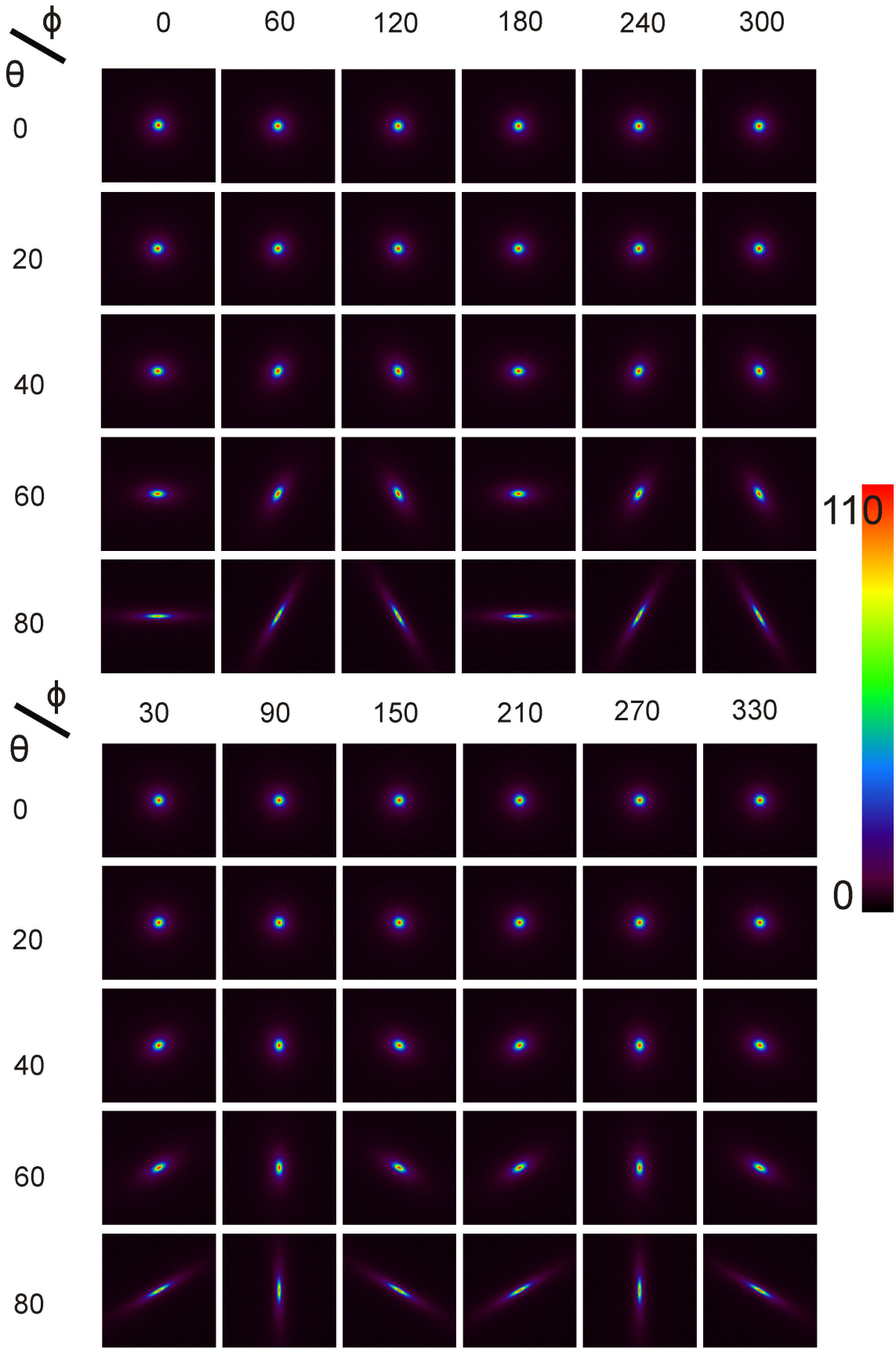


FIGURE 4. The 2D histograms of the number of particles during the simulated shower of a 10^6 particles at the center of a 64 m × 64 m square surface at different angles (ϕ, θ).

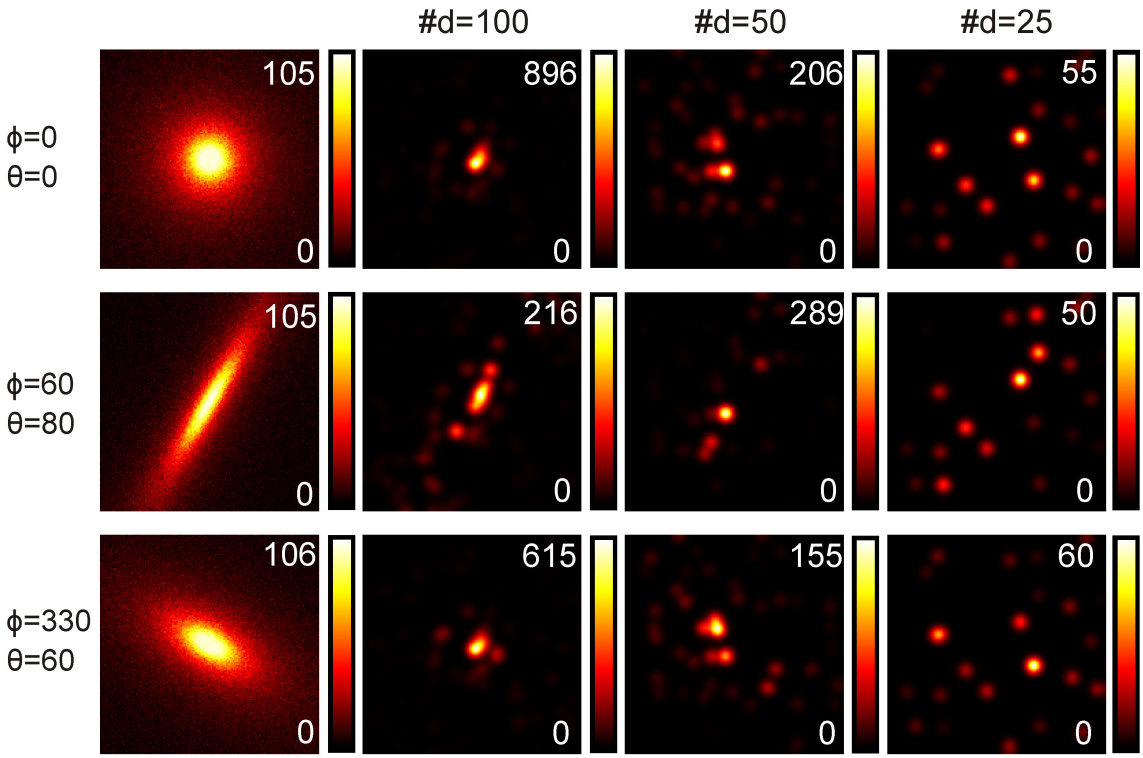


FIGURE 5. The 2D histograms of the number of particles during the simulated impact of a 10^6 muons at the center of a 16 × 16 m square surface at different angles denoted by (ϕ, θ). Density heat maps are displayed on a linear scale. The $\#d$ parameter specifies the number of detectors. The first column shows the complete particle distribution.

TABLE 1. The speed of the algorithm implementation in seconds for a given number of particles in the shower and the size of the simulation grid (in meters).

Simulation size (m x m)	Number of particles		
	10^4	10^5	10^6
16 x 16	0.85 ± 0.15	3.63 ± 0.03	37.05 ±0.21
32 x 32	0.94 ± 0.21	3.72 ± 0.06	37.19 ±0.31
64 x 64	1.01 ± 0.14	3.79 ± 0.04	37.75 ±0.41
128 x 128	1.26 ± 0.14	4.10 ± 0.04	38.00 ±0.43
256 x 256	2.50 ± 0.26	5.29± 0.16	39.34 ± 1.06
512 x 512	9.04 ± 0.91	11.40± 0.57	44.89 ± 1.99

parameter r_0 in practice has little effect on the area of the region in which the number of particles is greater than the background radiation. As r_0 increases, this area increases slightly (see Figure 3). The decisive factor for the shape of the area of increased radiation is the impact angle of the burst. As seen in Figure 4 they can have spherical or elliptical/spindle shape additionally rotated with respect to the impact center. Intuitively, the more detectors there are in an area, the closer the recorded shape of the burst particle histogram is to the real distribution (see Figure 5). With too few detectors, we can only say that radiation much larger than the background radiation was recorded, but we cannot say anything about the impact direction of the possible burst from which it came. This can be seen very well in the last column of Figure 5. In the case of cosmic ray shower observations, we do not yet have enough data to train a machine learning model in order to detect and classify events of showers or the angles of impact. Since we know the distribution of particles we were able to prepare an algorithm, which seems to be a fast and efficient method to prepare appropriate algorithms for experiments in CREDO and other projects related to the observation of cosmic ray showers. Our algorithm is much faster than available accurate simulations and its results are exactly what we need. In our previous CORSIKA based simulation discussed in [30] and performed on Prometheus supercomputer at CYFRONET computing center [34] an ensemble of $\sim$ 100 showers initiated by a particle of $10^{16} eV$ resulted in $\sim$ 10^4 muons and 10^5 electrons. These figures are compatible with simulation conditions discussed in this work. The time required to complete this simulation was of the order of a few days. Remarkably, the simulation of 10^4 muons at the ground level takes just a few seconds (see Table 1) using the approach discussed here. For 10^6 muons this figure is still around 1 minute. In the case of cosmic ray showers, typical data augmentation, which usually includes rotation, scaling or clipping, cannot be used because these are parameters that characterize a specific particle detection. Rotation is an important parameter that cannot be changed. Scaling (cone size) is determined by the number of particles of the bunch. Clipping is contained in the random distribution of the detectors.

VI. CONCLUSION

The resulting synthetic data will be used for further studies on the characteristics of the particle distribution of cosmic ray shower. We want to create an encoder-decoder based image

restoration algorithm that will allow PDF reconstruction of the entire shower based on the sampling done by the detectors. This will allow us to estimate the direction from which the particles struck. This is an extremely important issue especially in the field of fundamental research in astrophysics, where extremely desirable information is not only the fact that high-energy particles hit the ground, but also the possibility of at least approximate the source of their origin.

For the future, it is useful to speed up the algorithm. Certainly the generation of the background distribution can be parallelized. This should speed up the algorithm considerably, since the larger the image the more background needs to be drawn. At the moment, the speed of the implementation is sufficient to generate enough data for machine learning algorithms. The availability of tools realizing such simulations and enabling the providing of artificially generated data needed to develop effective analysis techniques and efficient machine recognition models of signals are particularly important in projects that implement the paradigm of large-scale distributed observation structure like CREDO or conceptually similar initiatives like CRAYFIS and DECO. Even the largest of the currently active ones do not yet have the number of active detectors to be able to realistically have an adequate volume of data to enable effective analysis and, in particular, learning of recognition models. The issue comes down not only to the still too low level of spatial saturation with mobile detectors on a global scale, but also to difficulties in developing reliable recognition methods on the basis of distributed observations in the absence of knowledge of ground truth. A slightly better situation exists in projects with a large scale fixed detector infrastructure like Auger, where the geometry of the detector structure is spatially constrained, fixed and thus the approach to detecting EAS scenarios can be implemented under more predictable conditions. Given all the problems and expectations discussed, reliable and time-efficient tools for simulating the distributions of observational data are an essential component to extend and accelerate research in projects like CREDO.

The proposed method of data generation for machine learning algorithms only simulates the distribution of particles recorded with a given detector setting. It is not a complete cosmic shower simulation as is the case in more advanced programs such as CORSIKA. This represents a trade-off between the completeness of the simulation and the speed of data generation. However, the literature we reviewed clearly indicates that the distribution obtained by the NKG method is equivalent to that which can be obtained using a complete cosmic shower simulation.

APPENDIX

LIST OF MAIN ACRONYMS

ML - Machine Learning
MC - Monte Carlo algorithm
EAS - Extensive Air Showers
CORSIKA - COsmic Ray SImulations for KAscade
CREDO - Cosmic-Ray Extremely Distributed Observatory
DECO - Distributed Electronic Cosmic-ray observatory
CRAYFIS - Cosmic Rays Found In Smartphones experiment
DNN - Deep Neural Network
AIRES - AIRshower Extended Simulations
NKG - Nakamura-Kamata-Greisen algorithm
CYFRONET - Academic Computer Centre of the University of Science and Technology in Krakow, Poland
PDF - Probability density function

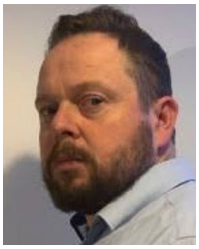
REFERENCES

[1] P. Homola et al., "Cosmic-ray extremely distributed observatory," *Symmetry*, vol. 12, no. 11, p. 1835, 2020.
[2] J. Vandenbroucke, S. BenZvi, S. Bravo, K. Jensen, P. Karn, M. Meehan, J. Peacock, M. Plewa, T. Ruggles, M. Santander, D. Schultz, A. L. Simons, and D. Tosi, "Measurement of cosmic-ray muons with the distributed electronic cosmic-ray observatory, a network of smartphones," *J. Instrum.*, vol. 11, no. 04, Apr. 2016, Art. no. P04019.
[3] D. Whiteson, M. Mulhearn, C. Shimmin, K. Cranmer, K. Brodie, and D. Burns, "Searching for ultra-high energy cosmic rays with smartphones," *Astroparticle Phys.*, vol. 79, pp. 1–9, Jun. 2016.
[4] T. Hachaj, Ł. Bibrzycki, and M. Piekarczyk, "Recognition of cosmic ray images obtained from CMOS sensors used in mobile phones by approximation of uncertain class assignment with deep convolutional neural network," *Sensors*, vol. 21, no. 6, p. 1963, Mar. 2021.
[5] O. Bar, Ł. Bibrzycki, M. Niedźwiecki, M. Piekarczyk, K. Rzecki, T. Sośnicki, S. Stuglik, M. Frontczak, P. Homola, D. E. Alvarez-Castillo, T. Andersen, and A. Tursunov, "Zernike moment based classification of cosmic ray candidate hits from CMOS sensors," *Sensors*, vol. 21, no. 22, p. 7718, Nov. 2021.
[6] M. Borisyak, M. Usvyatsov, M. Mulhearn, C. Shimmin, and A. Ustyuzhanin, "Muon trigger for mobile phones," *J. Phys., Conf.*, vol. 898, Oct. 2017, Art. no. 032048.
[7] M. Winter, J. Bourbeau, S. Bravo, F. Campos, M. Meehan, J. Peacock, T. Ruggles, C. Schneider, A. L. Simons, and J. Vandenbroucke, "Particle identification in camera image sensors using computer vision," *Astroparticle Phys.*, vol. 104, pp. 42–53, Jan. 2019.
[8] K. Nakamura, "Review of particle physics," *J. Phys. G: Nucl. Part. Phys.*, vol. 37, no. 7A, Jul. 2010, Art. no. 075021.
[9] T. Antoni et al., "The cosmic-ray experiment kascade," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 513, no. 3, pp. 490–510, 2003.
[10] W.-D. Apel et al., "Probing the evolution of the EAS muon content in the atmosphere with KASCADE-grande," *Astroparticle Phys.*, vol. 95, pp. 25–43, Oct. 2017.
[11] R. U. Abbasi et al., "Constraints on the diffuse photon flux with energies above 1018 eV using the surface detector of the telescope array experiment," *Astroparticle Phys.*, vol. 110, pp. 8–14, Jul. 2019.
[12] A. Aab et al., "Direct measurement of the muonic content of extensive air showers between 2×10^{17} and $2\times$ ^{1018} eV at the Pierre Auger Observatory," *Eur. Phys. J. C*, vol. 80, no. 8, p. 751, 2020.
[13] T. Bister, M. Erdmann, J. Glombitza, N. Langner, J. Schulte, and M. Wirtz, "Identification of patterns in cosmic-ray arrival directions using dynamic graph convolutional neural networks," *Astroparticle Phys.*, vol. 126, Mar. 2021, Art. no. 102527.
[14] A. Aab et al., "Deep-learning based reconstruction of the shower maximum X max using the water-Cherenkov detectors of the Pierre Auger observatory," *J. Instrum.*, vol. 16, Jul. 2021, Art. no. P07019.
[15] M. Piekarczyk et al., "CNN-based classifier as an offline trigger for the CREDO experiment," *Sensors*, vol. 21, no. 14, p. 4804, 2021.
[16] D. Heck, J. Knapp, J. N. Capdevielle, G. Schatz, and T. Thouw, "CORSIKA: A Monte Carlo code to simulate extensive air showers," Forschungszentrum Karlsruhe GmbH, Karlsruhe, Germany, Tech. Rep. FZKA 6019, 1998.
[17] S. J. Sciutto, "AIRES a system for air shower simulations," Departamento de Física and IFLP (CONICET), Universidad Nacional de La Plata, La Plata; Argentina, Tech. Rep. 19.04.00, 2019.
[18] S. Agostinelli, "GEANT4-a simulation toolkit," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 506, no. 3, pp. 250–303, 2003.

[19] S. N. Axani, K. Frankiewicz, and J. M. Conrad, "The CosmicWatch desktop muon detector: A self-contained, pocket sized particle detector," *J. Instrum.*, vol. 13, no. 3, Mar. 2018, Art. no. P03019.
[20] Ł. Bibrzycki, D. Burakowski, P. Homola, M. Piekarczyk, M. Niedźwiecki, K. Rzecki, S. Stuglik, A. Tursunov, B. Hnatyk, D. E. A. Castillo, K. Smelcerz, J. Stasielak, A. R. Duffy, L. Chevalier, E. Ali, L. Lakerink, G. B. Poole, T. Wibig, and J. Zamora-Saa, "Towards a global cosmic ray sensor network: CREDO detector as the first open-source mobile application enabling detection of penetrating radiation," *Symmetry*, vol. 12, no. 11, p. 1802, Oct. 2020.
[21] N. Gerasimova and G. Zatsepin, "Disintegration of cosmic ray nuclei by solar photons," *Sov. Phys. JETP*, vol. 11, no. 899, pp. 64–157, 1960.
[22] T. Wibig, "Small shower CORSIKA simulations," *Chin. Phys. C*, vol. 45, no. 8, Aug. 2021, Art. no. 085001.
[23] J. Knapp and D. Heck, "Extensive air shower simulation with CORSIKA: A user's manual," Kernforschungszentrum Karlsruhe, Forschungszentrum Karlsruhe GmbH, Karlsruhe, Germany, Tech. Rep. KfK 5196 B, 1993.
[24] J. G. Wilson and K. Greisen, *Progress in Cosmic Ray Physics*, vol. 3. Amsterdam, The Netherlands: North Holland, 1956.
[25] K. Kamata and J. Nishimura, "The lateral and the angular structure functions of electron showers," *Prog. Theor. Phys. Suppl.*, vol. 6, pp. 93–155, Feb. 1958.
[26] M. Senniappan, Y. Becherini, M. Punch, S. Thoudam, T. Bylund, G. K. Mezek, and J.-P. Ernenwein, "Signal extraction in atmospheric shower arrays designed for 200 GeV–50 TeV γ-ray astronomy," *J. Instrum.*, vol. 16, no. 7, Jul. 2021, Art. no. P07050.
[27] H. Nakada, A. Shiomi, M. Ohnishi, T. K. Sako, K. Hibino, and Y. Katayose, "Study of water Cherenkov detector to improve the angular resolution of an air-shower array for ultra-high-energy gamma-ray observation," *Experim. Astron.*, vol. 53, no. 3, pp. 991–1016, Jun. 2022.
[28] M. T. Dova, L. N. Epele, and G. A. Mariazzi, "Particle density distributions of inclined air showers," *Nuovo Cim. C*, vol. 24, pp. 745–750, Jan. 2001.
[29] K. Greisen, "Cosmic ray showers," *Annu. Rev. Nucl. Sci.*, vol. 10, no. 1, pp. 63–108, 1960.
[30] J. S. Pryga, W. Stanek, K. W. Woźniak, P. Homola, K. A. Cheminant, S. Stuglik, D. Alvarez-Castillo, Ł. Bibrzycki, M. Piekarczyk, O. Bar, T. Wibig, A. Tursunov, M. Niedźwiecki, T. Sośnicki, and K. Rzecki, "Analysis of the capability of detection of extensive air showers by simple scintillator detectors," *Universe*, vol. 8, no. 8, p. 425, Aug. 2022.
[31] J. A. Abraham, P. Abreu, M. Aglietta, C. Aguirre, C. Aguirre, E.-J. Ahn, D. Allard, I. Allekotte, J. Allen, P. Allison, and J. Alvarez-Muniz, "Atmospheric effects on extensive air showers observed with the surface detector of the Pierre Auger observatory," *Astroparticle Phys.*, vol. 32, no. 2, pp. 89–99, 2009.
[32] J. Hurtado and A. Barbat, "Monte Carlo techniques in computational stochastic mechanics," *Arch. Comput. Methods Eng.*, vol. 5, pp. 3–30, Jan. 1998.
[33] T. Lindeberg, "Scale-space for discrete signals," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 3, pp. 234–254, Mar. 1990.
[34] (2022). Cyfronet. *Prometheus—Computing Resources*. Accessed: Nov. 24, 2022. [Online]. Available: https://kdm.cyfronet.pl/portal/Prometheus:en

TOMASZ HACHAJ received the M.S. degree in computer science from the Krakow University of Technology, Poland, in 2006, the Ph.D. degree in computer science from the AGH University of Science and Technology, Krakow, Poland, in 2010, and the D.S. (Habilitation) degree in computer science from the Wrocław University of Science and Technology, Poland, in 2017.

He is the Head of the Department of Signal Processing and Pattern Recognition and the Deputy Director of Scientific and Organizational Matters with the Institute of Computer Science, Pedagogical University of Krakow, Poland. He has participated in various Polish national projects, being involved at both the technical/research and administrative levels. He is involved in the teaching activity of various second (master's) level courses in the field of computer science. He has authored and coauthored over 90 publications. His research interests include oriented to the development and application of signal processing and pattern recognition methods.



ŁUKASZ BIBRZYCKI received the Ph.D. degree in theoretical physics from the Institute of Nuclear Physics Polish Academy of Sciences, Kraków, Poland, in 2009. He is an Assistant Professor with the Institute of Computer Science, Pedagogical University of Krakow, Kraków. He is the author of more 20 research papers in scientific journals and on international scientific conferences. His research interests include computer modeling of nuclear and high energy particle reactions and the analysis radiation tracks with machine learning methods.

MARCIN PIEKARCZYK (Member, IEEE) received the graduate degree in automatics and robotics, in 2000, and the Ph.D. degree in computer science from the AGH University of Science and Technology, Krakow, Poland, in 2011. He is currently an Assistant Professor with the Institute of Computer Science, Pedagogical University of Krakow, Poland. He has published more than 50 papers in scientific conferences and peer-reviewed journals. His research interests include pattern recognition, machine learning, biometrics, information security, deep learning architectures, graph languages, gesture-based biometrics, and cosmic-rays detection. He is a member of IEEE CS, IEEE SMC, and ACM.