

RESEARCH

Open Access



Automated detection of meteor trails in all sky images using Radon transformation and convolutional neural network

Tomáš Suk^{1*} and Stanislava Šimberová²

*Correspondence:

Tomáš Suk

suk@utia.cas.cz

¹Institute of Information Theory and Automation, The Czech Academy of Sciences, Pod vodárenskou věží 1143/4, Praha 8

182 00, Czech Republic

²Astronomical Institute, The Czech Academy of Sciences, Fričova 298, Ondřejov 251 65, Czech Republic

Abstract

We present an automated pipeline for detecting meteor trails in single-frame, half-minute-cadence all-sky images acquired by the Digital Autonomous Fireball Observatories (DAFO) network. Unlike existing approaches that rely either on handcrafted feature extraction or purely data-driven methods, the proposed system combines classical image processing techniques (Radon transformation and Fourier-transformation-based filtering) with a convolutional neural network (CNN) for robust classification of candidate trails. The method is specifically designed for fish-eye camera images and operates without dependence on a particular hardware configuration. It addresses limitations of our previous approach Suk and Šimberová [1], namely high false-positive rates and insufficient robustness to ambiguous trails such as blinking satellites and airplanes. The proposed pipeline was evaluated on data from eleven nights across multiple observatories. For clearly identifiable meteor events, the method achieves a precision of 93.9% and a recall of 95.3%, with near-complete detection of all observable meteors not obscured by ground objects. The system also significantly reduces the number of false detections compared to earlier methods. These results demonstrate that combination of classical detection techniques with modern neural network classification provides an efficient and computationally feasible solution for large-scale automated meteor monitoring.

Keywords Meteor detection, Autonomous fireball observatories, Fish-eye camera, Radon transformation, Convolution neural network

1 Introduction

The detection of meteors in wide-field astronomical images is an important task for studying meteoroid flux, atmospheric entry processes, and the physical properties of small solar system bodies. Modern meteor observation networks, such as all-sky camera systems, generate large volumes of image data that require automated processing to enable efficient analysis 1.

Traditional approaches to meteor detection are typically based on handcrafted image processing techniques, such as thresholding, Hough or Radon transformations, and temporal differencing. These methods are computationally efficient and interpretable,



© The Author(s) 2026. **Open Access** This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

but they often struggle with complex observational conditions, including varying background illumination, cloud coverage, and the presence of artificial objects such as airplanes and satellites.

When a meteor is recorded by at least two observatories, we can compute its trajectory. This allows us both to identify potential meteorite fall locations and determine the origin of the corresponding meteoroid (a meteor for which both orbital information and recovered meteorite material are available is sometimes called meteorite with pedigree). When many trails from a meteor shower are recorded, they can contribute to our understanding of the physics. E.g. Joiret and Koschny [2] deal with the Southern Delta Aquariids meteor shower. A meteorite found thanks to DAFO is described in Spurný et al. [3].

In recent years, machine learning methods, particularly convolutional neural networks (CNNs), have been increasingly applied to astronomical image analysis, e.g. a new non-meteoritic paper [4] or an older one about meteors [5]. These approaches offer improved robustness and adaptability compared to purely handcrafted methods. However, fully data-driven solutions require large, well-labeled datasets and may lack transparency in decision-making. In the context of meteor detection, the available labeled data are often limited, and the variability of observed phenomena poses additional challenges.

The Digital Autonomous Fireball Observatories (DAFO) network provides a valuable source of all-sky fish-eye images with half-minute cadence. These data are characterized by strong geometric distortion, varying observational conditions, and the presence of numerous non-meteor trails, such as satellite tracks and airplane signals. Efficient processing of such data therefore requires methods that are both robust and computationally feasible.

In our previous work Suk and Šimberová [1], we developed a detection pipeline based primarily on classical image processing techniques. While this approach demonstrated practical applicability, it exhibited several limitations, including a relatively high rate of false detections and reduced robustness in the presence of ambiguous or weak trails.

The present work addresses these limitations by introducing a hybrid detection pipeline that combines classical signal processing methods with modern machine learning techniques. Candidate trails are first extracted using the Radon transformation and further filtered using the criteria based on the Fourier transformation. These candidates are then classified by a convolutional neural network into multiple classes representing different types of observed trails.

This hybrid approach leverages the strengths of both methodologies: the efficiency and interpretability of classical techniques and the classification power of neural networks. In contrast to purely data-driven approaches, the proposed method reduces the dependence on large training datasets while maintaining robustness to challenging observational conditions.

The proposed system is specifically designed for single-frame fish-eye images and does not rely on temporal information or specific hardware configurations. This makes it suitable for deployment in distributed meteor observation networks and for processing archival data.

The main contributions of this work can be summarized as follows:

- A hybrid meteor detection pipeline combining Radon transformation, Fourier-based filtering, and CNN-based classification.

- A multi-class classification framework capable of distinguishing meteors from various types of non-meteor trails, including satellites and airplanes.
- An evaluation on real observational data from multiple observatories, demonstrating high detection reliability and low false-positive rates under realistic conditions.

The paper is organized according to the algorithmic workflow. Section 2 describes data acquisition; Sect. 3 provides an overview of the algorithm; Sect. 4 discusses preprocessing (masking of ground objects and differencing of adjacent frames); Sect. 5 explains the Radon transformation application; Sect. 6 presents the Fourier test; Sect. 7 addresses terminal bursting meteors; Sect. 8 describes the neural network; Sect. 9 shows our results; and Sect. 10 concludes the paper.

1.1 State of the art

The automatic detection of meteor trails in visual data has been a long-standing challenge in astronomical image processing. Several approaches have been developed over the past decades, typically designed for specific image acquisition hardware, which often limits their general applicability.

1.1.1 Video-based detection systems

Many established systems focus on processing video sequences obtained from dedicated meteor detection networks. For example, Molau [6, 7], and the UFOCapture software by SonotaCo [8] have been widely used within the European and Japanese meteor observation communities. Further developments can be found in Gural et al. [9], Gural [10], and McGarvey et al. [11].

The CAMS (Cameras for Allsky Meteor Surveillance) network employs automated software to detect meteor showers in the night sky and validate entries in the IAU Working List of Meteor Showers, [12–14].

Similarly, the Croatian Meteor Network (CMN) has developed custom software for video-based meteor detection and trajectory reconstruction [15]. A new algorithm for the Allsky 7 data was published in Sennlaub et al. [16]. Anghel et al. [17] processes data from Romanian Meteorites Orbits Reconstruction by Optical Imaging (MOROI) network of video cameras.

Although effective within their respective environments, these solutions are closely tied to their camera systems and require precise tuning, limiting their portability.

1.1.2 Radar and radio-based methods

Alternative meteor detection strategies have also been developed using radar and radio techniques. The BRAMS (Belgian RADio Meteor Stations) project in Belgium [18] and MeteorScan are notable examples that enable meteor detection under cloud cover or during daytime. Mallama and Espenak [19] also employed similar methods. Grigg et al. [20] deal with the meteor ionisation trails in radio images. Vankawala et al. [21] investigate very low frequency radio emissions from meteors. While powerful, such methods lack the visual data needed for morphological classification and trajectory visualization. Li et al. [22] evaluates the data collected with the Jicamarca 50 MHz incoherent scatter radar by the convolutional neural network.

1.1.3 Classical image processing approaches

Before the widespread use of machine learning, many efforts relied on handcrafted feature extraction techniques. The Hough transformation has been used in works such as Trayner et al. [23], Cheselka [24], and Reffet et al. [25] to detect straight-line features associated with meteor trails. Our own earlier research [1, 26] applied the Hough and Fourier transformations to identify linear structures and periodic signals (e.g. dashes from electronic shutters).

1.1.4 Deep learning and neural network methods

In recent years, deep learning techniques have been introduced to overcome the limitations of classical methods. Roman and Buiu [27] explored self-organizing maps for classification, while Gural [28] applied recurrent neural networks on light curves and CNN on imagery chips for shallow layered and less computationally intense CNN to detect meteors. It actually operates on the single streak from multiple “stacked” video frames.

More recently, Cecil and Campbell-Brown [29] integrated CNNs into meteor detection pipelines within the Canadian Automated Meteor Observatory (CAMO). Another modern development is the work of Ciocan et al. [30], which applies CNN-based image analysis to data collected from weather balloons and airplanes, demonstrating resilience to camera motion. Shirasuna and Gradwohl [31] use the architecture of the network with an attention mechanism. Peña-Asensio et al. [32] combines Grad-CAM algorithm for meteor tracking and ResNet-34 neural network for their classification.

These learning-based methods show significant promise in improving generalization and reducing false detections. However, their application is often limited by the size and quality of available training datasets, the specific characteristics of video streams, and the lack of cross-system standardization.

2 Source of data

Our source of data is DAFO (Digital Autonomous Fireball Observatories) network thanks to Pavel Spurný and his team in the Czech Republic; details can be found in [33]. DAFO consists of 21 observatories distributed over Czechia, Slovakia, Austria and Germany distanced less than 200 km from each other. In addition to the necessary electronics, each autonomous observatory includes two specialized fish-eye digital cameras that observe the entire sky and operate in dual mode. The cameras are labeled as camera 0 and camera 1.

2.1 Digital camera parameters

Each camera alternates between 35 s of image capturing and a 25-second pause for data storage, with a 5-second overlap, see Fig. 1. The active phase of the camera is interrupted with a frequency of 15 Hz and a duty cycle of 50%, i.e., 1/30 s exposure and 1/30 s pause; every 15th pause is omitted. The older equipment had the mechanical rotating shutter providing these interruptions, now, it is purely electronic. Regardless, we still call it “electronic shutter”. Using these parameters, we can estimate the velocity of a moving object, i.e. each longer dash means the one second. When a long trail is not dashed, it is something slow.

The observation frequencies are configured so that a meteor’s luminous trail appears as a regularly interrupted curve, while slower-moving objects evince trails that are either

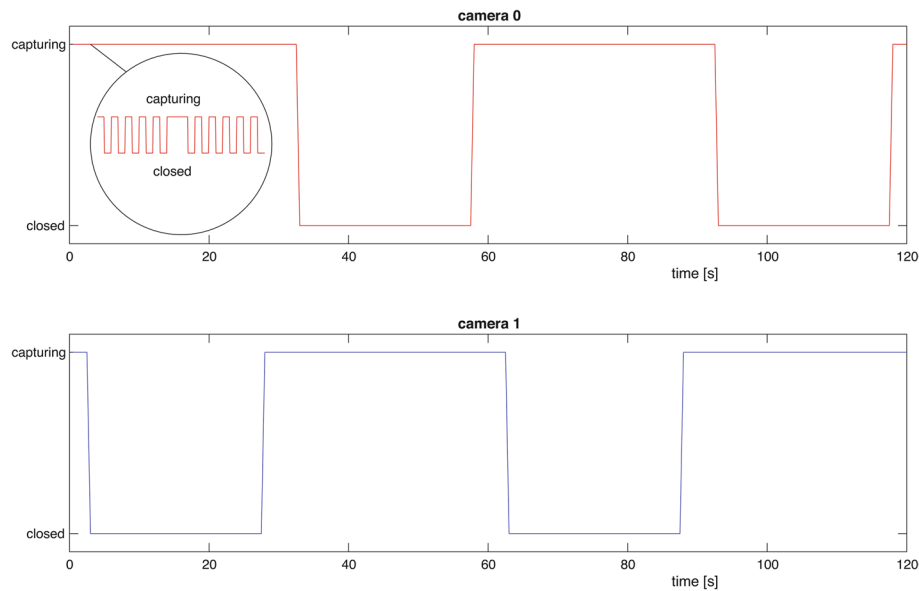


Fig. 1 Scanning curves of both cameras with the alternating capturing. The magnifying glass in the upper subimage shows interrupted 15-Hz capturing with omitted 15th pause

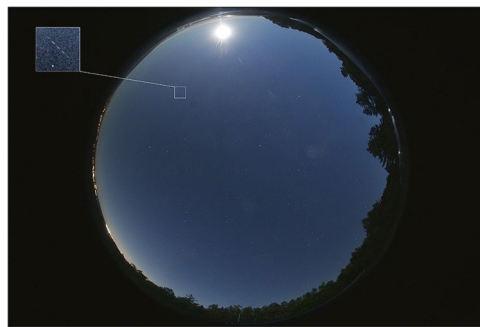


Fig. 2 Example of data from the Ondřejov observatory: image from 11 July 2022, 21:45:30, camera 1, with a meteor detail

irregularly interrupted (airplanes) or continuous (satellites). The size of the color 8-bit images is 5472×3648 pixels. The color information is of minor importance since meteor trails appear white; therefore, we process only the brightness channel defined as $Y = 0.299R + 0.587G + 0.114B$, where R , G , and B are the color channels. A captured object that is too colorful usually indicates a source other than a meteor. Such colorful trails are rare, so colorfulness is not explicitly tested in our processing. An example from the Ondřejov observatory is shown in Fig. 2.

2.2 Fish-eye camera model

DAFO uses fish-eye cameras, which can capture the entire sky. However, the originally straight trails of flying objects appear as arcs in the images, requiring distortion compensation. The formulas used for this compensation are provided in Appendix A.

The pinhole camera image would be significantly larger than that obtained using a fish-eye camera. Theoretically, the horizon would be mapped to infinity when the entire hemisphere is used. These extreme positions, however, are not used in practice because the real horizon lies higher. Nevertheless, the memory requirements and computational

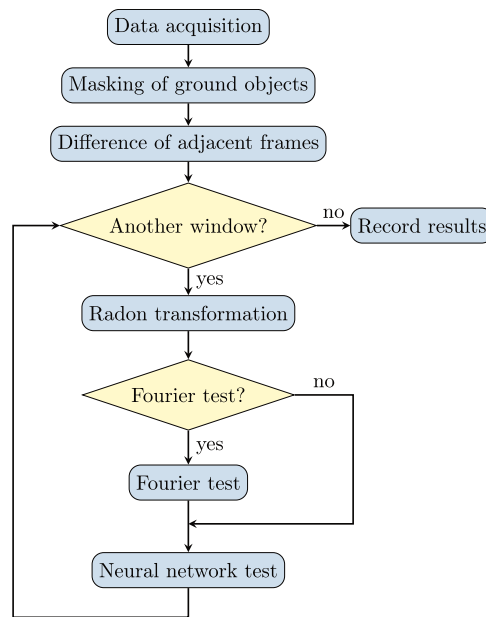


Fig. 3 Flowchart of the proposed procedure for one image

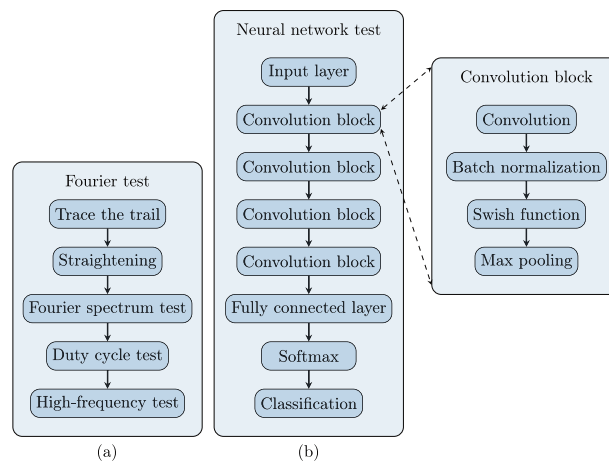


Fig. 4 Structure of **a** the Fourier test and **b** the neural network test

complexity remain high; therefore, computing the full pinhole camera image should be avoided whenever possible.

3 Proposed algorithm flowchart

The flowchart of the proposed method is shown in Fig. 3. The details of the simplified Fourier test are depicted in Fig. 4a, and the architecture of the neural network is shown in Fig. 4b. The algorithm consists of several components. First, during preprocessing, ground objects are masked and static sky objects are removed by subtracting adjacent images with compensation for the Earth's rotation. Next, linear trails are detected using the Radon transformation. The detected traces are subjected to the Fourier test, which can be omitted when short trails with only a single dash are also of interest. Finally, the trail undergoes testing by the neural network.

4 Preprocessing

In the preprocessing stage, we need to remove static objects from the snaps. These can be divided into two categories: objects static on the ground and objects static in the sky. One of these can be removed by subtracting adjacent snaps. Based on experiments, we decided to remove the static sky objects by subtracting adjacent snaps with compensation for the Earth's rotation, and to remove the static ground objects using a fixed mask. When we have adjacent images from a given sequence available, we do not need to use such sophisticated methods as Cohen and Lu [34], which work with only one image.

4.1 Ground objects

Ground objects in the image appear near the horizon and typically remain stable overnight (with the minor exception of occasional headlights from passing cars). These objects can generally be divided into illuminated (e.g. reflective surfaces, building windows) and non-illuminated (e.g. trees, poles). A light source such as a satellite or airplane passing behind a tree can resemble a dashed meteor trail. Similarly, distant road lights can sometimes appear as meteor-like traces. Therefore, the removal of ground objects is highly desirable.

We use a binary image as a mask to remove these objects. A separate mask must be created for each camera in the pair. It is better to make a specific mask for each night, but when the weather (mainly clouds) prevents from doing a good mask, it is better to use a good mask from a few days before or after the current day.

The flowchart for ground mask creation is shown in Fig. 5. The algorithm was developed for wide types of images from various seasons and luminous conditions. The inner radius of 1459 pixels and the outer sky radius of 1824 pixels were derived so we are sure to take really the inner and outer parts of the sky and no part of the sky is removed uselessly.

When we searched the adaptive coefficients a_1 , a_2 (see the block “adaptive coefficient” in Fig. 5), we took a few “critical” images, where we needed different optimal values of the adaptive coefficient A_C . So, we compute them by a linear regression with the result $A_C = a_1 T_C - a_2$, where $a_1 = 17$ and $a_2 = 2.27$. The neighborhood radius 115 pixels for the adaptive threshold computation was derived from the typical size of the ground objects.

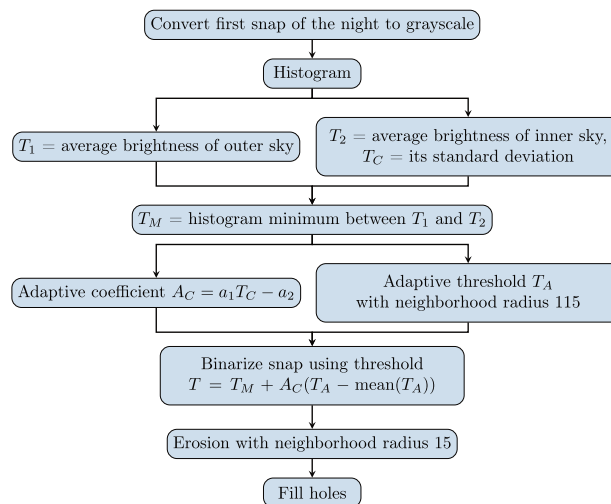


Fig. 5 Flowchart of the ground object mask creation

The erosion operation improves the results; we apply a morphological operation with a circular structuring element of radius 15 pixels. Finally, we fill the holes in both the sky and the background. The erosion radius (15 pixels) is a variable parameter, while the other parameters are considered optimal. A detailed example of a mask is shown in Fig. 6.

4.2 Subtraction of static sky objects

The goal of this step is to remove all static sky objects, especially the Moon, stars, and planets. For a nightly sequence of snaps, we subtract the adjacent snaps from the current one, thereby removing stable objects and preserving the moving ones. However, this process introduces a complication caused by the Earth's rotation. A one-minute difference between adjacent snaps from the same camera corresponds to a quarter of a degree of rotation, which represents up to 12 pixels in our image data and therefore cannot be neglected. Compensation for this rotation would be straightforward for a pinhole camera, but it is more complex for a fish-eye camera; we must use a little astrometry.

The coordinates of each pixel are first converted to the pinhole model as described in Appendix A. They are then rotated around the celestial north pole $\{x_o, y_o\}$ by the angle β and finally converted back to fish-eye coordinates. The rotation around the north pole is performed using the following formulas

$$\begin{aligned} x' &= x_o + (x - x_o) \cos \beta - (y - y_o) \sin \beta \\ y' &= y_o + (x - x_o) \sin \beta + (y - y_o) \cos \beta, \end{aligned} \quad (1)$$

where $\beta = 0.25^\circ$ for the previous snap to move it 1 min ahead and $\beta = -0.25^\circ$ for the next snap to move it 1 min back.

Because our fish-eye camera model is not perfect, the stars are not subtracted completely. To mitigate this, we apply a “maximum”-type filter to both the previous and next snaps. This filter replaces the value of each pixel with the maximum in its circular neighborhood with a diameter of 7 pixels. The filtered previous and next snaps are then subtracted from the current snap with suitable weights.

The overall brightness of the images changes mainly just after sunset and moonrise and just before sunrise and moonrise. It is useful to compensate for these effects, which

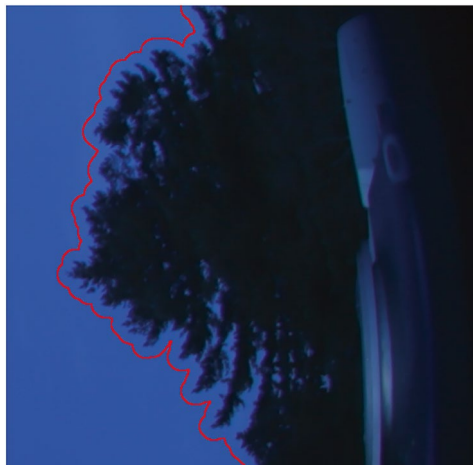


Fig. 6 Detail of the red ground mask boundary. Objects on the right of the red curve are deleted

affect the entire sequence. When we assume the changes are approximately linear, we can use the following formula

$$f'_c(x, y) = f_c(x, y) - \frac{1}{2} \left[(f_p(x, y) - \mu_p) \frac{\sigma_c}{\sigma_p} + \mu_c \right] - \frac{1}{2} \left[(f_n(x, y) - \mu_n) \frac{\sigma_c}{\sigma_n} + \mu_c \right], \quad (2)$$

where $f_c(x, y)$ is the current snap, $f_p(x, y)$ the previous snap, and $f_n(x, y)$ the next snap. The previous and next snaps are already compensated for the Earth's rotation, ground-object masking, and maximum filtering. The symbols μ_c , μ_p , and μ_n denote the mean brightness of the current, previous, and next snaps, respectively, while σ_c , σ_p , and σ_n represent their brightness standard deviations.

An example is in Fig. 7, where the central parts of the images are shown. Even if the original image includes the Moon and stars, we can work with the meteor only in the difference image. The finishing twilight can be seen on the background, therefore the contrast and brightness normalization help us. We work with $\mu_p = 0.1468$, $\sigma_p = 0.0627$, $\mu_c = 0.1797$, $\sigma_c = 0.0701$, $\mu_n = 0.1793$, and $\sigma_n = 0.0699$, so especially the brightness shift between the current and previous images is not negligible.

At the beginning of the sequence, when there is no previous snap, and at the end, when there is no next snap, we subtract only one snap without the coefficient 1/2. After subtraction, negative values are replaced by zeros. The movement of planets is negligible compared to the Earth's rotation; therefore, we remove them along with the stars and the Moon.

4.2.1 Center of rotation

A separate task is determining the location of the celestial north pole $\{x_o, y_o\}$ in the snaps, which serves as the rotation center. Since the pole is close to Polaris, we define a square region where Polaris is the only prominent star. The pixel with the maximum brightness detected within this square corresponds to the position of Polaris (see Fig. 8). An elliptical regression [35] of the Polaris trail is then computed, and the ellipse center represents the coordinates of the celestial north pole. This procedure is repeated for each camera separately. The determined coordinates can be reused for multiple nights, but since the

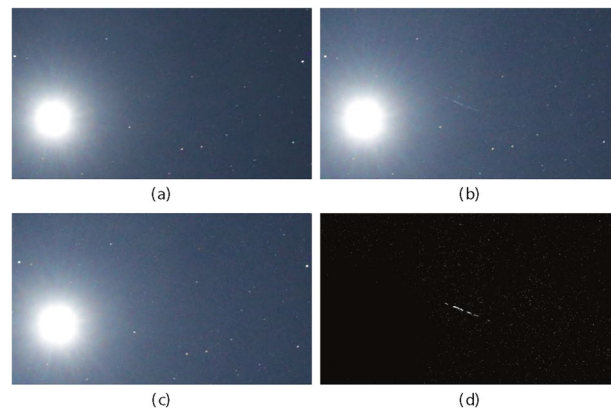


Fig. 7 Subtraction of the previous and next images from Ondřejov 12th March 2022, camera 0. **a** the previous image from 18:31:00, **b** the current image from 18:32:00 with a meteor, **c** the next image from 18:33:00, **d** The difference image. The original image contains the Moon and stars, regardless we can work only with the meteor in the difference image

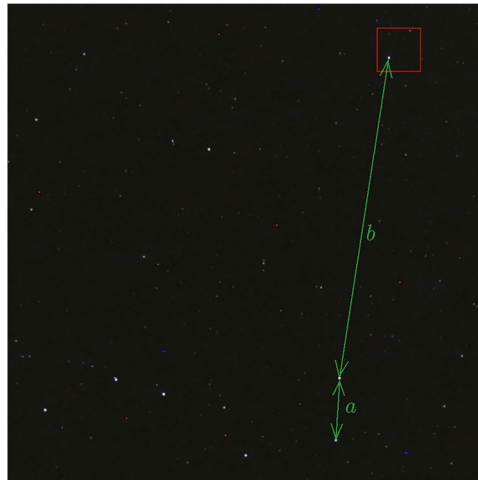


Fig. 8 Detail of the north pole search in a snap. Polaris is located as the maximum brightness pixel within the red window. The back side of the “Big Dipper” is visible in the lower part of the figure. Polaris can be found approximately five times the distance of the extension of the line connecting the rear stars of the Big Dipper ($b/a = 5.25 \approx 5$)

camera position can slowly change over time, we recommend repeating the computation at least once per month and after any camera manipulation. A clear night with visible Polaris should be used for this calibration.

By default, the square window has its upper-left corner at coordinates $x_r = 2736$, $y_r = 2700$. If the algorithm fails (i.e., the deviation of the elliptical regression is too large), we also test a square with the same x_r coordinate but $y_r = 936$. The default window size is 40×40 pixels. If the camera orientation is non-standard, both the position and the size of the neighborhood can be manually adjusted.

5 Trail search

After eliminating the ground objects using the mask and the celestial objects by subtracting adjacent snaps, only the trails of moving objects remain. To detect them, we employ the Radon transformation, which is suitable for identifying straight line segments in an image. The formulas of the Radon transformation are provided in Appendix B.

5.1 Radon transformation parameters

Applying the Radon transformation to the entire image is computationally inefficient, as a very small angular step is required, which increases processing time. Moreover, short and faint meteor trails are difficult to detect when analyzing the entire image at once. Therefore, we divided the image into 18×12 square windows, each of size 304×304 pixels, and computed the Radon transformation in each window separately.

A potential problem arises when a meteor trail lies on the border between two windows. In such cases, the fragments of the trail within individual windows may be too short to detect. To treat this, we apply a 10% overlap between windows, corresponding to 31 pixels. As a result, the windows in the central region of the image have dimensions 366×366 pixels. The empty windows in the first and last two columns are excluded from computation, so in practice, 12×12 windows are used.

We also considered to use the Radon transformation with the correction to the fish-eye camera. The result of many experiments is that the trail part in one window is almost

straight and can be well found by the traditional Radon transformation for the straight lines even if the complete trail in the whole image is arched.

Based on experimental optimization, we set the angular step θ to 0.5° and the shift step t to 3 pixels. A 3-pixel step for t integrates a 3-pixel-thick line into one array element, thereby improving the likelihood of detecting weak meteors. The global maximum and significant local maxima in the resulting array are then identified. After a maximum is found, its surrounding neighborhood of 59×9 elements is cleared to ensure sufficient separation between detected maxima.

6 Fourier test

The structure of the optional Fourier test is shown in Fig. 4a. Each line detected by the Radon transformation is traced to determine the corresponding trail. The curve is reparametrized such that adjacent points are spaced by one pixel (reparametrization by distance) and then straightened to become a straight line. An example of the light curve in a trail is in Fig. 9.

Because of a meteor trail may begin or end within the window, we apply a brightness threshold $\theta_b = 1.5\%$ of the maximum brightness. The portions of the trail near the borders that fall below this threshold are discarded, and only the section between the first and last above-threshold points is analyzed. This trail segment is then subjected to several tests to distinguish meteors from other phenomena.

Although the Fourier test is an older method and not always sufficient on its own, it effectively eliminates some false detections. Compared to [1], we simplified the approach and relaxed some thresholds. By default, the Fourier test is applied; omitting it increases the number of detected trails but also a number of false alarms. The test primarily identifies dashed-line trails resulting from the electronic shutter. To detect meteors consisting of a single dash, the Fourier test must be skipped.

6.1 Fourier spectrum test

We first identify the longest continuous segment of the trace, where gaps (pixels with brightness below θ_b) do not exceed 20 pixels. The 1D Fourier transformation of this segment, consisting of d samples s_i , is computed as

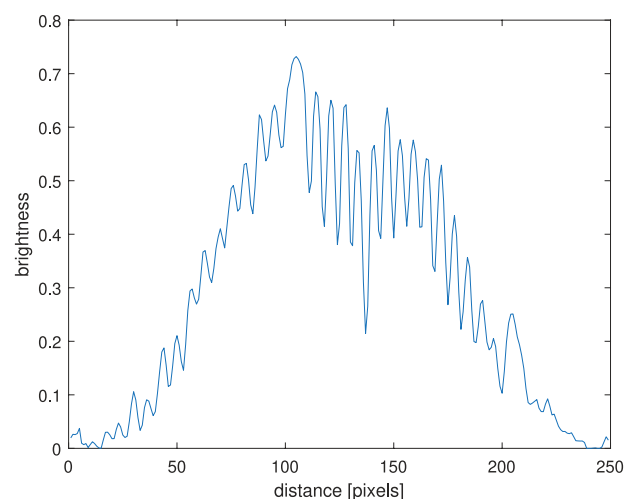


Fig. 9 Light curve of the tested trail from Ondřejov, 13 January 2014, 03:01:30, camera 0

$$a_j = \sum_{i=0}^{d-1} s_i e^{-2\pi i j \sqrt{-1}/d}, \quad j = 0, \dots, d-1. \quad (3)$$

The amplitude spectrum $|a_j|$ typically shows a dominant local (often global) maximum at $j = 0$, corresponding to the total brightness of the trail. For a meteor trace, another local maximum is expected at the frequency of the electronic shutter. The Fourier spectrum naturally decreases from the zero frequency towards to the higher frequencies. To partially compensate it, we use the spectrum normalized by the square root of the frequency index counting from one

$$a'_j = 100 |a_j| \frac{\sqrt{j+1}}{d+1}. \quad (4)$$

In the Fourier spectrum test, we first look for the first local minimum $p_n < 1.1$ for $3 < n < d$ (assuming the brightness is normalized to $\langle 0, 1 \rangle$). Next, we find the maximum p_m for $n < m < d_h$, where $d_h = \min\{d/2, 64\}$. The number of local maxima between n and d_h exceeding 0.7 is denoted p_x . The trail passes the Fourier spectrum test if $d > 20$, $p_n/p_m < 1.1$, $p_m > 0.4$, $d/m > 3$, and $1 \leq p_x \leq 3$. We can summarize that the trail must be sufficiently long, the first local maximum in the spectrum must be sufficiently significant in suitable position and the number of the local maxima must be from 1 to 3.

An example of the normalized amplitude spectrum a'_j used for the Fourier test is shown in Fig. 10. In this case, the spectrum exhibits a prominent local maximum $p_m = 14.85$ at $m = 40$. The other relevant parameters are $d = 351$, $p_n/p_m = 0.132$, $d/m = 10.97$, and $p_x = 1$.

6.2 Duty cycle test

The Fourier test also includes a duty cycle evaluation. The electronic shutter has a duty cycle of 50%, meaning that the opening time equals half the total period. The trail should exhibit a similar ratio: the illuminated portions (dashes) should occupy approximately 50% of the total trace length.

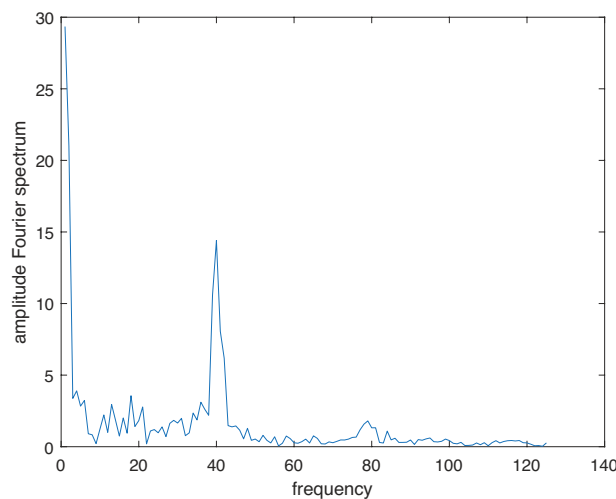


Fig. 10 Normalized amplitude Fourier spectrum of the tested trail from Ondřejov, 13 January 2014, 03:01:30, camera 0 with a peak on the electronic shutter frequency

Because of the brightness fluctuates along the trace, we detect points where a dash transitions into a dark gap. The reference level is determined as the midpoint between the local minimum (dark space) and maximum (bright dash) of the light curve (see Fig. 9). The light curve is smoothed using a Gaussian filter with a length of 21 and standard deviation of 2.5 pixels. This level is then subtracted from the original signal and shifted upward by $1/25$ of the global maximum to improve stability.

The duty cycle c is computed as $c = s_c/d_c$, where s_c is the number of pixels above the level and d_c is the distance between the first and last such pixel. The signal passes the test if $20\% \leq c \leq 70\%$.

6.3 High-frequency test

The high-frequency test, also part of the Fourier analysis, verifies that the spectrum does not contain excessive high-frequency noise. Meteor trails should not exhibit much higher frequencies than those associated with the electronic shutter, whereas airplane traces or clouds often do. The presence of the high frequencies is measured between quarter and half frequencies of the sampling (between Nyquist frequency and its half)

$$v = \frac{1}{d/2 - d/4 + 1} \sum_{j=d/4}^{d/2} a'_j. \quad (5)$$

It is then compared with the spectral value p_m at the shutter frequency m . A signal passes this test if $v/p_m < 0.4$. The spectrum in Fig. 10 satisfies this condition with $v/p_m = 0.032$.

7 Meteors with terminal burst

Some meteors are not detected because one dash in the trail is significantly brighter than the others (see Fig. 11a or c). This is likely caused by the terminal burst of the meteoroid as it passes through the atmosphere.

We developed a specific algorithm to remove these bright dashes. The trail image is divided into three parts: left, central, and right. The borders between them are iteratively shifted from 15 to 30 pixels from the left and right edges. If the brightest dash is more than six times brighter than the second brightest dash and is located within 25 pixels from the left or right edge of the original image, it is removed (see Fig. 11 b and d). Trails processed in this way are classified as “atypical meteors” (*meteorsatyp*).

Many meteors exhibit variations in dash brightness, but when the ratio between the brightest and second brightest dash is less than six, the meteor can be detected using the standard procedure without removing the brightest dash.

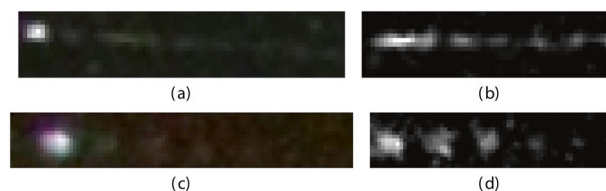


Fig. 11 Examples of meteor trails with one dash much brighter than the others. **a** and **b** show the event from 23 March 2022, 22:43:00, camera 0, Ondřejov observatory; **c** and **d** show the event from 13 August 2024, 00:15:30, camera 1, Svatouch observatory. **a** and **c**: original trails with bright dashes; **b** and **d**: trails after removal of the bright head. All images are normalized in brightness

8 Neural network

A neural network is employed for the final classification of trails, determining whether a given trail is a meteor or another object. Its architecture consists of the following layers (see also Fig. 4b):

- Image input layer
- Four convolutional blocks
- Fully connected layer
- Softmax layer
- Classification layer

8.1 Input images and normalization

The input images of the trails must have a uniform size. We use horizontal images with a length of 600 pixels and a width of 15 pixels. The maximum theoretical trail length in a window is 574 pixels (for a quarter-circle trajectory), so the value 600 is a safe, rounded-up estimate.

Various methods for size normalization were tested: zero-padding, scaling, repetition, mirrored repetition, and partitioning long trails into shorter 100-pixel segments. Finally, the symmetric side-balanced zero-padding performed best: trails are padded with zeros on both sides, keeping the signal centered. We evaluated it as the optimal method. The image input layer accepts gray-level images of this size.

8.2 Convolutional blocks

The convolution block consists of these layers:

- 2D convolution layer
- Batch normalization layer
- Swish layer
- 2D max pooling layer

The 2D convolution layer performs the convolution by 64 filters 3×3 with padding to maintain input size. The batch normalization layer normalizes a mini-batch of data across all observations for each channel independently. After normalization, the layer scales the input with a learnable scale factor α and shifts it by a learnable offset β . The swish activation layer applies the swish function on the layer inputs

$$f(x) = \frac{x}{1 + e^{-x}}. \quad (6)$$

The convolution block is finished by the 2D max pooling layer that downsamples the feature maps by dividing the input into squared pooling regions, then computing the maximum of each region. The factor of the downsampling is two, it means both pool size and stride are set to two. The summary is in Table 1.

8.3 Fully connected, softmax, and classification layers

The fully connected layer transforms data from the input size to the size equaling the number of classes K . It also multiplies the input by a weight matrix and then adds a bias vector. Currently, we use $K = 13$ classes, but it can be changed.

Table 1 Layers of the convolution block and their parameters

Layer	Parameters
2D convolution layer	64 filters 3×3 , padding to the same size
Batch normalization layer	-
Swish layer	-
2D max pooling layer	pooling 2, stride 2
fullyConnectedLayer	13 classes
softmaxLayer	-

Table 2 Training classes, their descriptions, the numbers of training samples (# t. s.) and the numbers of trails classified to the given classes (# trails)

Class name	Description	# t. s.	# trails
<i>clouds</i>	Residual moving clouds, not fully subtracted.	3127	23897194
<i>contrails</i>	Elongated narrow clouds, possibly airplane contrails.	2585	240623
<i>cosmicrays</i>	Violet trails of uncertain origin.	128	6558
<i>ground</i>	Residual effects from ground masks and moving clouds.	1708	2075581
<i>lights</i>	Ground lights exceeding the ground mask.	80	13660
<i>meteors</i>	Normal meteor trails.	1431	1093
<i>meteorsatyp</i>	Atypical meteors with the brightest dash removed.	27	51
<i>noise</i>	Empty sky with only noise.	289	16269343
<i>planes</i>	Long airplane trails.	1884	771158
<i>points</i>	Small groups of points, often parts of airplane trails.	708	2900119
<i>satelblink</i>	Blinking satellites, resembling dashed meteor trails.	339	2383
<i>satels</i>	Satellites or distant planes with continuous trails.	766	49172
<i>stars</i>	Isolated stars or single points.	1032	10298719

The softmax layer normalizes the data $\mathbf{x} = (x_1, \dots, x_K)$ into the interval $\langle 0, 1 \rangle$ by the softmax function

$$\sigma(\mathbf{x})_i = \frac{e^{a_i(\mathbf{x})}}{\sum_{j=1}^K e^{a_j(\mathbf{x})}}, \quad i = 1, \dots, K, \quad (7)$$

where $a_i(\mathbf{x}) = \ln(P(\mathbf{x}, \psi|c_i)P(c_i))$, with $P(\mathbf{x}, \psi|c_i)$ the conditional probability of class i and $P(c_i)$ the class prior.

The classification layer computes the cross-entropy loss. Its output is a vector of “scores”, representing the likelihood that a trail belongs to each class.

8.4 Training set

The training set was originally created manually and improved on the base of the experiments. We have tested various classes. Finally, the following thirteen classes were evaluated as the best. Their list in Table 2 is sorted alphabetically. Classes can be added or merged as needed. Examples from the training set are shown in Fig. 12, with contrast normalized to the full black–white range.

The class structure was designed to reflect the categories encountered during routine operation of the DAFO network. Rather than representing a theoretical taxonomy, the classes correspond to practical distinctions required for reliable automated processing and reduction of false detections.

The training set was built from a subset of trails detected via Radon transformation. Some classes had few samples, others had many. In spite of it, this training set yielded the best results even if it is unbalanced. The 27 samples of the *meteorsatyp* class looks

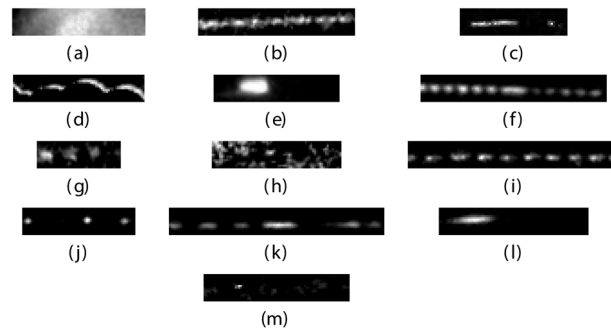


Fig. 12 Examples from the training set: **a** clouds, **b** contrails, **c** cosmicrays, **d** ground, **e** lights, **f** meteors, **g** meteor-satyp, **h** noise, **i** planes, **j** points, **k** satelblink, **l** satels, **m** stars

like too few, but these meteors are rare, so we have shortage of data here. We experimented with unifying of the *meteorsatyp* and *meteors* classes, omitting *meteorsatyp* class and other similar modifications. All other results were worse than that with the 27-sample *meteorsatyp* class.

8.5 Data splitting and evaluation strategy

To avoid potential bias due to repeated observations of the same meteor event, data were grouped at the night level when constructing training and evaluation subsets. This reduces the risk of information leakage between training and evaluation data.

The decision threshold for the neural network output was determined using a validation subset and subsequently fixed for all reported experiments. Due to the limited availability of labeled meteor data, full cross-validation (e.g., leave-one-night-out) was not performed. This limitation may lead to slightly optimistic performance estimates and should be addressed in future work with larger datasets.

8.6 Network training

The neural network was trained using stochastic gradient descent with momentum (SGDM) with a learning rate of 0.01. It minimizes the cross-entropy loss function on the validation data. Training was performed for 600 epochs, with 10% of the data reserved for validation and 10% for testing. Data augmentation was limited to horizontal and vertical flipping and 180°-rotation due to the specific structure of meteor trails. When we have them in narrow long rectangles, we cannot rotate them by an arbitrary angle.

The mini-batch size was not explicitly fixed but determined internally by the training framework based on the dataset size and available computational resources. We observed that the net used 352 batches per epoch, what would mean the effective batch size was approximately 172 augmented samples. Since the optimization procedure remained stable across experiments, the exact batch size was not found to have a significant impact on the final performance.

The validation frequency was once in 30 iterations. We compute all the epochs in every case and then we choose the best result with the minimum of the loss function. Usually, it is a few epochs before the end. The final confusion chart is in Fig. 13.

The blue fields on the matrix diagonal are well classified samples from the test set, while the brown fields outside the diagonal are errors. Here we have 44 errors together from 5642 samples, what means accuracy 99.22%. The substantial errors are between

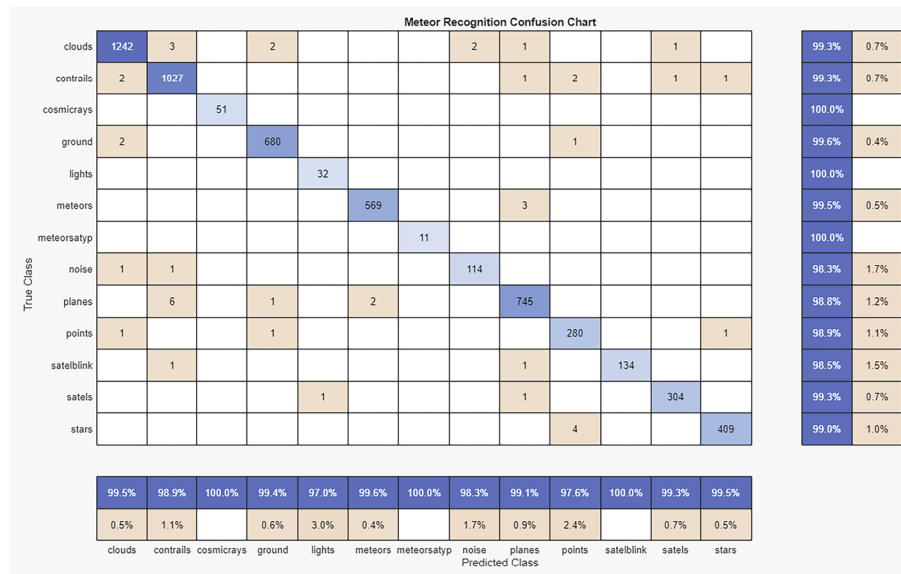


Fig. 13 The confusion chart on the validation data

meteors and non-meteors. Here 3 meteors were recognized as planes and 2 planes were marked as meteors, what is accuracy 99.91%.

8.7 Network output and score evaluation

The network outputs a “score” for each class, ranging from 0 to 1. Although the score approximates a probability, it does not strictly satisfy probability axioms. It is related to the need of high score approaching to the one to be sure it is really a meteor. Finally, we chose a common implicit threshold 0.999. Beside it, we can find an optimal threshold for the specific dataset.

Evaluation proceeds in two steps:

1. A trail is classified as a meteor if the score for *meteors* or *meteorsatyp* is the highest and exceeds a first threshold (implicitly 0.999). Optionally, the trail must also pass the Fourier test.
2. A trail is classified as a possible meteor if the higher of the *meteors* or *meteorsatyp* scores exceeds a second threshold (implicitly 0.99). In this step, the score need not be the highest, and the Fourier test is not required.

9 Results

The proposed method was evaluated on observational data obtained from multiple nights and observatories within the DAFO network. The dataset includes a range of conditions, from nights with clearly visible meteors to more challenging cases containing ambiguous trails such as satellites and airplanes.

We processed data from eleven nights captured at five observatories (see Fig. 14). The selected observation nights were chosen to represent a variety of typical and challenging conditions, including both periods with high meteor activity (e.g., Perseid maximum from 2024) and nights dominated by non-meteor phenomena such as satellite trails (data from 2022). This selection allows for a more comprehensive evaluation of the robustness of the proposed method.

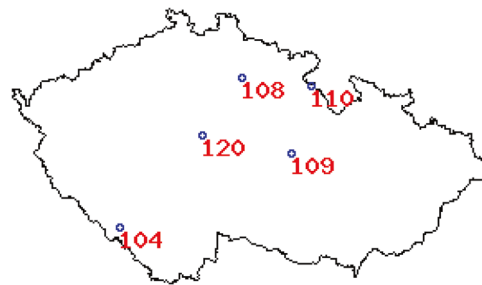


Fig. 14 Observatories from which the data were obtained: 104 Churáňov, 108 Jičín, 109 Svratouch, 110 Polom, and 120 Ondřejov

9.1 Evaluation strategy

A precise quantitative evaluation is inherently difficult due to the absence of a complete ground truth dataset. Therefore, the detected trails were categorized into three groups: *certain meteors*, *uncertain meteors*, and *certain non-meteors*, based on expert assessment.

We have 64 trails classified as certain meteors and 43 uncertain meteor trails. The non-meteor set contains 56 524 510 trails together (the sum of the trail numbers from Table 2), what is approximately 3740 trails per snap. The classes with the low number of samples contain almost all data that we have, only a few misleading examples were deleted. The other classes contain selection of many samples. The data then were divided randomly to training, validation and testing.

Due to overlapping captures (5-second overlap) and multi-observatory visibility, a trail may appear in multiple snaps, that is why the number of meteor trails does not equal to the number of meteors. One meteor trail can be divided among a few windows and a trail can also be found a few times by the Radon transformation. All these trails are counted separately in Table 2. Finally, we unify all detected parts of one meteor in one snap into one meteor trail.

The uncertain meteors are either weak or atypical. The weak meteors are visible only after extreme contrast enhancement ($\approx 10 \times$), which also increases noise. The trail of some short meteors is not divided into dashes, so we do not know, if it is very short meteor or some artificial body. The trails were sorted to the sets after consultation with an expert, so we take the set *certain meteors* as our ground true. The category of *uncertain meteors* highlights the intrinsic ambiguity present in real observational data. In many cases, even expert assessment cannot unambiguously distinguish between weak meteors and artificial objects. The performance of the proposed method on this category therefore reflects not only classification accuracy but also the fundamental limits of the data.

9.2 Network performance

The results for the implicit threshold are in Table 3. The column “night” is the date of the night beginning. The three non-found certain meteors were hidden behind trees, the ground mask erased them and the neural network had no opportunity to test them. The method successfully detects the vast majority of clearly identifiable meteor events in the evaluated dataset. The few missed cases are primarily associated with occlusion by ground objects or extremely weak signals near the detection limit.

Table 3 Meteor search—results for threshold 0.999

Night	Observatory	Found certain meteors	Non-found certain meteors	Found uncertain meteors	Non-found uncertain meteors	False alarms
2014-01-12	120	5	0	0	0	1
2014-12-23	104	6	0	1	0	0
2022-02-15	120	0	0	0	0	0
2022-02-23	120	1	0	0	2	0
2022-03-12	120	2	0	0	0	0
2022-03-28	120	0	0	0	0	1
2022-04-30	120	0	0	0	0	0
2022-07-02	120	2	0	1	0	0
2022-07-11	120	1	0	0	0	0
2022-08-29	120	0	0	0	0	1
2024-08-12	108	16	2	3	10	1
2024-08-12	109	13	0	5	10	0
2024-08-12	110	15	1	1	10	0
Total		61	3	11	32	4

We have also recognized 11 uncertain meteors from 43 as meteors and 32 as non-meteors. Finally, we have 4 false alarms, i.e. non-meteor trails found like meteors. From the table, the precision $61/(61+4) = 93.85\%$ and recall $61/(61+3) = 95.31\%$ can be calculated. F1-score is then 94.57% ¹. We omitted the uncertain meteors.

The achieved precision and recall indicate that the method provides a favorable balance between sensitivity and robustness. In particular, the high recall suggests that only a small fraction of clearly detectable meteors is missed, while the high precision indicates a low rate of false detections.

It is a question, if we should count the meteors behind trees as false negatives, when we intentionally erased the ground objects including potential meteors behind trees. If not, the recall would be 100% and the F1-score would increase to 96.83%.

The optimal threshold for a given trail set is that enables finding all certain meteors and minimum false positives. In our case the minimum score among detected certain meteors was 0.999439. Coincidentally, an uncertain meteor had a score of 0.999379, therefore we consider the threshold 0.9993 to be optimal for the given dataset. With this threshold, we are able to reduce the number of false positives to one, see Fig. 18b.

Fig. 15 shows the number of false positives and false negatives as a function of the score threshold from 0.9 to 1. The red curve includes uncertain meteors. The optimal threshold 0.9993 on the red curve is near the right margin, where we can find all detectable certain meteors and minimize the number of the false alarms.

The decision threshold was selected to balance the trade-off between false detections and missed events. While the optimal threshold may vary depending on the specific dataset and observational conditions, the chosen value provides stable performance across the evaluated nights.

False detections are primarily associated with specific challenging cases, such as airplanes partially obscured by clouds or blinking satellite trails. These cases exhibit visual characteristics similar to meteors, making them difficult to distinguish even for human

¹Precision = $TP/(TP+FP)$, Recall = $TP/(TP+FN)$, F1-Score = $2 \times \text{Precision} \times \text{Recall}/(\text{Precision} + \text{Recall})$, TP = True Positives, FP = False Positives, FN = False Negatives.

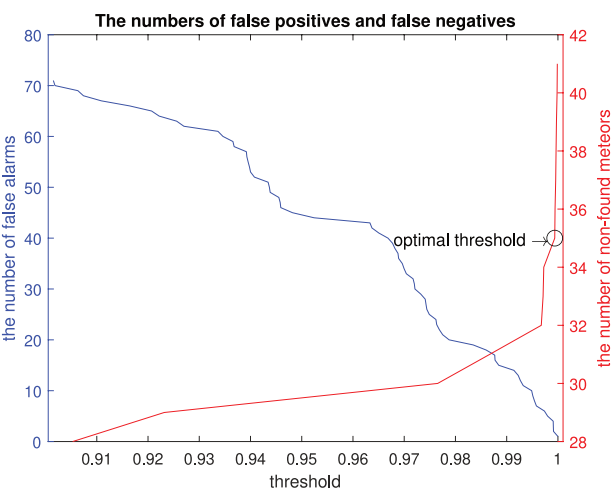


Fig. 15 The number of false positives (false alarms on the left blue axis) and false negatives (non-found meteors on the right red axis), including uncertain meteors, versus score threshold

Table 4 Meteor search—results for threshold 0.999 without the Fourier test

Night	Observatory	Found certain meteors	Non-found certain meteors	Found uncertain meteors	Non-found uncertain meteors	False alarms
2014-01-12	120	5	0	0	0	3
2014-12-23	104	6	0	0	1	1
2022-02-15	120	0	0	0	0	1
2022-02-23	120	1	0	0	2	2
2022-03-12	120	2	0	0	0	1
2022-03-28	120	0	0	0	0	0
2022-04-30	120	0	0	0	0	0
2022-07-02	120	2	0	1	0	0
2022-07-11	120	1	0	0	0	0
2022-08-29	120	0	0	0	0	0
2024-08-12	108	16	2	3	10	1
2024-08-12	109	13	0	6	9	0
2024-08-12	110	15	1	1	10	0
Total		61	3	11	32	9

observers. The low number of such false detections demonstrates the robustness of the proposed classification approach.

9.3 Results without Fourier test

In this section, we compare results with and without the Foutier test. While the over-all detection rate remains similar, the inclusion of the Fourier test significantly reduces the number of false positives, highlighting its importance in the overall system. We can compare the results without the Fourier test in Table 4 with that from Table 3.

The certain meteors were found all, similarly as with the Fourier test. Also, the uncertain meteors were evaluated by the similar way. There are only two differences: in the night from 23rd to 24th December 2014, observatory Churáňov, one uncertain meteor was recognized with the Fourier test as meteor and without the test as non-meteor. In the night from 12th to 13th August 2024, observatory Svratouch, one uncertain meteor



Fig. 16 The blinking satellite trails from the Ondřejov observatory, 12th March 2022; **a** 18:17:00, camera 0 **b** 18:17:30, camera 1

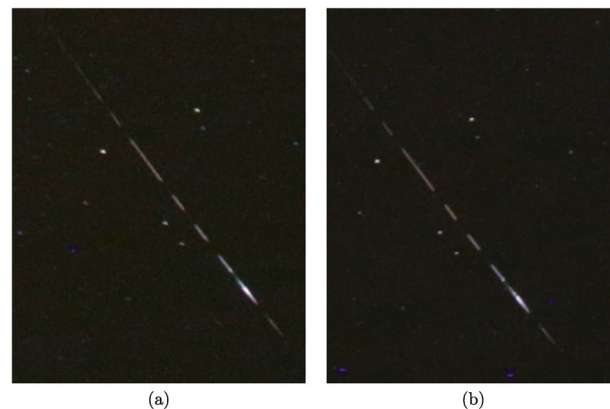


Fig. 17 The meteor trails from the Jičín observatory, 13th August 2024; **a** 00:19:00, camera 0 **b** 00:19:30, camera 1

was recognized with the Fourier test as non-meteor and without the test as meteor. The total numbers stayed the same.

The significant difference is in the false alarms, we have four ones with the Fourier test versus nine different false alarms without the Fourier test, it means the Fourier test decreased the number of false alarms from 9 to 4. That is why, we decided to preserve the Fourier test in the algorithm.

9.4 Sources of errors

In the meteor search, we must face up a number of problems. One of them are blinking satellites, see the trails in Fig. 16 captured by both cameras of one observatory during the five-second overlap of capturing. They look exactly as meteors, including one longer dash, but if we compare both images (a) and (b), the trails are shifted (compare the star above the trails), it means the body is slow, so it cannot be meteor. We recognized it as some type of a new blinking satellite. We can compare it with Fig. 17, where the trails are not shifted, the body is much faster, so it is the meteor. The dashes are long, so it was fast even among the meteors.

Another source of problems are airplanes. They have blinking navigation lights that can sometimes look like meteors, see Fig. 18a. If we can see only a part of the trail, we could hesitate, if it is not a meteor, but when we can see the whole trail and also its continuations in the previous and next snaps, it is clear that it is an airplane. Fig. 18b shows our only false alarm. It is an airplane behind a cloud, but the similarity with the meteor is surprising.

The problems are also with the very weak meteors, see example in Fig. 19. They are visible only after strong contrast increase. We are not sure, if they are meteors or not.



Fig. 18 Examples of the found trails: **a** Ondřejov 15th February 2022, 18:16:35, camera 0, it is certainly an airplane; **b** Ondřejov 28th March 2022, 19:39:00, camera 0, it is the airplane behind the clouds, our only false alarm. A plane behind a cloud really resemble a meteor. In the top part of the figure, there are crops from the original images, in the bottom part, there are the trails found by the Radon transformation and tested by the neural network

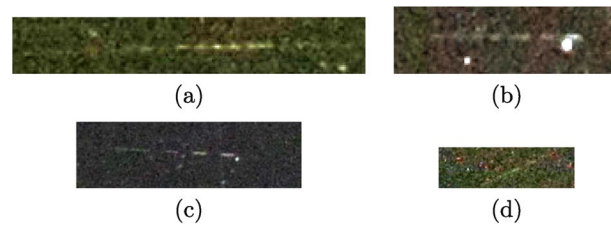


Fig. 19 Examples of the found trails: **a** Churáňov 24th December 2014, 02:39:55, camera 1, **b** Jičín 13th August 2024, 00:47:30, camera 1, **c** Jičín 13th August 2024, 01:53:30, camera 1, **d** Svatouch 12th August 2024, 23:49:00, camera 0. All the snaps have increased contrast about eight times, the trails are hardly visible in the original snaps

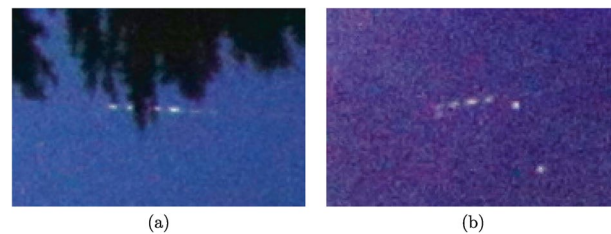


Fig. 20 The meteor from 13th August 2024, 02:10:30, cameras 1; **a** the Polom observatory, **b** the Jičín observatory

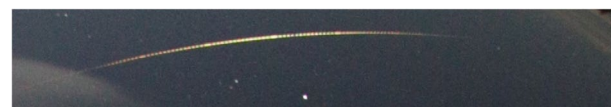


Fig. 21 Our longest meteor. It is from Ondřejov observatory, 13th January 2014, 03:01:30, camera 0

Finally, we consider the trails in Fig. 19a and b to be probably distant airplanes, while the trails in Fig. 19c and d to be probably meteors. The trails from Fig. 19a and c were evaluated by our algorithm as meteors, while that from Fig. 19b and d as non-meteors.

Some meteors are very low above the horizon, see e.g. Fig. 20a. The meteor is partly hidden behind a tree, thus it was erased by the ground mask. Fortunately, this meteor has been found, because it is fully visible on a snap from another observatory, see Fig. 20b

9.5 Interesting meteors

In this section, we would like to show a few interesting meteors. In Fig. 21, there is our longest meteor. The number of longer dashes indicates that the meteor's passage had

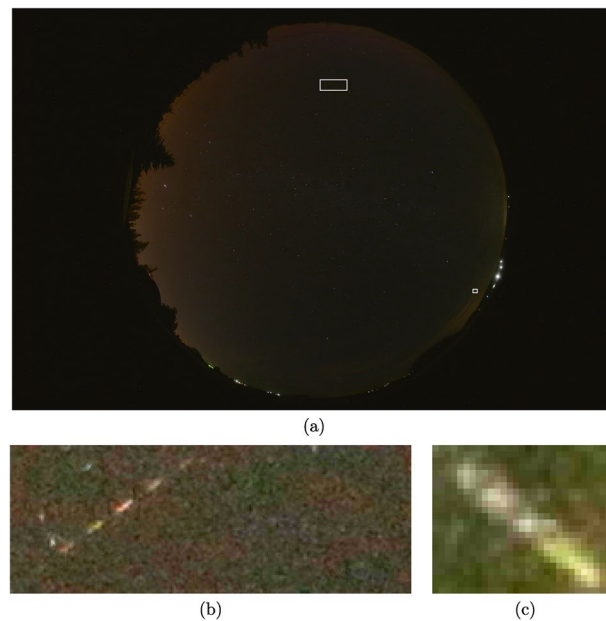


Fig. 22 Example of the Jičín observatory data, snap from 13th August 2024, 01:03:00, camera 0, **a** whole snap with two meteors in the white bounding boxes, **b** and **c** the details of the meteors with increased contrast

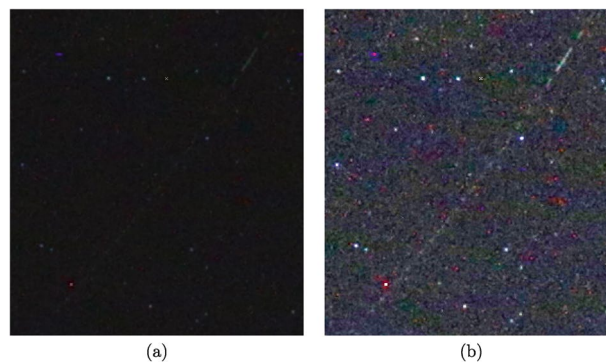


Fig. 23 The uncertain meteor from 12th August 2024, 20:22:00, camera 0, Svatouch observatory found thanks to our algorithm; **a** the original image, **b** the image with increased contrast 5.1 times

to last more than four seconds. Another interesting snap is in Fig. 22a. It contains two meteors, see details in Fig. 22b and c. Finally, there is an example of a very weak meteor, see Fig. 23. We overlooked it in “manual” search, it has been found only thanks to our algorithm. Nevertheless, we have not hundred-percent certainty that it is the meteor, therefore we categorized it as an uncertain meteor. The problem is its weakness, it is fully visible only after the strong contrast increase.

9.6 Time of computation

While the computing complexity of our algorithm is approximately linear $\mathcal{O}(n)$, where n is the number of processed snaps, the time of computation has relatively slow constant. The time also very depends on the computer. We tested it on the computer equipped by the processor Intel(R) Core(TM) i9-7940X CPU with 3.10 GHz, 32 GB memory and 64-bit Windows 11 Pro in Matlab language, version 2022a. During the review process, we moved from version 2022a to version 2025a, but the times in this section are from

Table 5 Meteor search—computation time

Batch	Camera	Collection time	Processing time
1	0	5 d 4 h 53 min	2 d, 8 h, 39 min 55 s
1	1	5 d 4 h 53 min	2 d, 14 h, 38 min 27 s
2	0	5 d 6 h 59 min	2 d, 5 h, 19 min 3 s
2	1	5 d 6 h 59 min	2 d, 6 h, 28 min 12 s

**Fig. 24** The meteor from Churáňov observatory, 23rd December 2014, 18:04:55, camera 1

the version 2022a. First, we trained the neural network, it took 17 h, 10 min and 23 s (400 epochs). Then, we computed our algorithm on all the data. The obtained times in days, hours, minutes and seconds are in Table 5.

We divided the computation to two batches, the batch 1 contains the nights from 12th January 2014, 23th December 2014, 15th February 2022, 23th February 2022, and 12th March 2022, the batch 2 contains the nights from 28th March 2022, 30th April 2022, 2nd July 2022, 11th July 2022, 29th August 2022, and 12th August 2024 (three observatories). Each batch is further divided to the snaps from camera 0 and those from camera 1. The summer nights are shorter, therefore the summer batches contain more nights.

The used computer have 14 processor cores, so we could launch the four batches parallelly. The times in Table 5 should not be added, the decisive is the longest one. The average time is about 54 s per snap. If we could process the data immediately after their capturing, we would be able to process them that day. For the sake of completeness, both cameras also collect the data parallelly, we can add the times for the batches 1 and 2, but not for the individual cameras 0 and 1.

The computation time depends on the specific image, a rich snap with many trails is processed longer than an almost empty snap with only few trails. We tested the time distribution on the snap from 23rd December 2014, 18:04:55, Churáňov observatory with the meteor in Fig. 24.

The time distribution of the entire algorithm is in Table 6. The maximum filter is part of the algorithm for the subtraction of the previous and next images, the parametrization of the trail is done so the adjacent pixels have then distances 1. The neural network test can be analyzed separately, how it is in Table 7. The times are balanced, so it is not easy to search another space for additional acceleration of the algorithm.

Table 6 The time distribution of the individual operations

Operation	Time
Neural network test	15.899 s
Rotation with the fish-eye correction	7.093 s
Radon transformation	6.354 s
Record of the results	5.933 s
Maximum filter	4.494 s
Interpolation	1.504 s
Reading the snaps	0.981 s
Removing the bright dashes	0.776 s
Parametrization of the trail	0.367 s
Conversion of the snaps to the gray levels	0.254 s
Rest of the algorithm	4.912 s
Total	48.567 s

Table 7 The time distribution of the individual components of the neural network test

Neural network test	
Suboperation	Time
Batch normalization	3.808 s
Swish function	3.433 s
Convolution	3.346 s
Max pooling	1.207 s
Fully connected layer	0.910 s
Classification layer	0.628 s
Input layer	0.507 s
Rest of the network	2.05 s
Total	15.899 s

10 Discussion and conclusion

The results demonstrate that the proposed hybrid approach provides a robust and practical solution for automated meteor detection in all-sky images. By combining classical image processing techniques with convolutional neural network classification, the method effectively balances computational efficiency with classification accuracy.

A key strength of the proposed pipeline is its ability to operate reliably under real observational conditions. The dataset used in this study includes a wide range of challenging scenarios, such as varying background illumination, cloud coverage, and the presence of artificial objects. The achieved performance indicates that the method is capable of handling these conditions without requiring extensive manual intervention.

The use of a multi-class classification framework allows the system to distinguish meteors from several types of non-meteor trails, including satellites and airplanes. This is particularly important in modern observation networks, where artificial objects represent a significant source of false detections. The results show that most false positives are associated with visually ambiguous cases that are difficult to classify even for human observers.

The inclusion of the Fourier-based filtering step plays an important role in reducing false detections. Although the overall detection rate remains similar without this component, the number of false positives increases, confirming that the combination of complementary methods improves overall system performance.

The category of uncertain meteors highlights the inherent limitations of the available data. In many cases, even expert assessment cannot unambiguously determine the

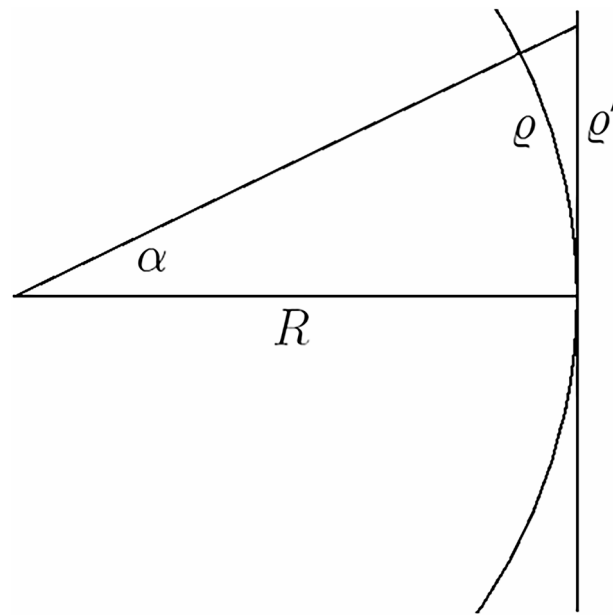


Fig. A1 Simplified model of the fish-eye camera

nature of weak or atypical trails. The performance of the proposed method in this category should therefore be interpreted as an indication of the intrinsic ambiguity of the problem rather than purely as a classification error.

At the same time, several limitations of the proposed approach must be acknowledged. The training dataset is relatively small and imbalanced, particularly for rare classes such as atypical meteors. This may affect classification reliability for these cases. Furthermore, the method relies on manually defined ground masks, which require maintenance when observational conditions change. Finally, the evaluation does not include full cross-validation, and the reported performance may therefore be slightly optimistic.

Despite these limitations, the results suggest that the proposed method is suitable for practical deployment in meteor observation networks. The computational complexity is manageable, and the system can significantly reduce the need for manual inspection of large image datasets.

Future work will focus on expanding the training dataset, improving the representation of rare classes, and exploring more advanced validation strategies. Another promising direction is the integration of temporal information from consecutive frames, which could further improve the discrimination between meteors and artificial objects.

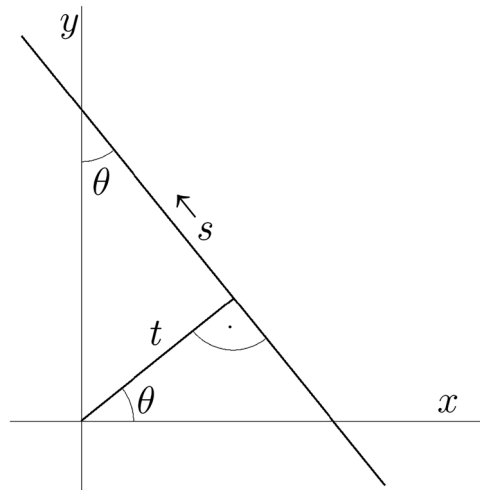


Fig. B1 Radon transformation

Appendix A Fish-eye camera model

We can model the fish-eye camera as projection on spherical surface, see Fig. A1.

A point that would be projected to the distance ϱ' from the optical axis of the standard pin-hole camera, is now projected to the distance ϱ in the fish-eye camera. We can easily derive $\varrho = R\alpha$ (α in radians) and $\varrho' = R \tan \alpha$, from that

$$\varrho' = R \tan\left(\frac{\varrho}{R}\right) \quad (\text{A1})$$

for conversion from fish-eye to pin-hole and

$$\varrho = R \arctan\left(\frac{\varrho'}{R}\right) \quad (\text{A2})$$

for conversion from a pin-hole to a fish-eye cameras. The sphere radius R was experimentally estimated as 0.3 of the smaller image size 3648, i.e. 1094.4 pixels.

The entire conversion procedure begins with conversion from the rectangular $\{x, y\}$ to the polar $\{\vartheta, \varrho\}$ coordinates

$$\begin{aligned} \vartheta &= \arctan\left(\frac{y - y_c}{x - x_c}\right) \\ \varrho &= \sqrt{(x - x_c)^2 + (y - y_c)^2}, \end{aligned} \quad (\text{A3})$$

then ϱ is converted to ϱ' by (A1) and back to Cartesian coordinates

$$\begin{aligned} x' &= x_c + \varrho' \cos \vartheta \\ y' &= y_c + \varrho' \sin \vartheta. \end{aligned} \quad (\text{A4})$$

The reverse conversion is analogous (we use (A2) instead of (A1)) assuming the optical center $\{x_c, y_c\}$ is at the real center of the image, i.e. $x_c=2735.5$, $y_c=1823.5$ in zero-based coordinates.

Appendix B Radon transformation

Radon transformation searches straight line segments in an image. The searched straight line is defined by the distance t from the coordinate origin and by the angle θ , see Fig. B1. The parameter s is the parameter of the parametrically defined straight line.

During computation, we use equations

$$\begin{aligned} s &= y \cos \theta - x \sin \theta \\ t &= x \cos \theta + y \sin \theta, \end{aligned} \quad (\text{B1})$$

from that

$$\begin{aligned} x &= t \cos \theta - s \sin \theta \\ y &= s \cos \theta + t \sin \theta. \end{aligned} \quad (\text{B2})$$

The Radon transformation of an image $f(x, y)$ is then done by integration

$$R(\theta, t) = \int_{s_1}^{s_2} f(t \cos \theta - s \sin \theta, s \cos \theta + t \sin \theta) ds. \quad (\text{B3})$$

We integrate over s and t is parameter of the result. The limits s_1, s_2 are given so we integrate over the whole image. When we find a local maximum in $R(\theta, t)$, the corresponding straight line is given by Eqs. (B2), where (θ, t) is position of the maximum and s is the parameter of the straight line.

If we have rectangular image or window, the minimums and maximums of s and t are in its vertices, therefore we search s_1, s_2 (limits of the integration) and also t_1, t_2 (limits of the output array) as limits of $R(\theta, t)$ in the vertices for all used values of θ .

Acknowledgements

We thank Pavel Spurný (Observatory Ondřejov) for kindly providing the DAFO Network data for experiments and Lenka Kotková for preparing the data sets from the archive.

Author contributions

S. Š. provided the data and professional astronomical expertise. T. S. wrote the software and performed the experiments.

Funding

The research was supported by the Czech Science Foundation (GAČR) under the grant No. GA24-10069S and The Czech Academy of Sciences (AV ČR) under the project No. RVO:67985815.

Data availability

The data that support the findings of this study are available from the Astronomical Institute but restrictions apply to the availability of these data, which were used under licence for the current study, and so are not publicly available. Data are however available from the authors upon reasonable request and with permission of the Astronomical Institute. The journal also requires that authors cite any publicly available data on which the conclusions of the paper rely. Data citations should include a persistent identifier (such as a DOI), should be included in the reference list using the minimum information recommended by DataCite, and follow journal style. Dataset identifiers including DOIs should be expressed as full URLs.

Code availability

Not applicable.

Declarations

Ethics approval

This study did not involve human participants, data, or animals; therefore, ethical approval was not required.

Consent to participate

Informed consent was obtained from all individual participants included in the study.

Consent for publication

The participants have given relentless consent to publish their individual data and both direct and indirect identifiers in this article.

Competing interests

The authors declare no competing interests.

Received: 17 June 2026 / Accepted: 23 September 2026

Published online: 09 October 2026

References

1. Suk T, Šimberová S. Automated meteor detection by all-sky digital camera systems. *Earth Moon Planet*. 2017;120:189–215. <https://doi.org/10.1007/s11038-017-9511-z>
2. Joiret S, Koschny D. On the origin of the Southern Delta Aquariids meteor shower. *Planet Space Sci*. 2023;232:105700. <https://doi.org/10.1016/j.pss.2023.105700>
3. Spurný P, Borovička J, Shrbený L. The [Z]’dárnad Sázavou meteorite fall: Fireball trajectory photometry dynamics. *Meteorit Planet Sci*. 2020;55(2):376–401. <https://doi.org/10.1111/maps.13444>
4. Guo X, Jiang Y, Liu Y. Ephemeris accuracy improvement for moons of gas giants: a deep learning based methods. *Discover Space*. 2024;128(1):1–14. <https://doi.org/10.1007/s11038-024-09554-8>
5. Roman VŞ, Bui C. Automatic analysis of radio meteor events using neural networks. *Earth Moon Planet*. 2015;116(2):101–113. <https://doi.org/10.1007/s11038-015-9473-y>
6. Molau, S. The meteor detection software MetRec. In: *Meteoroids 1998. Proceedings of the International Conference*, pp. 131–134. Tatranská Lomnica, Slovakia: Astronomical Institute of the Slovak Academy of Sciences; 1999, held in August 17–21, 1998.
7. Molau, S. MetRec Homepage. <https://www.metrec.org/> (2017). Accessed June 29, 2020.
8. SonotaCo: Software/Download. http://sonotaco.com/soft/e_index.html (2020). Accessed June 29, 2020.
9. Gural P, Otto P, Tedesco E. Moving object detection using a parallax shift vector algorithm. *Publ Astron Soc Pac*. 2018;130:074504–115. <https://doi.org/10.1088/1538-3873/aac1ff>
10. Gural P. Advances in the meteor image processing chain using fast algorithms, deep learning, and empirical fitting. *Planet Space Sci*. 2020;182:104847–18. <https://doi.org/10.1016/j.pss.2020.104847>
11. McGarvey K, Vilalta R, Samara M, Michell R. A pattern recognition system for the automated tracking and classification of meteors using digital image data. In: *Astronomical Data Analysis Software and Systems XX. ASP Conference Series*, vol. 442, San Francisco: Astronomical Society of the Pacific; 2011. pp. 349–352.
12. Gural PS. The California all-sky meteor surveillance (CAMS) system. In: *International Meteor Conference, IMC; 2010*, pp. 28–31.
13. Jenniskens P, Gural PS, Dynneson L, Grigsby BJ, Newman KE, Borden M, Koop M, Holman D. CAMS: Cameras for Allsky Meteor Surveillance to establish minor meteor showers. *Icarus*. 2011;216(1):40–61. <https://doi.org/10.1016/j.icarus.2011.08.012>
14. Jenniskens P, Nénon Q, Gural PS, Albers J, Haberman B, Johnson B, Holman D, Morales R, Grigsby BJ, Samuels D, Johannink C. CAMS confirmation of previously reported meteor showers. *Icarus*. 2016;266:355–370. <https://doi.org/10.1016/j.icarus.2015.08.014>
15. Gural P, Šegon D. A new meteor detection processing approach for observations collected by the Croatian Meteor Network (CMN). *WGN, the Journal of the IMO*. 2009;37(1):28–32.
16. Sennlaub R, Hofmann M, Hankey M, Ennes M, Müller T, Kroll P, Mäder P. Object classification on video data of meteors and meteor-like phenomena: Algorithm and data. *Mon Not R Astron Soc*. 2022;516:811–823. <https://doi.org/10.6084/m9.figshare.16451625>
17. Anghel S, Nedelcu DA, Birlan M, Boaca I. Single-station meteor detection filtering using machine learning on MOROI data. *Mon Not R Astron Soc*. 2023;518:2810–2824. <https://doi.org/10.1093/mnras/stac3229>
18. Calders S, Lamy H. Automatic detection of meteors in the BRAMS data. In: *Proceedings of the International Meteor Conference IMC’14; 2014*, pp. 1–3.
19. Mallama A, Espenak F. Automated meteor detection and the Leonid shower. *Publ Astron Soc Pac*. 1999;111:359–363.
20. Grigg D, Tingay SJ, Sokolowski M, Wayth RB. DUG insight: A software package for big-data analysis and visualisation, and its demonstration for passive radar space situational awareness using radio telescopes. *Astron Comput*. 2022;40:100619. <https://doi.org/10.1016/j.jascom.2022.100619>
21. Vankawala PM, Marshall RA, Vida D, Brown P. An observational investigation of very low frequency radio emissions originating from meteors. *Planet Space Sci*. 2024;245:105892. <https://doi.org/10.1016/j.pss.2024.105892>
22. Li Y, Galindo F, Urbina J, Zhou Q, Huang T-Y. Meteor detection with a new computer vision approach. *Radio Science*. 2022;57(10):1–15. <https://doi.org/10.1029/2022RS007515>
23. Trayner C, Haynes BR, Bailey NJ. Automatic meteor detection: an application of Hough transforms. In: *International Conference on Image Processing, ICIP’96*, vol. 2. New York: IEEE; 1996. pp. 693–696.
24. Cheselka M. Automatic detection of linear features in astronomical images. In: *Astronomical Data Analysis Software and Systems VIII. ASP Conference Series*, vol. 172, San Francisco: Astronomical Society of the Pacific; 1999. pp. 349–352.
25. Reffet B, Vaubaillon J, Colas F. A new meteor detection algorithm for shuttered photography. In: *Proceedings of the International Meteor Conference IMC’14; 2014*, pp. 1–3.
26. Šimberová S, Suk T. Automated detection of meteors in observed image sequence. In: *International Conference of Computational Methods in Sciences and Engineering, ICCMSE’15*, vol. 1702, Melville, USA: AIP; 2015. pp. 190009–11900094.
27. Roman VŞ, Bui C. Automatic detection of meteors using artificial neural networks. In: *Proceedings of the International Meteor Conference IMC’14; 2014*, pp. 122–125.
28. Gural PS. Deep learning algorithms applied to the classification of video meteor detections. *Mon Not R Astron Soc*. 2019;489(4):5109–5118. <https://doi.org/10.1093/mnras/stz2456>
29. Cecil D, Campbell-Brown M. The application of convolutional neural networks to the automation of a meteor detection pipeline. *Planet Space Sci*. 2020;186:104920–112. <https://doi.org/10.1016/j.pss.2020.104920>
30. Ciocan C, Kandeepan M, Cassagne A, Vaubaillon J, Zander F, Lacassagne L. A new meteor detection application robust to camera movements. in French (2023). <https://arxiv.org/abs/2309.06027>

31. Shirasuna VY, Gradwohl ALS. An optimized training approach for meteor detection with an attention mechanism to improve robustness on limited data. *Astron Comput.* 2023;45:100753 <https://doi.org/10.1016/j.ascom.2023.100753>
32. Peña-Asensio E, Trigo-Rodríguez JM, Grèbol-Tomás P, Regordosa-Avellana D, Rimola A. Deep machine learning for meteor monitoring: Advances with transfer learning and gradient-weighted class activation mapping. *Planet Space Sci.* 2023;238:105802 <https://doi.org/10.1016/j.pss.2023.105802>
33. Spurný P, Borovička J, Šrbený L. Automation of the Czech part of the European fireball network: equipment, methods and first results. In: *Proceedings of IAU Symposium*, vol. 236. Cambridge, UK: Cambridge University Press; 2007. pp. 121–130.
34. Cohen M, Lu W. A diffusion-based method for removing background stars from astronomical images. *Astron Comput.* 2021;37:100507 <https://doi.org/10.1016/j.ascom.2021.100507>
35. Fitzgibbon AW, Pilu M, Fisher RB. Direct least squares fitting of ellipses. In: *Proceedings of the 13th International Conference on Pattern Recognition, ICPR'96*, vol. 1, secretariat@iapr.org: IAPR; 1996. pp. 253–7.

Publisher's note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.