DETECTION AND CHARACTERIZATION OF SPACE DEBRIS USING VST/OMEGACAM ARCHIVAL DATA

E. Rachith, S. Hellmich, and J.-P. Kneib

École Polytechnique Fédérale de Lausanne, Laboratory of Astrophysics (LASTRO), 1015 Lausanne, Switzerland, Email: $\{$*elisabeth.rachith, stephan.hellmich, jean-paul.kneib*$\}$*@epfl.ch*

ABSTRACT

As space activity grows exponentially, the orbital environment is becoming increasingly congested with active satellites, inactive spacecraft, and debris of varied sizes and shapes. It has become evident that, in addition to precise monitoring of the debris population, active space debris removal is essential to ensure long-term access to Earth's orbits, requiring detailed knowledge of each object's orbit, shape, and rotational state. Despite the significant efforts by space surveillance networks to monitor and track a growing number of objects, the available data often consists only of positional information, leaving critical gaps in understanding the physical and rotational characteristics of defunct satellites and space debris. To bridge this knowledge gap, we are leveraging data from the VST/OmegaCAM archive, a unique dataset of over 400,000 high-precision observations spanning 12 years. While VST was designed for deep-sky surveys, space debris cross its field of view, leaving detectable traces in the images. The instrument's exceptional sensitivity allows us to detect objects as small as 5 cm in low-Earth orbit (LEO) and 30 cm in geostationary orbit (GEO). Processing this extensive archive requires advanced image analysis, for which we have developed a novel streak-detection method that combines a convolutional neural network with a Hough transform layer. We present our training dataset, algorithm design, and streak-detection workflow and discuss the detections realised with our algorithm on a set of OmegaCAM images. We discuss the performance of the algorithm in terms of completeness, sensitivity and precision. After detection, each streak is correlated with the catalogue of known orbital objects, and photometric reduction is applied to extract light curves, providing insights into the object's attitude and shape. The intensity profile of a streak is retrieved by placing a series of rectangular apertures along the streak and measuring the flux received from the target in each aperture. The retrieved flux is calibrated against photometric reference stars from GAIA DR2 with magnitude conversion and error rejection processes. This approach yields high-precision absolute magnitudes with errors of a few hundredths of a magnitude, enabling detailed characterization of non-resolved objects. Fourier analysis is applied to retrieve, when relevant, rotation rates of objects, providing valuable information about the operational status of the detected objects and potential retrieval possibilities. All data processed through our detection and reduction pipeline will be made available in an open-access data repository, offering the community a valuable resource for further investigation into the orbital debris population. Beyond supporting active debris removal efforts and enhancing knowledge about individual debris objects, this work will also enable deeper insights into key aspects such as the increasing interference with astronomical observations and a more accurate understanding of the size-frequency distribution of orbital debris.

Keywords: Space debris; Passive optical; Machine learning.

1. INTRODUCTION

Despite decades of space debris observations and significant advances in recent years, there remains a significant lack of understanding of many aspects of these objects, in particular their evolution over time and their interactions with the environment. This knowledge gap is a growing challenge as the accumulation of space debris in Earth's orbit continues to escalate, threatening active satellites, space missions and human safety. While considerable efforts have been made to monitor these objects, much remains unknown about their physical properties and long-term behavior. Addressing these uncertainties is critical to ensuring the sustainability of orbital activities and any additional data that can be obtained is therefore invaluable.

One underexplored avenue for gathering such data lies in wide-field astronomical surveys. These surveys, primarily designed for astrophysical research, often capture serendipitous observations of space debris as streaks in their images. While these streaks are typically considered nuisances by astronomers, they contain valuable information about space debris. However, analyzing archival astronomical data for space debris presents unique challenges: the debris signatures differ from targeted observations, and the volume of archival data spanning years is immense.

Proc. 9th European Conference on Space Debris, Bonn, Germany, 1–4 April 2025, published by the ESA Space Debris Office
Editors: S. Lemmens, T. Flohrer & F. Schmitz, (http://conference.sdo.esoc.esa.int, April 2025)

This study explores how to efficiently detect space debris streaks in wide-field astronomical images and extract valuable information from them. While targeted observations using dedicated instruments are highly effective and have significantly advanced our understanding of space debris, they are often costly, resource-intensive, and limited by acquisition time and observational constraints. In contrast, our approach leverages existing open-source data, such as that provided by the European Southern Observatory (ESO), to complement these efforts. Wide-field telescopes offer unique advantages due to their high sensitivity and resolution, enabling the extraction of critical information not commonly found in conventional debris catalogs. For instance, rotational status and rotation rates, which are essential parameters for planning active debris removal missions, can be derived from light curves obtained through this approach.

Although this study focuses on data from the VLT Survey Telescope (VST) and OmegaCAM imager, our broader aim is to create a framework adaptable to other astronomical surveys. This approach not only enhances SSA efforts but also demonstrates the potential for repurposing existing data to address critical challenges in space sustainability.

Traditional methods for detecting streaks in astronomical images include techniques such as the Hough transform and matched filtering. The Hough transform is particularly effective for identifying linear features, especially when preprocessing steps like background subtraction and edge detection are employed to isolate streaks from stars and other artifacts. Matched filtering, which aligns predefined streak templates with image data, has also been utilized to detect faint trails with high sensitivity [6][14].

More recently, machine learning and deep learning approaches have gained traction due to their ability to handle large datasets and complex patterns. Convolutional Neural Networks (CNNs) and architectures like U-Net have been widely applied, often outperforming traditional methods in terms of accuracy and robustness. For example, Liu et al. [12] demonstrated the potential of machine learning to identify satellite streaks in wide-field astronomical images, leveraging pre-trained models fine-tuned on domain-specific datasets to enhance detection sensitivity. Similarly, Jeffries and Acuña [8] developed a machine learning framework that integrates feature extraction and classification techniques to detect faint streaks while minimizing false positives. Additionally, ASTA (Automated Satellite Tracking for Astronomy) [16] combines deep learning with computer vision techniques for effective satellite trail detection.

Despite significant progress, challenges remain in reducing false positives, improving sensitivity to faint streaks, and processing the increasing volume of astronomical data efficiently. The integration of traditional algorithms with modern AI techniques offers a promising path forward in addressing these issues.

The remainder of this paper is organized as follows: Section 2 details the methods employed in this study, including the detection algorithm, dataset selection, artificial streak generation, and subsequent steps such as photometric reduction, astrometric calibration, and orbit fitting. The same section introduces the performance metrics used to evaluate the proposed approach, including precision, recall, and F1-score. Section 3 presents the results of the study, covering the learning rate search, performance outcomes, and photometric reduction findings. Finally, Section 4 provides a discussion of the implications of the results

2. METHODS

2.1. Detection algorithm

The algorithm developed for detecting satellite and space debris traces is a machine learning model based on the HT-LCNN (Hough Transform Lookup-based Convolutional Neural Network), a convolutional neural network architecture incorporating a trainable Hough transform prior block introduced by Lin et al [11]. The integration of the Hough transform module into the model is particularly valuable, as it equips the network with an inherent understanding of the geometric shapes to detect. The Hough transform, a well-established technique for detecting lines in images, has been extensively applied in the detection of satellite streaks, making it particularly relevant to this application.

Lin et al.'s original HT-LCNN algorithm was trained and evaluated using the Wireframe [7] and York Urban [2] datasets, which were designed for edge detection in architectural and interior imagery. The model was initially intended for tasks such as automatic orientation in smart devices. To adapt this approach for detecting satellite and space debris traces, modifications were made to handle astronomical images saved in the 16-bit FITS format, as opposed to the more common 8-bit format used in general machine learning applications. This adjustment was necessary because astronomical images typically contain higher bit-depth information to preserve subtle details critical for analysis.

2.2. Dataset selection

The preparation of a robust and diverse dataset was a critical step in training and validating our machine learning algorithm for detecting satellite and space debris traces. OmegaCAM [9], mounted on the VLT Survey Telescope (VST), was chosen for several compelling reasons.

- *Open Access and Extensive Archive:* OmegaCAM provides public access to its data, with archives spanning over a decade (since 2011), enabling the study of space debris evolution over time.

- *High Sensitivity:* Its exceptional sensitivity, enhanced by VST's 2.6-meter primary mirror, allows the detection of faint debris traces that might otherwise be missed.
- *High Resolution:* The high resolution of OmegaCAM, with its 16k×16k pixel array and a pixel scale of 0.21 arcseconds per pixel, facilitates accurate light curve reconstruction, crucial for characterizing debris properties.
- *Wide Field of View:* Covering one square degree per image, OmegaCAM captures a large number of debris traces in a single exposure, enhancing statistical reliability. A sample image can be seen in Figure 1.

Images were selected from the OmegaCAM archive spanning its entire operational timeline. Due to the one-year proprietary period at the time of selection, images from 2023 and later were excluded. To ensure versatility, images from all available filters were included, allowing the algorithm to detect streaks on raw images without requiring photometric adjustments despite filter differences. A random selection of 375 images ensured diversity in backgrounds, moon illumination levels, streak types, star density, and artifacts.

Given the large size of OmegaCAM mosaics (32 CCDs per image) and their high pixel count, subcrops measuring 512×512 pixels were extracted from individual CCD frames. This approach preserved fine details while enabling the algorithm to differentiate streaks from background artifacts more effectively.

Streaks in the selected images were manually annotated using Label Studio [17]. Due to the limited number of original streaks detected in the dataset (352 streaks across 12,000 CCD frames), artificial streaks were generated to augment the dataset, enabling better characterization of the algorithm's performance. While this ratio may seem low, it is important to note that the population of satellites and space debris has increased drastically in recent years. Consequently, fewer streaks were present in the earlier years of the telescope's operation, and we expect a higher density of streaks in more recent archival data. The process for creating synthetic streaks is detailed in the next section.

The preprocessing workflow involved several steps:

1. Opening original FITS images
2. Removing prescan and overscan regions
3. Adding artificial streaks
4. Cropping images into 512×512 pixel subframes
5. Saving images as 16-bit PNG files

The decision to save images as PNG files rather than using FITS directly was driven by computational efficiency.

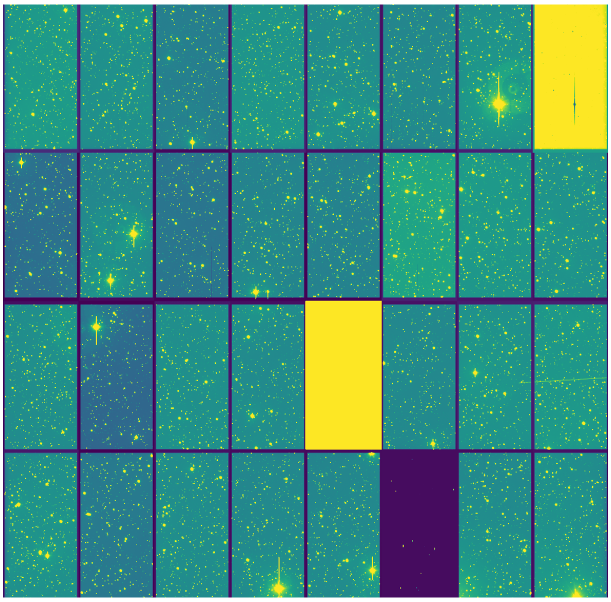


Figure 1: OmegaCAM detector layout.

PyTorch is optimized for standard image formats like PNG, whereas FITS files resulted in slower data loading. Ensuring 16-bit depth preserved all critical information for optimization without data loss.

Of the 375 images selected, 305 were allocated for training the algorithm, corresponding to a total of 312,320 individual PNG images. The remaining images were divided equally between the validation and test sets, with 35 images each, resulting in 35,840 PNG images per set. While the test set is not utilized in the current study, it will be employed in a subsequent phase of the analysis.

2.3. Artificial streaks generation

To ensure that the algorithm generalizes effectively across the OmegaCAM archive, it is crucial that the artificially generated streaks closely resemble real streaks observed in the images. To achieve this, we developed a streak generator that simulates realistic debris streaks through a multi-step process.

1. *Generation of a Space Debris Population:* A synthetic population of space debris was created based on orbital parameters and physical characteristics. This step ensures that the simulated streaks represent a diverse range of debris types.
2. *Computation of Signal-to-Noise Ratio (SNR):* For each debris streak, the signal-to-noise ratio (SNR) was calculated based on its brightness relative to the background noise in the image. This step ensures that the simulated streaks match the visibility conditions of real streaks, accounting for factors such as atmospheric conditions and image filter properties.

3. *Drawing of Streaks on Original Images:* Using the computed parameters, streaks were drawn directly onto the original images while preserving their background characteristics. The streaks were carefully rendered to mimic the appearance of real debris traces, including variations in brightness, length, and orientation.

Population generation To create a realistic yet flexible population for artificial streak generation, space debris objects were randomly selected from a predefined pool of altitudes and sizes, as summarized in Table 1. The pool was designed to include a higher proportion of smaller debris located closer to Earth's surface, reflecting the characteristics of the actual space debris population. However, instead of replicating the current known distribution of satellites and debris, we opted for this homogeneous pool to better evaluate the algorithm's performance across a wide range of scenarios. Once the debris objects were selected, each was assigned a corresponding streak. This involved generating random lines with varying lengths and orientations to simulate the motion of space debris across the detector. To align with real-world observations, where most streaks extend across the entire detector, 80% of the simulated streaks were designed to span the full width or height of the detector, while 20% were generated with one endpoint located within the detector's boundaries. This distribution ensures that the artificial streaks closely resemble those observed in actual OmegaCAM images while providing sufficient variety for robust algorithm training and validation.

SNR computation The sky background brightness for each image was first measured using a sigma-clipping approach to exclude bright sources and focus on the true background. The steps were as follows:

1. A sigma-clipping algorithm was applied to the image to compute a threshold for detecting sources, effectively isolating regions of interest.
2. Sources were detected using the computed threshold, and a segmentation image was created to identify their locations.
3. Detected sources were masked using circular footprints to exclude them from the background calculation.
4. Sigma-clipped statistics (mean, median, and standard deviation) were computed for the unmasked pixels, with the mean value used as the sky background brightness.

The signal from each space debris object was then calculated by assuming the debris to be spherical with a diameter taken from the predefined population and Lambertian scattering of light. The apparent brightness of the object was determined with the following formula.

Table 1: List of sizes and altitudes used to generate an artificial space debris population.

Size [m]	Altitude [km]
0.01	300
0.02	350
0.03	400
0.04	450
0.05	500
0.06	550
0.07	600
0.08	650
0.09	700
0.1	750
0.15	800
0.2	850
0.25	900
0.3	950
0.4	1000
0.5	1100
0.6	1200
0.7	1300
0.8	1400
1	1500
1.5	2000
2	3000
	4000
	5000
	10000
	20000
	35786
	36100

$$m_{sat} = M_{\odot} - 2.5 \cdot log_{10}(A \cdot \rho \cdot \phi(h)) + 5 \cdot log_{10}(h) + A_{\nu} \cdot \chi(z) \quad (1)$$

$M_{\odot}$ is the Sun's magnitude used as reference, A the cross sectional area of the object, ρ the object's albedo for which a value of 0.175 is assumed [13], h the altitude of the object, $\phi(h)$ the phase function, computed as

$$\phi(h) = \frac{1 - \cos(\alpha)}{2} \quad (2)$$

with

$$\cos(\alpha) = \frac{R_{\oplus} + h}{\sqrt{((R_{\oplus} + h)^2 + h^2)}} \quad (3)$$

and $R_{\oplus}$ the radius of the Earth. A_{ν} is the extinction in the specific filter band [3] and $\chi(z)$ the optical pathlength along a line of sight in units of air masses as a function of the zenith angle z.

$$\chi(z) = \frac{1}{\sqrt{1 - 0.96 \cdot \sin(z)^2}} \quad (4)$$

From the apparent magnitude m_{sat}, the flux of the space debris object was computed relative to a reference star with a magnitude of 20 and its corresponding flux value provided by ESO for each filter [4].

$$F_{sat} = F_{\star} \cdot 10^{-0.4 \cdot (m_{sat} - m_{\star})} \quad (5)$$

The total flux contribution of the space debris was then obtained by multiplying this flux by the exposure time of the streak. To account for seeing effects, the signal within the seeing aperture was calculated as

$$n_{aper} = \frac{n_{sat} \cdot 2 \cdot r_{FWHM}}{l_{streak} + 2 \cdot r_{FWHM}} \quad (6)$$

with n_{sat} the total flux contribution from the space debris, r_{FWHM} the radius corresponding to the full width at half maximum (FWHM) of the point spread function and l_{streak} the length of the streak. The SNR is finally computed as

$$SNR = \frac{n_{aper}}{\sqrt{(n_{aper} + n_{sky} \cdot p + p \cdot \sigma_R^2)}} \quad (7)$$

with n_{sky} the sky background flux, σ_R the readout noise of the imager and p the area factor defined as

$$p = 2 \cdot r_{FWHM}^2 \quad (8)$$

Before drawing streaks on images, any streaks with an SNR ≤ 2 were discarded. Such streaks are not visible to the naked eye and would not provide sufficient information for light curve analysis or meaningful training data for our algorithm. This ensures that only realistic and detectable streaks are included in our dataset.

Streak generation To accurately simulate the appearance of satellite streaks in astronomical images, it is essential to account for both the movement of the object and the atmospheric conditions that affect how the streak appears. The movement of the satellite or space debris creates the linear shape of the trace, while atmospheric effects such as scintillation and seeing influence its brightness and sharpness. Over long exposures, such as those in the OmegaCAM archive, seeing typically dominates for fixed objects, causing blurring. However, for fast-moving satellites or debris crossing the field of view, scintillation effects become locally significant, introducing rapid intensity fluctuations along the streak. Additionally, while the rotation of objects can create dashed or varying-intensity streaks, this effect was not modeled in the current study.

To incorporate these effects, we used Langevin dynamics—a stochastic framework that models systems influenced by both deterministic forces and random noise. Langevin dynamics is well-suited for modeling scintillation because it captures the random fluctuations caused by atmospheric turbulence while allowing for a stable equilibrium around a central position. Specifically, we modeled scintillation as a random walk within a potential well. Since seeing effects can be modeled using a Moffat profile, we used this profile as the harmonic potential in our Langevin dynamics framework.

The Langevin equation in one dimension is given by

$$\frac{dx}{dt} = -\frac{\partial V(x)}{\partial x} + \eta(t) \quad (9)$$

where $V(x)$ is the potential function and $\eta(t)$ is a stochastic noise term with properties

$$\langle \eta(t) \rangle = 0 \quad \text{and} \quad \langle \eta(t)\eta(t') \rangle = 2D\delta(t - t') \quad (10)$$

where D is the diffusion coefficient. For a Moffat profile, the potential is derived from its probability density function

$$f(x; \alpha, \beta) = \frac{\beta - 1}{\pi \alpha^2} \left(1 + \frac{x^2}{\alpha^2}\right)^{-\beta} \quad (11)$$

leading to the potential

$$V(x) = D \ln\left(1 + \frac{x^2}{\alpha^2}\right) \tag{12}$$

In two dimensions, this generalizes to

$$\frac{d\vec{x}}{dt} = -\nabla V(\vec{x}) + \vec{\eta}(t) \tag{13}$$

where

$$V(\vec{x}) = D \ln\left(1 + \frac{x^2 + y^2}{\alpha^2}\right) \tag{14}$$

and

$$\nabla V(\vec{x}) = D \cdot \frac{2\vec{x}}{\alpha^2 \left(1 + \frac{x^2+y^2}{\alpha^2}\right)} \tag{15}$$

By combining these equations with the computed SNR, we simulated realistic streaks that include scintillation-induced intensity variations and seeing-induced blurring.

The parameters α (scale parameter) and β (shape parameter) of the Moffat distribution can be derived from the seeing as

$$\alpha = \frac{FWHM}{2 \cdot \sqrt{2^{(1/\beta)} - 1}} \tag{16}$$

and a standard value of $\beta = 2.5$ was chosen for our model [18].

Since we could not identify a direct correlation between the diffusion coefficient D in the Langevin dynamics model and the physical parameters of atmospheric scintillation, we performed a fitting procedure using multiple simulated images. These images were generated with varying diffusion coefficients, and the wobbling of the streaks observed in each image was measured. Assuming that the objects are on circular orbits, we derived their velocities and, using known values of telescope resolution, aperture, and local windspeed, calculated their altitudes. The orbital velocity v can be derived from the measured wobble w and the scintillation timescale τ [14]

$$v = \frac{w}{\tau} \tag{17}$$

with

$$\tau = \frac{D_t}{v_w} \tag{18}$$

D_t is the telescope aperture and v_w the windspeed. The object's altitude can then be computed as

$$h = \sqrt[3]{\frac{\mu_\oplus}{(v \cdot \theta)^2}} - R_\oplus \tag{19}$$

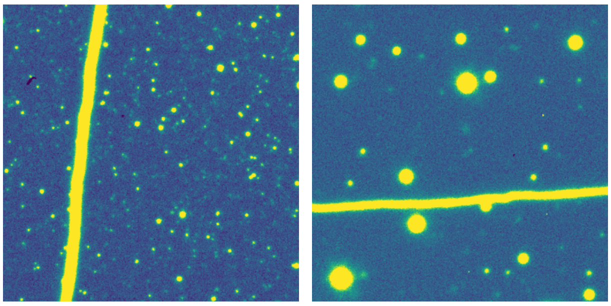
with θ the telescope resolution and $\mu_\oplus$ the standard gravitational parameter of the Earth.

After slight adjustments to the altitude-diffusion coefficient pairs through visual inspection, we derived the following relationship between the altitude h of an object on a circular orbit and its diffusion coefficient D

$$D = 12 \cdot \log(h) - 65 \tag{20}$$

Note: h is in [km] in Equation (20).

Figure 2 shows the final rendering of an artificial streak on an image.



(a) Artificial streak (b) Original streak

Figure 2: Comparison of an artificial and original streak for objects at similar altitudes.

2.4. Photometric reduction

To analyze the detected space objects, we are implementing an image processing pipeline that extracts astromentric and photometric measurements. While the astrometry is used to correlate the detections with known satellites, photometric lightcurves are extracted from the streaks to allow characterizing the tumbling states of the observed objects.

Detection refinement The output of the neural network is a bitmap that encodes for each pixel how likely it is to be contained within a streak. Extracting the actual start and end points of a detected streak from this bitmap sometimes results in these points not matching precisely to the actual observed streak. Thus, to enable precise astrometric measurements of the start and end points, the detections need to be aligned with the observation. Therefore, a cut-out of the detection is generated and rotated such that the streak is horizontally aligned. Then the streak is segmented into equally sized sections that are horizontally median combined to obtain cross-sectional profiles along the streak. These profiles are fitted to a Moffat distribution. The center of each fitted Moffat profile then corresponds to the center of the streak at the corresponding location. These centers are then fit to a straight line. To align the detected streak with the observed one, the cutout is then rotated according to the slope of the line so that it is horizontally aligned. This procedure is repeated until the slope of the fitted Moffat profile centers approaches zero. Finally, the adjusted end points are transformed back to the original image.

Astrometric calibration The raw images acquired with OmegaCAM already contain an astrometric solution. According to the OmegaCAM, this solution, however, corresponds to a rough estimate and should be refined to allow precise astrometric measurements. We use a local instance of Astrometry.net [1] to plate-solve the raw images and improve the astrometric solution. This improved solution is then used to obtain precise coordinates of the end-points for the detected streaks.

Object correlation Before the streaks can be further analyzed, they are correlated with all known objects that are contained in the space-track catalog [2]. Therefore, the two-line elements (TLEs) at epochs closest to the observation time of the images are propagated to the observation time to identify objects that crossed the detector throughout the observation. The identified candidates are then projected onto the image and assigned to the detected streaks. Due to inaccuracies in the TLEs, the projected positions do not precisely match the observed streaks. Therefore, thresholds in distance between observation and projection and the difference in position angle are used for the assignment.

Orbit fitting To align the orbit with the observation, the TLEs of the assigned objects are fit to the observed streak using the batch least squares orbit determination from the Orekit orbital dynamics library [1]. This process requires knowledge on the precise time-stamps for the end-points of the streaks. For detections that have start or/and end-point on the image, the time-stamps can be obtained from the start and/or end of exposure. To account for the OmegaCAM two-blade shutter, a pixel-dependent correction of the start and end of the exposure is implemented.

For streaks that cross the entire detector, the precise start and end time of the observation are unknown. In this case, the times that correspond to the minimum angular separations between the entry and exit points and the assigned TLE are computed and used as the initial time stamp for orbit fitting. Then, an interval of a few seconds around this initial observation time is defined and the orbit that has the minimum Mahalanobis distance [19] between the fitted orbit and the TLE is searched within this interval, and time stamps for the entry and exit points of the streaks are computed from this solution.

Lightcurve extraction Once the precise coordinates and corresponding observation times for the endpoints are determined, the lightcurves can be extracted from the streaks. Therefore, a series of rectangular apertures is placed along the streak, and the flux inside the apertures is measured. To obtain magnitudes for relative photometry with field stars, the corresponding exposure time (the time the object spent in the aperture) is computed from the fitted orbit.

The seeing effect introduces disturbance that results in the streaks deviating from a straight line. Because during most observations, the seeing is heavily over-sampled (OmegaCAM pixel size is 0.21 arcsec and the seeing FWHM for 50% of the observation time is larger than 0.8 arcsec), the excursions of the streak from a straight line can reach several pixels. In order to avoid clipping (the aperture only partly covers the streak) or excessively large apertures (to make sure the streak is entirely contained in each aperture), the apertures are centered on the streak. Therefore, a similar process as for the streak alignment is used: A cutout image of the streak is rotated to horizontally align the streak, and the streak is segmented into equally sized sections that are horizontally median combined to obtain cross-sectional profiles along the streak. Moffat distributions are fitted to the median combined cross-sectional profiles, and the centers of the Moffat fits define the center for the measurement aperture to obtain the flux. The size of the quadratic apertures is set to 2 times the average FWHM of the Moffat fits. To avoid measurement errors introduced by the rotation of the image, the centers of the apertures are transformed back to the original image before the fluxes are measured with photutils [10].

2.5. Performance metrics

In order to evaluate the results of the machine learning algorithm on the validation set, several metrics commonly adopted in the field are used and presented below.

Precision Precision is an indication of how many cases detected as positive are actually true positives. It tells us how good the algorithm is at making correct predictions when something is detected, and is calculated as follows.

$$Precision = \frac{TP}{TP + FP} \tag{21}$$

[1] https://astrometry.net/

[2] https://www.space-track.org/

Recall Recall gives an indication of how many positive detections, whether true or false, have been made by the model. It indicates the ability of the model to make positive detections. The formulation is as follows.

$$Recall = \frac{TP}{TP + FN} \tag{22}$$

F1-score The F1 score is the harmonic mean of precision and recall, offering a balanced measure of a model's performance by equally weighting both metrics. It is especially useful in situations where achieving perfect precision and recall is unrealistic. By combining these two metrics, the F1 score provides a comprehensive evaluation of a model's ability to make accurate predictions while effectively managing trade-offs between false positives and false negatives.

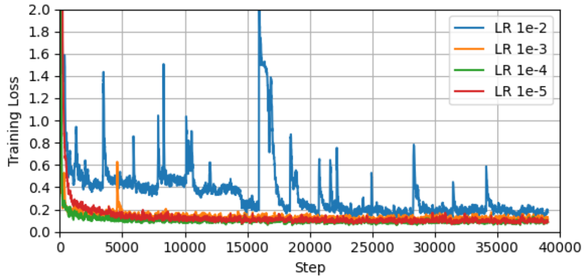
$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{23}$$

3. RESULTS

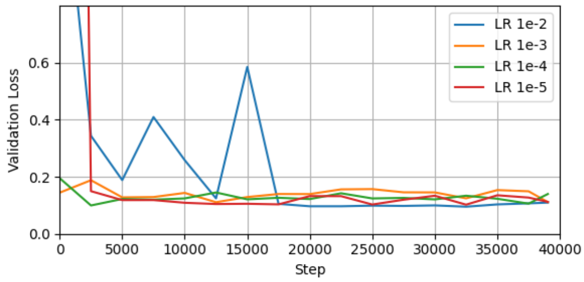
3.1. Learning rate search

Before training the final algorithm, we conducted a learning rate search to determine the optimal learning rate for our current configuration and dataset. To achieve this, we trained the model for one epoch using different learning rates and analyzed the training and validation losses. While training for only one epoch was sufficient due to the large size of our dataset, the batch size was limited to 8 because of memory constraints. As a result, the training loss curves were smoothed using an exponential moving average to better visualize trends, as the small batch size introduced noise into the loss curves.

From the results, it was evident that a learning rate of 1×10^{-2} was too high, as the training loss curve lacked a steep descent at the beginning, showed instability with visible peaks, and remained at a high loss throughout. The corresponding validation loss was also highly unstable. On the other end of the spectrum, a learning rate of 1×10^{-5} achieved better training loss but exhibited a similarly insufficiently steep descent at the start of training. Among intermediate values, 1×10^{-3} and 1×10^{-4} showed promising results. However, 1×10^{-4} was identified as the most appropriate learning rate as it achieved a lower training loss, slightly better validation loss, and greater stability compared to 1×10^{-3}. Based on this analysis, we selected 1×10^{-4} as the final learning rate for training the algorithm. A graph of the smoothed training and validation losses for all tested learning rates is shown in Figure 3.



(a) Training loss



(b) Validation loss

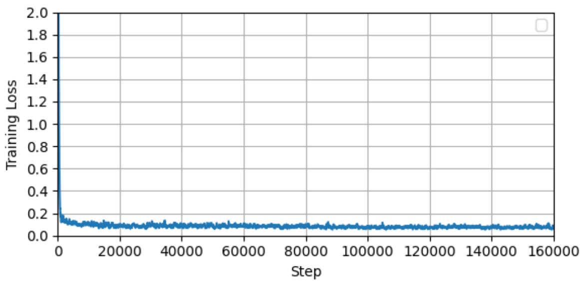
Figure 3: Learning rate analysis.

3.2. Performance results

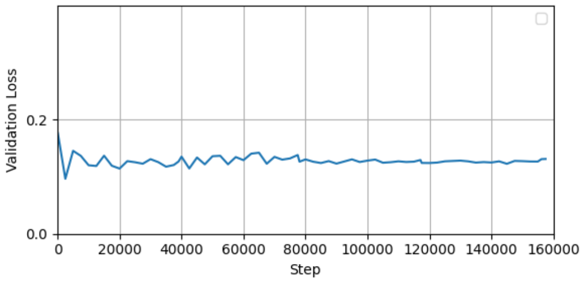
The algorithm was ultimately trained using the Adam optimizer with an initial learning rate of 1×10^{-4}. To address potential convergence issues, we employed the AMSGrad variant of Adam and applied L2 regularization with a factor of 2×10^{-5}. The training process spanned 4 epochs, which was feasible due to the large size of the dataset, and the learning rate was adjusted every 2 epochs. The training and validation losses throughout the training process are shown in Figure 4.

The evaluation of the model on the validation set produced the results shown in Table 2. Additionally, Figures 5 to 7 illustrate a selection of examples of correct detections made using our trained model, showcasing its performance on the validation data.

The performance results of our algorithm are highly promising, demonstrating its capability to effectively detect the majority of streaks across diverse conditions. While there is room for improvement in recall and precision, the achieved metrics are impressive given the wide variety of backgrounds, filters, shapes, and brightness levels of the streaks. Notably, the precision, recall, and F1 score are all consistently high and balanced, indicating the robustness of the algorithm without any significant bias toward over-detection or under-detection. A quick visual analysis revealed that most false positives are attributable to image artifacts, while most false negatives occur with extremely faint streaks or those located near image edges or crossing bright stars with pronounced diffraction spikes that confuse the model. Importantly, no difference was observed in the detection performance be-



(a) Training loss



(b) Validation loss

Figure 4: Training and validation loss.

Table 2: Performance of the trained model on the validation dataset

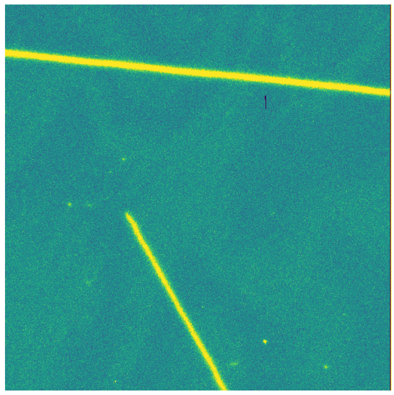
Precision	Recall	F1-score
96.4	96.5	96.4

tween real and artificial streaks, highlighting the strong generalization capacity of the algorithm. These findings underscore the reliability and adaptability of our approach in challenging scenarios.

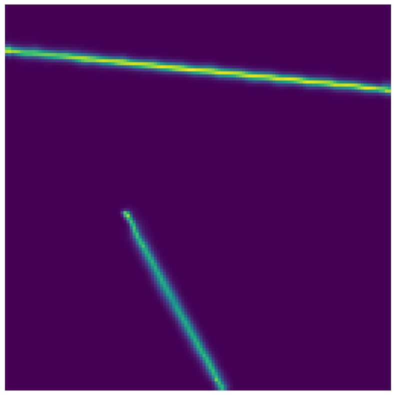
3.3. Photometric reduction results

We tested the reduction pipeline using the results of a previous version of the detection algorithm. Of the 3630 detected streaks in this dataset that corresponded to one month of VST observations in r-band, 1182 could be correlated with cataloged objects. These were processed by the photometric reduction pipeline to extract their lightcurves.

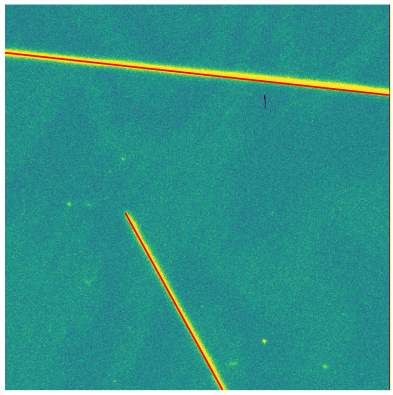
Figure 8 shows an example of poorly detected end points and how alignment is improved after detection refinement. This streak was detected with a previous version of the network and we expect that with the current version the precision of the detected end points is much higher. However, this example shows that even in extreme cases, we are able to exactly align the detection with the observed streak.



(a) Original image



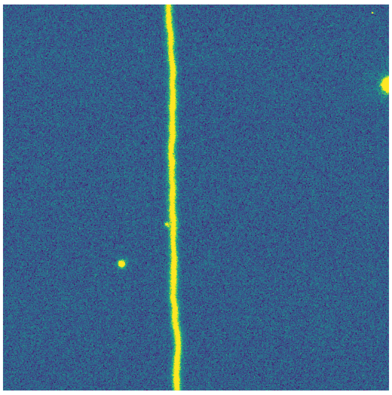
(b) Detection heatmap



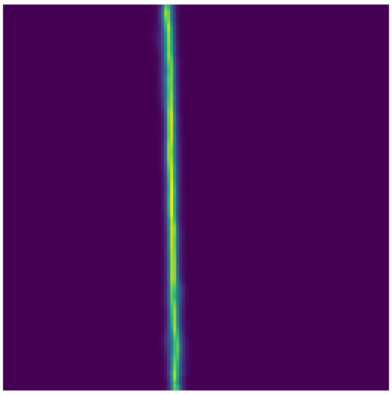
(c) Detected lines

Figure 5: Detection in an image with two streaks. The end of one of the streaks is within the image boundaries.

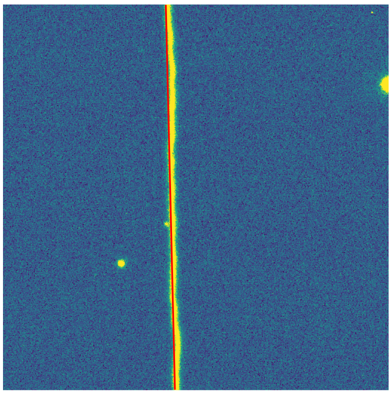
Figure 9 shows how the incorporation of the Mahalanobis distance into the orbit fitting improves the determined orbit. For this example, we used a streak that crossed several tiles of the detector but was already in the field of view when the exposure started. The blue line corresponds to the projected orbit on the image and indicates where the object would be according to its latest TLE. The offset between the observed streak and the TLE projection corresponds to several tens of arcseconds across- and along-track. We then used the entry and exit points of the tiles that were crossed throughout the exposure to see how much closer the fitted orbit reproduces the ac-



(a) Original image

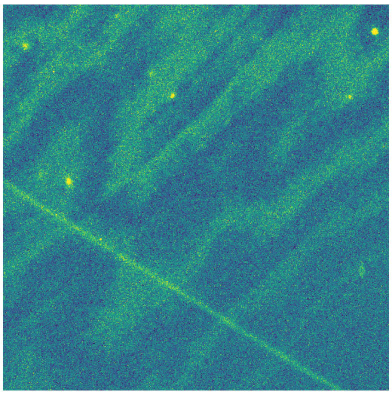


(b) Detection heatmap

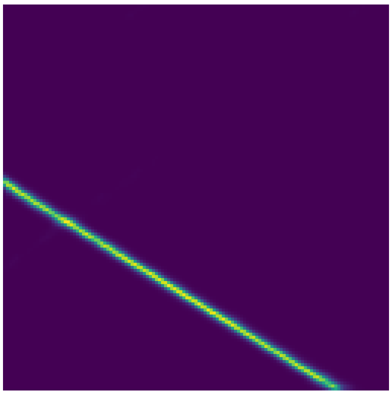


(c) Detected lines

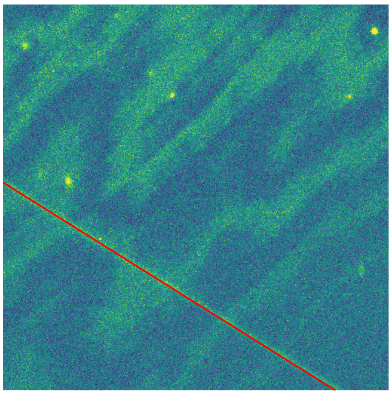
Figure 6: Detection of a slow object.



(a) Original image



(b) Detection heatmap

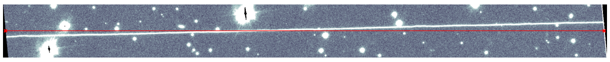


(c) Detected lines

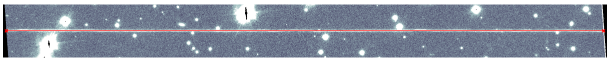
Figure 7: Detection of an object with low SNR (SNR 3.8). The background has visible filter fringe patterns.

tual start of the streak (without using the start point and start of exposure for orbit fitting). First, the initial time stamps that correspond to the minimum angular separation between the entry and exit points and the projected space-track TLE were used to fit the orbit to the observation. The red dashed line in panel (a) of figure 9 corresponds to the projection of the resulting orbit. While the across-track error is completely compensated, the along-track error remains and the fitted orbit fails to reproduce the actual start of the streak (green cross). The lower panel of figure 9 shows the orbit with the minimum Mahalanobis distance to the space-track TLE in an interval of a few seconds of the initial time stamps. The along-track is significantly decreased, and the fitted TLE is able to reproduce the start of the observed streak within a less than two arcseconds. This test demonstrates that our procedure to fit orbits of streaks that cross the whole detector produces very accurate orbits.

Satellites and space debris in LEO can cross the detector in as little as 1 second, which means that for such objects only very short lightcurves are obtained, and many of them have small SNRs because of their high angular velocities. Still, robust brightness measurements were extracted, and an indication of rotation is visible for some

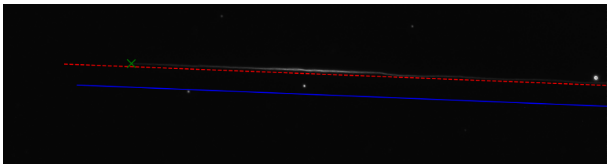


(a) Initial detection of a streak's end points.

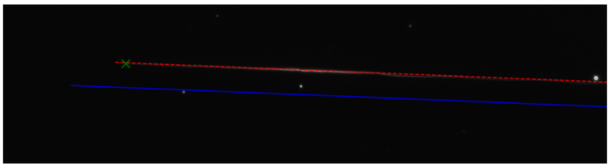


(b) Improved alignment of the detection after postprocessing.

Figure 8: Detection alignment.

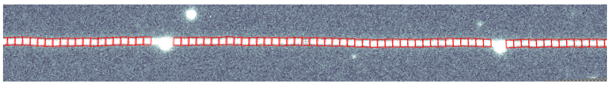


(a) Start point of the observed streak (green x), space-track (blue) TLE and initial fit (red).

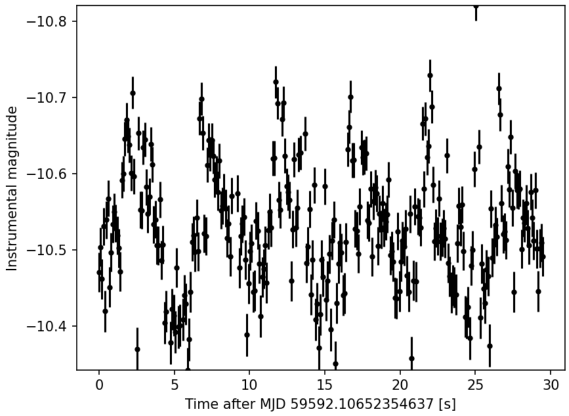


(b) Space-track (blue) TLE and orbit with the smallest Mahalanobis distance to the space-track TLE (red).

Figure 9: Results of orbit fitting using the Malananobis distance.



(a) Measurement apertures.



(b) Lightcurve.

Figure 10: Placement of the measurement apertures along a streak (a). Each aperture is centered on the peak of the Moffat profile fitted to the data. Apertures contaminated by field stars are automatically detected as outliers in distribution of the Moffat parameters and removed. Raw lightcurve extracted from the streak (b).

of the observed objects in LEO. Satellites in higher orbits cross the image much more slowly. For satellites in GEO, lightcurves of several minutes were captured. Figure 10a shows how the measurement apertures are centered on the streak. The parameters of the Moffat profiles (except for the amplitude) are expected to be similar throughout the streak, and the center only slightly changes from one aperture to another. Apertures that are contaminated by cosmics, background stars, or other image artifacts result in outliers in the distribution of Moffat parameters and thus can easily be detected to exclude them from further processing. The raw lightcurve in instrumental magnitude extracted from this streak is shown in Figure 10b. Refinement of lightcurves into higher-order data products, such as rotation periods or phase curves, is currently done manually [5] but will be incorporated into the processing pipeline. An example of a period determination of a GEO satellite that was found in 11 images in the preliminary detection network results dataset was presented in a previous work [15].

4. DISCUSSION

The results of this study are highly promising, demonstrating that our method is effective in detecting satellite and space debris traces in wide-field images, with good performance scores. However, further analysis is needed to better understand the limitations of the algorithm, particularly in terms of undetected streaks. Investigating factors such as limiting signal-to-noise ratio (SNR) and refining the post-processing pipeline to minimize false positives could significantly enhance performance. Additionally, the test set mentioned earlier, which exclusively contains real streaks, will serve as a definitive evaluation of the algorithm's effectiveness on raw data without artificial streaks. With these final adjustments in place, the next exciting phase will involve applying the algorithm to archival data to identify previously undetected space debris. Preliminary findings from this study, along with similar work by others, suggest that many objects not listed in existing catalogs can be identified, offering new opportunities for the field of Space Situational Awareness (SSA). Beyond discovering new debris, this approach also holds potential for studying the historical evolution of debris populations by analyzing past observations, providing valuable insights into long-term trends and dynamics in Earth's orbital environment.

In addition to detecting streaks, we show that we are able to obtain precise astrometric and photometric measurements from these detections. Incorporating the Mahalanobis distance in the orbit-fitting process allows us to fit accurate orbits even when the streak crosses the entire field of view. This capability is crucial for tracking faint and fast-moving objects. However, in the current implementation, the mean anomaly is not accounted for when calculating the Mahalanobis distance. To further improve our orbit-fitting procedure and enhance its accuracy, we plan to incorporate the mean anomaly into future iterations of our method.

As mentioned previously, analysis of the preliminary results indicates that many detections cannot be identified with any known objects. Most of these detections are relatively faint streaks that cross the entire detector, and we expect them to be caused by untracked debris in low Earth orbit (LEO). These detections contain very important information on this poorly observed population. However, without an orbit that we can associate with these streaks, no further analysis—such as brightness measurements—is possible because the exposure times (the duration objects spend on the images) are unknown. To address this limitation, we have started exploring how atmospheric seeing effects could be utilized to estimate the angular velocity of objects causing these streaks. By assuming circular orbits, this angular velocity could provide an estimate of an object's altitude. To achieve this, we used observing condition data (e.g., seeing FWHM, temperature, wind speed, and atmospheric coherence time), combined with streaks from identified objects (for which we know angular velocities) and synthetic streaks, to train machine learning architectures such as convolutional neural networks (CNNs) and transformers. Although both algorithms performed well on synthetic data, their performance on real data was limited. We concluded that this shortfall was likely due to insufficient training data for capturing how streak appearances depend on observing conditions. As more real detections become available, we plan to repeat these experiments with an expanded dataset.

To ensure that our findings contribute meaningfully to ongoing research efforts in SSA and related fields, we are planning to publish all detections and light curves in a dedicated repository called Orbital Debris Lightcurve Inventory (ODLI). This database and its web interface are currently under development and will be finalized once the detection algorithm reaches its final version. Furthermore, we are collaborating with the IAU Centre for the Protection of the Dark and Quiet Sky from Satellite Constellation Interference (CPS) to define interfaces between our tools and their data repositories. This collaboration will ensure that our results can be seamlessly integrated into other databases and made accessible to a broader scientific community.

ACKNOWLEDGMENTS

The authors would like to extend their gratitude to Arthur Devaux, Aymeric Deslarzes and Seifeddine Ajmi, who contributed significantly to the annotation of our training dataset. Their diligent work was invaluable in preparing the data for analysis and we appreciate their dedication to this project. We further thank Sarah Marciniak for her support in implementing the orbit fitting procedures, Dofri Vidarsson for investigating how the seeing effect could be used to estimate the angular velocity of a observed object and Léonard Flückiger for his initial work on the modelization of atmospheric scintillation.

We wish to express our sincere appreciation to the ESA Academy for their generous sponsorship, which enabled our participation in this conference.

We acknowledge the use of AI-assisted technologies, specifically Perplexity AI, during the preparation of this manuscript. The AI was utilized to improve readability and refine language. All content has been reviewed and validated by the authors to ensure accuracy and integrity.

REFERENCES

1. B. Cazabonne and P. Cefola. Towards accurate orbit determination using semi-analytical satellite theory. In *31st AAS/AIAA Space Flight Mechanics Meeting*, 2021.
2. P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. In *Computer Vision – ECCV 2008*, pages 197–210. Springer Berlin Heidelberg, 2008.
3. ESO. La silla extinction coefficients. Accessed 10.03.2025.
4. ESO. Omegacam exposure time calculator. Accessed 10.03.2025.
5. S Hellmich, E Rachith, B. Y. Irureta-Goyena, and J.-P. Kneib. Harvesting Large Astronomical Data Archives For Space Debris Observations. In *2nd NEO and Debris Detection Conference*, volume 2. ESA Space Debris Office, 2023.
6. P. Hickson. A fast algorithm for the detection of faint orbital debris tracks in optical images. *Advances in Space Research*, 62(11):3078–3085, 2018.
7. K. Huang, Y. Wang, Z. Zhou, T. Ding, S. Gao, and Y. Ma. Learning to parse wireframes in images of man-made environments. In *Computer Vision – ECCV 2018*, pages 626–635, 2018.
8. C. Jeffries and R. Acuña. Detection of streaks in astronomical images using machine learning. *Journal of Artificial Intelligence and Technology*, 4(1):1–8, 2023.
9. K. Kuijken. OmegaCAM: ESO's newest imager. *The Messenger*, 146:8, 2011.
10. Bradley L., Sipőcz B., Robitaille T., Tollerud E., Vinícius Z., Deil C., Barbary K., Wilson T. J., Busko I., Donath A., Günther H. M., Cara M., Lim P. L., Meßlinger S., Burnett Z., Conseil S., Droettboom M., Bostroem A., Bray E. M., Bratholm L. A., Jamieson W., Ginsburg A., Barentsen G., Craig M., Pascual S., Rathi S., Perrin M., and Morris B. M. astropy/photutils: 2.2.0, 2025.
11. Y. Lin, S. L. Pintea, and J. C. van Gemert. Deep hough-transform line priors, 2020.
12. L. Liu, S. Rongyu, and Z. He. Identification of megaconstellations in wide-field astronomical images with machine learning. *The Astronomical Journal*, 169(4):205, 2025.

13. M. Mulrooney, M. Matney, M. Hejduk, and E. Barker. An investigation of global albedo values. 2008.

14. G. Nir, B. Zackay, and E. O. Ofek. Optimal and efficient streak detection in astronomical images. *The Astronomical Journal*, 156(5):229, 2018.

15. E Rachith, S Hellmich, and B. Y. Irureta-Goyena. A Novel Machine Learning Based Algorithm for Efficient Streak Detection. In *2nd International Orbital Debris Conference*. NASA Orbital Debris Program Office, 2023.

16. Stoppa, F., Groot, P. J., Stuik, R., Vreeswijk, P., Bloemen, S., Pieterse, D. L. A., and Woudt, P. A. Automated detection of satellite trails in ground-based observations using u-net and hough transform. *A&A*, 692:A199, 2024.

17. M. Tkachenko, M. Malyuk, A. Holmanyuk, and N. Liubimov. Label Studio: Data labeling software, 2020-2022. Open source software available from https://github.com/heartexlabs/label-studio.

18. I. Trujillo, J.A.L. Aguerri, J. Cepa, and C.M. Gutiérrez. The effects of seeing on sérsic profiles – ii. the moffat psf. *Monthly Notices of the Royal Astronomical Society*, 328(3):977–985, 2001.

19. A Vananti and T Schildknecht. Metric for object association in the Keplerian orbits space. In *71st International Astronautical Congress (IAC)*, 2020.