Università degli Studi di Napoli
"Parthenope"

International PhD Programme / UNESCO Chair

"Environment, Resources and Sustainable Development"

Automatic identification of solar and extra-solar moving and transiting objects in astronomical images with Artificial Intelligence

SUPERVISOR
Prof. Alessandra Rotundi

CANDIDATE
Stefano Fiscale

TUTORS
Prof. Laura Inno
Prof. Alessio Ferone

Cycle XXXVIII

Author's address Stefano Fiscale
Department of Science and Technology,
Parthenope University of Naples
stefano.fiscale001@studenti.uniparthenope.it

Supervisor(s) Prof**.** Alessandra Rotundi
Department of Science and Technology
Parthenope University of Naples

Tutor(s) Prof. Laura Inno
Prof. Alessio Ferone

Reviewer(s) Prof. David John Armstrong
Prof. Özlem Polat
Prof. Tiziano Zingales
Prof. Valerio Bozza
Dr. Stefano Cavuoti

Reviewers

Due Docenti Comunità Europea per la Label PhD European Comunity:

1) Prof. David John Armstrong
D.J.Armstrong@warwick.ac.uk
Department of Physics, University of Warwick, Gibbet Hill Road, Coventry, CV4 7AL, UK

2) Prof. Özlem Polat
ozlem.polat@cumhuriyet.edu.tr
Teknoloji Fakültesi, Mekatronik Mühendisliği Bölümü, Elektronik Sistemleri A.B.D.

Tre Docenti Atenei nazionali:

1) Prof. Tiziano Zingales
tiziano.zingales@unipd.it
Università degli Studi di Padova, Dipartimento di Fisica e Astronomia, Vicolo dell'Osservatorio 3, 35122 Padova, Italia

2) Prof. Valerio Bozza
vbozza@unisa.it
Dipartimento di Fisica "E.R. Caianiello", Università di Salerno, Salerno, Italia

3) Dr. Stefano Cavuoti
stefano.cavuoti@inaf.it
INAF, Osservatorio Astronomico di Capodimonte, Salita Moiariello 16, 80131, Napoli, Italia

Abstract

The rapid growth of photometric data from space- and ground-based surveys made crucial the use of Artificial Intelligence (AI) in the search for moving objects within and beyond the Solar System. This doctoral thesis addresses two critical challenges in modern astrophysics: optimizing Deep Learning (DL) architectures for exoplanet detection and providing the observational constraints to characterize the dust environment of long-period comets in support of the ESA/Comet Interceptor mission. The core thesis of the exoplanetary research is that the high complexity and overparameterization of state-of-the-art Convolutional Neural Networks (CNNs) are redundant for transit detection. Since planetary transits are essentially low-dimensional features, I demonstrate that minimizing architectural depth, width, and resolution mitigates overfitting and enhances generalization. Through a methodology involving compact CNNs and dimensionality reduction techniques, I designed models with cross-mission compatibility, specifically demonstrated using data from both NASA's *Kepler* and the Transiting Exoplanet Survey Satellite (TESS). Results show that simplified architectures perform at least on par with complex ones, providing a robust framework for future surveys like ESA's PLAnetary Transits and Oscillations of stars (PLATO). In parallel, in support to the selection of the target comet for the ESA/Comet Interceptor mission, I served as principal observer for the acquisition and analysis of photometric data of long-period comets using the Copernico Telescope (Asiago, Italy) and the Telescopio Nazionale Galileo (TNG, Canary Islands, Spain). I reduced these data by using a pipeline written in Interactive Data Language to derive $Af\rho$ curves. These curves were subsequently used to impose constraints on the study aimed at characterizing the composition and activity of the dust tails of long-period comets located more than 4 astronomical units from the Sun.

Contents

Abstract ii

iii

1 Introduction 1

2 Exoplanets detection with Deep Learning 1

2.1 Introduction . . . 1

2.1.1 How do we find exoplanets? . . . 1

2.1.2 Space-based telescopes . . . 2

2.1.3 Manual classification of transit signals . . . 5

2.1.4 Related Works . . . 6

2.1.5 Main catalogs of Threshold Crossing Events . . . 7

2.2 Aims . . . 9

2.2.1 Definition of model complexity . . . 9

2.2.2 Rationale behind minimizing model complexity . . . 10

2.3 Results for Exoplanets . . . 13

2.3.1 "Identifying exoplanets in TESS data by Deep Learning" [26]. . . 13

2.3.2 "DART-Vetter: A Deep LeARning Tool for automatic triage of exoplanet candidates" [23]. . . . 24

2.3.3 "Detection of Exoplanets in Transit Light Curves with Conditional Flow Matching and XGBoost" [24] . . . 45

3 Cometary observations in support of the ESA/Comet Interceptor space mission 72

3.1 Introduction . . . 72

3.2 Aims . . . 74

3.3 Results for Comets . . . 75

3.3.1 "Inbound evolution of twenty long period comets [6]" . . . 75

3.3.2 My contribution to the paper . . . 93

4 Conclusions **101**
4.1 Conclusions on Exoplanet detection with Deep Learning . . . 101
4.1.1 Open issues and future development . . . 104
4.2 Conclusions on Cometary observations in support of the ESA/Comet Interceptor space mission . . . 106

5 Additional research activities **107**
5.1 List of co-authorship manuscripts . . . 107
5.2 List of talks and posters . . . 109
5.3 Collaborations . . . 109
5.4 Research stay at Laboratoire d'Astrophysique de Marseille . . . 110
5.5 Teaching experience . . . 111
5.6 Scientific seminars . . . 111
5.7 Public Outreach . . . 113
5.8 Additional Training . . . 113
5.9 Reviewer activities . . . 114
5.10 Prizes / Awards . . . 114

A Exoplanet detection with Deep Learning **122**
A.1 Detection of Exoplanets in Transit Light Curves (ongoing work) . . . 122

B Night report examples **133**
B.1 Date: 2025-01-03. Telescope: Copernico. . . . 133
B.2 Date: 2024-12-01. Telescope: TNG . . . 134
B.3 Date: 2024-11-07. Telescope: Copernico . . . 136
B.4 Date: 2024-10-30. Telescope: Copernico . . . 136

List of Tables

A.1 Comparison of VGG-19 (with two different pooling sizes (psz) of 5 and 3), Resnet-34 and Resnet-18 architectures. The best values are highlighted in boldface. 125
A.2 Classification metrics obtained with a Support Vector Machine, radial basis function kernel, applied on the two-dimensional test data. 128
A.3 Resnet-18 in inference mode. Evaluation metrics on test set by varying training epochs on the grid {40, 50, 60}. 131

B.1 Seeing quality: ∼4. Humidity (%): [80,100] 133
B.2 Information about the contents of each FITS file. The prefix of each of the following files is LRS.2024-12-03. 134
B.3 Information about the contents of each FITS file. 136
B.4 Summary of data collected during the night in terms of number of exposures and duration of each. 137
B.5 Information about the contents of each FITS file. 138
B.6 Summary of data collected during the night in terms of number of exposures and duration of each. 138

List of Figures

2.1 Stacked bar chart reporting the cumulative number of exoplanets discovered per year and detection method. Image Credit: NASA Exoplanet Archive 3
2.2 A set of simulated global views projected onto a two-dimensional space. The data are fitted with a Support Vector Machine with radial basis function kernel and linear kernel to reproduce a scenario of overfitting (left panel) and underfitting (right panel), respectively. 10
2.3 Comparison of the network architectures presented in Paper 1 [26] and Paper 2 [23]. 26
2.4 Architectural diagram of our new model, based on three main modules: feature extraction, dimensionality reduction and classification. The model is trained on 1×201 global views to produce a multi-class classification in astrophysical false positive (EB, purple box), planet candidate (PC, green box) and non-transiting phenomenon (J, yellow box). 46

3.1 Schematic representation of the Comet Interceptor flyby trajectory (not to scale). Spacecraft A performs the most distant passage from the comet's nucleus, while Probes B1 and B2 follow closer paths. Both probes will transmit real-time data to Spacecraft A, where it will be collected and stored. Figure credit: Jones G.H. et al. (2024) [44]. 74
3.2 An example of bias (left panel) and flat field (right panel) calibration files acquired at TNG during an observing run. 96

3.3 Assessing exposure times for capturing images of comets requires analysis of the relevant star fields. (Left panel) In regions of the sky free of star clusters and bright stars, the exposure time can be set to at least 10-15 minutes. (Center panel) With a very bright star in the upper left, the exposure time should not exceed 15 minutes due to the potential risk of image saturation and corruption of the portion of the image affected by the comet. (Right panel) If the comet is located in regions of the sky with many stars, some of which are very bright, it is imperative to set very low exposure times in the order of 5 minutes, at least during the first exposure. 97
3.4 Computation of the median image for bias and flat field. The FITS files are loaded, obtaining a $n \times w \times h$ tensor. Trimming is applied to remove the regions at the edges of the image. Then, the median is computed at each pixel across the tensor with the IDL command `medarr`. 98
3.5 Determining the two starless areas for sky background removal. (Left panel) The image is opened in IDL and the area where the comet is located is identified (orange box). (Middle panel) A zoom in on the comet area is performed until a clear view of the sky arount it is achieved. (Right panel) The color of the image is inverted and the saturation is significantly increased in order to assess the maximum extent of the coma and tail. Finally, the two regions of interest in the sky are defined (red rectangles). Comet: C/2020 O2, image acquired at TNG on August 24, 2022. 99
3.6 An example of $Af\rho$ curve computed for the comet C/2020 O2. The image of the comet was acquired at the TNG on July 23, 2022, with an exposure time of 480 seconds. The data reduction process was carried out on July 9, 2024. 100

A.1 Training curves of the loss function, F1-score, and AUC obtained by training VGG-19 for 100 epochs with pooling size (psz) set to 3 and 5. . 123
A.2 Two-dimensional distributions of the feature vectors obtained by training VGG-19 with Adam, as a function of pooling size and training epochs. The top row shows results with pooling size 3 after 40, 60, and 100 epochs (from left to right). The bottom row shows the corresponding distributions for pooling size 5. Purple points indicate samples belonging to the EB class, green points represent samples from the PC class, and yellow points correspond to the J class. 124

A.3 Two-dimensional distributions of the feature vectors obtained by training VGG-19 with SGD, as a function of pooling size and training epochs. The top row shows results with pooling size 3 after 40, 60, and 100 epochs (from left to right). The bottom row shows the corresponding distributions for pooling size 5. Purple points indicate samples belonging to the EB class, green points represent samples from the PC class, and yellow points correspond to the J class. 124
A.4 Two-dimensional distributions of the training set feature vectors obtained with our best four network configurations. VGG-19 with pooling sizes 3 and 5 (first two panels), ResNet-34 (third panel), and ResNet-18 (fourth panel). These distributions were produced using the t-Stochastic Neighbor Embedding algorithm. Purple points indicate samples belonging to the EB class, green points represent samples from the PC class, and yellow points correspond to the J class. 126
A.5 Two-dimensional distributions of training and test set features. (Top row) The first panel shows the normalized data projected with t-SNE, while the second shows the relative mapping learned by the MLP on the training set. The last panel shows the test set features projected with the MLP. (Middle row) The three panels show the mapping learned by the MLP for each class on the training set. (Bottom row) Distribution of test data when projected with the MLP. 127
A.6 Two-dimensional distributions of training and test set features produced by the MLP, with the first two principal components provided for each class. 128
A.7 Two-dimensional distributions of the training set feature vectors within the embeddings generated by t-SNE (left panel) and by the MLP (middle panel). The right panel shows the feature vectors of the test set produced by the MLP. 130
A.8 Three-dimensional distributions of the training set feature vectors within the embeddings generated by t-SNE (left panel) and by the MLP (middle panel). The right panel shows the feature vectors of the test set produced by the MLP 130
A.9 Training and validation metrics when training Resnet-18 on the balanced dataset. 131
A.10 Comparison of the network architectures presented in Paper 3 [24] and Paper 4 (Fiscale et al. 2025, under preparation). 132

Listings

Chapter 1

Introduction

The research conducted during my PhD focused on the collection and analysis of photometric data from moving objects within and beyond the Solar System, such as *comets*, moving and active minor bodies of the Solar System, and *exoplanets*, planets orbiting stars other than the Sun.

During my PhD course I developed Deep Learning models for the identification of exoplanet candidates in data collected by NASA's *Kepler* [46] and Transiting Exoplanet Survey Satellite (TESS) [71] space telescopes.

In parallel, I worked as the principal observer in the frame of the Italian team supporting the selection of the comet that will be the target for the European Space Agency (ESA)/Comet Interceptor space mission [44]. I carried out photometric observations with the Copernico Telescope, Asiago, Italy, and the Telescopio Nazionale Galileo (TNG), La Palma, Canary Islands, Spain, and contributed to the subsequent data reduction.

Although the two research topics I explored during my PhD are quite distinct in principle, they actually provide complementary perspectives on the same research field: planetary systems formation. In fact, comets play an important role in the process of solar system building block formation, while exoplanets contribute to learn about planetary bodies diversity, thus to understand planetary systems final architectures. Therefore, the results presented in this thesis, i.e. the development of novel Deep Learning approaches for exoplanets identification and ground-based observation in support of ESA/Comet Interceptor space mission, contribute to the broader effort of reconstructing the evolution from a gas+dust proto-planetary disc to the emergence of a planetary systems.

The remainder of this thesis is organized as follows: Chapter 2 presents the work conducted in the context of identifying the best planetary candidates for further study; Chapter 3 describes the work performed in support of ESA/Comet Interceptor space mission; the conclusions drawn from both lines of research are reported in Chapter 4; Chapter 5 outlines additional activities carried out during my PhD course, including the list of co-authored publications, presentations at international and national conferences, and outreach initiatives.

Chapter 2

Exoplanets detection with Deep Learning

2.1 Introduction

This chapter reports the results I obtained within the main project of my thesis: the development of Deep Learning models for the detection of exoplanet candidates.
In order to define the background of my work I describe in this introductory section the main exoplanets detection methods, the contribution of space-based telescopes in discovering potential exoplanetary signals, and the challenges associated with manual and automatic classification of these signals. Section 2.2 defines the research objectives, explaining the concept of model complexity and the rationale behind its minimization. Section 2.3 describes, unfolding the reasoning and implementation behind the deployment of the proposed models and illustrating their progressive evolution across different studies, the results I obtained and published in three referenced papers, reported in this Chapter.

We will refer to these three papers, attached at the end of Sections 2.3.1, 2.3.2, and 2.3.3, as Paper 1 [26], Paper 2 [23], and Paper 3 [24], respectively.

2.1.1 How do we find exoplanets?

The discovery of planets beyond the Solar System has revolutionized our understanding of the formation and evolution of planetary systems, revealing a diversity of worlds far greater and more complex than imagined just a few decades ago.

In 1995, Michel Mayor and Didier Queloz published a paper announcing the discovery of 51 Pegasi b, the first extrasolar planet orbiting a Sun-like star [58]. The excitement

surrounding the discovery attracted attention in the field, and since then, the number of detected exoplanets has increased dramatically as shown in Figure 2.1. In the following fifteen years, most of the identified planets were gas giants. The improvement in observational capabilities brought by dedicated space missions – such as *Kepler* and TESS – has led to the discovery of rocky, Earth-sized planets that are potentially habitable [11].

Among the wide range of methods used to identify extrasolar planets, five of them led to the most discoveries: transit, radial velocity, direct imaging, microlensing and pulsar timing. Transit method [17] is based on measuring dims in star's light originating from the passage of celestial objects directly between the parent star and an observer. Radial velocities [58] measure the observable shift in the color of the star's light wobbling in space due to orbiting objects. Direct imaging [57] is based on gathering photometric observations capturing planetary images. This method requires planet be large enough to be visualized. Microlensing [56] is based on measuring light bent from a distant star when a planet passes between the star and Earth. Pulsar timing is the method used by Aleksander Wolszczan and Dale Frail in the discovery of the first exoplanet in 1992 [91], and it consists in leveraging on the highly regular, precise pulses from pulsars – rotating neutron stars – to detect exoplanets. Differences measured in pulse intervals might suggest the presence of an exoplanet.

Among the 6042 confirmed exoplanets[1], 74% has been identified with the transit method, 19% with radial velocities, 4.2% with microlensing and 1.4% with direct imaging. Based on the transit method, *Kepler* and TESS have played a crucial role in this achievement, contributing to the discovery of the vast majority of exoplanets known to date. Only about 3% of these exoplanets are terrestrial, highlighting how current hardware and software instrumentation still pose limitations in the detection of new Earth-like worlds. This limitation is expected to be overcome with upcoming space missions, such as ESA's PLAnetary Transits and Oscillations of stars (PLATO) telescope [69], whose primary goal is to detect and characterize terrestrial exoplanets in orbits extending up to the habitable zone.

2.1.2 Space-based telescopes

This section provides details on the three missions that produced the data analyzed in this thesis project: *Kepler*, TESS, and PLATO.

The aim of these missions is to develop telescopes capable of collecting photometric observations of a set of target stars in order to identify, using the transit method, signals

[1] Data acquired from NASA Exoplanet Archive on November 10th, 2025.

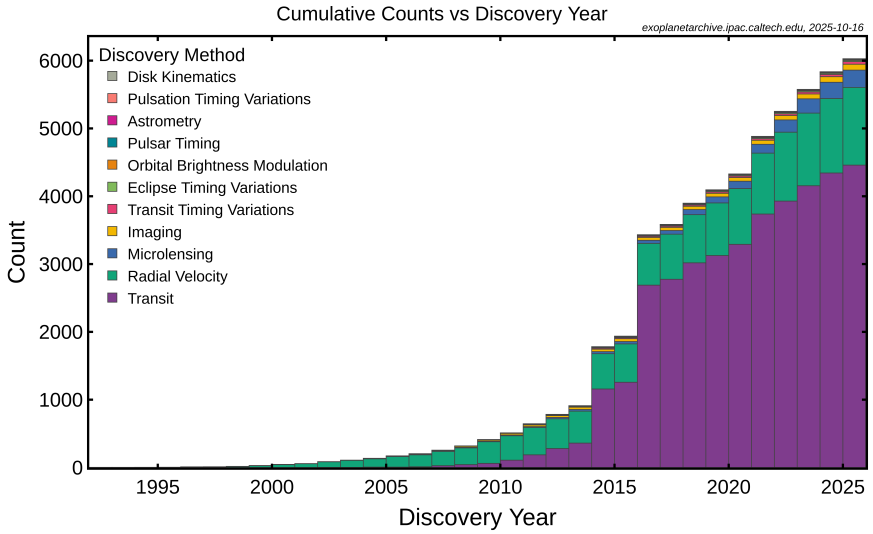

Figure 2.1: Stacked bar chart reporting the cumulative number of exoplanets discovered per year and detection method. Image Credit: NASA Exoplanet Archive

potentially attributable to planets, known as Threshold Crossing Events (TCEs).

Threshold Crossing Events (TCEs)

Before being analyzed by Machine Learning (ML) models developed in this thesis, the raw photometric data collected by these space missions undergo an initial processing phase aimed at identifying potential transit-like events. This is achieved through dedicated detection pipelines that search for periodic dims in stellar brightness consistent with the passage of a planet across the stellar disk. Each candidate signal that satisfies a predefined signal-to-noise ratio threshold is labeled as a TCE. These signals, which can originate from either planetary transits or various types of false positives (e.g. eclipsing binaries, instrumental artifacts), represent the input data analyzed by the models presented in this thesis. Technical details on TCEs detection and pre-processing are discussed in Section 2 of Paper 2.

The Kepler mission

The *Kepler* spacecraft was launched by NASA in March 2009 with the primary goal of estimating the fraction of stars in our galaxy that host Earth-like planets.

Designed to observe approximately 150,000 target stars in the Cygnus constellation, *Kepler* monitored over 530,000 stars during its nine-year mission, leading to the discovery of more than 2600 confirmed exoplanets [10]. The spacecraft's observing strategy consisted of continuously monitoring a single stellar field of 110 square degree to collect brightness measurements and identify TCEs. The telescope was equipped with a 0.95

m effective aperture and an array of 42 CCDs, each with a pixel scale of about 3.98 arcseconds. Images were obtained with two different sampling rates: long cadence (29.4 min) and short cadence (58.9 s).

After the failure of two out of the four reaction wheels on the *Kepler* spacecraft, the observation of the target field of view was no longer possible. The mission was named K2 [39], and the new observation strategy involved collecting photometric observations near the ecliptic, observing fields sequentially as the spacecraft orbited the Sun. This strategy regularly identified new and well-characterized target regions, enabling the observation of scientifically important objects across a wide range of galactic latitudes in both celestial hemispheres.

For each star, light curves were generated using aperture photometry, with apertures optimized to maximize the signal-to-noise ratio and minimize contamination from nearby stars. The data reduction pipeline included corrections for systematic trends, cosmic rays, and pointing variations. These details are extensively described in Section 2.1 of Paper 2 [23].

The TESS mission

On April 2018, the successor of *Kepler*, NASA's TESS telescope, was launched. TESS surveys the entire sky through a sector-based scanning strategy, with each sector observed for approximately 27 days. It is equipped with four CCD cameras, each covering a 24° × 24° field of view with a resolution of 21 arcseconds/pixel. As a result, each observed sector corresponds to a sky area of 24° × 96°. The original cadence of TESS photometric observations for its ∼200,000 target stars was 2 minutes. Following the mission's success and the need to observe additional targets, new sampling rates of 20 seconds, 10 minutes, and 30 minutes were introduced.

To date, TESS has contributed to the discovery of more than 400 confirmed exoplanets and thousands of TCEs still awaiting classification [32, 95].

As in the case of *Kepler*, light curves are derived from the integrated photometric flux within optimal apertures computed by the Science Processing Operations Center (SPOC) [42], which systematically corrects for background variations, instrumental noise, and pointing drifts. Details on how TESS processes its light curves are provided in Section 2.2 of Paper 2.

The ESA's PLAnetary Transits and Oscillation of stars mission

ESA's PLATO mission [69] is expected to be launched in December 2026, and represents the next generation of space-based telescopes involved in the discovery and

characterization of extrasolar planets.

To this aim, the spacecraft is equipped with 26 ultrasensitive cameras (81.4-megapixel and 15.0 arcseconds/pixel each) designed to work together and capture the tiniest variations in the intensity of a star's light. As the mission is designed to detect rocky planets orbiting Sun-like stars in the habitable zone, the cameras will stare at the same region of the sky continuously for a minimum of two years. With the combination of data from such a large number of cameras, PLATO will provide us with high quality photometry and a total field of view of 2232 square degrees, by far larger than has been possible with previous missions. Images for more than 200,000 target stars will be collected with a sampling rate of 25 seconds, and every 2.5 seconds for the two 'fast' cameras.

2.1.3 Manual classification of transit signals

Once new TCEs are detected by a telescope, the next step consists in understanding whether they originate from planets transiting the target star or from other sources.

Traditionally, this detection and a first automated vetting rely on algorithms such as Box Least Squares (BLS) [48] and Transit Least Squares (TLS) [37]. BLS is highly effective at identifying periodic box-like signals, making it a standard for detecting gas giants. TLS improves upon this by incorporating stellar limb-darkening profiles, significantly increasing the detection sensitivity for smaller, Earth-sized planets. However, these methods are essentially template-matching techniques: their performance degrades when dealing with transits whose shape differs from that of the adopted model.

This procedure – known as *vetting* – requires the manual examination of a set of photometric data and diagnostic parameters provided for each TCE [55]. Transit shape, depth, duration, differences between odd and even transits, secondary eclipses, as well as stellar and planetary parameters relevant for TCEs classification are among the set of data examined. As of today, vetting represents the most reliable method for TCEs classification.

Despite its reliability, vetting presents two major drawbacks: slowness and lack of objectivity.
First, the human analysis requires a long period of training before the ability to discriminate between different types of signal is mastered. But some signals might be very difficult to be classified because of the bad quality of data and the lack of sufficient information. As a consequence, even a team of expert vetters can struggle in such analysis, spending from a few minutes up to several days in classifying a single TCE, creating substantial bottleneck in the process. At this pace, even the classification of

a reasonable amount of signals can take weeks. The huge amount of data produced by *Kepler* and TESS emphasized on the unsustainability of a process exclusively relying on visual inspections.
Second, labels provided by humans on these signals might vary depending on subjective factors – level of expertise, attention involved in the process and emotional state – making it difficult to quantify biases introduced in classification. In other words, the same expert might provide different labels on the same signal during different evaluations, thus introducing a degree of uncertainty in TCEs label that is unpredictable.

These limits, combined with the intrinsic rigidity of classical algorithms, stressed the need to deploy automatic approaches based on Deep Learning. Unlike BLS or TLS, which assume a specific transit shape, Deep Learning models can learn to distinguish the subtle morphological differences between a true planetary transit and instrumental artifacts that often mimic the signatures of a transit.

2.1.4 Related Works

This section provides an overview on the evolution of Machine Learning models for the classification of TCEs, starting from the first approaches developed for the analysis of *Kepler* data in 2014. An in-depth discussion of related works can be found in Section 1 of Paper 2 and Section 2 of Paper 3, while detailed performance comparisons with our models are reported in Sections 7.1 (Paper 2) and 6.4 (Paper 3).

The large volume of data produced by the *Kepler* mission made Machine Learning-based vetting approaches an attractive and scalable alternative to visual inspection of TCEs. The first automated classifiers, Autovetter [59] and Robovetter [19], relied on Random Forest (RF) [12] algorithms to perform a binary classification of *Kepler* TCEs into planet candidates and false positives. Scalar features, including planetary and stellar parameters, differences between odd and even transit depths, centroid motion measurements, and differential image statistics, were input into these models for each TCE [2]. The application of these classifiers to *Kepler* data played a key role in producing labelled catalogs of TCEs – such as the Kepler Q1–Q17 Data Releases (DR) 24 and 25 [16, 82] – that later served as the foundation for training more advanced Machine Learning architectures.

Hybrid approaches combining Self-Organizing Maps (SOMs) [47] and RFs were applied on data from K2 and Next Generation Transit Survey missions [3, 4, 90], with the aim of exploiting either the capacity of SOM in representing transit shapes in two-dimensional topological spaces, and the discriminative power of decision trees. Findings from these

[2] The complete list of 114 attributes input into the RF is provided in Section 5 of [16].

works demonstrated that classification accuracy of unseen TCEs benefits when properly merging different techniques.

With the advent of Deep Learning, Astronet [75] was the model marking the turning point in automatic exoplanet candidates detection. Astronet, a Convolutional Neural Network (CNN) [50], was designed to process two representations for each TCE, known as global and local views, consisting in two light curves folded on the TCE's period with different binning size. An example of global and local view is depicted in Figure 12.1 of Paper 1. Based on the promising results achieved by Astronet (∼95% of recall measured on a very small set of planets from Kepler Q1-Q17 DR 24), CNNs became the standard classification model with several works showing their application on data from different surveys [20, 94, 67].

Further implementation were based on including additional stellar and planetary features as input for the network, with the aim of training the models on the knowledge useful for humans in TCEs classification [2, 66]. This issue is detailed in Section 2.2.2.

To date, the most advanced classifiers are ExoMiner [83], developed on *Kepler* data, and Astronet-Triage-v2 [81], optimized for TESS. Both CNN-based architectures are wide and deep, capable of simultaneously processing multiple representations of TCEs. While these models significantly improved classification accuracy, they also introduced increasing architectural and computational complexity, resulting in substantially longer code development, training, and inference times. Detailed comparisons between the performance of these models and ours are provided in Paper 2 and Paper 3.

In parallel, research efforts have also moved toward improving model interpretability. A notable contribution in this direction is provided by [72, 73], who proposed a Transformer-based model [85] for the binary classification of TESS TCEs, designed to process simultaneously global and local views, centroid information, and astrophysical features.

Overall, the transition from manual to automated vetting has drastically enhanced the speed, objectivity, and reproducibility of classifications, while reducing dependence on human judgment. Today, research aims not only to further improve accuracy but also to develop models that are lighter, more interpretable, and generalizable – capable of adapting to multiple missions and operating effectively in heterogeneous and noisy data environments.

2.1.5 Main catalogs of Threshold Crossing Events

This section introduces the main catalogs of TCEs used in this work, highlighting the distinction between official mission releases and literature-based datasets. Section 3 of

Paper 2 and Section 3.1.2 of Paper 3 provide quantitative information for each of these catalogs.

In the development of supervised Machine Learning classifiers for exoplanet candidates identification, training and testing data are typically obtained by combining stellar light curves with information extracted from catalogs of TCEs (as described in Section 3 of [75]). For a given TCE, the global and local views are generated by folding and binning the corresponding light curve according to the orbital parameters provided in the catalog. This data preparation process, common to most models operating on photometric data, is described in detail in Section 3.1.3 of Paper 3 and references therein.

Among the available datasets, the Kepler Q1–Q17 DR24 and DR25 catalogs[3] have served as the reference standard for training Deep Learning models such as Astronet and ExoMiner, respectively. These catalogs contain, for each TCE, detailed information including orbital parameters, stellar characteristics, and a classification label. Both DR24 and DR25 are considered *official* catalogs, as their TCEs were analyzed and validated over several years by multiple research teams within the *Kepler* collaboration. The Exoplanet Follow-up Observing Program (ExoFOP) [1] catalog is the analogous case for the TESS mission. For this reason, the associated labels are regarded as highly reliable, with minimal classification uncertainty.

In contrast, several studies have employed *non-official* or *literature-based* catalogs, i.e., collections of TCEs labeled by the same research group publishing the analysis. An example is provided by [81], where the authors manually inspected and labeled the TESS TCEs used in their study. As reported in their Section 2.4, this process required more than two years, underscoring the time-intensive nature of human vetting, as we mentioned in Section 2.1.3.

Considering the overall goal of our work of developing models capable of classifying data from multiple missions, we relied on both types of datasets: official archives such as Kepler Q1-Q17 DR24 and DR25, and literature-based catalogs such as that of [81]. This approach ensured the robustness of the models across surveys characterized by different data quality, noise properties, and labeling methodologies.

[3]Given that many TCEs in these two datasets refer to the same planetary candidates, we opted in our works to conduct independent training and evaluation phases for Kepler DR24 and DR25. This approach ensures that the performance metrics of our models remain unbiased by any overlap between the two data releases.

2.2 Aims

In developing our Machine Learning models for exoplanet candidates detection, we focused on designing simple network architectures. This design choice is in line with the *Occam's razor* heuristic (William of Occam, 1285–1349), stating that we should prefer simpler models to more complex models, and that this preference should be traded off against the extent to which the models fit the data [7].

This approach finds wide application in the field of Machine Learning, where numerous studies have demonstrated that less complex models can ensure better *generalization* capabilities. With generalization, we will refer to the model's performance when applied in inference mode to data not seen during training.

The remainder of this section is organized as follows. Section 2.2.1 provides the definition of *model complexity*, while Section 2.2.2 explains the rationale behind the choice of progressively developing models with reduced complexity.

2.2.1 Definition of model complexity

The concept of model complexity has been extensively studied since the 1990s, initially with reference to classical models such as decision trees [9, 14, 93], and has received increasing attention in the context of Deep Learning [5, 68, 53]. To date, it remains one of the central challenges in the field, yet it lacks a universally accepted definition, as complexity strongly depends on the nature of the model under consideration. For instance, the complexity of a decision tree is commonly associated with the number of levels and leaves, whereas such an analysis cannot be directly extended to Deep Learning models, whose representations are more abstract and distributed over several architectures.

In [40], model complexity is defined as the complexity of the distributions that can be represented by Deep Learning models with a given parametrization. The complexity of a deep neural network is generally correlated with the number of trainable parameters[4], which is determined by several factors:

- **Model architecture.** The network architecture is the primary determinant of model complexity. Depth (i.e., the number of hidden layers) and width (i.e., the number of channels or neurons per layer) have a direct impact on the total number of trainable parameters. In CNNs, both the number and the size of convolutional filters contribute significantly to complexity, as they represent additional parameters to be learned during training.

[4] In this thesis, we use the terms *model complexity* and *number of trainable parameters* interchangeably, as they are closely related.

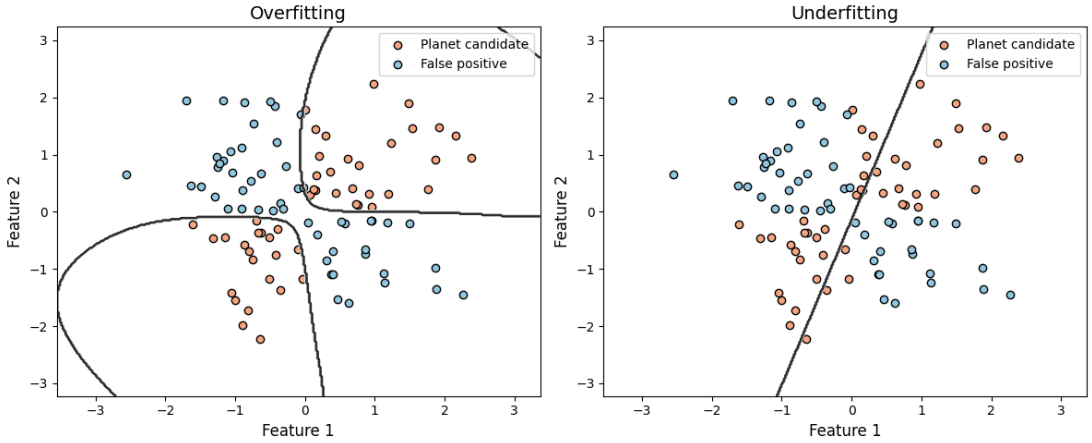

Figure 2.2: A set of simulated global views projected onto a two-dimensional space. The data are fitted with a Support Vector Machine with radial basis function kernel and linear kernel to reproduce a scenario of overfitting (left panel) and underfitting (right panel), respectively.

- **Data complexity.** The input dimensionality directly affects model complexity. When dealing with high-dimensional data, the number of neurons in the input layer increases accordingly, leading to a higher total number of parameters to optimize.

Models with an excessive number of parameters tend to *overfit* the training data, capturing noise and thus degrading performance on the test set. Conversely, simple models suffer from *underfitting*, being unable to capture discriminative patterns within the data. An example of overfitting and underfitting scenarios is provided in Figure 2.2. According to [7], to maximize generalization performance model complexity should ideally lie in an intermediate regime, where both bias and variance are jointly minimized.

In summary, in the design of our models, we focused on minimizing both architectural and data complexity to mitigate the risk of overfitting and underfitting scenarios. Details are provided in the next section.

2.2.2 Rationale behind minimizing model complexity

The reasons behind the decision to focus on minimizing model complexity are twofold: technical and application-related.

1. Technical aspect. From a technical standpoint, the need arose after observing that the development of models for classifying TCEs was evolving toward an architectural

extension in width, depth, and resolution[5].

Following the methodology proposed by [2], many studies have found that including additional features — such as stellar and planetary parameters or multiple representations of transits — can improve the classification capabilities of CNNs. At the same time, [80] demonstrated that for an increase not properly scaled in CNNs complexity in depth, width, and/or resolution, classification performance grows logarithmically until it reaches saturation[6]. In this regard, [87, 88] proved that CNNs for exoplanetary candidates detection with a single input view perform on par with their most complex counterparts.

The adoption of the approach introduced by [2] has led to the inclusion of an increasing number of features, often without an adequate feature selection phase [33]. The result has been a significant increase in data dimensionality, with marginal benefits in terms of accuracy and a higher risk of overfitting.

An increase in network width limits the potential extension of the model to data from different missions, forcing researchers to develop new mission-specific architectures.

For example, Exominer processes six representations of *Kepler* TCEs as input, along with stellar, planetary, and diagnostic parameters. In Section 10 of [83], the authors present the extension of their model to TESS data. However, this probably required substantial modifications, such as writing a new preprocessing pipeline to read and adapt the new data to the *Kepler* scale. The model architecture was also modified. The result was a decline in performance measured on the test set. Recall, the parameter measuring the fraction of correctly classified planets, declined from 93% to 73%.

While increasing the depth of the network by adding multiple fully-connected layers can enhance its expressivity, it also increases complexity and the risk of overfitting. In this work, a single layer for classification was preferred, as the high-level representations learned by the preceding convolutional layers proved sufficient for the task. Indeed, experimental evidence demonstrated a clear separability among the extracted feature vectors according to their class membership, effectively simplifying the subsequent classification process. This will be further discussed in Paper 3 through Figure 4 and Appendix A.1, Figure A.4.

Typically, such an extension of the architectural depth significantly contributes to an increase in the complexity of the model. This is because fully-connected layers hold most of the parameters of a neural network. If a network includes H fully-connected layers with n neurons each, the number of connections — i.e., parameters — is equal

[5] According to [80], with resolution we refer to the input size.
[6] This saturation of accuracy can be seen in Figure 3 of [80].

to $(H-1)\cdot n^2$. This implies that, even with a minimal increase in depth, the total number of parameters grows quadratically.

In summary, increasing the depth and/or width of a model results in a corresponding rise in the number of trainable parameters. In this research field, where large datasets are often unavailable (the largest publicly available dataset of labelled TCEs is the Kepler Q1-Q17 DR25, containing ∼35,000 samples), excessively large models trained on relatively small datasets are prone to poor generalization performance on unseen data.

2. Implementation and Usability. From a practical standpoint, focusing on minimizing model complexity is of critical importance, as it enables us to provide the scientific community with a product that can be executed directly on a standard laptop.

High-Performance Computing (HPC) infrastructures are often not easily accessible or user-friendly for non-computer-science communities: the possibility to perform analysis in local environments represents a significant advantage. The reduced number of parameters in our models ensures that both training and inference processes can be completed within very short timescales. In Table A.1, we provide quantitative estimates of these processes. Also, saving model weights locally poses no risk of storage saturation: while state-of-the-art architectures often require several gigabytes of disk space (e.g., ∼2.5 GB for a 660M parameter model like ConvNeXt V2-H [92]), our optimized model requires only a few megabytes (e.g., ∼2 MB for ∼0.5M parameters), facilitating effortless distribution and integration into portable scientific pipelines.

In conclusion, based on the findings of [80] and [87, 88], we adopted a model design strategy focused on processing exclusively photometric input data. This approach allowed us to easily combine datasets from the *Kepler* and TESS missions and to enhance the generalization capability of a single network architecture able to classify TCEs from multiple space missions. These design choices also led to models that are straightforward to use and highly practical for the broader scientific community, enabling reliable analysis to be performed efficiently even on personal computers without the need for HPC infrastructure.

2.3 Results for Exoplanets

This section aims to introduce the models presented in my first-author peer review papers, reported in the following three sections, and to outline the architectural choices made during their development. The rationale behind each design decision is hereafter discussed, providing insight into the evolution of the proposed approaches.

2.3.1 "Identifying exoplanets in TESS data by Deep Learning" [26].

To develop this model, published in one of my first-author papers ([26], Paper 1, attached after this section), we used Astronet as base network: this architecture was the standard for binary classification of TCEs.

Analyzing Astronet architecture for improvements, we noted it was based on VGG-19 [76], a CNN widely used in several research topics ranging from computer vision to speech recognition [89]. In fact, Astronet is equipped with convolution blocks each with pairs of convolutional layers alternated by pooling layers. The Rectified Linear Units (ReLU) [38] activation function processes the output of each convolution. The main advantage of this configuration is in the consecutive application of two non-linear rectification layers, which makes the decision function more expressive and discriminative than using a single convolutional layer. This concept is detailed in Section 2.3 of [76].

Following the methodology of [75], we exclusively input into the network phase-folded light curves, i.e., the global and local views (depicted in Figure 12.1, Paper 1). These views consist in two representations of the TCE folded on its period, differing in bin size each other. As mentioned in Section 2.2.2, this choice ensures greater extensibility of the model to data from different missions.

Conversely to Astronet, our aim was to deploy a model able to process either *Kepler* and TESS data without any architectural modification. Because catalogs of labelled TESS TCEs were not large enough to train a Deep Learning model on, we used data from Kepler Q1-Q17 DR24. To mitigate the risk of overfitting on *Kepler* data, we employed a Transfer Learning approach with regularizing effect. More precisely, we pre-trained our model on Kepler Q1-Q17 DR24 TCEs and fine-tuned the classification layer on the small TESS dataset available. The goal was to improve predictive performance on unseen TESS TCEs. We used a Transfer Learning [86] approach to avoid generating TESS simulated data. Our findings revealed a boost in classification accuracy on TESS TCEs in a previous work employing such an approach [25].

Since both *Kepler* and TESS catalogs of TCEs were strongly imbalanced towards the

false positive class, we used class weighting to address this issue. To further improve generalization, we equipped the model with Dropout [77] and Batch Normalization [41] layers.

Astronet was trained with Crossvalidation [28, 63, 78] to ensure a robust generalization. Our choice to use Dropout allowed us to achieve the same goal, but with a single trained model, thus saving training and inference times, as well as memory consumption. The advantages resulting from the use of Dropout are detailed in Section 1.1(a) and Section 5.3 of Paper 2.

Conversely to the base network, we used a single fully-connected layer for classification, obtaining a reduction in number of trainable weights and mitigating overfitting risks. The resulting network architecture we used in this work is shown in Figure 12.2 of Paper 1.

With these optimizations, our model achieved a recall of 81% and precision of 87% on a test set of more than 5000 TESS TCEs.

Chapter 12
Identifying Exoplanets in TESS Data by Deep Learning

Stefano Fiscale, Laura Inno, Angelo Ciaramella, Alessio Ferone, Alessandra Rotundi, Pasquale De Luca, Ardelio Galletti, Livia Marcellino, and Giovanni Covone

Abstract Deep Learning algorithms are currently widely used in astronomy for providing accurate predictions in classification problems, and more specifically to the search for planets outside the Solar System, called "exoplanets" (e.g., [16] and references therein). For instance, NASA's Transiting Exoplanet Survey Satellite (TESS) has already observed 76% of the sky for over 3 years, producing millions of light curves but only ~ 221 planets have been confirmed to date. That's because the light curves need to be individually inspected by humans in order to validate the pres-

S. Fiscale (✉) · L. Inno · A. Ciaramella · A. Ferone · A. Rotundi · P. De Luca · A. Galletti · L. Marcellino
Department Science and Technology, University of Naples Parthenope, Centro Direzionale di Napoli, Naples 80143, Italy
e-mail: stefano.fiscale001@studenti.uniparthenope.it

L. Inno
e-mail: laura.inno@uniparthenope.it

A. Ciaramella
e-mail: angelo.ciaramella@uniparthenope.it

A. Ferone
e-mail: alessio.ferone@uniparthenope.it

A. Rotundi
e-mail: alessandra.rotundi@uniparthenope.it

P. De Luca
e-mail: pasquale.deluca@uniparthenope.it

A. Galletti
e-mail: ardelio.galletti@uniparthenope.it

L. Marcellino
e-mail: livia.marcellino@uniparthenope.it

L. Inno · G. Covone
INAF, Osservatorio Astronomico di Capodimonte, Salita Moraliello, 16, Naples 80131, Italy
e-mail: giovanni.covone@unina.it

G. Covone
Department of Physics Ettore Pancini, University of Naples Federico II, Naples, Italy

A. Esposito et al. (eds.), *Applications of Artificial Intelligence and Neural Systems to Data Science*, Smart Innovation, Systems and Technologies 360,
https://doi.org/10.1007/978-981-99-3592-5_12

ence of planetary transits in them. To remove the human-interaction element that is slowing down the process, we developed a Convolutional Neural Network (CNN) that can automatically identify the exoplanets from TESS light curves. Given the scarcity of labeled TESS light curves, we use Transfer Learning (TL) to train our model on a more robust and larger training set, i.e., the ∼15,700 labeled light curves from the NASA's Kepler survey. We then used the learned representation as basic knowledge and fine-tuned the CNN upper layers by making them task dependent on the TESS labeled samples. Moreover, we use the dropout and adaptive reparameterization techniques to prevent our model from overfitting and approximately combining exponentially many different neural network architectures efficiently. We show that the use of TL in parallel with regularization techniques significantly improves the performance of the net, which achieves a precision of 87% and a recall of 81%, thus outperforming similar algorithms already available in the literature (see, e.g., [15, 21]).

12.1 Introduction

The research of new planets outside the Solar System is one of the most challenging and active research fields of science. Recently, modern space-based telescopes like NASA's Kepler and Transiting Exoplanet Survey Satellite (TESS) have significantly revolutionized this research field by allowing us to discover thousands of new planets by searching the overwhelming volume of data they collected. In fact, these telescopes constantly monitor the luminosity of stars in a given field of view, looking for a decrease in its observed flux. Periodic decreases, called "events", can be produced by different phenomena, including an object transiting in front of that star (as seen from the Earth). We define a planet candidate (PC) an event caused by a planet transiting in front of the star; with eclipsing binaries (EB) one caused by a multiple stars system, in which one of the companions in transiting in front of the other; and with non-transiting phenomenon (NTP) all events that are not produced by an astrophysical source, such as instrumental noise and artifacts. All events detected in Kepler and TESS light curves are indicated as Kepler Object of Interest (KOI) and TESS Object of Interest (TOI), respectively. When such events are detected in a photometric sequence, assessments need to be implemented in order to determine their nature as PC, EB, or NTP. Typically, the assessments are done by a team of expert astronomers visually inspecting the data and voting for a final disposition in what is known as the "vetting process". However, these visual investigations present the following limitations: (1) they are highly time-consuming, requiring from a few hours up to several weeks for event; (2) they are prone to subjectivity as different astronomers could make different dispositions; (3) they are challenging to be performed on the light curves of very bright stars because of the low contrast. In order to overcome the above limitations, automatic and objective analytic tools are needed. Indeed, Machine Learning (ML) methods have proved to be an effective alternative. Several algorithms have been proposed in this context [5, 15, 16, 21]. Here we present

a Convolutional Neural Network (CNN) that we developed to perform a binary classification of all events detected in TESS data in PC and not-planet candidate (NPC, including EB and NTP).

The paper is organized as follows: in Sect. 12.2 we describe the creation of the dataset used in this study; Sect. 12.3 describes the architecture and training specifics of our neural network; in Sect. 12.4 the model generalization performance is evaluated over a test set of events; conclusions are drawn in Sect. 12.5.

12.2 Building the Dataset

We want to build a CNN that can automatically vet TESS data by providing a disposition for each TOI as a PC or NPC. This means that we need to work on pre-processed light curves, in which events have been already identified and characterized in terms of their period, epoch and duration. These events are called Threshold Crossing Events (TCE) and listed in the Mikulski Archive for Space Telescopes (MAST[1]). However, since ML algorithms require a large amount of training data, we cannot build a large enough training set by only using the labeled dataset currently available for TESS ($\mathcal{D}_{T1}$), which consists of ∼5,500 TCEs [1]. Besides this dataset provided, there is another TESS labeled dataset available in literature,$\mathcal{D}_{T2}$, assembled by [21], including an additional ∼15, 000 TCEs. We decided to use $\mathcal{D}_{T1}$ as our independent test set, and $\mathcal{D}_{T2}$ for the model fine-tuning. The actual training is instead carried out by using the TCEs from Kepler DR24 [4], $\mathcal{D}_K$, which contains over 15,700 labeled events that have been cross-validated by different research groups over the past years [12, 13]. We then use Transfer Learning (TL) in order to apply the model trained on this dataset to TESS data, and, more specifically, we to let our model learn the target conditional probability distribution $P(\mathbf{Y}_\mathcal{T} \mid \mathbf{X}_\mathcal{T})$ in $\mathcal{D}_\mathcal{T} = \{\mathbf{X}_\mathcal{T}, \mathbf{Y}_\mathcal{T}\}$, i.e., the overall set of TESS data ($\mathcal{D}_{T1} \cup \mathcal{D}_{T2} \subset \mathcal{D}_\mathcal{T}$), where $\mathbf{X}_\mathcal{T}$ is the set of TCEs related to the TOIs and $\mathbf{Y}_\mathcal{T}$ is the set of the dispositions assigned to each TOI.

However, the marginal probability distributions of Kepler and TESS domains are different:

$$P(\mathbf{X}_K) \neq P(\mathbf{X}_\mathcal{T}), \tag{12.1}$$

as they are two different instruments. To overcome this issue, we manipulated the light curves in a common format. The light curve corresponding to each event is folded with the transit event positioned at the center, at phase equal zero (as the light curve is shown in the range of phases $= [-0.5, 0.5]$) and rescaled to have mean equal zero and standard deviation equal to 1. Finally, for each of these light curves, we generate two binned representations: a global view, with length of 201 bins, which shows the transit event on the entire observing window, and a local view with length of 61 bins, showing a zoomed-in representation of the transit event. These two binned phase-folded views represent the input of our model (Fig. 12.1). Due to the

[1] https://archive.stsci.edu/.

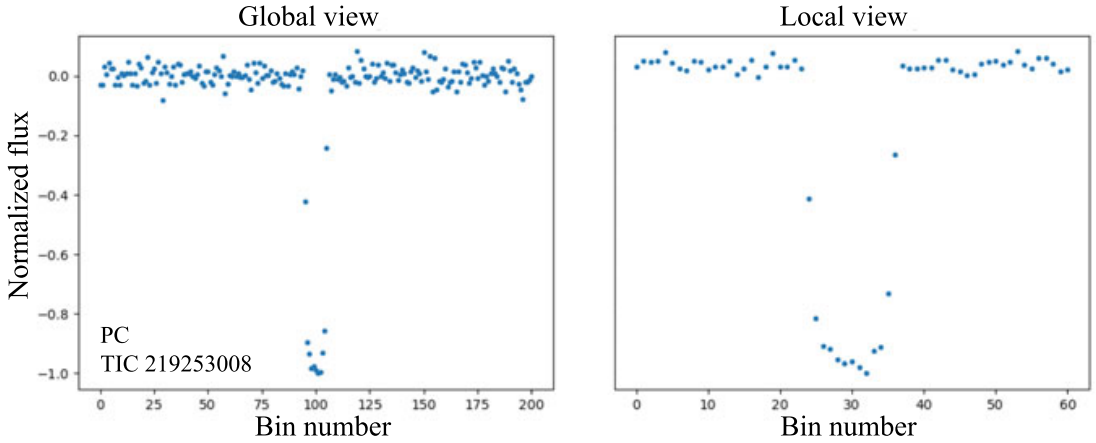

Fig. 12.1 Our input representations. The global view (left panel) describes the transit event on its entire orbital period. The local view (right panel) is a close-up vision of the transit event

large number of light curves to process, thanks to the involving of high-performance architectures used in different research fields [6, 7], we also deployed a GPU-based parallel algorithm to reduce the computational time for the pre-processing [9].

12.3 The Convolutional Neural Network Architecture

We defined the structure of our network based on the work proposed by [17]. The key idea is to deploy a deeper model while reducing the number of the total parameters. As suggested in [17], we used several pairs of convolutional layers with a number of filters increasing by power of two. The ReLU is applied after each convolutional layer. The application of these two non-linearity allows us to detect more complex features with respect to a single bigger filter. The filter's dimension of the convolutional layers is set to 5×5. We determined this value experimentally. The pairs of convolutional layers are alternated by a pooling layer with fixed stride of two. The underlined implementation details aim to minimize the number of parameters of the network. However, the network can still be sensitive to overfitting. To prevent our model from overfitting, we introduce the Dropout and Batch Normalization [11, 18]. Specifically, we apply the Dropout after each convolutional layer in order to randomly drop the features map and to promote independence between them. The batch normalization is applied after each pooling layer. We will refer to *convolution block* as the block containing two convolutional layers alternated with dropout which ends with pooling and batch normalization. This approach is applied to both input views. Specifically, the two input views are processed through disjoint convolutional blocks in order to extract and shrunk the main features. The learned features are merged and classified in the dense block. The *dense* block takes as input the flattened features and performs the classification task by means of a single fully connected layer. We randomly drop the units of this layer and finally apply the batch normalization. The output layer is

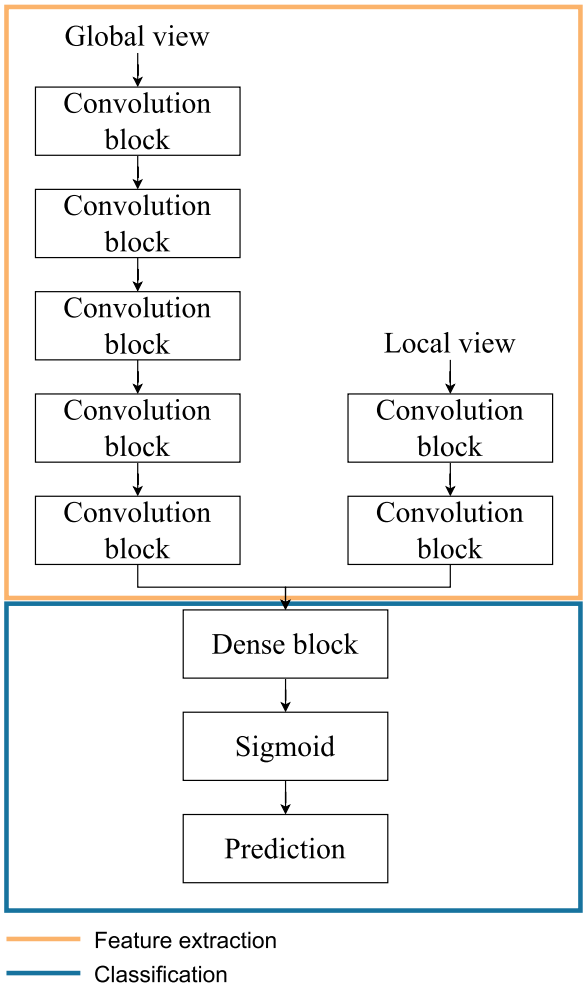

Fig. 12.2 Our CNN architecture. We separately process the two input views to perform feature extraction on each of them (orange panel). We flattened the computed features thus classifying them by means of a multi-layer perceptron (blue panel)

composed of a single neuron. We used the sigmoid activation function on this layer to obtain a result in [0, 1] that represents a probabilistic value for which the input event belongs to each class.

The full network architecture is shown in Fig. 12.2.

12.3.1 Training

During training, the parameters of our model are adjusted to minimize the cost function. We used the Adam optimization algorithm to minimize the binary cross-entropy over the training and validation set. We trained our model for 10 epochs on the Kepler DR24 dataset with a batch size of 64. With this dataset, we allow our network to acquire a robust knowledge through the training epochs. The use of dropout allows us to perform a single network training process. We initially set the learning rate $\alpha = 10^{-5}$. However, dropout introduces a significant amount of noise in the gradients compared to standard stochastic gradient descent [18]. Therefore, a lot of gradients tend to cancel each other. In order to make up for this, we tried values of

α 10–100 times bigger. The value $\alpha = 10^{-3}$ turned out to be the best for us. Another way to reduce the effect of the dropout noise is to use a high momentum. We set the two momentum of the Adam algorithm to 0.9 and 0.999. Through experimental tests we found the best value for the dropout rate $p = 0.2$. We noticed that by increasing the value of p to 0.5 the model performance becomes worse.

12.3.2 Fine-Tuning

The application of TL to our model can be done by following two different approaches: The first is to fix the parameters our network learned from $\mathcal{D}_K$ as basic knowledge and then fine-tune the entire model on the $\mathcal{D}_{T2}$ training set (model A). The second is to perform the fine-tuning only on the classification block (model B). We performed several tests (see next Section) and found that this second approach is preferable, as it allows the net to achieve a better performance. Therefore, once we trained the model on Kepler DR24, we fine-tuned the parameters of the network on $\mathcal{D}_{T2}$. We freeze the parameters of the feature extraction block (Fig. 12.2) in order to keep the knowledge that our model learned on Kepler DR24. We make sure to set this block in inference mode to avoid dropout omits some feature maps [10]. By doing so, we also force the batch normalization layers to not update their batch statistics. This prevents the batch normalization layers from undoing all the training we have done so far. We then run a 10 epochs training on the classification block of our model. When adapting the dense block parameters we reduced the learning rate α from 10^{-3} to 10^{-5}.

12.4 Evaluation of the CNN Performance

We evaluate the performance of our network on a test set, which consists in a new set of TOIs that have not been used for training: $\mathcal{D}_{TEST}$, which includes $\mathcal{D}_{T1}$ and a subset of data from [21] randomly excluded from $\mathcal{D}_{T2}$, for a total of 3, 404 PCs and 1, 867 NPCs. In order to assess the performance of the CNN, we estimated the following metrics by applying the model to $\mathcal{D}_{TEST}$: confusion matrix, i.e., the number of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) identified by the model, precision, recall, and F-score. In Table 12.1, we show the results obtained by a model only trained on $\mathcal{D}_{T2}$, (model A) and then we compare them with the ones obtained by the same network architecture but trained with TL (models B and C). Finally, we also compare the results of our models with the ones of *Astronet* [21], which is currently the best vetting model available in the literature. More details are given in the next subsection.

Table 12.1 Performance comparison between (A) the model without TL, (B) the model completely fine-tuned, (C) the model for which we fine-tuned the dense block and *Astronet*, the best vetting model currently available in the literature [21]

Model	TP	TN	FP	FN	Precision	Recall	F-score	Accuracy
A	0.76	0.77	0.22	0.23	0.83	0.82	0.82	0.77
B	0.72	0.79	0.20	0.27	0.86	0.72	0.78	0.75
C	0.81	0.78	0.21	0.19	0.87	0.81	0.84	0.80
Astronet*	N.A.	N.A.	N.A.	N.A.	0.65	0.45	0.53	N.A.

*Note that this model has been tested on a different dataset

12.4.1 TL Applied to the Entire Model Versus Only on the Classification Layer

We compare the two different approaches described in the previous section by analyzing the network performance on the test set. In the first case, the fitted parameters that our network learned from $\mathcal{D}_K$ are used as basic knowledge. We run a 10 epochs training process on the entire model with the $\mathcal{D}_{T2}$ training set. The learning rate has been set to $\alpha = 10^{-3}$. The performance of this model (B) is summarized in the second row of Table 12.1.

Data listed in Table 12.1 for model B shows that, despite the good parameters' initialization obtained from training on $\mathcal{D}_K$, its performance deteriorates. This can be explained by the fact that some of the learned knowledge is lost during the fine-tuning, making the CNN more inclined to classify everything as NPC. More specifically, when running a new training, the model starts exploring the optimization surface looking for a deeper local minimum and can move away from the one already reached when training on $\mathcal{D}_K$. Moreover, the new training process forces the batch normalization layers to update their mean and variance statistics. By doing so, the knowledge they had gained is disrupted. As a result, this model finds a high fractions of TN and FN, and has lower values of recall and F-score when compared with model C, which is the one fine-tuned only for the classification block. In fact, model C finds a higher fractions of TP and TN, thus showing a better capability in distinguish PCs from NPCs. Moreover, training only the classification block makes the optimization process computationally less expensive, as the number of parameters to adapt decreases from ∼1,927,000 to ∼1,256,000.

Finally, if we compare the precision, recall, and F-score of our best model (C) with the ones of *Astronet* when choosing the same value for the threshold score, we note that our model shows an improvement of almost a factor two for each metric.

12.5 Conclusions

In this work, we present a CNN that performs a fast and accurate classification of TESS TCEs in planet candidates and non-planet candidates. In order to overcome the problem of the limited size of TESS training data, we used domain adaption techniques, and, specifically, Transfer Learning. This technique allowed us to exploit additional data from a different telescope, Kepler, and double the training set cardinality, from ∼15,000 to ∼30,000 labeled samples. We find that the model achieves the best performance when trained on Kepler data and fine-tuned on TESS data only for the classification block. The fact that a Transfer-Learned model performs better than a model only trained on TESS data can be explained by considering both the larger number of examples from which the net can learn and the higher quality of Kepler's photometric data with respect to TESS. However, we find that this improvement is maximally capitalized when the TL is applied only on the classification block, as this will allow the net to preserve the knowledge already acquired in the first step of the training, while also reducing the number of parameters to be optimized. Moreover, by adding the Dropout and Batch Normalization in both the convolutional and the dense blocks, we improved the model generalization capabilities. By comparing the performance of our best model with a similar model available in the literature, we find that it achieves significant better values of precision and recall. This means that our model can provide a valuable aid in the search of new planets by vetting thousands of new TCEs in just a few seconds. In the next future, the authors will explore and compare feature selection [14], dimensionality reduction [8], Bayesian framework [3]), and data integration methodologies [2] for a more robust characterization of the light curves. We will also investigate the possibility to move to more novel neural network models, such as Transformer [20], which have the additional advantage of being explainable (see, e.g., [19]).

Concluding, our experimental results show that TL can be very efficient in improving the network performance in scientific problems in which the number of examples for the training is limited, as ours.

References

1. Akeson, R., Christiansen, J.: TESS follow-up observing program working group (TFOP WG): the ExoFOP-TESS website. In: American Astronomical Society Meeting Abstracts# 233, vol. 233, pp. 140-09 (2019)
2. Ciaramella, A., Nardone, D., Staiano, A.: Data integration by fuzzy similarity-based hierarchical clustering. In: BMC Bioinformatics, vol. 21 (2020)
3. Ciaramella, A., Staiano, A., Cervone, G., Alessandrini, S.A.: A Bayesian-based neural network model for solar photovoltaic power forecasting. Smart Innov. Syst. Technol. **54**, 169–177 (2016)
4. Coughlin, J.L., Mullally, F., Thompson, S.E., Rowe, J.F., Burke, C.J., Latham, D.W., Zamudio, K.A.: Planetary candidates observed by Kepler. VII. The first fully uniform catalog based on the entire 48-month data set (Q1-Q17 DR24). Astrophys. J. Suppl. Ser. **224**(1), 12 (2016)

5. Dattilo, A., Vanderburg, A., Shallue, C.J., Mayo, A.W., Berlind, P., Bieryla, A., Yu, L.: Identifying exoplanets with deep learning. ii. two new super-earths uncovered by a neural network in k2 data. Astron. J. **157**(5), 169 (2019)
6. De Luca, P., Formisano, A.: Haptic data accelerated prediction via multicore implementation. In: Arai, K., Kapoor, S., Bhatia, R. (eds.), Intelligent Computing. SAI 2020. Advances in Intelligent Systems and Computing, vol. 1228. Springer, Cham (2020)
7. De Luca, P., Galletti, A., Marcellino, L.: Parallel solvers comparison for an inverse problem in fractional calculus. In: 2020 Proceeding of 9th International Conference on Theory and Practice in Modern Computing (TPMC 2020), pp. 197–204 (2020)
8. Ferone, A., Petrosino, A.: A rough fuzzy perspective to dimensionality reduction. In: Lecture Notes in Computer Science, vol. 7627. Springer, Berlin (2015)
9. Fiscale, S., Luca, P. D., Inno, L., Marcellino, L., Galletti, A., Rotundi, A., Quintana, E.: A GPU algorithm for outliers detection in TESS light curves. In: International Conference on Computational Science, pp. 420–432. Springer, Cham (2021)
10. https://keras.io/guides/transfer_learning/
11. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456. PMLR (2015)
12. Morton, T.D., Bryson, S.T., Coughlin, J.L., Rowe, J.F., Ravichandran, G., Petigura, E.A., Batalha, N.M.: False positive probabilities for all Kepler objects of interest: 1284 newly validated planets and 428 likely false positives. Astron. J. **822**(2), 86 (2016)
13. Mullally, F., Coughlin, J.L., Thompson, S.E., Christiansen, J., Burke, C., Clarke, B.D., Haas, M.R.: Identifying false alarms in the Kepler planet candidate catalog. Publ. Astron. Soc. Pac. **128**(965), 074502 (2016)
14. Nardone, D., Ciaramella, A., Staiano, A.: A sparse-modeling based approach for class specific feature selection. Peer J. Comput. Sci. **5**, 1–25 (2019)
15. Osborn, H.P., Ansdell, M., Ioannou, Y., Sasdelli, M., Angerhausen, D., Caldwell, D., Smith, J.C.: Rapid classification of TESS planet candidates with convolutional neural networks. Astron. Astrophys. **633**, A53 (2020)
16. Shallue, C.J., Vanderburg, A.: Identifying exoplanets with deep learning: a five-planet resonant chain around kepler-80 and an eighth planet around kepler-90. Astron. J. **155**(2), 94 (2018)
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015 (2014)
18. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
19. Valizadegan, H., Martinho, M.J., Wilkens, L.S., Jenkins, J.M., Smith, J.C., Caldwell, D.A., Oza, N.C.: ExoMiner: a highly accurate and explainable deep learning classifier that validates 301 new exoplanets. Astrophys. J. **926**(2), 120 (2022)
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, p. 30 (2017)
21. Yu, L., Vanderburg, A., Huang, C., Shallue, C.J., Crossfield, I.J., Gaudi, B.S., Quinn, S.N.: Identifying exoplanets with deep learning. III. Automated triage and vetting of TESS candidates. Astron. J. **158**(1), 25 (2019)

2.3.2 "DART-Vetter: A Deep LeARning Tool for automatic triage of exoplanet candidates" [23].

The evolution of our model is described in Paper 2 [23] (attached after this section) with a CNN named DART-Vetter. In this work, we focused on further reducing the model complexity, with modifications on input data and convolutional layers.

Removing the local view. The innovation consists in eliminating the channel processing the local view, preserving the global view as the only input. Previous studies proved the impact of CNNs with a single input view in achieving classification performance on par with two views based CNNs [87, 88].

Removing one convolutional layer from each block. In Paper 1, each convolutional block had a pair of convolution layers. During model selection, we found that performance improved when using a single convolution layer for each block.

Reducing the convolutional filters size. We also reduced the size of the convolutional filters from 5×1 to 3×1. The effectiveness of small convolutional filters was first demonstrated by [18], and later consolidated by [76], who showed that smaller filters – when properly integrated into a given architecture – not only reduce the total number of trainable parameters but also increase the expressive power of the network. The consecutive application of non-linear operators enhances the discriminative capability of the decision function, while the receptive field preserves a size comparable to that achieved using single, larger convolutional filters. The model architecture is illustrated in Figure 4 of Paper 2.

In our case, the use of these smaller filters allows the model to capture more complex relationships within the data compared to larger filters, as they increase sensitivity to local variations in the transit signal. This is a crucial aspect given that, in some cases, the transit may be represented by a single point in the input signal (as shown in Figure 3 of [75]).

Impact of architectural simplification. The removal of the channel responsible for processing the local view, combined with the reduction in filters size, yielded several advantages. The total number of trainable parameters was significantly reduced from ∼1,927,000 to ∼527,000 – corresponding to a 73% decrease. As a result, the risk of overfitting was notably lowered, along with the required training and inference times. A comprehensive overview of the innovations introduced by this work is provided in Section 1.1 of Paper 2.

Dealing with class imbalance. To deal with class imbalance in our dataset, we used

data augmentation and class weighting techniques, as described in Section 4 and in Section 5.3. of Paper 2.

We changed the training strategy from the previous work, training the model on *Kepler* and TESS data simultaneously, rather than adopting a Transfer Learning approach. This strategy resulted in a significant improvement in the model's ability to minimize the fraction of misclassified planets. The recall measured on the test set reached 91%, that is ∼10% higher than the values obtained in the previous study.

Compared to Paper 1, we enhanced classification accuracy across data from different missions with a model showing ∼1,400,000 parameters less. This finding represents a substantial step toward the development of cross-mission deep neural networks grounded in the heuristic principle of Occam's razor.

Figure 2.3 shows a comparison between the two network architectures we presented in Paper 1 and Paper 2; a quantitative analysis is provided in Table 6 of Paper 2.

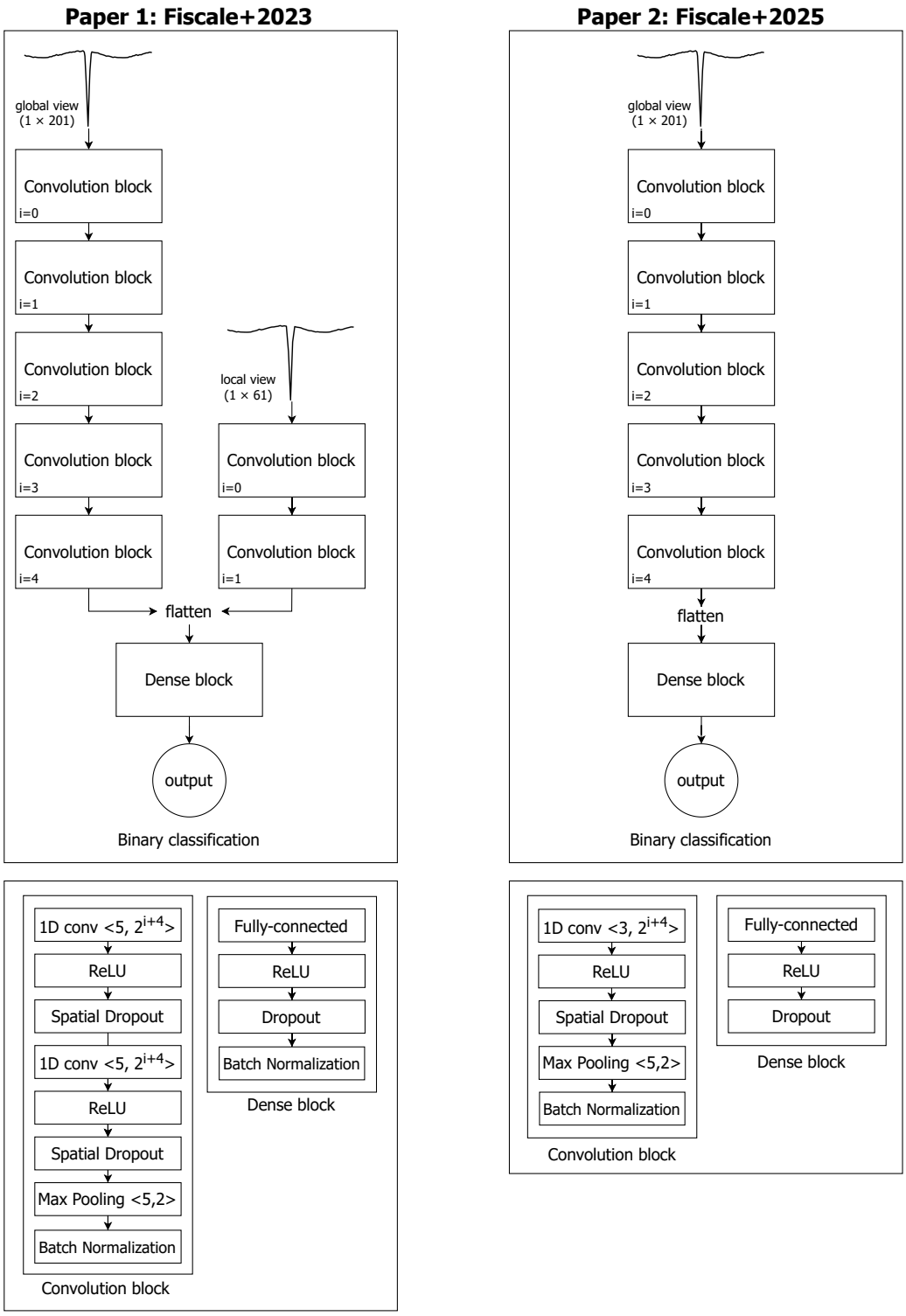

Figure 2.3: Comparison of the network architectures presented in Paper 1 [26] and Paper 2 [23].

The Astronomical Journal, 170:73 (18pp), 2025 August https://doi.org/10.3847/1538-3881/addf4d

DART-Vetter: A Deep Learning Tool for Automatic Triage of Exoplanet Candidates

Stefano Fiscale[1,2,3], Laura Inno[2,3], Alessandra Rotundi[1,2], Angelo Ciaramella[2], Alessio Ferone[2], Christian Magliano[3,4], Luca Cacciapuoti[5], Veselin Kostov[6,7], Elisa V. Quintana[6], Giovanni Covone[3,4,8], Maria Teresa Muscari Tomajoli[1,2], Vito Saggese[4], Luca Tonietti[1,2,3,9], Antonio Vanzanella[10], and Vincenzo Della Corte[3]

[1] UNESCO Chair "Environment, Resources and Sustainable Development," Department of Science and Technology, Parthenope University of Naples, Italy; stefano.fiscale001@studenti.uniparthenope.it
[2] Department of Science and Technology, Parthenope University of Naples, Centro Direzionale di Napoli, Naples, I-80143, Italy
[3] INAF, Osservatorio Astronomico di Capodimonte, Salita Moiariello, 16, Naples, I-80131, Italy
[4] Department of Physics "Ettore Pancini," University of Naples Federico II, Naples, Italy
[5] European Southern Observatory, Karl-Schwarzschild-Strasse 2 D-85748 Garching bei Munchen, Germany
[6] NASA Goddard Space Flight Center, 8800 Greenbelt Road, Greenbelt, MD 20771, USA
[7] Citizen Scientist, Planet Patrol Collaboration, Greenbelt, MD 20771, USA
[8] INFN Section of Naples, Via Cinthia 6, 80126, Napoli, Italy
[9] Department of Biology, Federico II University of Naples, Naples, Italy
[10] National Centre for Nuclear Research, Pasteura 7, 02-093, Warsaw, Poland
Received 2024 June 5; revised 2025 May 2; accepted 2025 May 30; published 2025 July 7

Abstract

In the identification of new planetary candidates in transit surveys, the employment of deep learning models proved to be essential to efficiently analyze a continuously growing volume of photometric observations. To further improve the robustness of these models, it is necessary to exploit the complementarity of data collected from different transit surveys such as NASA's Kepler, Transiting Exoplanet Survey Satellite (TESS), and, in the near future, the ESA Planetary Transits and Oscillation of stars mission. In this work, we present a deep learning model, named `DART-Vetter`, that is able to distinguish planetary candidates from false positives signals detected by any potential transiting survey. `DART-Vetter` is a convolutional neural network that processes only the light curves folded on the period of the relative signal, featuring a simpler and more compact architecture with respect to other triaging and/or vetting models available in the literature. We trained and tested `DART-Vetter` on several data sets of publicly available and homogeneously labelled TESS and Kepler light curves in order to prove the effectiveness of our model. Despite its simplicity, `DART-Vetter` achieves highly competitive triaging performance, with a recall rate of 91% on an ensemble of TESS and Kepler data, when compared to `Exominer` and `Astronet-Triage`. Its compact, open source, and easy to replicate architecture makes `DART-Vetter` a particularly useful tool for automatizing triaging procedures or assisting human vetters, showing a discrete generalization on threshold-crossing events with multiple event statistic > 20 and orbital period < 50 days.

Unified Astronomy Thesaurus concepts: Convolutional neural networks (1938); Astronomy data analysis (1858); Exoplanet detection methods (489)

1. Introduction

Among the various methods used for discovering exoplanets, the transit technique from space telescopes has emerged as the most productive, contributing to the discovery of over three-quarters of the ∼5800 currently confirmed exoplanets.[11] Considering that a little more than 30 years ago no exoplanet had been detected yet, this represents tremendous progress, which has been especially boosted by the two dedicated NASA's space missions: Kepler (D. G. Koch et al. 2010) and its successor the Transiting Exoplanet Survey Satellite (TESS; G. R. Ricker et al. 2014).

Since its launch in 2009 and throughout its nine years in space, Kepler detected more than 34,000 events (J. D. Twicken et al. 2016) in the light curves of over 150,000 stars. Among these events, about 4700 were dispositioned as candidate planets and nearly 2700 of them turned out to be confirmed as planets (see e.g., T. D. Morton et al. 2016; F. Mullally et al. 2016; S. E. Thompson et al. 2018). TESS, launched in 2018, is actively scanning almost the entire sky—divided into 26 sectors—every two years by using a wide-field strategy. Each sector is observed for about 27 days, allowing TESS to capture high-precision brightness measurements of approximately six million stars. As of today, TESS has detected more than 7500 candidate planets, 622 of which have been confirmed as true planets.[12]

This large gap between the amount of collected data and the number of exoplanets found is due to the intervening time-consuming process that allows us to rule out common false positive (FP) scenarios and identify potential exoplanet candidates, called *vetting*. Vetting consists of carefully examining and validating the detected transits to distinguish transiting planets from astrophysical (e.g., eclipsing binaries or stellar variability) or instrumental (e.g., spacecraft momentum dumps) FPs.

While the exact analysis steps needed to promote an observed transit to a candidate and, possibly, a confirmed planet might vary depending on the instrument and detection method adopted, the team performing it, and the specific system under analysis, the basic steps for vetting transit signals

[11] https://exoplanetarchive.ipac.caltech.edu/docs/counts_detail.html

[12] These numbers were retrieved from the NASA Exoplanet Archive (https://exoplanetarchive.ipac.caltech.edu/) on 2025 May 2.

are common. Initially, vetters focus on the light-curve data in which a transit-like feature has been detected, by using all available information (e.g., transit shape, difference between primary and secondary eclipses, etc.) to rule out FPs. At this stage, it is possible to discard clear FPs for which further analysis will not need to be carried out. This operation is often referred to as *triage*.

For candidates surviving triage, mostly transiting planets and eclipsing binaries, additional observations are collected and used to perform an independent validation or clearing out specific FP scenarios. To remove a FPs scenario, such as background eclipsing binaries, a pixel-level analysis—based on the examination of difference images or centroids offset for the target stars—is performed (S. T. Bryson et al. 2013; A. Hadjigeorghiou & D. J. Armstrong 2024). When such ancillary data are not available or not enough, statistical tools, as employed by C. Magliano et al. (2022), can be used to validate transits (T. D. Morton 2015; S. Giacalone et al. 2021). These tools assess the significance of the observed transit signal and provide a quantitative estimation of validation confidence. Finally, the confirmation of the exoplanet identification is achieved through the radial velocity detection of its mass. While the last two steps in the process are strongly human-dependent, the first two, i.e., triage and vetting, can be automatized to some extent and several authors have proposed automated procedures that can support the validation process at different stages.

Initial works focused on analyzing Kepler's data set in order to produce a binary classification of transit events in planet candidates (PCs) and FPs. Noteworthy examples include the decision-tree-based machine learning (ML) codes such as `Autovetter` (S. D. McCauliff et al. 2015), `Robovetter` (J. L. Coughlin et al. 2016), and `SIDRA` (D. Mislis et al. 2016). However, as more and more diverse data were acquired with the advent of new missions, such as K2 (S. B. Howell et al. 2014) and TESS, the complexity of the classification problem grew and deeper model started to become appealing. C. J. Shallue & A. Vanderburg (2018, hereinafter SV18) introduced `Astronet`, a convolutional neural network (CNN) designed for vetting PCs identified by the Kepler mission but potentially applicable to other surveys as well. Indeed, `Astronet` and similar architectures have been applied to new data sets produced by surveys such as K2 (A. Dattilo et al. 2019), WASP (N. Schanche et al. 2019), and NGTS (D. J. Armstrong et al. 2018; A. Chaushev et al. 2019). Adjustments to the methodology, incorporating new input information, and refining data representation (M. Ansdell et al. 2018; M. Jara-Maldonado et al. 2020; H. Valizadegan et al. 2022, 2023; H. Liao et al. 2024) have resulted in enhanced classification performance.

The CNN by SV18 was later customized for TESS by L. Yu et al. (2019, hereinafter YU19). They obtained good performance for one of their trained models specialized on discarding all nonastrophysical signals, called `Astronet-Triage`, which was then incorporated into the TESS Quick-look Pipeline (QLP) since 2019, as outlined by N. M. Guerrero et al. (2021).

Recently, E. Tey et al. (2023, hereinafter TEY23) found that the screening process performed with the YU19 model led to the exclusion of a significant number of potentially valid PCs. Hence, they proposed a new model, called `Astronet-Triage-v2`, which is trained to identify five different classes, including the one of interest, "eclipsing signals," in order to preserve more information for the next steps. To achieve this goal, they had to further increase the complexity of the model, providing seven different representations of the light curve and additional scalar information as input to the network. While this approach produces higher performance, and hence effectively reduces the number of "lost" planets, it has some drawbacks: a) the increased complexity of the model makes it more difficult to apply and test on other data sets, and b) it has a final classification that requires further interpretation or considerations. Concerning the issue of model complexity (X. Hu et al. 2021), recent studies by K. Visser et al. (2022a, 2022b) have shown that increasing the number of trainable parameters of neural networks can actually reduce their predictive performance. The authors of these studies presented `Genesis`, a CNN similar to those presented in SV18 and M. Ansdell et al. (2018) but with much less trainable parameters (or weights), and demonstrated that their simpler model performs as well as its more complex counterparts. This is an interesting notion—reducing the complexity of a deep learning model can improve the generalization capability, prevent overfitting, and is more easily exportable to be verified and used by different teams (S. Fiscale et al. 2025).

In this work, we present a simplified CNN—designed to be easily replicable—to perform the initial sorting of signals into PCs and not-planet candidates (NPCs), named `DART-Vetter`. In fact, based on the results from `Genesis`, we explored the opportunity to further simplify the architecture with respect to state-of-the-art CNNs used to perform triaging and vetting, and found that we can significantly reduce both the input dimensionality (A. Ferone & A. Petrosino 2017) and the number of layers without degrading the model's performance. Overall, `DART-Vetter` shows robust predictive performance on TESS planetary candidates, while good generalization capabilities are achieved on Kepler data sets for threshold-crossing events (TCEs) with multiple event statistic (MES) > 20 and orbital periods < 50 days. The MES estimates the signal-to-noise ratio (S/N) of the signal against the measurement noise.

Our paper is organized as follows. In Section 2, we explain how light-curve data from the Kepler and TESS mission are collected, automatically processed, and searched for potential events, which we use to prepare the input. In Section 3, we detail the transit signals and corresponding light curves used as the training and test sets for our model, `DART-Vetter`, along with the preprocessing steps used to standardize them in Section 4. The architecture of the neural network and the training process are covered in Section 5. We present the quantified results of our classifier in Sections 6 and 7, and discuss the implications in Section 8. Our conclusions are drawn in Section 9.

1.1. Contributions and Novelty of This Work

The approach we present with the deep learning-based model `DART-Vetter` offers several advantages in the field of exoplanets detection, as summarized below:

a. Faster training and TCEs predictions: We extensively employ dropout (N. Srivastava et al. 2014), a regularization technique that not only mitigates overfitting and underfitting issues but also enables the training and

assessment of an exponential number of network architectures in a computationally inexpensive manner. In contrast to other model selection methods, such as *K*-fold cross-validation, dropout allows the model to be trained only once, which provides faster performance and other advantages, as discussed in Section 5.3.

b. Balancing adaptability and reliability: The use of a compact input representation facilitates both data collection and preparation pipelines, and the application of our model on cross-mission data. We demonstrate in Section 7.1 that `DART-Vetter` can achieve competitive triage capabilities on Kepler and TESS data. However, this versatility comes with limitations. In particular, since the model relies on a reduced set of input features, it shows reduced performance when classifying Kepler TCEs with MES < 20 or with orbital periods longer than 50 days, as discussed in Section 6.3.

c. Better generalization with reduced model complexity: With only 527,329 parameters, compared to state-of-the-art models with more than 100 million, `DART-Vetter` is significantly more compact. This reduction in complexity enhances generalization on unseen TCEs, minimizes the risk of overfitting on small data sets, and maintains competitive F1-scores across multiple data sets. We refer the reader to Sections 5 and 7.2 for a more detailed discussion of the importance of reducing model complexity.

d. Multiclass classification capability: Although it was initially designed for binary classification, `DART-Vetter` can be quickly adapted to perform multiclass classification. As detailed in Section 6.2, this expands its applicability beyond binary classification tasks.

These features make `DART-Vetter` a versatile and easily reproducible tool that can ensure a correct minimization of the fraction of misclassified planetary transits.

2. From Light Curves to TCEs

The data used in this work are light curves in which a TCE, i.e., a potential transiting planet signature with a minimum S/N of 7, is detected (J. M. Jenkins et al. 2002; J. L. Christiansen et al. 2012).

In this work, we aim to build a model that is potentially able to automatically classify all TCEs detected in the light curves collected by a generic transiting survey. Since we want to develop a survey independent approach, we adopted all labeled TCEs publicly available from both Kepler and TESS missions in order to obtain the largest and most diverse possible training set. Given that data can be collected and processed differently depending on the survey characteristics and data flow, we summarize here for the reader's convenience how TCEs are detected in both Kepler and TESS data.

2.1. Kepler Data

The Kepler spacecraft collected photometric observations for ∼156,000 target stars at 29.4 minutes cadence (long cadence) and for 512 target stars at a shorter cadence of 58.85 s (D. G. Koch et al. 2010). These observations were divided into quarters, each lasting approximately 93 days. In each quarter, the telescope focused on a specific field of view in the sky, continuously monitoring the brightness of all its target stars within that field.

As described in J. M. Jenkins et al. (2010), the Kepler Science Operation Center (KSOC) pipeline processed long cadence data by performing the following steps: the raw data are calibrated at the pixel levels (removal of CCD bias voltage and cosmic rays, flat field corrections, etc.), and sky background subtracted, hence a simple aperture photometry (SAP) is performed for each target star. The optimal aperture is determined automatically to maximize the S/N for the given star (S. T. Bryson et al. 2010). This process results in the extraction of the stellar flux as a function of time, indicated as the SAP light curve, which is then corrected from systematics and other errors (J. D. Twicken et al. 2010, presearch data conditioning (PDC) SAP light curve) and searched for events by the Transiting Planet Search (TPS) module (J. M. Jenkins et al. 2010).

Once a TCE is detected, the light curve is searched again for additional events, after all occurrences of the first are masked. This is done iteratively, until up to 10 TCEs are identified. For an extensive description of the procedure of transit detection, we refer the interested reader to J. M. Jenkins et al. (2002), J. M. Jenkins et al. (2010), and J. L. Christiansen et al. (2012). For all TCEs, a set of supplementary statistics are computed by the Data Validation (DV) module, which can aid the vetters in assigning dispositions to them (see e.g., J. L. Coughlin et al. 2016; T. D. Morton et al. 2016; F. Mullally et al. 2016; S. E. Thompson et al. 2018)).

All the Kepler data used in this paper can be found in the Mikulski Archive for Space Telescopes (MAST; STScI 2016).[13]

2.2. TESS Data

TESS was designed to perform photometric observations for ∼200,000 target stars at 2 minutes cadence (short cadence data) and simultaneously collect images of its entire field of view every 30 minutes (full-frame Images, long cadence data). The sampling rate of full-frame images was reduced from 30 to 10 minutes starting from the first TESS Extended Mission (2020 July). The features related to each observed star are stored in the TESS Input Catalog (TIC; K. G. Stassun et al. 2019). This catalog is periodically updated and its information (e.g., stellar properties, ancillary data) are computed by using Gaia DR2 (Gaia Collaboration et al. 2018) as a base catalog and other photometric catalogs, such as APASS (J. L. Tonry et al. 2018), and UCAC4 and UCAC5 (N. Zacharias et al. 2013, 2017). The data for the target stars are reduced and analyzed by the Science Processing Operations Center (SPOC) pipeline (J. M. Jenkins et al. 2016), which is strongly based on the KSOC pipeline described above, with some modifications (e.g., removal of spurious signals due to the absence of shutters on the TESS cameras). Calibrated images are computed both for short and long cadence data. The optimal aperture for target stars sampled at 2 minutes is determined as in KSOC so that the SAP light curves (J. D. Twicken et al. 2010; R. L. Morris et al. 2020) are extracted. The SPOC PDC (J. C. Smith et al. 2012) module applies a set of corrections to the SAP light curve (e.g., rejection of outliers and instrumental systematics). This PDCSAP light curve is searched for TCEs by the TPS module as in KSOC. The DV module computes a set of supplementary features for each TCE identified by TPS, publicly available on MAST.[14] Full-frame Images are processed by the MIT's QLP

[13] http://archive.stsci.edu/missions/kepler/lightcurves/tarfiles/DOI_LINKS/Q0-17_LC+SC/

[14] https://exoplanetarchive.ipac.caltech.edu

that extracts light curves for all stars with TESS magnitude $\leqslant 13.5$ (C. X. Huang et al. 2020a, 2020b; M. Kunimoto et al. 2021, 2022).[15] In accordance with the methodology employed by SPOC, QLP generates normalized light curves from the optimal aperture (SAP light curves). This pipeline removes stellar variability from SAP light curves by fitting a basis spline (B-spline) with spacing in the range [0.3, 1.5] (A. Vanderburg & J. A. Johnson 2014). This process is repeated iteratively until convergence. At each iteration, 3σ outliers are identified and discarded before refitting the B-spline. The QLP searches the detrended light curves (KSPSAP light curves) for TCEs using the Box Least Squares (BLS; G. Kovács et al. 2002) algorithm. In our work, we use the normalized, detrended light curve generated by all these pipelines, phase-folded over the period provided by the TCEs, and binned. We show in Section 6 that our results are independent of the pipeline used to extract the light curves. Hence, for a planetary transit detected from a generic survey to be vetted by our model, we just need to have the standardized photometric data, the epoch of the observed transit-like signal, and the period of the candidate planet.

All the TESS data used in this paper can be found at MAST (MAST Team 2021).[16]

3. Catalogs of Currently Available Labeled TCEs

To achieve our goal of developing a simple and adaptable automatic triage model, we demonstrate its flexibility by evaluating it on various data sets, including Kepler and TESS TCEs, used both individually and in combination. Details of the data sets used are provided below.

3.1. Kepler Data Set

The Kepler data set we used is the Autovetter Planet Candidate Catalog (J. H. Catanzarite 2015) for Q1–Q17 (S. Seader et al. 2015) Data Release 24 (DR24) and Data Release 25 (DR25, S. E. Thompson et al. (2018)).

1. The DR24 catalog consists of 20,367 TCEs detected by the KSOC pipeline in Kepler PDCSAP light curves. These TCEs have been automatically labeled by Autovetter (J. H. Catanzarite 2015). To enhance the accuracy of the training set labels, we followed the approach proposed by SV18 and removed all TCEs classified as unknown. In order to binarize the labels, we associated 3600 TCEs PCs to the PC class and the remaining 12,137 TCEs, including 9596 astrophysical FPs and 2541 nontransiting phenomenon, to the NPC class.
2. Kepler Q1–Q17 DR25 contains 34,032 transit signals detected in the 17 quarters of Kepler primary mission data set. The detection process of these TCEs is described in J. D. Twicken et al. (2016). To generate our set of PCs and NPCs TCEs from this catalog, we performed the following steps. First, we removed all TCEs with rogue flag set to 1. These are TCEs with less than three detected transits, added to the DR25 catalog because of a bug in the Kepler pipeline. For our PC class, we considered the set of 2726 confirmed and 1382 candidate planets from the Cumulative KOI catalog. This catalog was created with the intent of providing in one place the most accurate information about all the Kepler TCEs dispositioned as confirmed/candidate planet or FP. We included into our NPC class the 3946 TCEs labeled as FP in the Cumulative KOI table and the 21,098 TCEs from the DR25 catalog that are not in the Cumulative KOI table.

3.2. TESS Data Set

We used four catalogs of TESS TCEs: the Exoplanet Follow-up Observation Program (ExoFOP), TESS Triple 9, and the catalogs provided by YU19 and TEY23.

1. The catalog from the Exoplanet Follow-up Observation Program (ExoFOP).[17] From this catalog, we discarded 1,453 TCEs overlapping with the data set YU19. We also removed ∼560 TCEs for which there is no indication of the detection pipeline, i.e., SPOC or QLP. The remaining set of TCEs, which we define as $\mathcal{D}_E$, contains 3133 planets (including known and confirmed planets, PCs, ambiguous planetary candidates and nearby PCs[18] that we labeled as PC) and 419 FPs (including false alarms and eclipsing binaries, which we labeled as NPC) with dispositions assigned by the TESS Follow-up Observing Program Working Group (R. Akeson & J. Christiansen 2019).
2. The TESS Triple 9 catalog ($\mathcal{D}_{TT9}$, L. Cacciapuoti et al. 2022; C. Magliano et al. 2023). This catalog has been compiled within the framework of the NASA Citizen Science project Planet Patrol,[19] an initiative engaging both scientists and amateur astronomers in the vetting process to expedite the analysis of a large number of TCEs within a short time frame. Initially, this catalog contained 999 TCEs for which L. Cacciapuoti et al. (2022) provided dispositions. A new set of 999 TCEs has been recently labeled and added to the catalog by C. Magliano et al. (2023). In both cases, the authors determined the TCEs dispositions by manually examining the outputs produced by DAVE (V. B. Kostov et al. 2019) for each TCE. The samples of this catalog have been labeled as candidate planet, FP, and potential FP. The latter label is used when human vetters are not fully convinced of the nonplanetary nature of the given TCE. In the process of binarizing these labels, we decided to discard all potential FPs (amounting to 251 TCEs) to reduce the level of uncertainty affecting our test set. By further removing 187 TCEs in common with YU19, we obtain a final catalog of 1314 PCs (including candidate planets) and 276 NPCs (including FPs) homogeneously vetted.
3. The catalog of YU19 ($\mathcal{D}_{YU19}$) contains ∼16,700 TCEs detected in QLP SAP light curves from TESS sectors [1–5]. In total, there are 505 events labeled as PCs, 2247 as eclipsing binaries, 976 as V-shape, 816 as instrumental systematics, and 12,170 events labeled as junk. The authors have manually ascribed these dispositions through the implementation of a predefined set of rules,

[15] QLP data can be found at https://archive.stsci.edu/hlsp/qlp (C. X. Huang et al. 2020b).
[16] https://archive.stsci.edu/tess/bulk_downloads/bulk_downloads_ffi-tp-lc-dv.html#lc
[17] We downloaded the catalog from https://exofop.ipac.caltech.edu/tess/view_toi.php on 2023 January 12.
[18] Nearby PCs are TCEs consistent with a PC signature but their detected transit occurs around a nearby star.
[19] https://www.zooniverse.org/projects/marckuchner/planet-patrol/

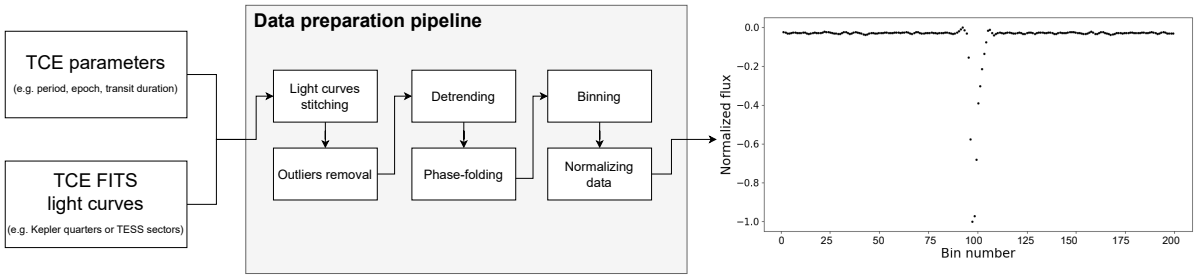

Figure 1. Data flow of our preparation pipeline. For each TCE, we downloaded the related FITS light-curve files from MAST. These light curves are concatenated in a single signal that is cleaned from not-a-numbers and outliers. The variability in the observed flux not related to the TCE's transits is flattened in the detrending module. The resulting signal is folded over the TCE's period. We finally binned and normalized the signal in order to produce a 201-length global view with median flux and transit depth set to 0 and −1, respectively.

Table 1
Composition of the Different TCEs Data Sets used in this Work

Data Set	Name	Positive Class	Negative Class	Imbalance Ratio (%)	Task	Reference
Kepler Q1–Q17 DR24	$\mathcal{D}_{K24}$	7200	12,136	37 : 63	binary	(1)
Kepler Q1–Q17 DR25	$\mathcal{D}_{K25}$	8216	24,657	24 : 76	binary	(2)
TESS ExoFOP	$\mathcal{D}_{E}$	2976	576	84 : 16	binary	(3)
TESS Triple 9	$\mathcal{D}_{TT9}$	1314	527	71 : 29	binary	(4)
TESS YU19	$\mathcal{D}_{YU19}$	996	16,154	6 : 94	binary	(5)
TESS TEY23	$\mathcal{D}_{TEY23}$	4830(E)	700(B);12,318(J)	4 : 27 : 69	multiclass	(6)
Data Set combination						Source
TESS (total)	$\mathcal{D}_{T}$	5286	17,257	23 : 77	binary	$\mathcal{D}_{E} \cup \mathcal{D}_{TT9} \cup \mathcal{D}_{YU19}$
TESS + Kepler DR24	$\mathcal{D}_{TK}$	12,486	29,393	30 : 70	binary	TESS $\cup$ $\mathcal{D}_{K24}$

Notes. For each data set, we provide: the name that we use to refer to them in this work, the distribution of samples and imbalance ratio between the two classes, and the type of task (binary or multiclass classification) for which they were used to train and test the model. The combined data sets are also listed, along with their respective class distributions and catalog sources.
[a] The number of positive training samples was doubled with data augmentation techniques, as described in Section 4.
References. (1) J. H. Catanzarite (2015); (2) S. E. Thompson et al. (2018); (3) R. Akeson & J. Christiansen (2019); (4) L. Cacciapuoti et al. (2022); C. Magliano et al. (2023); (5) L. Yu et al. (2019); (6) E. Tey et al. (2023).

aiming to guarantee uniformity in the labeling process (see Section 2.2 of their paper for further details). During the binarization step, we consider as PCs the 505 TCEs they classify as PCs and all other 16,209 TCEs as NPCs.

4. The catalog provided by TEY23 ($\mathcal{D}_{TEY23}$), consisting of a subset of TCEs detected by QLP in TESS long cadence data that the authors have manually vetted. This subset is composed of 24,952 TCEs (including 8992 TCEs detected in Sector 13, 13,372 TCEs detected from Sectors 14 to 26, and 2588 TCEs from Sectors 27 to 39). The authors classified these TCEs as: "periodic eclipsing signal," "single transit," "contact eclipsing binaries," "junk," and "not-sure" (see Section 2.4 of their paper for further details on the labeling process). By discarding (i) 5340 TCEs for which the authors did not provide a consensus label and (ii) all the TCEs labeled as "single transit" and "not-sure," we obtained 2613 periodic eclipsing signals, which include PCs and noncontact eclipsing binaries, and 16,5290 not-planet, including 738 "contact eclipsing binaries" and 15,791 "junk."

Because the TESS catalogs exhibit an imbalance toward one of the two classes, we generated a single catalog of TESS TCEs by merging data from the $\mathcal{D}_{E}$, $\mathcal{D}_{TT9}$, and $\mathcal{D}_{YU19}$ data sets. We then obtained a more balanced catalog of TESS ($\mathcal{D}_{T}$) TCEs from which we could derive significantly more reliable model evaluation metrics. In the discussion of the results obtained on this test set (Section 6.1.3), we have beforehand discarded 355 samples whose disposition differs between $\mathcal{D}_{E}$ and $\mathcal{D}_{TT9}$ catalogs. To evaluate the predictive capabilities of the model when trained-tested on cross-mission data, we also generated a catalog containing TCEs from $\mathcal{D}_{T}$ and $\mathcal{D}_{K24}$. We discuss the motivation for creating this data set in Section 5.1.

Table 1 summarizes the details of the different catalogs presented in this section.

4. Standardization of Input Data

Here, we describe how we standardized the light curves for each TCE in the data sets listed in Table 1. The data flow of our preprocessing pipeline is illustrated in Figure 1.

We represent each TCE as a phase-folded light curve divided into 201 bins. We first create a multisector light curve by stitching light-curve segments from the different Kepler quarters or TESS sectors. For Kepler TCEs, we concatenate all the related PDCSAP light curves as in SV18. For TESS TCEs, we only stitch the SAP light-curve segments in the same year in which the TCE was discovered. For example, if an event was discovered during TESS observation year 2, we use the light curves segments from TESS sectors [14, 26] to generate the final light curve. This is done to prevent the uncertainty on the transit parameters propagating onto distant observations leading in a mismatch of the folded data, as shown in Figure 2.

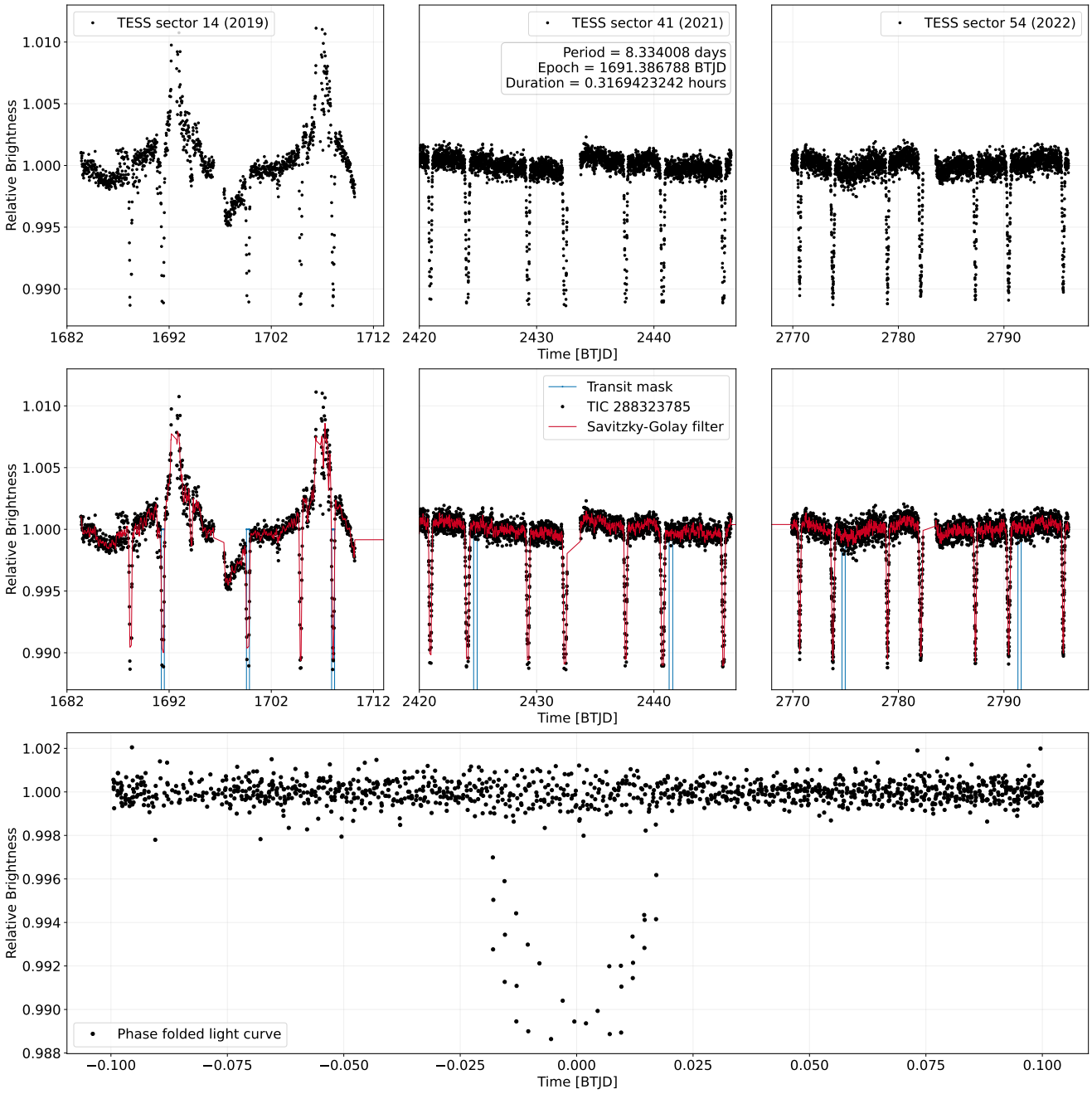

Figure 2. An example of incorrect generation of the global view. For TIC 288323785, TESS acquired photometric data over a span of 3 yr (top panel). The transit mask (blue solid line), generated by using the period, epoch, and duration from the TCE, fits the data from sector 14 well (middle-left panel) but not the later ones (second and third middle panels). When flattening the light curves (black dots) dividing by the interpolating polynomial (red dashed line), transits from sectors 41 and 54 are not preserved. As a result, the light curve is not correctly phase-folded (bottom panel).

Then, we remove all outliers (flux measurements above $\pm 3\sigma$ from the median flux) from the stitched light curve (S. Fiscale et al. 2021) and any long-term variability in the signal that is not related to the TCE.

We perform this by dividing flux values over the interpolating polynomial computed using the Savitzky–Golay filter (P. De Luca et al. 2022). The flux values corresponding to the TCE transits are masked during this step so that they are preserved. This detrending process aligns with recent data-driven approaches proposed for astrophysical applications (M. G. Orsini et al. 2025). Our detrended signal is phase-folded on the relative period and binned with a time bin size of 30 minutes. As in YU19, we linearly interpolate over any void bin to generate a uniform input signal of length 201. Following SV18, the final step of our data preparation pipeline consists of normalizing the binned flux so that the median value is zero and the maximum transit depth is fixed at -1. By implementing this data normalization, we enhanced the numerical stability of the model; however, the information on the transit depth is lost. The binned and normalized light curve is called the *global view* and it represents the input of our model.

In order to perform the data augmentation, we also produce a horizontal reflection of the PC's global views to obtain

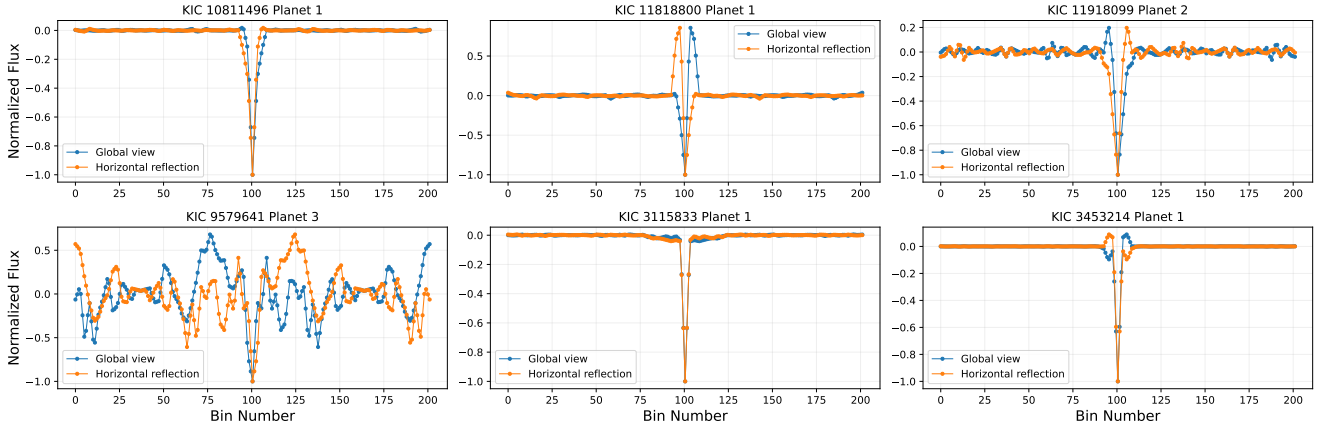

Figure 3. An example of the output obtained by the horizontal reflection. From top left to bottom right we display six global views (blue lines) generated for six PCs KOIs from Kepler Q1 to Q17 DR25. For each global view, we provide its horizontally reflected version (orange lines). The horizontal reflection procedure is applied over all the data sets highly imbalanced toward the class of NPC.

additional samples (Figure 3). We applied this strategy specifically to the unbalanced data set in Table 1, where the number of negative samples far exceeds that of positive ones, to mitigate the imbalance between positive (PC) and negative (NPC) TCEs.

Due to the significant computational cost required by this pipeline, we employed parallel distributed strategies to optimize computational time (P. De Luca et al. 2019).

5. Network Architecture

The architecture of our model is a thinned version of the one proposed by K. Simonyan & A. Zisserman (2015), which is a one-dimensional CNN. This kind of network belongs to the class of deep learning models and have proved their effectiveness in processing time-series data in different research fields (see e. g. B. Zhao et al. 2017; D. M. Durairaj & B. K. Mohan 2022). A CNN is composed by two main branches: (i) a *feature extraction* branch—in which ever more complex features and nonlinear relationships (defined as feature maps) are extracted from input data, and (ii) a *classification branch*—where the output of the previous branch is processed feed-forwardly through at least one layer of neurons. Typically, the sigmoid activation function is used to map the output of the classification branch in the range [0, 1].

Our model takes the global view as input and processes it through the feature extraction branch. This branch is composed of five convolutional blocks. Each block has one convolutional layer with filter size 3 and a number of convolutional filters increasing by power of two in the range [16, 256]. During the convolutions, we preserve the length of the input feature maps by using the padding technique. Otherwise, the length of the feature maps generated by the convolutional blocks would progressively diminish to 0, resulting in a complete information loss. We then apply the rectified linear unit (ReLU) activation function to the output provided by the convolutional layer. At this stage of processing, we prevent feature maps of the model from overfitting by using the spatial dropout with dropout probability rate ρ fixed to 0.2. The convolutional block structure ends with a max-pooling layer and a batch normalization layer (S. Ioffe & C. Szegedy 2015). The max-pooling layer reports the maximum output within a 5-length neighborhood. The batch normalization layer regularizes the output of the block by introducing both additive and multiplicative noise.

The feature tensor produced by the feature extraction branch is flattened and delivered to the classification branch. This branch is composed of a single dense block containing a fully connected layer of 512 neurons and a dropout layer.

We used a single fully connected layer because it is adequate to guarantee the best approximation property (G. Cybenko 1989; K. Hornik et al. 1989). K. Visser et al. (2022b) showed that by reducing the number of model's parameters (i.e., the model's complexity), its performance is not subject to any deterioration. Typically, the model's complexity mainly depends on the number of the fully connected layers composing the classification branch. A network with a single fully connected layer of n neurons shows n^2 connections (i.e., parameters). By increasing the number of fully connected layers to H, the number of parameters to be optimized becomes $n^2 \times (H - 1)$. To understand this, consider that $n \times n$ is the number of connections between two consecutive layers, and with H layers the number of connections between them is $H - 1$. Typically, the density of connections in a fully connected network grows quadratically with the number of n neurons. The higher the model's complexity, the larger the training set size required. Since for labeled TCEs we cannot rely on such a large training set yet, we have to reduce the model's complexity to prevent the model from overfitting or underfitting.

The output layer of our model is a single neuron provided with the sigmoid activation function. We binarize the value produced by this function through a thresholding process. For output values greater than 0.5, the input TCE is labeled as PC. Otherwise, it is labeled as NPC. Figure 4 depicts the architecture of our model.

The source code necessary for the deployment and training of the CNN architecture, as well as that essential for the generation of input samples, are available on GitHub.[20]

5.1. Combining Training Data from Different Space Missions

In ML, exploiting data from different domains can help the model to reduce overfitting to specific data sets. In our case, training the model on Kepler and TESS TCEs brings the following benefits: (i) it allows us to increase the number of TCEs available for training, and (ii) it allows us to improve the network generalization performance. Hence, using TCEs from both Kepler and TESS data sets allows us to train a more

[20] https://github.com/stefanofisc/dartvetter_apj

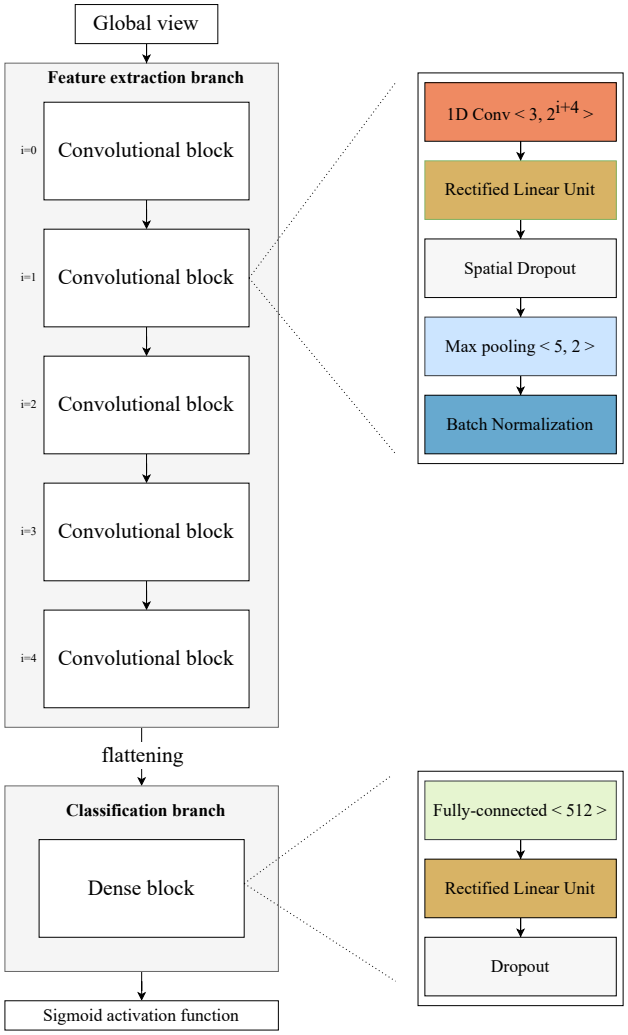

Figure 4. The architecture of the CNN model. The feature extraction branch is responsible for features extraction from the global view. The feature tensor produced by the last convolutional block is subjected to flattening. Then, classification is executed through utilization of a multilayer perceptron within the classification branch.

robust and versatile model, improving its performance in the detection of exoplanets across different surveys. In S. Fiscale et al. (2023), we optimized the parameters of the model by combining Kepler and TESS TCEs using Transfer Learning (R. Ribani & M. Marengoni 2019) and observed that the optimal performance (F1-score of 84%) was achieved by pretraining the model on Kepler TCEs and subsequently fine-tuning on TESS TCEs. However, during the process to further simplify the model's architecture, we found that `DART-Vetter` performs better when trained on the entire set of TCEs at once.

5.2. Model Selection

The network architecture of `DART-Vetter` was determined through a careful and extensive process of *model selection*, aimed at determining the optimal model in minimizing the fraction of misclassified planetary signals.

First, we defined a base configuration consisting of a neural network with fixed hyperparameters. We undertook a grid search process to train and test a set of configurations by varying the number of convolutional blocks, dense block structure, and the values of the hyperparameters, such as learning rate, batch size, and dropout probability rate. All these configurations shared the same training and test sets. Each configuration was evaluated based on its ability to balance model complexity and predictive capabilities. This evaluation allowed us to identify the best model able to generalize on signals from different transiting surveys. We provide further details of the model selection phase in Appendix A. The resulting model, named `DART-Vetter`, combines a compact structure with high predictive performance, making it a robust tool for the classification of TCEs.

5.3. Training Details

Once we fixed the hyperparameters, we initialized the parameters of our network drawing values from a uniform distribution (G. Klambauer et al. 2017). These parameters are optimized by using the Adam's optimization algorithm (D. P. Kingma & J. Ba 2015) to minimize the loss function. The learning rate value used at this stage is α=10^{-3} and the batch size is 128. As loss function, we employ the binary cross-entropy, which measures the difference between the real dispositions and the ones predicted by the model. As highlighted in the "Imbalance Ratio" column of Table 1, all our data sets exhibit an imbalance toward one of the two classes. To address class imbalance, we use the class weighting technique. More precisely, to differently weigh the model's predictive error on positive and negative samples, we employed the inverse class frequency method (V. Lertnattee & T. Theeramunkong 2004).

We make use of the dropout technique, which allows us to perform the training only once: given a network with n neurons, or units, the dropout allows to train 2^n different subnetworks.[21] During the training, each subnetwork is obtained by dropping some units, with a probability ρ, from the entire network architecture, but since all subnetworks share the same parameters, we save a significant amount of memory (Y. LeCun et al. 2015) with respect to other networks in which the training is done with standard model averaging (G. Claeskens et al. 2008). In that case, it is common to use ensembles of $K = 5$ to 10 neural networks (C. Szegedy et al. 2015), each with their own set of parameters. Because of the independence of these parameters between any two different networks of the ensemble, all K models have to be trained separately. In the context of exoplanet detection, this technique has been widely adopted, starting with SV18. However, training multiple independent models is highly time consuming, particularly for large architectures such as `ExoMiner` and `AstroNet-Triage-v2`. These models rely on K-fold cross-validation, requiring K separate training and evaluation runs, with the final TCEs prediction achieved by averaging the K independent classifications. This process inevitably complicates the reproducibility of these models.

For these reasons, the use of model averaging can be very expensive in terms of computational complexity, while dropout allows us to train many more than K networks but in the same time required to train a single network, i.e., $\sim$27 s in our case. During the model selection, we tested values for ρ

[21] The feasibility of training all 2^n networks relies on the condition that this quantity enables the generation and training of all possible subnetworks within a reasonable amount of time.

Table 2
Confusion Matrices Computed on all the Test Sets

Data Set	TN (%)	FP (%)	FN (%)	TP (%)	Accuracy	Misc. Rate
Kepler Q1–Q17 DR24	2056 (0.84)	376 (0.15)	136 (0.09)	1304 (0.90)	0.86	0.13
Kepler Q1–Q17 DR25	4645 (0.94)	292 (0.05)	669 (0.40)	979 (0.59)	0.85	0.14
TESS ExoFOP	11 (0.09)	111 (0.90)	8 (0.01)	581 (0.98)	0.83	0.16
TESS Triple 9	16 (0.16)	80 (0.83)	15 (0.05)	258 (0.94)	0.74	0.25
TESS YU19	2291 (0.71)	923 (0.28)	26 (0.12)	190 (0.87)	0.72	0.27
TESS	2250 (0.65)	1196 (0.34)	37 (0.03)	957 (0.96)	0.72	0.27
TESS + Kepler DR24	4478 (0.77)	1330 (0.22)	192 (0.07)	2307 (0.92)	0.81	0.18

Note. For each data set, we show the number of TNs, FPs, FNs, and TPs along with the relative fraction (%) to allow a fair comparison among all data sets. Accuracy and misclassification rates are also reported.

in the range [0.2, 0.5] as suggested in N. Srivastava et al. (2014) and found the best value of $\rho = 0.2$.

6. Experimental Results

In this section, we present the experimental results, assessing the network performance across various data sets, tasks, and evaluation metrics. The results for binary classification task are detailed in Section 6.1, while Section 6.2 focuses on the performance of `DART-Vetter` in the context of multiclass classification.

We use the conventional metrics in the field of machine learning in order to evaluate the model, namely:

1. *Accuracy*. The ratio of correctly predicted TCEs to the total number of samples in the data set:

$$\frac{\text{true positives} + \text{true negatives}}{\text{true positives} + \text{true negatives} + \text{false positives} + \text{false negatives}},$$

where true positives (TPs) and true negatives (TNs) are TCEs correctly identified as positive and negative by the classifier, respectively; FPs are TCEs identified as positive but labeled negative, and false negatives (FNs) are the samples identified as negative but labeled positive.

2. *Misclassification rate*. The fraction of TCEs inaccurately classified by the model, computed as:

$$\text{Misclassification rate}: \frac{\#\text{ incorrect predictions}}{\#\text{ total predictions}} = \frac{\text{false positives} + \text{false negatives}}{\#\text{ total}},$$

where 'total' is the total number of samples of a given data set, i.e., the sum of TP, FN, FP, and FN.

3. *Precision*. Fraction of planets correctly classified by the model among all TCEs that the model has classified as PCs:

$$\text{Precision}: \frac{\text{true positives}}{\text{true positives} + \text{false positives}}.$$

4. *Recall*. Fraction of planets correctly classified by the model:

$$\text{Recall}: \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}.$$

5. *F1-score*. Precision and recall are averaged to obtain the *F1-score*, which is often used to assess the overall effectiveness of a classifier's performance:

$$\text{F1-score}: 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

6.1. Binary Classification on Single/Cross-domain Data

We report the performance of the model in solving the task of binary classification on TCEs belonging either to a specific or multiple surveys. All the tests presented in the subsections below were conducted following this methodology:

STEP 1: We fixed a data set from Table 1, a training-test set split of 80–20 and a grid of training epochs of [10, 30, 50, 80, 150].

STEP 2: For each value of the grid, `DART-Vetter` was trained and evaluated. At the end of each training-test phase, we computed the scores for all the metrics. This procedure returned 10 values for each evaluation metric (five corresponding to the training set and five to the test set).

STEP 3: We analyzed the variation across these five values to assess the model's convergence speed in solving the task at hand on the fixed data set. To conduct this analysis, we computed the mean and standard deviation for each group of five values. For a given metric, high standard deviation values indicate that the model struggles to converge toward optimal performance. We illustrate the average and variation of values for precision, recall, and F1-score computed on each data set in Figure 6.

The confusion matrices computed on all the test sets are reported in Table 2. Additionally, in Appendix B we examined the variation of model performance when training and testing on different data set splits. The training-test splits considered in this analysis were 60–40, 70–30, 80–20, and 90–10. The results are displayed in Figure 8.

Here, we want to remark that we provide a complete set of evaluation metrics and explicitly detail their computation as we strive for transparency in our work. Our approach aims to provide a comprehensive and well-documented evaluation framework, which ensures greater clarity and facilitates reproducibility.

6.1.1. Test 1. Application on Kepler Q1–Q17 Data Release 24

The first set of TCEs on which we evaluated the predictive capabilities of the model is $\mathcal{D}_{K24}$. The results we obtained at the end of the three outlined steps are detailed below.

The model achieves optimal performance when trained for 80 epochs, obtaining a precision of 0.77, recall of 0.90, and

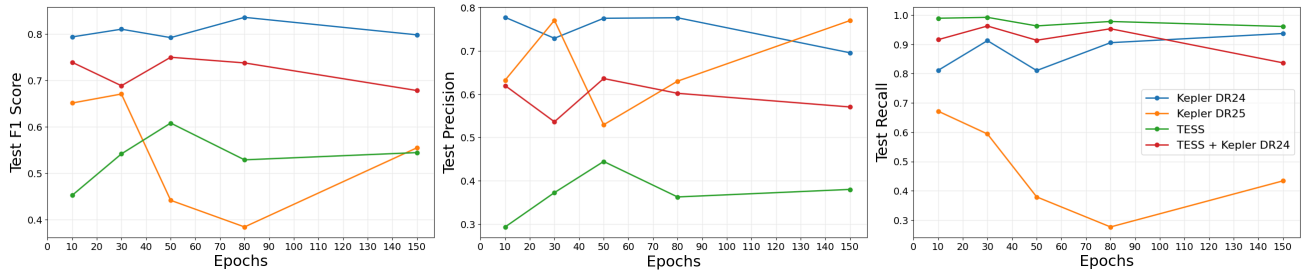

Figure 5. Variation of the main evaluation metrics (F1, precision, and recall) as the number of training epochs increases. These values have been computed on the test set of each data set. For sake of clarity, concerning TESS data we display the performance of `DART-Vetter` on the data set TESS that includes ExoFOP, Triple 9, and YU19.

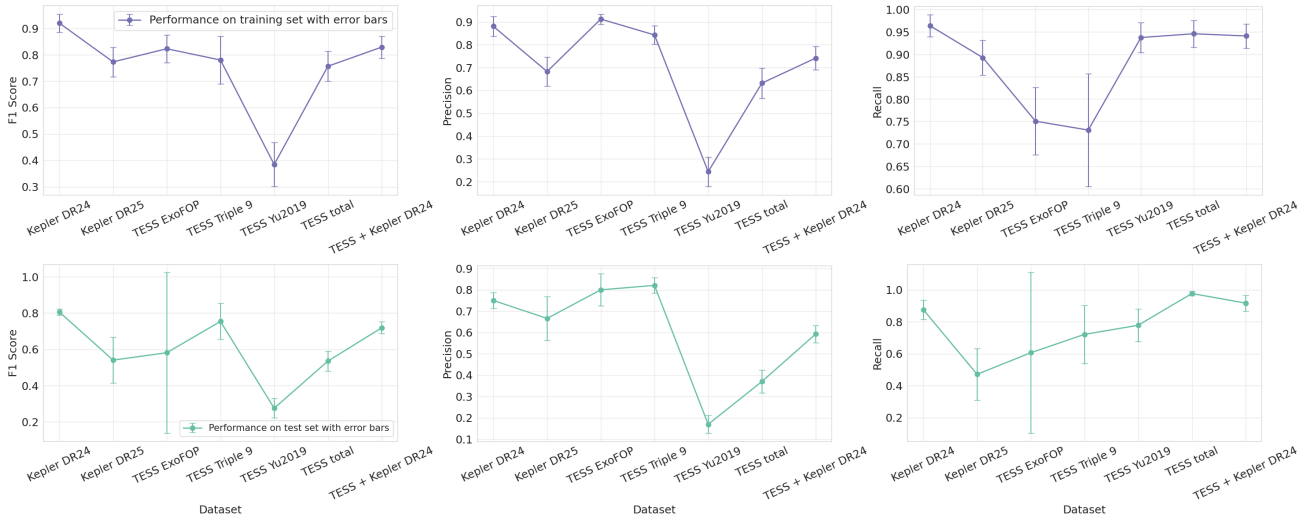

Figure 6. Mean values and error bars for F1-score, precision, and recall across all data sets. (Top panel, purple lines) Average scores and error bars computed for the specified evaluation metrics on the training sets. (Bottom panel, green lines) Corresponding values obtained on the test sets.

F1-score of 0.83. The values of the metrics remain stable as training epochs vary on both training and test sets (blue lines in Figure 5). On the test set, precision varies by 0.08 and recall by 0.12, with F1-score ranging of 0.04, reaching its maximum value for 80 training epochs. These small variations, visible with the relative error bars in Figure 6, indicate a rapid convergence of the model to returning reliable predictions on this data set. The fractions of TN and TP are 0.84 and 0.90, respectively. These rates are listed in the first row of Table 2.

On this set of TCEs, `DART-Vetter` proves to be a robust model for performing triage due to the low FN rate of 0.09 and misclassification rate of 0.13.

6.1.2. Test 2. Application on Kepler Q1–Q17 Data Release 25

The Kepler Q1–Q17 Data Release 24 catalog does not include many long-period and low-S/N TCEs, which are instead included in the latest Kepler TCEs catalog: the Kepler Q1–Q17 Data Release 25. We evaluate `DART-Vetter` also on this catalog and detail the results below.

The best performance on the test set is obtained when the model is trained for 30 epochs, but the triage capabilities of DART-Vetter on this data set are not reliable. The second row of Table 2 reports the confusion matrix calculated on the test set. The precision, recall, and F1-score on the test set reach values of 0.77, 0.59, and 0.67, respectively. In fact, the model is not able to learn relevant information during training, obtaining a precision of 0.64 and a recall of 0.86, which are 26% and 13% lower than the respective values obtained during the training on Kepler DR24.

As the training epochs increase, there are significant changes in the values of F1-score, precision, and recall on the test set (orange lines, Figure 5). As a result, the error bars calculated on these metrics and shown in Figure 6 are larger than those obtained on the Kepler DR24 test set. The wide variations observed in the metrics as the training epochs increase suggest that the model does not converge smoothly to a correct classification. Such fluctuations in the values motivate the high FN rate of 0.40.

6.1.3. Test 3. Application on TESS Data

We now examine how `DART-Vetter` performs on TESS data. We initially trained and tested the model separately on ExoFOP, Triple 9, and YU19, following steps 1, 2, and 3. As seen from Table 1, all three catalogs are heavily unbalanced toward one of the two classes. Although we handled this issue by adopting class weighting during training, such an imbalance still hampers the generalization capabilities of the model. For example, on test sets strongly unbalanced toward the PC class, `DART-Vetter` would obtain high recall and F1-score values simply by classifying all samples as PCs. For this reason, we present here the performance obtained by the model only on the TESS data set (described in Table 1), while the confusion

Table 3
(Left Block) Confusion Matrix Displaying the Performance of our Model on Multiclass Classification

	Class	B	E	J	Class	Precision	Recall	F1-score
Real	B	10 (0.13)	37 (0.50)	26 (0.35)	B	0.833	0.158	0.261
	E	0 (0.00)	322 (0.67)	156 (0.32)	E	0.587	0.674	0.629
	J	2 (0.001)	189 (0.15)	1053 (0.84)	J	0.860	0.846	0.853
		This work's dispositions			Average	0.776	0.776	0.776

Note. We computed this matrix on the test set of $\mathcal{D}_{TEY23}$, corresponding to 10% of the entire data set. (Right block) Precision, recall, and F1-score for each class. The weighted average values computed on the entire test set are reported in the last row of the table.

matrices, accuracy, and misclassification rate scores are listed in Table 2.

The best results on the test set are achieved on 50 training epochs, with the model showing robust triage capabilities, since it achieves a recall rate of 0.96, which is higher with respect to any other data set.

The green lines of Figure 5 show a stability close to 1 in the recall on the test set, as the number of training epochs increases. In contrast, the precision scores exhibit significant oscillations, indicating challenges in the model's ability to correctly classify TCEs belonging to the NPC class. This difficulty is reflected on the high FP rate of 0.34, as shown in the second to last row of Table 2.

6.1.4. Test 4. Application on Cross-domain Data

ML models can achieve more robust predictive capabilities when trained on data sets from multiple domains. In order to do this, the data need to be standardized, ensuring invariance to the noise characterizing the original distributions from which the data are drawn. In this study, we perform this normalization by preprocessing the TCEs using the pipeline described in Section 4. The model was trained and tested on the $\mathcal{D}_{TK}$ data set, following the methodology outlined in Steps 1, 2, and 3. The results obtained on this set of TCEs are presented below.

Optimal performance on $\mathcal{D}_{TK}$ is achieved after 50 training epochs, with precision and recall rates of 0.63 and 0.91 on the test set. The harmonic mean, i.e., the F1-score, is 0.75. The generalization capabilities of the model do not show significant variations along the training epochs (red lines, Figure 5). Moreover, the model achieves better performance than that obtained on the TESS data set. Comparing the scores obtained between $\mathcal{D}_T$ and $\mathcal{D}_{TK}$, a great improvement is found in the overall values of F1-score and precision, while the recall value remains very high. The standard deviation calculated on the F1-score is reduced compared to that calculated on $\mathcal{D}_T$, indicating that the addition of TCEs from $\mathcal{D}_{K24}$ is useful to improve generalization. The capabilities in performing a reliable triage on cross-mission data are confirmed by the low fraction of FNs, corresponding to 0.07. The confusion matrix on test set is given in the last row of Table 2.

Finally, we emphasize the benefit of training the model on the combined Kepler and TESS data set. Using this data set results in a significantly more reliable model, with recall stabilizing at approximately 92%, while precision increases by ∼20%. Additionally, as shown in Table 2, the misclassification rate drops by 9%, from 0.27 to 0.18, further demonstrating the effectiveness of this approach.

6.1.5. Comparison of Network Performance on Different Domains

In this section we summarize and analyze the results from the above tests. Figure 6 shows the trend of the average values of F1, precision, and recall as a function of the domain.

On the training set, the best performance is obtained on Kepler DR24, with precision, recall, and F1-score close to 1. The error bar on the F1-score is smaller compared to the others, meaning that the model quickly converges. The performance worsens on Kepler DR25, on which `DART-Vetter` obtains an F1-score ∼13% lower than on Kepler DR24. The wider error bars also indicate a slower convergence. This difference in performance on the two Kepler data sets reflects the difference in the S/N of TCEs in the two catalogs. The high-precision values achieved on ExoFOP and TESS Triple 9 are related to the bias of the training set toward PC TCEs. In fact, the recall is relatively low and indicates that the model misses many PCs. More in-depth considerations of the true planetary nature of the PC TCEs are provided in Section 8.2. The opposite happens on the YU19 data set, which is biased toward the NPC class. Finally, the performance of `DART-Vetter` on the entire TESS data set is stable, with error bars decreasing and a second highest score on the training recall; which instead is the highest on the test set. But the addition of Kepler DR24 TCEs to the TESS data produces even more robust performance: the model shows a faster convergence to optimal discrimination capabilities, as discussed in the previous paragraph.

We do not discuss the corresponding results on the test sets (depicted by the green lines in Figure 6) because they are in line with those presented on the training data.

6.2. Multiclass Classification on TESS Data

In order to compare the performance of our model with the ones available in literature, we test our network architecture for multiclass classification by using the $\mathcal{D}_{TEY23}$ data set, following the approach described in TEY23. We split the data set into 90% for training and 10% for testing. As shown in Table 1, this data set also exhibits a strong imbalance between classes, with one predominant class, J, including astrophysical signals such as stellar variability, artifacts, and scattered light. To address this problem, we used class weighting during training.

To perform multiclass classification, only minor changes to the architecture were required. We increased the number of output neurons from 1 to 3 and used cross-entropy as the loss function. The model was trained for 35 epochs.

The confusion matrix is given in Table 3, together with the precision, recall, and F1 rates for each class, and the averaged values on the entire test set. Given that our goal is to maximize the fraction of recovered planetary candidates, the class of

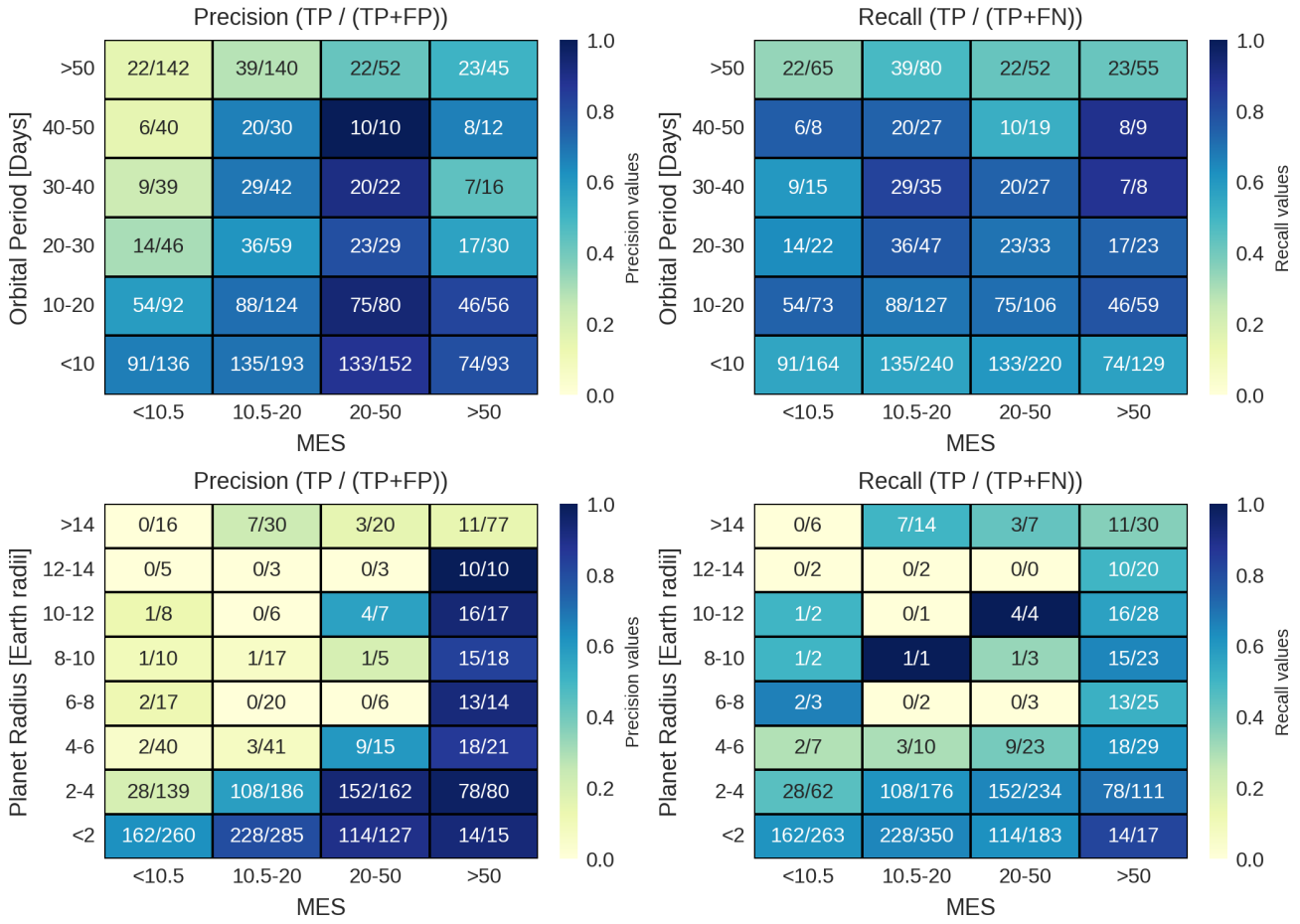

Figure 7. Performance analysis of `DART-Vetter` on the Kepler Q1–Q17 Data Release 25 data set in the space defined by the parameters MES, Orbital Period, and Planet Radius. (Top row) Fractional values of precision (left) and recall (right) as a function of MES and Orbital Period. (Bottom row) Same values obtained as a function of MES and Planet Radius.

particular interest is E, which includes planetary transits and noncontact eclipsing binaries. The fraction of correctly classified eclipsing signals is 0.67, while the rest of the samples are misclassified as J. The precision for class E is 0.587 and the recall is 0.674.

On the entire test set, the values of weighted average precision and recall are 0.776, indicating an overall good performance of the model in filtering out most nonplanetary signals.

6.3. Evaluating Model's Performance on MES-orbital Periods and MES-planet Radius

We analyze the performance of our model as a function of the parameter pairs MES-Orbital Period and MES-Planet Radius, focusing on the data set Kepler Q1–Q17 Data Release 25, which consists in the set of TCEs where our model struggles more in achieving robust generalization on unseen samples. The goal of this analysis is to identify those regions of the parameters space where the model's performance cannot be considered reliable because precision and recall values are less than $\sim$0.60. The results discussed below are illustrated in four heatmaps in Figure 7.

The model exhibits weak classification capabilities on TCEs with MES lower than 10.5 or with orbital period longer than 50 days. As displayed in the top left heatmap, for MES values lower than 10.5, the precision decreases from 0.66 (orbital period shorter than 10 days) to 0.15 (orbital period longer than 50 days). Conversely, for TCEs with orbital periods longer than 50 days, the precision improves as the MES increases, with precision values that remain below 0.51, indicating a predominance of FPs in this region of the parameters space.

The top right heatmap shows that for TCEs with low MES ($<$10.5), our model is not able to correctly classify signals with orbital periods out of the range [10, 50], where the recall assumes values lower than 0.55. These values remain very low ($<$0.48) in the region of space defined by TCEs with long orbital periods ($>$50 days), regardless of an increase in MES values. In these regions, there is a higher fraction of FNs than TPs.

Considering the bottom left heatmap, showing the performance of our model as a function of MES-Planet Radius [Earth radii, $R_\oplus$], the precision is higher than 0.85 in two particular regions: (i) for TCEs with MES larger than 50 and radius less than 14 $R_\oplus$, and (ii) for TCEs with MES values in [20, 50] and radius less than 4 $R_\oplus$. Outside these regions, the model shows a marked difficulty in distinguishing PCs from FPs, especially for TCEs with MES lower than 10.5 or radius greater than 14 $R_\oplus$.

Table 4
Performance Comparison of Different Classifiers When Tested on a Set of Unseen Kepler/TESS TCEs

Model	Input Size	Precision	Recall	F1-score	Test Set Distribution	Source	Reference
Binary classification on Kepler data							
Exominer	1×780	0.99	0.93	0.96	24 : 76	$\mathcal{D}_{K25}$	(1)
Exominer++	$1 \times 19{,}560$	0.97	0.96	0.97	9 : 91	$\mathcal{D}_{K25}$	(2)
DART-Vetter	1×201	0.77	0.59	0.67	25 : 75	$\mathcal{D}_{K25}$	This work
DART-Vetter	1×201	0.77	0.90	0.83	37 : 63	$\mathcal{D}_{K24}$	This work
Binary classification on TESS/Kepler data							
Astronet-Triage	1×262	0.74	0.97	0.83	2 : 98	$\mathcal{D}_{YU19}$	(3)
Astronet-Vetting	$1 \times (\sim 323)$	0.39	0.89	0.54	2 : 98	$\mathcal{D}_{YU19}$	(3)
Exominer-basic	1×773	0.88	0.73	0.79	1167 TOIs	$\mathcal{D}_{E}$	(1)
Exominer++	$1 \times 19{,}560$	0.93	0.95	0.94	7 : 93	TESS	(2)
DART-Vetter	1×201	0.63	0.91	0.75	30 : 70	$\mathcal{D}_{TK}$	This work
Multiclass classification on TESS data							
Astronet-Triage-v2	1×5248	0.75	0.97	0.85	9 : 91	$\mathcal{D}_{TEY23}$	(4)
DART-Vetter	1×201	0.77	0.77	0.77	26 : 73	$\mathcal{D}_{TEY23}$	This work

Note. For each model, we provide input size, evaluation metric scores, test set distribution, data source, and corresponding references. Input sizes were calculated by flattening all input vectors and supplementary features processed by the models.
References. (1) H. Valizadegan et al. (2022); (2) H. Valizadegan et al. (2025); (3) L. Yu et al. (2019); (4) E. Tey et al. (2023).

We observe in the bottom right heatmap a high percentage of FNs for TCEs with MES < 10.5 and radius between 2 and 6 $R_{\oplus}$, as well as for radius larger than 14 $R_{\oplus}$. However, we cannot accurately assess the performance of the model on TCEs with MES < 50 and radius > 6 $R_{\oplus}$ due to a small number of samples in this region of the parameter space.

From the analysis of the four heatmaps, we find that MES is a determining factor in the correct classification of TCEs, as is orbital period. In fact, performance improves as the MES values increase, as does for orbital periods lower than 50 days. In contrast, the radius does not seem to systematically influence the performance of the model, since no trends in classification performance are identified as the radius increases. For example, in the region with MES values in the range [20–50] and radius in [10–20] the model recovers all four PCs. However, for radius up to values greater than 14 $R_{\oplus}$, even at high MES, the recall decreases significantly, and the model misclassifies four of the seven PCs that fall in that parameter space.

7. Discussion

Here, we compare `DART-Vetter's` performance with the CNNs in literature, grouped by application domain (Kepler or TESS) and type of task (binary or multiclass). For each group of models, we pick the most effective configuration of `DART-Vetter` to perform a comparison.

7.1. Performance Comparison with State-of-the-Art Models

Table 4 shows for each model the scores[22] achieved on the main evaluation metrics described in Section 6, and information about the related test sets (distribution of samples among the classes and source catalog). The input dimension of each model is also provided. Since all the models presented in Table 4 have been trained and evaluated on slightly different distribution of samples, the comparison of their predictive performance has to be done taking this into consideration.

In the binary classification of Kepler TCEs, `DART-Vetter` shows discrete but inferior performance than `Exominer`, with an F1-score 13% lower than the CNN developed by H. Valizadegan et al. (2022). Yet, with a recall of 0.90 with respect to 0.93 of `Exominer`, `DART-Vetter` shows comparable triage capabilities. But the input dimensionality[23] of our model is significantly reduced (1×201 versus 1×780), indicating a better computational efficiency and generalization on cross-mission data.

For binary classification on TESS TCEs, the best-performing CNN currently available to our knowledge is still `Astronet-Triage` by L. Yu et al. (2019). By working solely on the global and local views, this model achieves appreciable performance on the test set, with a precision of 74% and a recall of 97%, when including eclipsing binaries in the TP class. `Astronet-Vetting` is a slightly more complex model with respect to `Astronet-Triage` because it processes a larger number of input features, but achieves lower performance, supporting the finding by K. Visser et al. (2022b) and subsequently by this study: an increase in input dimensionality does not always yield benefits in terms of a model's performance. Both versions of Astronet have been trained and tested on TESS TCEs only. On the other hand, `Exominer-basic` is trained on TCEs from the Kepler Q1–Q17 Data Release 25 catalog (J. D. Twicken et al. 2016) and tested on 1167 TESS TCEs for which N. M. Guerrero et al. (2021) provided dispositions. This test set is composed of TCEs labeled as KP, CP, FPs, and false alarms. The authors deliberately decided to not include the TCEs dispositioned as PC because they state the following: "We did not include the TOIs with PC disposition because this disposition is not conclusive." Since the number of test samples for each class is not detailed in H. Valizadegan et al. (2022), we only report the total test set size in Table 4. `Exominer-basic` reaches noteworthy performance by working on time-series (including the global view) and stellar parameters, thus showing that such

[22] The scores reported in Table 4 for each model are taken from the related papers as indicated in the Reference column.

[23] We computed all the input sizes described here and reported in Table 4 by flattening the entire set of input vectors and supplementary features processed by the models.

CNNs are able to perform well on cross-mission TCEs (see also S. Fiscale et al. 2023). In this context, DART-Vetter stands out as the model that needs the lower input dimensionality. Its triage capabilities are comparable with those of Astronet-Triage but evaluated on a more carefully balanced data set.

We include also the comparison with the improvement of the model provided by H. Valizadegan et al. (2022), Exominer ++ (H. Valizadegan et al. 2025). This work, available on arXiv, has been developed to improve performance on TESS data. The authors extend the architecture of this CNN by introducing five additional aspects of input information. The model processes as input photometric data for a total dimension of 19,522 and 38 scalar features. In the binary classification on Kepler data, Exominer++ improves its previous F1-score by 1%, establishing itself as the best model currently presented in this context. Predictive performance improves significantly on TESS data. As we found with DART-Vetter, Exominer++ also improves its generalization capabilities on TESS when trained on a data set combining Kepler and TESS data. With a recall of 95%, the model is inferior only to Astronet-Triage in the correct classification of planets, while it proves to be the best in terms of generalization with an F1-score of 94%. We calculated the distributions of the test sets on which Exominer++ is evaluated based on the number of TCEs given in Tables 2 and 4 of their paper, and considering that the model is trained and tested with 10-fold cross-validation.

In multiclass classification, the F1-score of DART-Vetter is 8% lower than that of the optimal model Astronet-Triage-v2. Even in this case, the model stands out for its extremely compact input, using only 1×201 features compared with 1×5248 of Astronet-Triage-v2.

Thus, the minimized complexity of our model ensures it can be adapted on binary or multiclass classification of signals of any domain, with triage capabilities comparable to those of the currently available models specific to a given transit survey. However, it is important to consider that such a reduction in complexity, due in part to the use of input photometric data alone, limits the potential of our model, which shows degrading performance in classifying Kepler TCEs with MES < 20 or with orbital periods > 50 days.

On the other hand, the weakness of state-of-the-art models lies in the high computational complexity required by the processes of (i) data collection, (ii) preprocessing, and (iii) network architecture implementation. With DART-Vetter, we strive to achieve a good trade-off between predictive performance and ease of implementing.

7.2. Rationale Behind Choosing a Less Complex Model

Minimizing model complexity is a fundamental aspect of the DART-Vetter design because it enhances training efficiency, improves generalization to unseen data, and increases interpretability. A cornerstone in this area is the seminal work by I. Guyon & A. Elisseeff (2003). Their work addresses the importance of variable and feature selection, which allows for processing only linearly independent features, thereby eliminating redundant information in the data that adds unnecessary complexity to the problem. This is a very relevant topic in deep learning and recent studies are continuing to focus on it (M. S. Im & V. R. Dasari 2022; N. Pudjihartono et al. 2022).

As extensively described in X. Hu et al. (2021), in the context of deep learning, model complexity refers to different factors (e.g., expressive capacity and effective model complexity). For DART-Vetter, we consider the following set of factors: the number of weights to be optimized during the training process, the number of hyperparameters to be set during the fine-tuning process, the computational complexity required by the process of preparing the input data set, and the flexibility of the model in terms of application on data from different surveys. Models with a large number of weights need to be trained on very large data sets to avoid overfitting and underfitting. In an overfitting scenario, the model stores such specific details about the training set that it cannot generalize to unseen data. In contrast, in an underfitting scenario, which generally occurs for very complex models trained for a few epochs on a relatively small data set, the model is unable to learn from the training data the features necessary to generalize over the test set. Occam's razor principle suggests that, among models that achieve similar performance, the simplest model (with the least number of weights) should be preferred, since this choice minimizes the risk of overfitting and underfitting.

As described in Section 5, the use of a single classification layer dramatically reduces the number of network parameters (or weights), leading to a crucial advantage: keep model complexity low while retaining high predictive performance.

With 527,329 parameters, DART-Vetter is significantly smaller than more complex models such as Astronet-Triage-v2, which has over 100 million parameters. Despite this reduction in complexity, DART-Vetter achieves competitive F1-scores on different domains, demonstrating that a less complex network can achieve performance comparable to that of larger models, as previously shown by K. Visser et al. (2022a). This architectural simplification reduces the risks of overfitting and underfitting by improving the ability of the model to generalize on unseen data. It also limits the need for extensive hyperparameter fine-tuning or significant architectural modifications to deal with data sets from different missions, such as Kepler, TESS, or Planetary Transits and Oscillation (PLATO). Indeed, although models such as Exominer and Astronet-Triage-v2 are highly effective on TCEs belonging to the same domain as their respective training sets, their application on data from different missions would require substantial and computationally expensive architectural changes. An example of this is Exominer-basic, the version of Exominer adapted for TESS data. This adaptation from Kepler to TESS results in a decay of model performance by about 17% in terms of F1-score. These changes not only require an almost complete redesign of the network architecture but also significantly compromise model performance.

With DART-Vetter, the reduction in the number of trainable weights results in greater computational efficiency, both during training and in the preparation of inputs. In particular, high-dimensional inputs result in a larger number of parameters to be optimized and a high cost for data set generation because data from different sources need to be consistently combined.

Thus, our approach not only reduces computational costs but also makes the model inherently flexible in terms of adaptability to data from different missions, ensuring effective, efficient, and portable triage.

8. Current Limitation and Future Perspectives

Here, we address the current factors that constrain the predictive accuracy of our model with the aim of improving its performance and interpretability in future applications (A. Maratea & A. Ferone 2021).

8.1. Representing the Transit Shape

The global views we generated for each TCE consist in light curves that are binned and phase-folded over the TCEs periods. In this work, we found the same problem discussed in E. Tey et al. (2023). More precisely, binning and phase-folding operations involve aggregating flux measurements into discrete bins, leading to a reduction in the overall information content. Although these processes aid in the generation of a more compact input representation, they introduce uncertainties since they degrade the precision of the representation of the transit shape. In this study, we did not investigate the quantification of the impact resulting from the loss of information during phase-folding and binning. This issue remains a matter of interest and a particular focus will be given in future works.

8.2. The Bias of TCEs Dispositions

Our CNN for exoplanet detection is inevitably affected by biases introduced by human-generated labels, and its predictions lack explainability. During the vetting process, human vetters could inadvertently introduce their own biases or subjective judgments. This can result in a biased training data set and subsequently affects the performance of our model trained on such data. To address this issue, it is necessary to employ deep learning models specifically designed to learn from both labels and intrinsic patterns in the data. Among these models, we plan to use graph neural networks (GNNs; F. Scarselli et al. 2008; Z. Wu et al. 2020). Transforming an m-dimensional input vector (i.e., light curve) into a graph, with the aim of processing it by means of a GNN, could allow us to detect more complex nonlinear relationships and patterns within the input (W. Prummel et al. 2023). Consequently, the use of GNNs might enhance the model's interpretability and provide insights into patterns contributing to the discrimination of exoplanets from FPs.

9. Conclusions

The CNNs for exoplanets detection have demonstrated remarkable predictive performance, as shown by the works of H. Valizadegan et al. (2022), E. Tey et al. (2023), and H. Valizadegan et al. (2025), while also maintaining simplicity in implementation, as seen in the study conducted by K. Visser et al. (2022b). Building upon these achievements, we developed an effective and easily replicable CNN to automatically distinguish planetary transits from FPs.

Starting from the promising outcomes obtained in our previous work (S. Fiscale et al. 2023), we deployed `DART-Vetter`, a triage model capable of performing both binary and multiclass classification on TCEs from different surveys. Our results emphasize the ability of the model in solving these tasks when assessed on single-domain (Kepler and TESS) and cross-domain (TESS + Kepler DR24) data sets, achieving recall values on test sets greater than 92%. On the other hand, our model does not achieve good performance on Kepler DR25 TCEs, with a high fraction of FNs of 40%. As shown in Section 6.3, `DART-Vetter` struggles with the classification of Kepler TCEs orbiting their stars in more than 50 days, or with MES lower than 20.

By comparing `DART-Vetter` with other CNNs for exoplanet detection we found that the performance is lower in terms of absolute accuracy with respect to state-of-the-art models such as `Exominer`, `Exominer++` and `Astronet-Triage-v2`. But the strength of our model lies in its straightforward adaptability across binary/multiclass classification on several data sets. This is accomplished thanks to the reduced model complexity, which we obtained mainly by minimizing the input dimensionality and the number of classification layers, together with the relative number of neurons. Unlike top performing models, which require high computational complexity for data collection, preprocessing, and network implementation, `DART-Vetter` offers an optimal balance between predictive performance and implementation simplicity, minimizing the FNs fraction for planets with short orbital periods (except for the TCEs of $\mathcal{D}_{K25}$ where more in-depth evaluations have been presented in Section 6.3) by needing only period-folded and binned light curves.

Appendix A provides the details related to the model selection phase, while in Appendix B we show how the model's performance varies when training and testing on different data set splits.

Acknowledgments

This paper includes data collected by the Kepler mission and obtained from the MAST data archive at the Space Telescope Science Institute (STScI). Funding for the Kepler mission is provided by the NASA Science Mission Directorate. STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS 5-26555.

This paper includes data collected with the TESS mission, obtained from the MAST data archive at the Space Telescope Science Institute (STScI). Funding for the TESS mission is provided by the NASA Explorer Program. STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS 5-26555.

This research has made use of the Exoplanet Follow-up Observation Program (ExoFOP; DOI: 10.26134/ExoFOP5) website, which is operated by the California Institute of Technology, under contract with the National Aeronautics and Space Administration under the Exoplanet Exploration Program.

This research has made use of the NASA Exoplanet Archive, which is operated by the California Institute of Technology, under contract with the National Aeronautics and Space Administration under the Exoplanet Exploration Program.

Appendix A

The model selection phase is aimed at determining the best-performing network on the test set and it involves training and evaluating multiple models with varying architectures and hyperparameters (e.g., learning rate, batch size). We conducted this phase on the Kepler DR24 data set as the one characterized by a more balanced number of positive and negative TCEs, with a discrete S/N when compared to other sources of the signals. The hyperparameters and the relative grids of values are listed in Table 5, where the values in bold represent the suboptimal

Table 5
List of Hyperparameters and their Respective Values on Which we Based the Grid Search Process

Hyperparameter	Values
Activation function	**ReLU**, Leaky ReLU
Batch size	64, **128**, 256, 512, 768, 1024
Dropout rate	**0.2**, 0.3, 0.4, 0.5
Training epochs	10:10:100, 150, 200, 300, 400, 500
Fully connected neurons	256, **512**, 768
Layer weights initializer	Uniform, **Lecun Uniform**, Random normal, Zero, Glorot Normal, Glorot Uniform,He Normal, He Uniform
Learning rate	1e-2, **1e-3**, 1e-4, 1e-5
Optimizer	**Adam**

Note. For each hyperparameter, a set of discrete values is defined. The grid search process trained the model based on all possible combinations among the distinct values that each hyperparameter can assume. We highlight in bold the values we used to train the best model.

configuration returned by the grid search. In Table 5, the training epochs increase by 10 in the interval [10, 100], and its optimal value depends on the input data set. The upper limit of fully connected neurons was set at 768 because it corresponds to the output size of the feature extraction branch. The dropout rate is evaluated on a discrete grid in [0.2, 0.5] because this is the interval suggested in N. Srivastava et al. (2014). The underlined configuration is defined as suboptimal because the fixed grids assume values in finite and discrete intervals.

We began with the base network architecture designed by S. Fiscale et al. (2023). This architecture processes two views of the same TCE: the global view and local view, which is a zoomed-in representation of the first. To create a simplified architecture, denoted as $\mathcal{M}$, we initially (i) removed the branch processing the local view and (ii) reduced the size of the convolutional filters from 5×1 to 3×1. We removed the local view because it is obtained from the global view using a different binning size. Thus, the local view is linearly dependent with respect to the global view. In I. Guyon & A. Elisseeff (2003), it was shown that processing linearly dependent input data adds further complexity to the problem but no additional information is gained by considering them. We adopted the use of smaller convolutional filters after D. C. Ciresan et al. (2011) and K. Simonyan & A. Zisserman (2015) proved their effectiveness in capturing more complex relationships in the input data than larger filters. The use of a smaller convolutional window exactly suits our needs since the most relevant information in our input samples (the transit points) might also be represented by a single point in the global view (see Figure 3 of C. J. Shallue & A. Vanderburg 2018).

Table 6
Summary of the Characteristics of the Model Before (S. Fiscale et al. 2023) and After (This Work) the Model Selection Phase

	S. Fiscale et al. (2023)	This Work
Input		
# Views	2	1
Feature Extraction branch		
# Convolutional blocks	5+2	5
Convolutional filters	5×1	3×1
Classification branch		
# Dense blocks	1	1
# Neurons	768	512
Batch normalization	yes	no
Number of parameters		
Feature Extraction branch	∼670,000	∼127,000
Classification branch	∼1,260,000	∼400,000
Total	∼1,927,000	∼527,000

Note. The element denoted with the "+" symbol corresponds to the values for the two branches processing the global and local views.

The model selection process was then conducted in two steps:

STEP 1. Based on $\mathcal{M}$, we trained a set of models by varying their depth (from five to three convolutional blocks) and evaluating the use of batch normalization layers within the convolutional and classification blocks.

STEP 2. We modified each convolutional block by removing one of the two triplets (1D convolution layer—Rectified Linear Unit—Dropout) and repeated STEP 1 for one iteration.

This methodology led to the architecture of the model presented in this study, `DART-Vetter`, characterized by ∼1,400,000 parameters less to be optimized during the training process. The differences in network architecture between the base model proposed in S. Fiscale et al. (2023) and `DART-Vetter` obtained through model selection are summarized in Table 6.

Appendix B

As depicted in Figure 8, the values of precision, recall, and F1-score computed on each test set do not exhibit significant changes when varying the data splits used for training and testing the model. This outcome highlights the ability of `DART-Vetter` in guaranteeing optimal performance on the data set under analysis, even when the training set is relatively small. Such a stability in performance across different training and test set fractions can be mainly attributed to the reduced number of trainable parameters of the model. Overall, optimal performance in terms of F1-score is achieved when training-testing the model on the data split 70–30.

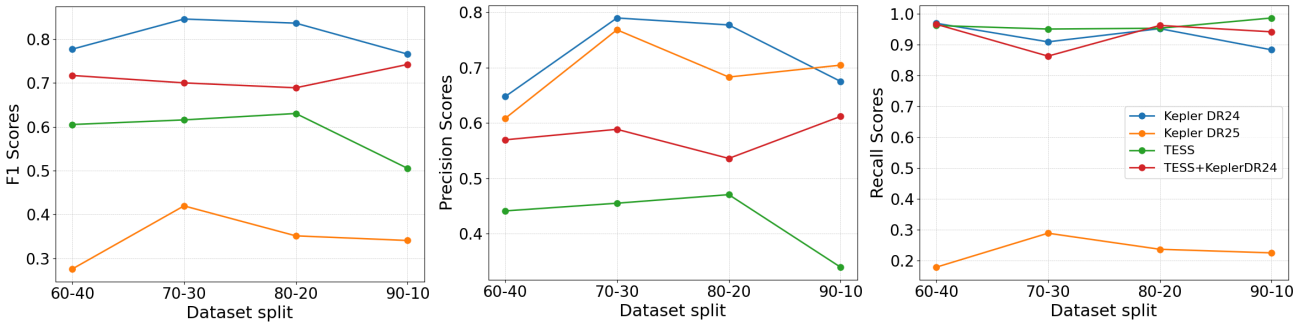

Figure 8. Variation of the main evaluation metrics as the training and test data sets are split according to the different following ratios: 60–40, 70–30, 80–20, and 90–10. Each line shows the scores of F1 (left panel), precision (middle panel), and recall (right panel) for a given data set.

ORCID iDs

Stefano Fiscale https://orcid.org/0000-0001-8371-8525
Laura Inno https://orcid.org/0000-0002-0271-2664
Alessandra Rotundi https://orcid.org/0000-0001-5467-157X
Luca Cacciapuoti https://orcid.org/0000-0001-8266-0894
Veselin Kostov https://orcid.org/0000-0001-9786-1031
Giovanni Covone https://orcid.org/0000-0002-2553-096X

References

Akeson, R., & Christiansen, J. 2019, AAS Meeting, 223, 140.09
Ansdell, M., Ioannou, Y., Osborn, H. P., et al. 2018, ApJL, 869, L7
Armstrong, D. J., Günther, M. N., McCormac, J., et al. 2018, MNRAS, 478, 4225
Bryson, S. T., Tenenbaum, P., Jenkins, J. M., et al. 2010, ApJL, 713, L97
Bryson, S. T., Jenkins, J. M., Gilliland, R. L., et al. 2013, PASP, 125, 889
Cacciapuoti, L., Kostov, V. B., Kuchner, M., et al. 2022, MNRAS, 513, 102
Catanzarite, J. H. 2015, Autovetter Planet Candidate Catalog for Q1-Q17 DataRelease 24 (KSCI-19090-001), Tech. Rep.
Chaushev, A., Raynard, L., Goad, M. R., et al. 2019, MNRAS, 488, 5232
Christiansen, J. L., Jenkins, J. M., Caldwell, D. A., et al. 2012, PASP, 124, 1279
Ciresan, D. C., Meier, U., Masci, J., Gambardella, L. M., & Schmidhuber, J. 2011, in IJCAI'11, Twenty-second Int. Joint Conf. Artificial Intelligence, ed. T. Walsh (Washington, DC: AAAI Press), 1237
Claeskens, G., Hjort, N. L., et al. 2008, Model Selection and Model Averaging (Cambridge: Cambridge Univ. Press)
Coughlin, J. L., Mullally, F., Thompson, S. E., et al. 2016, ApJS, 224, 12
Cybenko, G. 1989, MCSS, 2, 303
Dattilo, A., Vanderburg, A., Shallue, C. J., et al. 2019, AJ, 157, 169
De Luca, P., Fiscale, S., Landolfi, L., & Di Mauro, A. 2019, in Lecture Notes in Computer Science, 1874, 12th Int. Conf. on Internet and Distributed Computing Systems (IDCS 2019), ed. R. Montella et al. (Berlin: Springer), 369
De Luca, P., Galletti, A., & Marcellino, L. 2022, in 2022 16th Int. Conf. on Signal-Image Technology & Internet-Based Systems (SITIS), ed. J.-L. Dekeyser & M. Mokhtari (New York: IEEE), 530
Durairaj, D. M., & Mohan, B. K. 2022, Neural Comput. Appl., 34, 13319
Ferone, A., & Petrosino, A. 2017, in WILF 2016, Fuzzy Logic and Soft Computing Applications, ed. A. Petrosino, V. Loia, & W. Pedrycz (Berlin: Springer), 116
Fiscale, S., Ferone, A., Ciaramella, A., et al. 2025, Electronics, 14, 1738
Fiscale, S., Luca, P. D., Inno, L., et al. 2021, in in Lecture Notes in Computer Science, 12745, Int. Conf. Computational Science (ICCS 2021), ed. J. F. G. de Freitas et al. (Berlin: Springer), 420
Fiscale, S., Inno, L., Ciaramella, A., et al. 2023, Applications of Artificial Intelligence and Neural Systems to Data Science (Berlin: Springer), 127
Gaia Collaboration, Babusiaux, C., van Leeuwen, F., et al. 2018, A&A, 616, A10
Giacalone, S., Dressing, C. D., Jensen, E. L. N., et al. 2021, AJ, 161, 24
Guerrero, N. M., Seager, S., Huang, C. X., et al. 2021, ApJS, 254, 39
Guyon, I., & Elisseeff, A. 2003, JMLR, 3, 1157, https://www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf
Hadjigeorghiou, A., & Armstrong, D. J. 2024, MNRAS, 527, 4018
Hornik, K., Stinchcombe, M., & White, H. 1989, NN, 2, 359
Howell, S. B., Sobeck, C., Haas, M., et al. 2014, PASP, 126, 398
Hu, X., Chu, L., Pei, J., Liu, W., & Bian, J. 2021, KAIS, 63, 2585
Huang, C. X., Vanderburg, A., Pál, A., et al. 2020a, RNAAS, 4, 204
Huang, C. X., Vanderburg, A., Pál, A., et al. 2020b, RNAAS, 4, 206
Im, M. S., & Dasari, V. R. 2022, Math. Milit., 25, 1, https://core.ac.uk/download/524878644.pdf
Ioffe, S., & Szegedy, C. 2015, in Int. Conf. on Machine Learning, 37, ed. F. Bach & D. Blei (Cambridge, MA: PMLR), 448
Jara-Maldonado, M., Alarcon-Aquino, V., & Rosas-Romero, R. 2020, in Mexican International Conf. on Artificial Intelligence, ed. L. Martínez-Villaseñor, O. Herrera-Alcántara, H. Ponce, & F. A. Castro-Espinoza (Berlin: Springer), 50
Jenkins, J. M., Caldwell, D. A., & Borucki, W. J. 2002, ApJ, 564, 495
Jenkins, J. M., Caldwell, D. A., Chandrasekaran, H., et al. 2010, ApJL, 713, L87
Jenkins, J. M., Chandrasekaran, H., McCauliff, S. D., et al. 2010, Proc. SPIE, 7740, 77400D
Jenkins, J. M., Twicken, J. D., McCauliff, S., et al. 2016, Proc. SPIE, 9913, 99133E
Kingma, D. P., & Ba, J. 2015, in Proc. 3rd Int. Conf. Learning Representations (ICLR 2015), ed. Y. Bengio & J. Ba (San Diego: CA: ICLR / OpenReview.net), 2015
Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S. 2017, in 31st Annual Conference on Neural Information Processing Systems (NIPS 2017), ed. I. Guyon, V. Von Luxburg, S. Bengio et al. (Red Hook, NY: Curran Associates), 972
Koch, D. G., Borucki, W. J., Basri, G., et al. 2010, ApJL, 713, L79
Kostov, V. B., Mullally, S. E., Quintana, E. V., et al. 2019, AJ, 157, 124
Kovács, G., Zucker, S., & Mazeh, T. 2002, A&A, 391, 369
Kunimoto, M., Huang, C., Tey, E., et al. 2021, RNAAS, 5, 234
Kunimoto, M., Tey, E., Fong, W., et al. 2022, RNAAS, 6, 236
LeCun, Y., Bengio, Y., & Hinton, G. 2015, Natur, 521, 436
Lertnattee, V., & Theeramunkong, T. 2004, in Proc. IEEE Int. Symp. Communications and Information Technology, 2, (ISCIT 2004), ed. T. Odake, P. B. L. Wimalasundera, & S. Sutivong (Piscataway, NJ: IEEE), 1171
Liao, H., Ren, G., Chen, X., Li, Y., & Li, G. 2024, AJ, 167, 180
Magliano, C., Covone, G., Dobal, R., et al. 2022, MNRAS, 519, 1562
Magliano, C., Kostov, V., Cacciapuoti, L., et al. 2023, MNRAS, 521, 3749
Maratea, A., & Ferone, A. 2021, in WILF 2021, International Workshop on Fuzzy Logic and Applications, ed. A. Ciaramella, C. Mencar, S. Montes, & S. Rovetta (Aachen: CEUR Workshop Proceedings), 3074
MAST Team 2021, TESS Light Curves - All Sectors, STScI/MAST, doi:10.17909/t9-nmc8-f686
McCauliff, S. D., Jenkins, J. M., Catanzarite, J., et al. 2015, ApJ, 806, 6
Mislis, D., Bachelet, E., Alsubai, K. A., Bramich, D. M., & Parley, N. 2016, MNRAS, 455, 626
Morris, R. L., Twicken, J. D., Smith, J. C., et al. 2020, in Kepler Data Processing Handbook: Photometric Analysis, KSCI-19081-003, ed. J. M. Jenkins, 6 (Moffett Field, CA: NASA Ames Research Center), 107
Morton, T. D., 2015 VESPA: False Positive Probabilities Calculator, Astrophysics Source Code Library, ascl:1503.011
Morton, T. D., Bryson, S. T., Coughlin, J. L., et al. 2016, ApJ, 822, 86
Mullally, F., Coughlin, J. L., Thompson, S. E., et al. 2016, PASP, 128, 074502
Orsini, M. G., Ferone, A., Inno, L., et al. 2025, A&C, 52, 100964
Prummel, W., et al. 2023, in Proc. 2023 IEEE Int. Conf. Image Processing (ICIP), ed. P. S. Chung, M. Etoh, & H. Fujiyoshi (Piscataway, NJ: IEEE), 2730
Pudjihartono, N., Fadason, T., Kempa-Liehr, A. W., O'Sullivan, J. M., et al. 2022, Front. Bioinform., 2, 927312
Ribani, R., & Marengoni, M. 2019, in 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials, SIBGRAPI-T 2019 (Piscataway, NJ: IEEE), 57

Ricker, G. R., Winn, J. N., Vanderspek, R., et al. 2014, Proc. SPIE, 9143, 914320
Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. 2008, ITNN, 20, 61
Schanche, N., Collier Cameron, A., Hébrard, G., et al. 2019, MNRAS, 483, 5534
Seader, S., Jenkins, J. M., Tenenbaum, P., et al. 2015, ApJS, 217, 18
Shallue, C. J., & Vanderburg, A. 2018, AJ, 155, 94
Simonyan, K., & Zisserman, A. 2015, in Proc. 3rd Int. Conf. Learning Representations (ICLR 2015), ed. Y. Bengio & Y. LeCun (San Diego, CA: Computational and Biological Learning Society), 1
Smith, J. C., Stumpe, M. C., Van Cleve, J. E., et al. 2012, PASP, 124, 1000
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. 2014, JMLR, 15, 1929
Stassun, K. G., Oelkers, R. J., Paegert, M., et al. 2019, AJ, 158, 138
STScI 2016, Kepler LC+SC, Q0-Q17, STScI/MAST, doi: 10.17909/T98304
Szegedy, C., Liu, W., Jia, Y., et al. 2015, in Proc. the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), ed. R. Szeliski, B. Freeman, & M. Hebert (Piscataway, NJ: IEEE), 1
Tey, E., Moldovan, D., Kunimoto, M., et al. 2023, AJ, 165, 95
Thompson, S. E., Coughlin, J. L., Hoffman, K., et al. 2018, ApJS, 235, 38
Tonry, J. L., Denneau, L., Flewelling, H., et al. 2018, ApJ, 867, 105
Twicken, J. D., Chandrasekaran, H., Jenkins, J. M., et al. 2010, Proc. SPIE, 7740, 77401U
Twicken, J. D., Jenkins, J. M., Seader, S. E., et al. 2016, AJ, 152, 158
Valizadegan, H., Martinho, M. J., Jenkins, J. M., et al. 2025, arXiv:2502.09790
Valizadegan, H., Martinho, M. J. S., Jenkins, J. M., et al. 2023, AJ, 166, 28
Valizadegan, H., Martinho, M. J. S., Wilkens, L. S., et al. 2022, ApJ, 926, 120
Vanderburg, A., & Johnson, J. A. 2014, PASP, 126, 948
Visser, K., Bosma, B., & Postma, E. 2022a, JAI, 11, 2250011
Visser, K., Bosma, B., & Postma, E. 2022b, A&C, 41, 100654
Wu, Z., Pan, S., Chen, F., et al. 2020, TNNLS, 32, 4
Yu, L., Vanderburg, A., Huang, C., et al. 2019, AJ, 158, 25
Zacharias, N., Finch, C., & Frouard, J. 2017, AJ, 153, 166
Zacharias, N., Finch, C. T., Girard, T. M., et al. 2013, AJ, 145, 44
Zhao, B., Lu, H., Chen, S., Liu, J., & Wu, D. 2017, JSEE, 28, 162

2.3.3 "Detection of Exoplanets in Transit Light Curves with Conditional Flow Matching and XGBoost" [24]

We introduced a new approach and extended binary classification to multi-class classification of *Kepler* and TESS TCEs in Paper 3 [24] (attached at the end of this section). Figure 2.4 illustrates the architectural scheme of our new model consisting of three main modules: feature extraction, dimensionality reduction and classification.

We used VGG-19 as feature extractor as VGG-like architectures already proved their effectiveness in processing global and local views [75]. We then take the feature vectors produced by the last convolutional layer of VGG-19, and project them in a two-dimensional space by using the t-Stochastic Neighbor Embedding algorithm (t-SNE, [54]). The two-dimensional feature vectors are classified with a diffusion model consisting in a Conditional Flow Matching [43].

With this approach, we achieved a significant improvement in the model's predictive performance, reaching an overall TCE classification accuracy of ∼98%. The use of t-SNE to project high-dimensional feature vectors into a two-dimensional space introduced a substantial advantage in terms of interpretability: the distribution of data in the projected plane can be visually inspected, allowing for an intuitive understanding of how the model separates different classes based on the learned features. This aspect is particularly relevant in the context of TCE analysis, where transparency in the decision-making process is crucial for the validation of potential planet candidates.

The decision to employ the Conditional Flow Matching as classifier was motivated by the results presented in [43], where the authors combined the classification capability of diffusion-based models [64, 51], with a training scheme based on eXtreme Gradient Boosting trees. This configuration is particularly well-suited for handling tabular data, such as those produced after the t-SNE projection.

By integrating these components, the proposed model achieved a well-balanced trade-off between accuracy, efficiency, and interpretability, marking a significant step toward the development of modular architectures for the multi-class classification of TCEs originating from different space missions.

Hardware specifications. The computational experiments and the performance benchmarks described in the following paper were conducted on a high-end workstation designed to represent a locally accessible environment for the scientific community. The system is equipped with an Intel Core i9-13900KF processor (13th Gen) featuring 24 cores and 32 threads with a maximum clock speed of 5.8 GHz, supported by 64 GB of RAM. High-speed data I/O was ensured by a Corsair MP600 CORE XT NVMe SSD. Although the models are optimized for CPU efficiency, an NVIDIA discrete GPU was

available in the system for general-purpose acceleration. This setup confirms that the proposed AI pipeline can be executed efficiently on standard, non-HPC infrastructure.

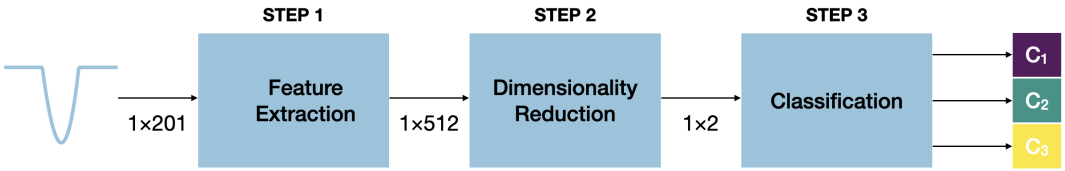

Figure 2.4: Architectural diagram of our new model, based on three main modules: feature extraction, dimensionality reduction and classification. The model is trained on 1×201 global views to produce a multi-class classification in astrophysical false positive (EB, purple box), planet candidate (PC, green box) and non-transiting phenomenon (J, yellow box).

Overview of the main findings. In summary, the evolution of the three models developed reflects a path of progressive architectural simplification, moving towards modularity and interpretability.

Our results have quantitatively demonstrated that this process of seeking efficiency and generalization should constitute a true design philosophy. Conclusions on this topic are drawn in Chapter 4.

electronics

MDPI

Article

Detection of Exoplanets in Transit Light Curves with Conditional Flow Matching and XGBoost

Stefano Fiscale [1,2,*], Alessio Ferone [3], Angelo Ciaramella [3], Laura Inno [1,2,3], Massimiliano Giordano Orsini [1], Giovanni Covone [2,4,5] and Alessandra Rotundi [1,3]

[1] UNESCO Chair "Environment, Resources and Sustainable Development", Department of Science and Technology, Parthenope University of Naples, 80133 Naples, Italy; laura.inno@uniparthenope.it (L.I.); massimiliano.giordanoorsini001@studenti.uniparthenope.it (M.G.O.); alessandra.rotundi@uniparthenope.it (A.R.)
[2] Istituto Nazionale di Astrofisica, Osservatorio Astronomico di Capodimonte, Salita Moiariello, 16, 80131 Naples, Italy
[3] Department of Science and Technology, Centro Direzionale di Napoli, Parthenope University of Naples, 80143 Naples, Italy; alessio.ferone@uniparthenope.it (A.F.); angelo.ciaramella@uniparthenope.it (A.C.)
[4] Department of Physics "Ettore Pancini", University of Naples Federico II, 80138 Naples, Italy; giovanni.covone@unina.it
[5] INFN Section of Naples, Via Cinthia 6, 80126 Naples, Italy
* Correspondence: stefano.fiscale001@studenti.uniparthenope.it

Abstract: NASA's space-based telescopes Kepler and Transiting Exoplanet Survey Satellite (TESS) have detected billions of potential planetary signatures, typically classified with Convolutional Neural Networks (CNNs). In this study, we introduce a hybrid model that combines deep learning, dimensionality reduction, decision trees, and diffusion models to distinguish planetary transits from astrophysical false positives and instrumental artifacts. Our model consists of three main components: (i) feature extraction using the CNN VGG19, (ii) dimensionality reduction through t-Distributed Stochastic Neighbor Embedding (t-SNE), and (iii) classification using Conditional Flow Matching (CFM) and XGBoost. We evaluated the model on two Kepler and one TESS datasets, achieving F1-scores of 98% and 100%, respectively. Our results demonstrate the effectiveness of VGG19 in extracting discriminative patterns from data, t-SNE in projecting features in a lower dimensional space where they can be most effectively classified, and CFM with XGBoost in enabling robust classification with minimal computational cost. This study highlights that a hybrid approach leveraging deep learning and dimensionality reduction allows one to achieve state-of-the-art performance in exoplanet detection while maintaining a low computational cost. Future work will explore the use of adaptive dimensionality reduction methods and the application to data from upcoming missions like the ESA's PLATO mission.

Keywords: exoplanet detection; deep learning; dimensionality reduction; diffusion models; decision trees

Academic Editor: Ricardo Martins

Received: 7 March 2025
Revised: 19 April 2025
Accepted: 21 April 2025
Published: 24 April 2025

Citation: Fiscale, S.; Ferone, A.; Ciaramella, A.; Inno, L.; Giordano Orsini, M.; Covone, G.; Rotundi, A. Detection of Exoplanets in Transit Light Curves with Conditional Flow Matching and XGBoost. *Electronics* **2025**, *14*, 1738. https://doi.org/10.3390/electronics14091738

1. Introduction

Since the discovery of 51 Pegasi b [1], the identification of exoplanets—planets orbiting stars other than the Sun—has become one of the most rapidly evolving research fields combining a wide range of expertise from astrophysics to data science [2]. Over the past two decades, space-based telescopes such as NASA's Kepler [3] and the Transiting Exoplanet Survey Satellite (TESS) [4] have revolutionized this field by collecting photometric measurements from hundreds of thousands of stars. By using the transit method [5], these telescopes have identified a large number of periodic signals due to real planets,

astrophysical events (e.g., eclipsing binaries and stellar variability), and other phenomena (e.g., instrumental systematics). The human-based analysis is the most reliable approach to classify these signals as expert astronomers can handle a wide range of possible scenarios based on their expertise [6,7]. However, the manual examination of these signals presents two major drawbacks. First, human judgment is not objective, and some astronomers might disagree on labels assigned to some signals. Second, this process is highly time-consuming considering that astronomers need to be trained on this task [8] and that labeling a single signal might require from a few hours up to several days as the visual examination of Data Validation reports provided for the signal of interest [9] is required at least.

To address these issues, Convolutional Neural Networks (CNNs) [10] became the standard in classifying these signals [11–15], from their first implementation by Shallue C. and Vanderburg A. [16] (hereinafter SV18) with Astronet. These CNNs detect the most relevant patterns in transit signals through a feature extraction block—which is typically based on the architecture of the CNN VGG19 [17]—thus performing classification leveraging the universal approximation property of a Multi-Layer Perceptron (MLP) [18–20]. Over time, the architecture of these networks has been optimized, leading to significant performance gains up to 99% of classification accuracy on real exoplanet signals [21,22].

However, such CNNs are designed assuming that features useful for humans in their analysis are equally relevant to the model in solving the task at hand. Processing linearly dependent input features unnecessarily increase the complexity of the network [23], both in terms of data collection and preparation, and model's parameter optimization. Moreover, the higher the number of model's parameters requiring optimization, the larger the volume of training data needs to be in order for the optimization algorithm to converge to a stable local minimum; in this field, the ratio of the dataset size to the number of model parameters is still heavily skewed toward the latter. CNNs face two major limitations: their classifier is highly complex, consisting of hundreds of thousands to millions of parameters, and they lack interpretability, which is crucial here to understand the reasoning behind model predictions.

Decision trees such as Random Forests (RFs) [24] and Gradient Boosted Trees (GBTs) [25,26], including XGBoost, have demonstrated their effectiveness in approximating complex distributions with lower computational costs than MLPs. These models are universal approximators like MLPs [27] but obtain particularly better classification performance on tabular data [28]. Previous efforts demonstrated the effectiveness of RF classifiers in classifying planetary candidates across ground- and space-based surveys [29–32].

Another promising approach to preserve classification accuracy while reducing model complexity is dimensionality reduction (DR). Methods like t-Distributed Stochastic Neighbor Embedding (t-SNE) [33] can effectively project high-dimensional data into lower-dimensional embedding while preserving data structures, making them highly suitable for data processing before classification. Integrating the potential of these models can facilitate the development of a more efficient classifier. In this context, Armstrong D. et al. [30] and Schanche N. et al. [32] introduced innovative approaches, respectively, employing the combination of RFs and a Self-Organizing Map (SOM) for classifying transit signals in the Next-Generation Transit Survey (NGTS) [34] data and RFs coupled with a CNN for processing data from the Wide Angle Search for Planets (WASP) [35] survey.

In this paper, we propose an innovative approach to perform a multi-class classification of the signals detected in Kepler and TESS data, in planet candidate (PC), astrophysical false positive (AFP), and non-transiting phenomenon (NTP). Our approach is based on a model consisting of three main components:

1. Feature extraction, performed using the widely adopted CNN VGG19, which transforms input signals into high-dimensional feature vectors;

2. Dimensionality reduction, performed by the t-SNE method, which maps the high-dimensional features to a lower dimensional space, where they can be most effectively classified;
3. Classification, implemented by Conditional Flow Matching (CFM) and XGBoost [36].

Our model achieves competitive performance compared to the state of the art, with an F1-score of 99% on Kepler data and 100% on a TESS dataset, operating on very small inputs in size terms compared to other approaches in the literature.

Our results reveal that the application of t-SNE on the feature vectors produced by VGG19 enhances classification capabilities of the model than classical VGG-based CNNs classifying feature vectors with a MLP.

With this model, we continue to build on the most relevant architectures in the context of exoplanets detection (i.e., CNNs and decision trees), with the innovation of merging them in a single data processing pipeline. This work highlights the advantages of combining deep learning with dimensionality reduction and decision tree classifiers, offering an effective and efficient solution for exoplanet detection.

The rest of the paper is organized as follows: Related works are presented in Section 2, where we also define the contribution and novelty of our approach. The Kepler and TESS data used in this work are described in Section 3.1, while a theoretical background on the three main components of our model is provided in Section 3.2. We explain our model's architectural details in Section 4, showing the results in Section 5. A discussion is reported in Section 6, including a comparison with related works, and conclusions are drawn in Section 7.

2. Related Works

This section provides an overview of the evolution of ML models for the classification of TCEs. Since numerous contributions have been made in this field, we summarize in Table 1 the specifics about the most relevant model architectures in order to highlight the key differences between prior studies and our approach.

The first ML models developed for the binary classification of TCEs were based on RFs, namely, Autovetter [29] and Robovetter [37]. These models were employed to classify thousands of Kepler TCEs and played a key role in generating two of the largest labeled datasets available for this survey: Kepler Q1–Q17 Data Release 24 and 25. Both approaches were designed to process a broad set of inputs, including scalar planetary features, centroid motion and difference image analysis, odd–even transit differences, secondary view, and phase-folded light curves (all these features are described in the caption of Table 1).

Subsequent efforts demonstrated that integrating different ML techniques lead to better classification performance. SOMs were applied to Kepler and K2 [38] data [39], while RF and SOM (RFC + SOM) combinations were tested on NGTS [30,34] data, and a model based on RF and CNN was applied on WASP data [32,35]. These hybrid approaches leveraged the strengths of different methods to improve robustness in TCE classification. These models used a limited set of input features, with SOM processing phase-folded light curves representing the transit shapes, while RFC + SOM integrates planetary parameters without centroid or secondary eclipse information. Although centroid information could help in identifying some false-positive scenarios such as background eclipsing binaries, in their work, Armstrong D. et al. [30] decided to not use this feature due to the risk to discard blended transiting planets, which instead remain candidates worthy of further analysis.

The introduction of deep learning revolutionized TCE classification, with CNN-based models becoming the standard. Astronet [16] was one of the first CNNs specifically designed for Kepler TCEs classification, processing a global view and its zoomed-in represen-

tation (the local view). Since then, CNN-based models have been used on various surveys, including K2 [11] and TESS [13,40]. Some models further increased input dimensionality by incorporating stellar and transit parameters alongside multiple light curves representations [14,41]. The evolution of CNNs culminated in two of the best models: Exominer [21] for Kepler and Astronet-Triage-v2 [22] for TESS. As shown in Table 1, both models process a combination of light curve representations (e.g., global and local views and secondary eclipse) and stellar and transit parameters to classify TCEs. Given that the input information processed by these models corresponds almost entirely to that used by astronomers during manual vetting, efforts to enhance their interpretability have become increasingly relevant. In this context, a notable contribution was proposed by Salinas H. et al. [42]. The authors of this study presented a Transformer-based approach for the binary classification of TESS TCEs, processing global and local views, centroid information, and stellar and planetary features.

Table 1. Comparison of different model architectures. For each model, we report the network architecture and the classification task (binary: 2c, or multi-class: >2c) and, if the model uses stellar features (St.f), planetary features (Pl.f, including transit period, transit duration, transit depth difference, etc.), centroid information (C, pixel-level information about the location of the variation in brightness for the detected transit), phase-folded flux (Pff, consisting of light curves with different data binning such as global and local views), odd–even (two consecutive transits), secondary view (Sv, consisting of the dip in star brightness when the detected object passes behind its star), difference image (Diff.img, used to evaluate whether the transit occurs out of the central pixel where the target star is supposed to be). The ✓ symbol indicates that the corresponding feature is used as an input parameter in the model; conversely, the symbol × indicates that the feature is not employed by the model.

Model [Ref.]	Architecture	Task	St.f.	Pl.f.	C	Pff	Odd–Even	Sv	Diff.img.
Robovetter	Decision Tree	3c	×	✓	✓	✓	✓	✓	✓
Autovetter	Decision Tree	3c	✓	✓	✓	✓	✓	✓	✓
Armstrong D. et al. [39]	SOM	2c	×	×	×	✓	×	×	×
Armstrong D. et al. [30]	RFC + SOM	2c	×	✓	×	×	×	×	×
Astronet	CNN	2c	×	×	×	✓	×	×	×
Astronet-K2 [11]	CNN	2c	✓	✓	×	✓	×	×	×
Exonet [41]	CNN	2c	✓	×	✓	✓	×	×	×
Genesis [43]	CNN	2c	✓	×	✓	✓	×	×	×
Astronet-Triage [13]	CNN	2c	×	×	×	✓	×	×	×
Astronet-Vetting [13]	CNN	2c	×	✓	×	✓	×	✓	×
Astronet-Triage-v2	CNN	5c	✓	✓	×	✓	×	✓	×
Exominer	CNN	2c	✓	✓	✓	✓	✓	✓	✓
Salinas H. et al. [42]	Transformer	2c	✓	✓	✓	✓	×	×	×
This work	CNN + DR + CFM + XGBoost	3c	×	×	×	✓	×	×	×

Our approach integrates the key strengths of previous models into a unified framework. We adopt CNN architecture based on VGG19 as feature extractor, a choice consistent with previous works such as Astronet, Astronet-Triage, and Astronet-Triage-v2. However, instead of relying on the CNN for end-to-end classification as previous works already tested, we show that classification performance improves when features are projected into lower-dimensional spaces by t-SNE and that decision trees are used to exploit their capabilities in discriminating tabular data. We simplified the input to only the global view, deciding not to process as input the other light curves that differ from the global view in binning size (e.g., local view). Furthermore, the dimensionality reduction using t-SNE enhances computational efficiency and enables interpretability through the visualization of data in a two-dimensional space, providing insight into model predictions. A performance comparison with state-of-the-art models is provided in Section 6.4.

3. Background

Since our aim is developing a model to classify signals detected in transit light curves, this section is intended to provide all the necessary background information about the data used in this work (Section 3.1) and the operation of the methods that make up our model (Section 3.2).

3.1. Data

We work with light curves in which periodic transits of potential planetary nature, called Threshold-Crossing-Events (TCEs), were detected. Section 3.1.1 briefly explains how light curves are produced by the Kepler and TESS telescopes and how TCEs are detected. For a more detailed overview on this, we refer the reader to Jenkins J. et al. [44] and Jenkins J. et al. [45]. Section 3.1.2 provides details about the composition of TCEs datasets used. The process of input data preparation is described in Section 3.1.3.

3.1.1. From Light Curves to Threshold Crossing Events

The Kepler and TESS space telescopes were designed to gather photometric observations for a set of target stars from which light curves are extracted with a technique called aperture photometry. This technique consists of extracting the flux values of a target star from each photometric observation by summing the pixel values within a predefined aperture, optimized to maximize the signal-to-noise ratio (SNR). The resulting one-dimensional signal represents the variation in stellar brightness as a function of time, forming a light curve. An example of Kepler light curve is depicted in the top panel of Figure 1.

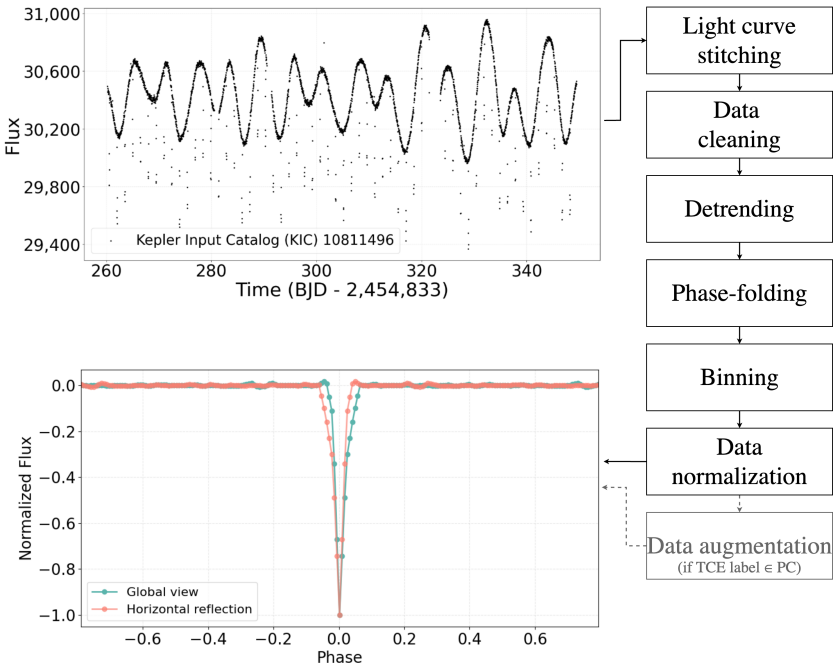

Figure 1. Our data preparation pipeline processes each TCE through several steps. First, we retrieve the corresponding light curve files from the MAST archive (for sake of simplicity, we show a single Kepler quarter for Kepler Input Catalog (KIC) star 10811496). These light curves are merged into a single signal (light curve stitching step), from which not-a-numbers and outliers have been filtered out (data cleaning step). During the detrending step, we eliminate any non-transit flux variation. The cleaned signal is phase-folded according to the TCE's period (phase-folding). Finally, we bin and normalize the data to create a 1×201-length global view, where the median flux is set to 0 and the maximum transit depth to -1 (binning and data normalization steps). The resulting signal is horizontally reflected if it belongs to the PC class (data augmentation step).

During its nine year service until 2018, Kepler monitored approximately 156,000 stars in a fixed region of the sky up to 4000 light years from the Earth, in the constellations of Lyra and Cygnus. The stellar flux measurements were sampled at 29.4 min cadence. Observations for each target star were divided into quarters of ∼90 days, after which the spacecraft rotated 90 degrees to maintain its solar panels pointed toward the Sun.

The observation strategy has changed with TESS, with the telescope scanning the entire sky in a 200-light-year range, dividing the sky into 26 sectors each observed for 27 days. TESS is designed to observe ∼200,000 target stars at a 2 min cadence (short cadence data) and to collect Full-Frame Images of each sector at 10 and 30 min cadences (long cadence data).

Both missions employ similar pipelines for data reduction and TCEs detection: the Kepler Science Operations Center (KSOC) [44] and the Science Processing Operations Center (SPOC) [45] for Kepler and TESS, respectively. These pipelines perform bias and flat field calibration, aperture photometry, and systematic corrections before identifying TCEs via the Transiting Planet Search module [46]. While SPOC is responsible for processing TESS short cadence data, long cadence data are processed by the MIT Quick Look Pipeline (QLP) [47].

3.1.2. Catalogs of Threshold Crossing Events Used in This Work

When TPS detects periodic dimming, i.e., TCEs, in the pre-processed stellar light curves, each TCE is examined by astronomers through automated tools [48] whose outputs are visually analyzed so that a label can be assigned to it [6,7]. This process, intent on determining the nature of each TCE, is called vetting.

In this study, we use three catalogs of labeled TCEs from which we produce the input representations we fed to our model. These catalogs provide for each TCE a set of physical (both planetary and stellar) and statistical parameters used for its classification and characterization. The parameters of these catalogs we used are those related to transit properties, i.e., transit period, duration, and the time of the first detected transit (defined as epoch), along with the column defining the label of the TCE. We employ these transit parameters during the pre-processing method, as described in Section 3.1.3.

- Kepler Q1–Q17 Data Release 24 ($\mathcal{D}_{K24}$). This catalog comprises 20,367 TCEs identified by the KSOC pipeline in Kepler light curves. These TCEs were automatically classified by Autovetter [29,49] into planet candidates (PCs), astrophysical false positives (AFPs), non-transiting phenomena (NTP), and unknown (UNK). To minimize the uncertainty of our dataset labels, we adopted the approach employed for the first time by SV18 by discarding all TCEs labeled as UNK. This filtering resulted in a final dataset of 3600 PCs, 9596 AFPs, and 2541 NTPs.
- Kepler Q1–Q17 Data Release 25 ($\mathcal{D}_{K25}$). This set is the final version of TCEs detected by the Kepler mission [50], comprising 34,032 TCEs automatically dispositioned by the Robovetter algorithm [37], that is an ensemble of decision trees trained on a dataset of labeled transits.

  The primary distinction between this catalog and $\mathcal{D}_{K24}$ lies in a higher number of long-period TCEs (approximately 372 days), resulting in $\mathcal{D}_{K25}$ being a TCE dataset characterized by a lower SNR. With longer orbital periods, the number of observed transits decreases, limiting the increase in SNR during our data preparation pipeline. Before generating model inputs from this catalog, we performed the following filtering operation. We removed all TCEs with the rogue flag set to 1, which correspond to cases with fewer than three detected transits, erroneously included in this catalog due to a bug in the Kepler pipeline. For our PC class, we selected the 2726 confirmed and 1382 candidate planets from the Cumulative KOI catalog (The Cumulative KOI catalog

contains the most precise information on all the Kepler TCEs labeled as confirmed and candidate planet, as well as false positive. Further information about Kepler tables of TCEs can be found at the following link: https://exoplanetarchive.ipac.caltech.edu/docs/Kepler_KOI_docs.html, accessed on 9 August 2024). Our AFP class includes the 3946 TCEs labeled as false positive in the Cumulative KOI table, while the NTP class contains the 21,098 TCEs from the Kepler Data Release (DR) 25 catalog that do not appear in the Cumulative KOI table.

- TESS TEY23 ($\mathcal{D}_{TEY23}$). This catalog contains a subset of 24,952 TCEs detected by QLP in TESS long cadence data for which Tey E. et al. [22] (hereafter TEY23) provided dispositions across a three-year vetting process. The authors used five labels to classify these TCEs: "periodic eclipsing signal", "single transit", "contact eclipsing binaries", "junk", and "not-sure" (see Section 2.4 of their paper for further details on the labeling process). To improve the reliability of our dataset, we filtered out (i) 5340 TCEs for which the authors did not provide a consensus label and (ii) all the TCEs labeled as "single transit" and "not-sure", thus obtaining 2613 periodic eclipsing signals (we will identify as E), which include both planet candidates and non-contact eclipsing binaries, 738 contact eclipsing binaries (B) and 15,791 junk (J).

We divided each dataset in 80% training and 20% test splits. In dividing the dataset into the training test, we created splits by preserving the same percentage for each class as in the complete set. By doing so, we avoided getting unbalanced splits toward one of the classes. Table 2 summarizes the composition of these datasets.

Table 2. Composition of the three datasets used in this study. For each class *, we report the total number of TCEs and their distribution in training (80%) and test (20%) sets. * Classes. PC: Planet Candidate, AFP: Astrophysical False Positive, NTP: Non-Transiting Phenomenon, E: Periodic Eclipsing Signal, B: Contact Eclipsing Binary, and J: Junk. [a] The number of samples for the PC classes was doubled as described in Section 3.1.3.

Dataset [Ref.]	Class	Total	Training	Test
Kepler Q1–Q17 DR24 [49]	PC	[a] 7200	5760	1440
	AFP	9596	7676	1920
	NTP	2541	2033	508
	Total	19,337	15,469	3868
Kepler Q1–Q17 DR25 [50]	PC	[a] 8216	6573	1643
	AFP	3946	3162	784
	NTP	21,098	16,950	4148
	Total	33,260	26,685	6575
TESS TEY23 [22]	E	2613	1647	966
	B	738	598	140
	J	15,791	13,327	2464
	Total	19,142	15,572	3570

3.1.3. Data Preparation

The TCEs detected by the KSOC, SPOC, and QLP pipelines still remain signals dominated by the brightness of their host star, in our case, representing noise. To prevent our model from learning the noise, we generate a standard one-dimensional representation for each TCE: a binned and phase-folded light curve devoid of any variability except that of the transit of interest. The methodology we adopted to generate such representations has been widely used in this context of exoplanets detection with Machine Learning (ML) since it was proposed by SV18, and it is described below.

This data preparation pipeline consists of two main blocks: data cleaning—where inconsistent data such as not-a-number and outliers are removed—and data smoothing—where any variability in light curve brightness except that caused by TCE is flattened.

First, we download from the Mikulski Archive for Space Telescopes (MAST) (https://archive.stsci.edu/, accessed on 19 June 2020 for $\mathcal{D}_{K24}$, 20 May 2023 for $\mathcal{D}_{TEY23}$ and 9 August 2024 for $\mathcal{D}_{K25}$) the light curves of the stars around which the TCEs of the catalogs $\mathcal{D}_{K24}$, $\mathcal{D}_{K25}$, and $\mathcal{D}_{TEY23}$ orbit. For each TCE, we apply the following operations:

- Stitching the light curves. A TCE can be associated with multiple segments (Kepler quarters or TESS sectors) of the light curve of its host star. This depends mainly on the observing strategy of the telescope. We generate a single light curve by sequentially appending segments, which we then normalize by the median value calculated over the entire signal;
- Data cleaning. From the resulting light curve, we discard all not-a-numbers and outliers beyond $\pm 3\sigma$ of the stellar flux;
- Detrending. In order to remove any non-TCE-related variability, we divide the cleaned flux data by an interpolating polynomial of degree 3 computed using the Savitzky–Golay method with filter window set to 11. During detrending, we preserve flux measurements related to TCE transit by applying a mask calculated based on TCE transit period and duration;
- Phase-folding and binning. This detrended signal is folded on the relative TCE period and binned with a time bin size of 30 min (When developing this data pre-processing pipeline, we tested time bin sizes of 2, 10, and 30 min, which correspond to the data sampling rates of the Kepler and TESS telescopes. The best results in terms of the shape of the resulting transit were obtained using the 30 min value).
  Following the same methodology used by SV18 and Yu L. et al. [13], we linearly interpolate any empty bin so as to generate an input signal of length 201;
- Normalizing the binned signal. The binned signal is then normalized to 0-median and maximum transit depth to -1. We define the binned and normalized transit as *global view*, consisting of the one-dimensional input we fed to our model;
- Data augmentation on the PC class. Since our main goal is to train a model able to minimize the number of misclassified planets, we double the number of samples belonging to this class in the $\mathcal{D}_{K24}$ and $\mathcal{D}_{K25}$ datasets. More precisely, we apply a horizontal reflection to the global views of the PC TCEs. We decided to not adopt the same procedure to the eclipsing signals (E class) of $\mathcal{D}_{TEY23}$ since Tey E. et al. [22] declared that this set of planets also contains a fraction of non-contact eclipsing binaries, and we want to minimize the risk of increasing the number of eclipsing binaries contaminating the E class because of our purpose of identifying exoplanets.

The schema of this data preparation pipeline is depicted in Figure 1.

This pipeline is highly time-consuming because of (i) the large number of light curves to be processed (∼64,000); (ii) the multiple scans to be performed on each of them; (iii) the high number of data points for each light curves, up to 70,000. To speed up this process, we parallelized this pipeline by distributing the workload over multiple nodes as described in Fiscale S. et al. [51] as the operations on different light curves are independent.

3.2. Components of Our Model

Our model combines neural networks and other methods for feature extraction, dimensionality reduction, and the classification of global views. Below, we provide a theoretical background of each component of our model.

3.2.1. Convolutional Neural Network and VGG19

A Convolutional Neural Network [10,52] is a deep learning model consisting of two main blocks: (i) feature extraction—where the input is subjected to a series of operations such as application of convolutional filters, non-linear activation functions such as Rectified Linear Unit (ReLU), and spatial dimensionality reduction, i.e., pooling. The aim is to extract increasingly complex features from the data and identifying possible patterns, undetectable by manual examination, useful in solving the task at hand; (ii) classification—where the features extracted from the previous block are processed through a classifier, which is typically a MLP.

In this work, we exploited the feature extraction block of VGG19 to extract from our input data, the global views, and their most relevant features. The details of this process are described in Step 1 of Section 4.

3.2.2. Dimensionality Reduction and t-SNE

Dimensionality reduction methods aim to project high-dimensional data in a lower-dimensional space while preserving as much of the significant structure of the data as possible. Among the several dimensionality reduction algorithms (e.g., Isomap [53], Locally Linear Embedding [54], and Laplacian Eigenmaps [55]), we used t-SNE. It is a non-linear method that improves the Stochastic Neighbor Embedding algorithm [56] in terms of cost function optimization and solving the crowding problem (A comprehensive technical description of the t-SNE algorithm, including a quantitative analysis of how it preserves the local and global structure of high-dimensional data, can be found in the original paper by Van der Maaten L. and Hinton G. [33]. In particular, Sections 3.2 and 3.3 and Figures 1 and 2 of the aforementioned paper illustrate how the algorithm addresses the crowding problem and enhances cluster separation in the embedded space). t-SNE generates a two- or three-dimensional representation of input data as follows.

Let $\mathcal{X} = \{\vec{x}_i\}_{i=1}^{m} \in \mathbb{R}^N$ the set of input data belonging to the N-dimensional space and $\mathcal{Y} = \{\vec{y}_i\}_{i=1}^{m} \in \mathbb{R}^d$ the counterpart set of projections into the output lower-dimensional space, where $d \in \{2, 3\} : d \leq N$. In the input space, the pairwise similarities are modeled as conditional probabilities $p_{j|i}$, with

$$p_{j|i} = \frac{\exp(-||\vec{x}_i - \vec{x}_j||^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-||\vec{x}_i - \vec{x}_k||^2/2\sigma_i^2))} \tag{1}$$

where σ_i is the variance of the gaussian distribution centered in the i-th sample. These conditional probabilities are symmetric since $p_{ij} = (p_{j|i} + p_{i|j})/2m$.

In the output space, similarities are modeled by using the t-Student distribution with one degree of freedom:

$$q_{ij} = \frac{(1 + ||\vec{y}_i - \vec{y}_j)||^2)^{-1}}{\sum_{k \neq l}(1 + ||\vec{y}_k - \vec{y}_l)||^2)^{-1}} \tag{2}$$

The aim is to determine the points $\vec{y}_i$ so that the Kullback–Leibler (KL) divergence [57] between the two conditional probabilities distributions is minimized:

$$\mathrm{KL}(P||Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{3}$$

The minimization of the quantity $\mathrm{KL}(P||Q)$ is computed with the gradient descent algorithm with momentum and adaptive learning rate (see Equation (5) in Van der Maaten L. and Hinton G. [33]). The momentum term increases during the gradient descent iterations, while the learning rate is updated according the method provided by Jacobs R.A. [58]. A full derivation of the t-SNE gradient can be found in Appendix A of the original paper. As the authors described in their Section 3.4, the optimization process is improved by the use of the "early exaggeration" trick, which forces the method to model large distances among the low dimensional representation of the samples based on their membership cluster. In other words, the distance between two samples of different clusters is maximized, as we found in our experiments and show in Figure 2.

We relied on t-SNE to reduce the dimensionality of feature vectors obtained from VGG19. Step 2 of Section 4 reports the details of this operation. We demonstrate in Section 5 that this dimensionality reduction places the data in a space where the classifier is able to define better separation surfaces that are highly discriminative than those learned from state-of-the-art CNNs, which work in higher dimensionality embeddings. The limitations of this technique are discussed in Section 6.5.

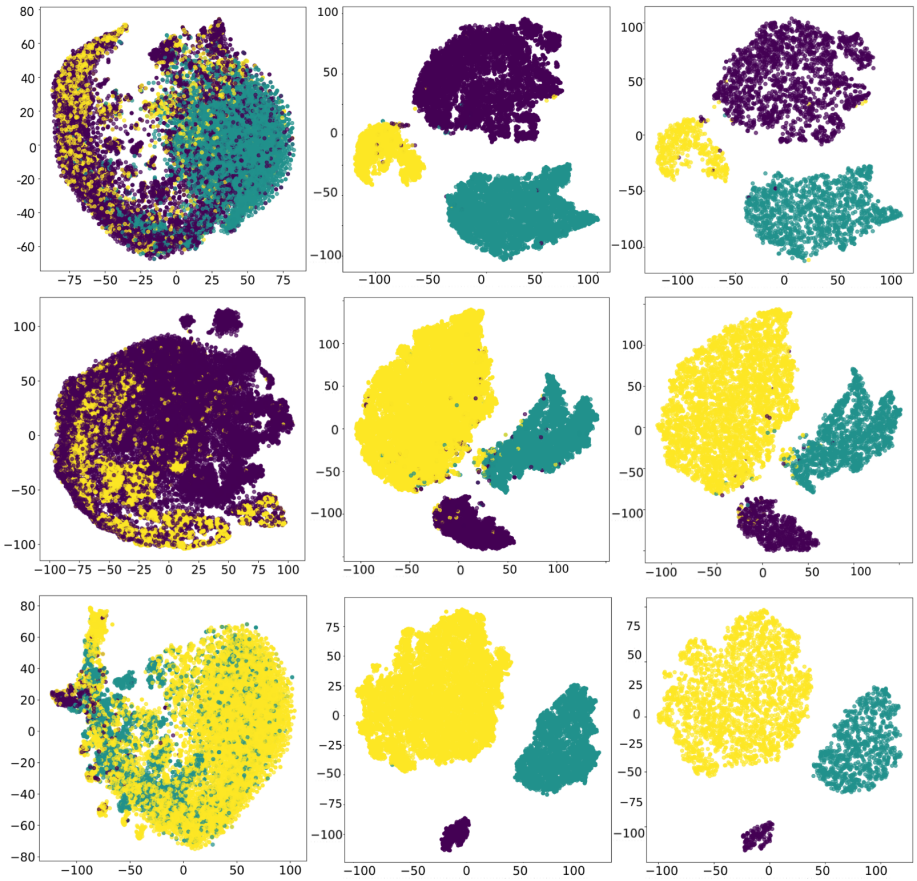

Figure 2. For each dataset, we show the representations of the global views ($\mathcal{D}$, left column), $\mathcal{D}_{2-train}$ (middle column), and $\mathcal{D}_{2-test}$ (right column) in the two-dimensional space defined by t-SNE. (First row) Kepler Q1–Q17 Data Release 24; (second row) Kepler Q1–Q17 Data Release 25; (third row) TESS TEY23. The application of VGG19 for extracting features from the global views ensures a highly effective separation among the three clusters of TCEs on each dataset. Purple points indicate samples belonging to the AFP class (or class B), green points represent samples from the PC class (or class E), and yellow points correspond to the NTP class (or class J).

3.2.3. Gradient-Boosted Trees and XGBoost

Gradient-boosted trees (GBTs) consist of a set of sequentially trained decision trees highly robust in the classification of tabular data [59–62]. Each new tree is constructed with the aim of correcting errors made by previous trees. In a classification context, the final prediction on a given sample is obtained by the majority vote from the predictions of all the trees. In this study, we used eXtreme Gradient Boosting (XGBoost) [63], which differs from GBTs as the trees are constructed in parallel, rather than sequentially. In addition, XGBoost extends traditional gradient boosting by including regularization elements in the objective function, thus improving generalization and preventing overfitting issues, which is very important in exoplanet detection as the majority of datasets are highly imbalanced toward the not-PC classes.

3.2.4. Diffusion Models and Conditional Flow Matching

Diffusion models are the new frontier of generative models, long characterized by the domain of Generative Adversarial Networks [64]. Diffusion models iteratively transform input samples with the injection of noise and then learn to reverse this process, reconstructing the original samples distribution, by solving a Stochastic Differential Equation.

Conditional Flow Matching [36] is a diffusion model used in both generative and classification tasks [65]. CFM transforms an input sample through a learned vector field varying in time $t \in [0, 1]$, mapping its original distribution to a Gaussian distribution over n_t steps. This mapping is achieved by applying a series of invertible transformations (e.g., affine and planar transformations) on the sample, constrained by the labels of input data in supervised learning scenarios. At step n_t, the sample follows the reverse process by solving an Ordinary Differential Equation (ODE) so that the original data distribution can be reconstructed.

XGBoost plays a dual role in this framework. During the forward process, a new XGBoost model is trained at each step to estimate the vector field, providing a more efficient alternative than using a classical neural network [36,66]. Additionally, a final XGBoost is applied on the reconstructed sample at the end of the reverse process to perform classification.

We implemented CFM and XGBoost within our model by exploiting the approach provided by Jolicoeur-Martineau A. et al. [36].

4. Method

This section details the pipeline that we designed for the classification of global views, starting with feature extraction through a CNN, followed by dimensionality reduction via t-SNE, and concluding with classification using Conditional Flow Matching and XGBoost. We leverage the strengths of each component in our pipeline to achieve robust generalization performance in lower-dimensional spaces.

Let

$$\mathcal{D} = \{(\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots, (\vec{x}_n, y_n)\}$$

be the input dataset composed by n samples. Each sample $\{\vec{x}_i\}_{i=1}^{n} \in \mathbb{R}^N$ consists of a global view of size $N = 201$, with

$$\{y_i\}_{i=1}^{n} \in \mathcal{Y} = \{\text{AFP}, \text{PC}, \text{NTP}, \text{E}, \text{B}, \text{J}\}$$

indicating its label. The datasets $\mathcal{D}_{K24}$, $\mathcal{D}_{K25}$, and $\mathcal{D}_{TEY23}$ are separately processed through the following steps.

1. Feature extraction. We extract the features from the global views with the feature extraction block of VGG19. This model is independently trained on each dataset until

overfitting on the global views. Since VGG19 is exclusively used as feature extractor, its training can be extended until overfitting the dataset in order to guarantee the most representative features are extracted. For each of the n global views, the VGG19's feature extraction branch produces a one-dimensional feature map of size 2560, once flattened. We trained VGG19 for 300 epochs on each dataset, with a learning rate of 1×10^{-3}, batch size of 128, and pooling size and stride fixed to 3 and 2, respectively, by using Adam [67] as the optimization algorithm. As highlighted from the number of TCEs for each class in Table 2, all our datasets are imbalanced toward one of the three classes. Typically, such an imbalance is toward the class of non-astrophysical transits (classes NTP and J). To address class imbalance, we used class weighting when training VGG19. The weights for each class were computed using the Inverse Class Frequency technique [68].
The n 2560-length feature vectors, we denote as $\mathcal{D}_1^{n\times 2560}$, are saved at the end of the last training epoch.

2. Dimensionality reduction. The resulting feature vectors are projected into a two-dimensional embedding defined by t-SNE. By processing $\mathcal{D}_1$, t-SNE produces (In our experiments, we also evaluated the classification performance of our model by processing three-dimensional data produced by t-SNE, rather than exclusively two-dimensional data. However, the best performance was obtained by processing data in two dimensions) a representation $\mathcal{D}_2^{n\times 2}$.
As shown in Step 2 of Figure 3, the input $\mathcal{D}_1$ of t-SNE is divided in two subsets (We discuss the application of this strategy in Section 6.5):

   - $\mathcal{D}_{1-train}$ (80% of data), used to generate $\mathcal{D}_{2-train}$, which will be used as training set for the Conditional Flow Matching;
   - The entire dataset $\mathcal{D}_1$, from which $\mathcal{D}_2$ is obtained. We extract from this representation the set $\mathcal{D}_{2-test}$, containing the data that will be used when assessing the Conditional Flow Matching performance.

   Our experiments revealed that running t-SNE for 3000 iterations, with a perplexity of 50, best maximized the separation of TCEs classes in the two-dimensional space.
   The two-dimensional projections obtained by t-SNE for training and test data are shown in middle and right panels of Figure 2, respectively.
   We emphasize that a quantitative assessment of how well t-SNE preserves the clustering structure of the data—particularly in terms of local and global neighborhood relationships—is thoroughly discussed in the original work by Van der Maaten L. and Hinton G. [33]. In our study, we focus on the practical impact this dimensionality reduction has in the context of TCE classification, showing the related evidence in Figures 2 and 4 and Table 3.

3. Classification with CFM and XGBoost. Following the methodology described in Jolicoeur-Martineau A. et al. [36] and Li A. et al. [65], we performed TCE classification by processing $\mathcal{D}_2$ with CFM and XGBoost (Step 3). Each sample of $\mathcal{D}_2$ is mapped into the vector field of the CFM from $t = 0$ to $t = 1$ in $n_t = 50$ steps. At each step, an XGBoost is trained to estimate the vector field. The sample at time $t = 1$ is processed with an ODE, returning the output sample that is fed to an additional XGBoost, responsible for the TCE classification [65].
   Due to the low dimensionality of the input and the good separability between classes of TCEs provided by t-SNE, a very accurate classification performance is obtained as early as n_t = 50 noise levels. Each of the n_t XGBoosts has 100 decision trees of maximum depth of 2 and has been trained for 30 epochs with 63 steps per epoch. An extended discussion on finding the sub-optimal hyperparameters configuration is provided in Section 6.

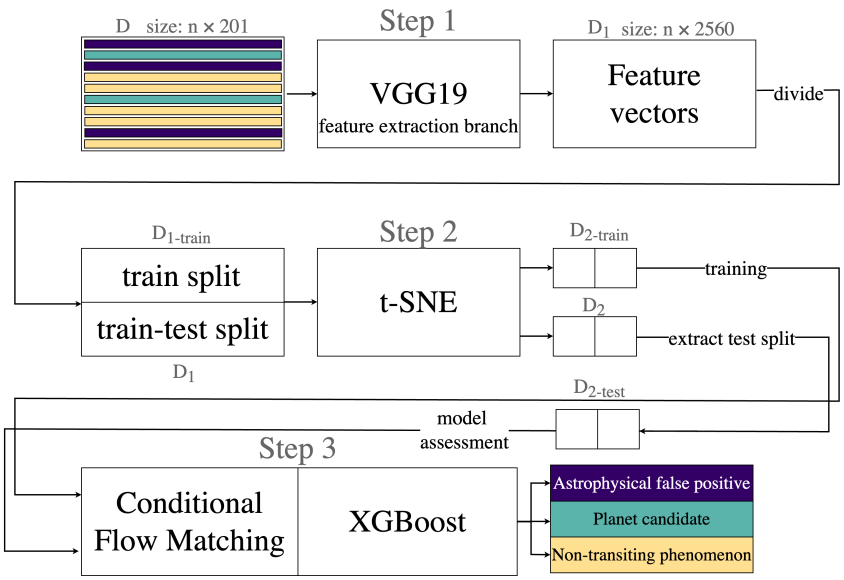

Figure 3. Architecture of our model. *STEP 1*: The input dataset $\mathcal{D}$, containing n global views labeled as AFP (or B, purple rectangles), PC (or E, green rectangles), NTP (or J, yellow rectangles), is processed by VGG19. For each global view, VGG19 produces a feature vector of size 2560. We define the entire set $\mathcal{D}_1$. *STEP 2*: We generate two splits of data from $\mathcal{D}_1$: $\mathcal{D}_{1-train}$, containing the 80% of feature vectors to be used for training in STEP 3, and $\mathcal{D}_1$, which corresponds to the entire dataset obtained in STEP 1. We use t-SNE to project the two splits separately into a two-dimensional space, obtaining $\mathcal{D}_{2-train}$ and $\mathcal{D}_2$. *STEP 3*: We train the CFM with XGBoost on $\mathcal{D}_{2-train}$ and evaluate its performance on $\mathcal{D}_{2-test}$, the subset of $\mathcal{D}_2$ containing only the test data. XGBoost performs multi-class classification of TCEs into AFP (or B), PC (or E), and NTP (or J).

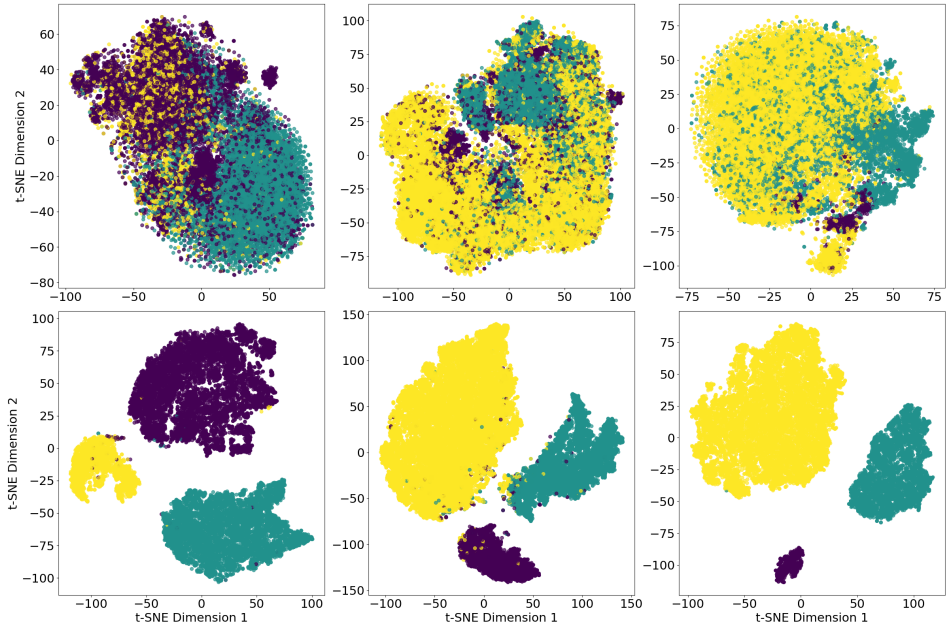

Figure 4. Visual comparison between the features extracted by DART-Vetter (**top row**) and VGG19 (**bottom row**) in the two-dimensional embedding defined by t-SNE. The features extracted from the global views of $\mathcal{D}_{K24}$, $\mathcal{D}_{K25}$, and $\mathcal{D}_{TEY23}$ are depicted in the left, middle, and right panels, respectively. Purple points indicate samples belonging to the AFP class (or class B), green points represent samples from the PC class (or class E), and yellow points correspond to the NTP class (or class J).

Table 3. Performance of different vetting models. Our precision, recall, and F1-scores for Kepler data are computed by averaging the scores of Table 4 obtained on each class. Other model scores are taken from the reference manuscripts. The best results on Kepler and TESS datasets are highlighted in boldface.

Model [Ref.]	Survey	Precision	Recall	F1-Score
SOM [39]	Kepler	0.864	0.865	0.864
SOM [39]	K2	0.945	0.972	0.958
RFC + SOM [30]	NGTS	0.901	0.914	0.907
Exominer [21]	Kepler	0.968	0.974	0.971
Exominer-Basic [21]	TESS	0.88	0.73	0.79
Astronet-Triage-v2 [22]	TESS	0.84	0.99	0.909
Transformer [42]	TESS	0.809	0.8	0.805
This work	Kepler	**0.974**	**0.987**	**0.980**
This work	TESS	**1.0**	**1.0**	**1.0**

Table 4. Classification performance of the model across three datasets: Kepler Q1–Q17 Data Release (DR) 24, Data Release 25, and TESS TEY23. The metrics, computed on test samples, show the ability of our model in distinguishing between TCEs of different natures, including Astrophysical False Positives (AFP), Planet Candidates (PC), and Non-Transiting Phenomena (NTP) in the Kepler datasets. For the TESS dataset, the classification involves TCEs whose nature could be non-contact eclipsing binaries (B), eclipsing signals (E), and Junk (J). For Kepler DR24, individual class misclassification rates are provided, showing particularly strong performance in identifying planet candidates (0.42% misclassification rate). On Kepler DR25, our model exhibits a global misclassification rate of 2.1% across all classes, while on TESS TEY23, it achieves robust predictions performance, with a 0% misclassification rate.

Dataset	Class	Precision	Recall	F1-Score	Misclass. Rate (%)
Kepler Q1–Q17 DR24	AFP	0.9943	0.9932	0.9937	1.25
	PC	0.9972	0.9986	0.9979	0.42
	NTP	0.9803	0.9803	0.9803	3.93
Kepler Q1–Q17 DR25	AFP	0.910	0.985	0.946	2.1
	PC	0.971	0.996	0.983	
	NTP	0.997	0.972	0.984	
TESS TEY23	B	1.000	1.000	1.000	0.0
	E	1.000	1.000	1.000	
	J	1.000	1.000	1.000	

5. Results

The results obtained from the model on the three datasets under this study are presented in this section. On each dataset, we evaluated the classification performance in terms of precision, recall, F1-score, and misclassification rate for each class [69]. Table 4 reports these results. In general, the discriminatory capabilities of the model on each dataset are competitive with those obtained from state-of-the-art models [21,22].

5.1. Application on Kepler Q1–Q17 Data Release 24

The model achieves high predictive accuracy across all three classes of $\mathcal{D}_{K24}$, with noteworthy results in the identification of planets. The results are reported in the first block of Table 4 and discussed below.

On the PC class, we obtain a precision of 0.9972, a recall of 0.9986, and an F1-score of 0.9979, with a misclassification rate of 0.042, indicating a very robust distinction between planetary signals and false positives or non-transiting phenomena. On the AFP class, the precision is 0.9943, recall 0.9932, and F1-score is 0.9937, resulting in a misclassification

rate of 0.0125. The NTP class, while more challenging due to its variability in the nature of transits (including any transit not consistent with astrophysical ones), maintains optimal classification metrics, with a precision, recall, and F1-score of 0.9803 and a misclassification rate of 0.0393.

Analyzing the confusion matrix in left panel of Figure 5, we observe 25 TCE misclassified. The majority of these misclassifications occur between the AFP and NTP classes, counting 19 samples. These samples correspond to the purple and yellow points in the top right panel of Figure 2, which are located in regions of the two-dimensional space close to the center of a cluster to which they do not belong. For the PC class, our model misclassifies only two planets, labeling them as AFP. Meanwhile, there are only three false positives: two AFPs and one NTP, which are the purple and yellow points in the green cluster of the top right panel in Figure 2.

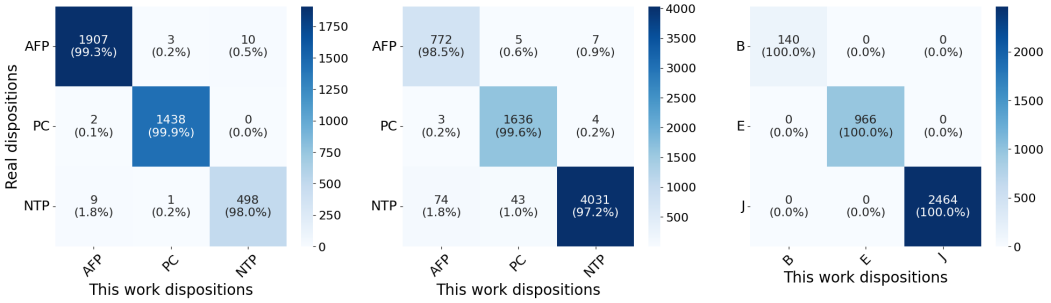

Figure 5. Confusion matrices computed on the test sets of $\mathcal{D}_{K24}$ (**left panel**), $\mathcal{D}_{K25}$ (**middle panel**), and $\mathcal{D}_{TEY23}$ (**right panel**). The high classification performance of Conditional Flow Matching with XGBoost is evident from the diagonal elements of each matrix, with a percentage of correctly classified samples for each class ranging from 97% to 100%. On both Kepler and TESS data, our model retrieves at least 99.6% of planets during classification.

These results demonstrate the effectiveness of our model in distinguishing the TCEs of the three classes, particularly in the identification of planet candidates.

5.2. Application on Kepler Q1–Q17 Data Release 25

Compared to the DR24 dataset, the higher number of long-period TCEs with a lower SNR in $\mathcal{D}_{K25}$ slightly affects the classification accuracy of our model. Nevertheless, discrete classification performance is achieved on this dataset as well. The second block of Table 4 displays the results we present below.

For the AFP class, the model achieves a precision of 0.910, a recall of 0.985, and an F1-score of 0.946. For the PC class, very high scores are obtained: precision is 0.971, recall is 0.996, and F1-score is 0.983. For the NTP class, the model exhibits the highest precision of 0.997, along with a recall of 0.972 and an F1-score of 0.984.

The middle panel confusion matrix in Figure 5 shows that the number of misclassified planets is very low. The model successfully identifies 1636 planets out of 1643 total samples, with only seven misclassifications. Three are associated with the AFP class and four to the NTP class. Regarding these four TCEs, we focus the reader's attention on the middle right panel of Figure 2. Among the five planets (green points) projected by t-SNE near regions dominated by NTPs, only four are classified as NTP. Our model is able to retrieve one of them during classification, despite the fact that its position in the two-dimensional space seemed to compromise its classification. The number of false positives is relatively high (48, including 43 NTPs and 5 AFPs). These samples are visible in middle right panel of Figure 2. The presence of a small cluster of yellow points and five purple points falling into the green cluster of the PCs can be observed. For these points, the model is unable to provide the correct label.

Overall, the misclassification rate of 0.021 further confirms the robust discrimination capabilities of the model on all three classes of TCEs, despite their imperfect separation in the two-dimensional space.

The results obtained on the two Kepler datasets show that the greatest uncertainty in the model lies in the discrimination between AFP and NTP class samples. However, the percentage of misclassifications between these two classes is extremely low (∼2%) and involves samples located at the edges of the clusters (as shown in the rightmost panels of Figure 2), suggesting that further analysis on these cases would not make significant contributions to the overall evaluation of the model. We recall that the main goal is to minimize the fraction of misclassified PCs as they represent the signals of greatest scientific interest. In this regard, our model performs very well: the maximum percentage of misclassified planets is 0.4% in $\mathcal{D}_{K25}$, a value that is very small. As mentioned in Section 3.1.2, $\mathcal{D}_{K25}$ is known to contain a higher fraction of long-period planets than $\mathcal{D}_{K24}$, resulting in fewer available transits and, consequently, a lower SNR of TCEs. This aspect may justify a slight increase in the misclassifications from $\mathcal{D}_{K24}$ (0.1%) to $\mathcal{D}_{K25}$ (0.4%). In Section 6.6, we discuss instead the problem of label noise, whereby a TCE of a given class (e.g., AFP) may change its disposition over time or be labeled differently by different teams of astronomers.

5.3. Application on TESS TEY23

As highlighted in third block of Table 4, our model shows impressive performance in classifying non-contact eclipsing binaries, eclipsing signals, and junk. The model correctly classifies all samples with no misclassified TCEs. As a result, precision, recall, and F1-score all reach their maximum value of 1, with a misclassification rate of 0%. The confusion matrix on this dataset is shown in right panel of Figure 5. The correct classification of all samples is mainly due to their perfect separation in two-dimensional space, as visible in the bottom right panel of Figure 2.

The results on the TESS dataset demonstrate that the prediction of our model is consistent with the labels assigned by experts, which we considered as ground truth. For the two Kepler datasets, our predictions align with the automated labels produced by Autovetter and Robovetter. These findings suggest that the proposed method exhibits strong robustness when applied to real data.

6. Discussion

6.1. The Contribution of VGG19 and t-SNE in TCEs Classification

The use of VGG19 for extracting features from the global views proved to be crucial in ensuring a highly discriminative representation of TCEs.

Initially, we evaluated the use of the CNN provided by Fiscale S. et al. [15] as a feature extractor that we will call DART-Vetter. This model, developed to classify Kepler and TESS TCEs, processes the global view through five convolutional blocks. Each block consists of a one-dimensional convolutional layer followed by ReLU activation, spatial dropout, max pooling, and batch normalization. The number of filters in the convolutional layers increases exponentially from 16 to 256. The extracted features are then flattened and classified through a single fully connected layer. Feature extraction from the global views of the three datasets was performed following the same procedure described in Section 4 (Steps 1 and 2). DART-Vetter was trained until overfitting on each dataset, and the extracted features were stored at the final training epoch.

However, the classification performance on feature vectors produced by DART-Vetter was poor. For this reason, we evaluated the use of VGG19. This network has already been widely used in the context of exoplanet identification, demonstrating its effectiveness since

the introduction of Astronet, whose network architecture is clearly based on that of VGG19, as shown in Figure 7 of their paper [16].

Figure 4 presents a visual comparison of the features obtained from DART-Vetter (top panel) and VGG19 (bottom panel) for the $\mathcal{D}_{K24}$ (left panel), $\mathcal{D}_{K25}$ (middle panel), and $\mathcal{D}_{TEY23}$ (right panel) datasets. The results indicate that VGG19 provides a significantly more effective class separation, with minimal overlap between different classes, unlike the features extracted by DART-Vetter, where class overlap is more noticeable.

The left column of Figure 2 shows, for each dataset, the two-dimensional representations of global views before features are extracted from them using VGG19. Middle and right columns depict the two-dimensional representation—defined by t-SNE—of the features extracted from this network on the training and test data, respectively. The clear class separation achieved after feature extraction and dimensionality reduction confirms that VGG19 captures robust discriminative features, while t-SNE further enhances their separability, as highlighted by the results of Table 3. This near perfect separability facilitated the training of Conditional Flow Matching and XGBoost, enabling them to learn well-defined decision surfaces resulting in a very low misclassification rate.

Thus, the high classification accuracy of our model is mainly due to the key role of combining feature extraction and dimensionality reduction for the identification of most relevant patterns within the global views.

6.2. Reducing the High Computational Complexity and Memory Demand When Training the Conditional Flow Matching with XGBoost

In this section, we discuss a key factor we had to handle during the design of the model and which directed us toward the use of dimensionality reduction methods.

Training our CFM with XGBoost on the feature vectors extracted from VGG19 would have required an extremely high computational cost, making the whole process shown in Figure 3 impractical. We trained CFM and XGBoost with the methodology provided by Jolicoeur-Martineau A. et al. [36], which requires training n_t models, where the levels of noise n_t should take values in [50, 100]. In addition, each model needs to be trained on a duplication of the input dataset. For example, on Kepler Q1–Q17 Data Release 25, it would have been necessary to replicate the dataset at least 50 times, bringing the overall size to more than 1.6 million samples, each with 2560 features. This would have required significantly more memory resources than were available on our machine and training times on the order of several days, also hampering the hyperparameters fine-tuning. An even more critical issue concerned the scalability of the model: even if we had obtained good results, such an onerous process would have made it difficult for other users to exploit the model on their own laptops. The application of t-SNE proved its effectiveness also in addressing this issue by compressing the input vectors from 2560 to only 2 features.

This compression allowed us to preserve the discriminative patterns learned from VGG19 on the global views while ensuring efficient training of CFM with XGBoost and easily reproducible model configuration on common hardware.

To further assess the ability of t-SNE in mapping the feature vectors into a two-dimensional space, we made a comparison with Principal Component Analysis (PCA). As illustrated in Figure 6, t-SNE outperforms PCA in preserving class separability within the lower dimensional embedding. Left and central panels of Figure 6 (top row) demonstrate that when features from the $\mathcal{D}_{K24}$ and $\mathcal{D}_{K25}$ datasets are projected using PCA, they are partially overlapped and spread across significantly larger regions, resulting in poor cluster definition. For each dataset, the standard deviations (σ_1, σ_2) of the first two principal components are $\mathcal{D}_{K24}$: (2903.8, 1551.8); $\mathcal{D}_{K25}$: (407.0, 224.2); $\mathcal{D}_{TEY23}$: (18.0, 5.8).

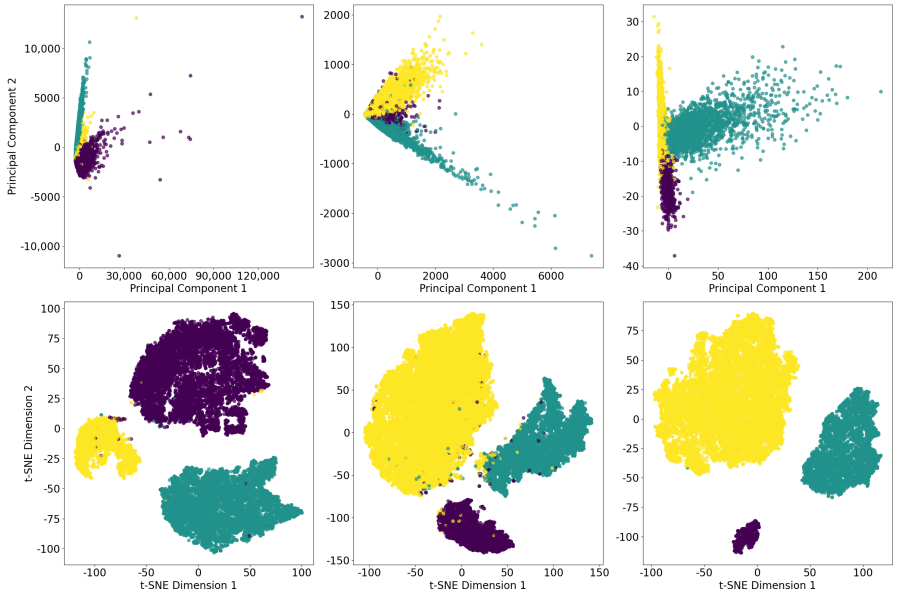

Figure 6. Comparison between t-SNE (**bottom row**) and Principal Component Analysis (PCA (**top row**)) in dimensionality reduction. Two-dimensional features for $\mathcal{D}_{K24}$, $\mathcal{D}_{K25}$, and $\mathcal{D}_{TEY23}$ are shown in the left, middle, and right panels, respectively. Purple points indicate samples belonging to the AFP class (or class B), green points represent samples from the PC class (or class E), and yellow points correspond to the NTP class (or class J).

This wider spreading occurs because data with PCA are linearly mapped to the directions with largest variance, causing high variance classes to distribute across larger regions.

On the other hand, t-SNE maps the same features into more compact embedding while effectively preserving the separation between the three TCE classes. The better ability of t-SNE to project TCEs features into well-defined regions corresponding to their respective classes justifies our choice to adopt this method for dimensionality reduction in our pipeline.

6.3. Finding the Hyperparameter Configuration Optimizing Classification Accuracy

During the experiments, we conducted a systematic analysis of classification performance by varying the Conditional Flow Matching and XGBoost hyperparameters. As suggested in Jolicoeur-Martineau A. et al. [36], we tested noise levels in the range [50, 100], with discrete increments of 10 units. The optimal number of decision trees was evaluated by considering sets in [100, 500], each time increasing by 100 units. Given the two-dimensional inputs of our Step 3, an extensive exploration of the maximum depth of the trees was not necessary, limiting the analysis to architectures with depths of no more than four levels. The number of training epochs of decision trees was estimated in [10, 100], with increments of 10. The batch size for each epoch was chosen by preferring powers of 2 in order to optimize computational efficiency and maximize the use of available hardware resources. The value of 63 steps for each epoch allowed us to process batch sizes of similar size to 256. The experiments conducted revealed that the performance does not improve as model complexity increases. This result finds a natural interpretation in Occam's razor heuristic, according to which, given equal performance, simpler models are preferable to their more complex counterparts.

6.4. Comparison with State-of-the-Art Vetting Models

In this section, we compare the predictive performance of our model with those achieved by state-of-the-art vetting models. Table 5 reports the technical details regarding the models we compare with, while Table 3 summarizes the performance of these compari-

son models. Direct comparisons should be made with caution as these models are trained and tested on datasets from different surveys, varying in training and test set size and pre-processing methods. Additionally, each model applies a specific classification threshold, which is typically optimized on recall to minimize the fraction of misclassified planets.

Table 5. Technical details regarding the deep learning models we compare for classification performance in Table 3. For each model, we provide information about the architecture and training hyperparameters. Input branches indicate the number of input channels through which the model processes data. The column Figure Ref. refers to the figure in the original article where the network architecture is shown. We derived the information displayed in this table from reference articles (and from related source codes when available). We were not able to retrieve the related information for the element denoted with the "-" symbol.

Hyperparameters	**Models**		
	Exominer	Astronet-Triage-v2	Salinas H. et al. [42]
Architecture	CNN	CNN	Transformer
Figure Ref.	Figure 9	Figure 8	Figure 2
Input branches	8	7	3
Activation	ReLU	ReLU	Attention Mechanism *
Regularization	Dropout	Dropout	Dropout
Fully-connected	4×128	4×512	Linear ($X \rightarrow 2$) *
Output	Sigmoid	Sigmoid	Softmax *
Optimizer	Adam	Adam	Adam
Training	-	20,000 steps	60 epochs
Learning rate	6.73×10^{-5}	1×10^{-3}	1×10^{-3}
Batch size	-	64	100
Model selection	10-fold CV	10-fold CV	10-fold CV
	Armstrong D. et al. [39]		Armstrong D. et al. [30]
	Kepler	K2	
Architecture	SOM	SOM	RFC + SOM
Grid dimension	20×20	8×8	20×20
Radius	20	<20	20
Training epochs	500	500	300
Learning rate	0.1	0.1	0.1

* Softmax in the final layer, Attention Mechanism within the Transformer blocks. X represents the concatenated output dimension of the encoders.

Exominer achieves a precision of 0.96 and recall of 0.97, while Astronet-Triage-v2 attains 0.84 and 0.99, respectively. Despite their ability to minimize the fraction of false negatives, these models were designed to process human-relevant input features, some of which are linearly dependent. For example, the local view is essentially a global view with a different bin size. Including such redundant features increases model complexity without significantly improving generalization capabilities [23,43,70]. In addition, high-dimensional inputs increase the risk that the network will need substantial architectural changes to be applied to data from different surveys. This limitation is evident in the degraded performance of Exominer when applied to TESS data (Exominer-Basic), where its precision and recall drop to 0.88 and 0.73, respectively. While reducing input redundancy is crucial for optimizing model effectiveness, astronomers often prioritize interpretability to understand the reasoning behind model predictions. Salinas H. et al. [42] introduced a Transformer-based approach designed to enhance interpretability. However, its performance remains below that of Exominer and Astronet-Triage-v2, achieving a precision of 0.809 and recall of 0.8.

The SOM-based model proposed by Armstrong D. et al. [39] proves its robustness on K2 data, with an F1-score = 0.958, but the performance deterioratea on the Kepler dataset, where the F1-score is 0.864.

Compared to all previous approaches, our model achieves the highest scores on both Kepler and TESS datasets, with an F1-score of 0.980 and 1.0, respectively. Notably, this performance is achieved without increasing input dimensionality, consequently making our model easily transferable across surveys. Our implementation choices supported by promising results suggest that projecting the learned features into lower dimensional spaces and then classifying them by exploiting the capabilities of decision trees as universal approximators may be sufficient to outperform more complex models.

6.5. Current Limitations of Our Model

The current limitation of our approach is the use of t-SNE for dimensionality reduction. Unlike deep learning models, t-SNE lacks optimizable parameters and, therefore, cannot learn dynamic mappings from high- to low-dimensional space through a training phase. Consequently, to guarantee that the two-dimensional projection of training data is not influenced by test data during dimensionality reduction (as described in Step 2 of Section 4), we had to split $\mathcal{D}_1$ into distinct subsets.

Additionally, t-SNE is computationally expensive on large datasets as it requires computing pairwise distances. Although this method presents these drawbacks, it proved to be effective in preserving class separability in the lower-dimensional embeddings of both our TCE datasets and benchmark datasets of images and handwritten digits [71].

To address these limitations, we plan to replace t-SNE with methods that can learn dynamic mapping, such as Variational Autoencoder (VAE).

6.6. The Noise Affecting TCE Labels and Lack of Benchmark Dataset

In this section, we discuss what we consider to be a central challenge in the field of exoplanet detection: the presence of label noise affecting the classification of TCEs. It is well known by the exoplanet community that TCE labels are subject to uncertainty as they may evolve over time with the availability of new observations and through manual vetting by experts. Consequently, a certain degree of ambiguity is to be expected. For example, this issue is mentioned in Exominer and found in Astronet-Triage-v2, where Table A1 in Tey E. et al. [22] highlights certain disagreements among astronomers regarding TCE dispositions, with some cases lacking a Consensus Label. Further discrepancies in the TCE label can be found in Cacciapuoti L. et al. [6] and Magliano C. et al. [7], who independently examined and relabeled subsets of TCEs from the ExoFOP catalog, sometimes diverging from the labels provided by the TESS Follow-up Program Observing Group (TFOPWG; [72]). Labels change over time, observable in the "View all TFOPWG Disposition" field of ExoFOP, further confirming this underlying ambiguity.

Supervised models for exoplanet detection are evaluated on these datasets, labeled by different research teams, and a universally accepted "ground-truth" dataset for model assessment does not yet exist. This constitutes a significant limitation that makes direct comparisons across studies inherently difficult. This remains an open issue in the field and warrants further attention from the exoplanet community.

7. Conclusions

We presented a model to distinguish planetary signals from false positives in Kepler and TESS transit light curves. Our approach combines deep learning, dimensionality reduction, diffusion models, and decision trees. More precisely, we used VGG19 for feature extraction, t-SNE for dimensionality reduction, and Conditional Flow Matching with XGBoost for

classification. The proposed model was evaluated on three datasets achieving F1-scores of 98% on Kepler data and 100% on the TESS dataset TEY23, which represents a performance improvement over the best-performing models on Kepler (1% better than Exominer) and on TESS (10% better than Astronet-Triage-v2). The architecture we designed guarantees low computational complexity in data collection, preparation, and processing. Our Python code (version 3.10.15), implemented using the PyTorch library (version 1.9.5) [73], is freely available at the following link: https://github.com/stefanofisc/dartvetter_cfm.

We relied on the effectiveness of VGG19 as a feature extractor, as illustrated in Figure 4, and the results reported in Table 3 proved that t-SNE significantly enhances class separability. While VGG19 extracts highly discriminative patterns, these features lie in a high-dimensional space, which can affect the ability of the classifier to define optimal decision boundaries. By contrast, projecting these features into a two-dimensional space via t-SNE facilitates the learning of well-defined separation surfaces. This enables CFM and XGBoost to achieve high classification accuracy with substantially lower computational and memory demands. We carefully considered the potential risk of overfitting while designing our pipeline. Consequently, we employed several strategies to reduce this risk, particularly in preventing the model from being biased toward the majority class, as detailed in Section 4. Furthermore, without dimensionality reduction, training CFM on high-dimensional feature vectors, with the method developed by Jolicoeur-Martineau A. et al. [36], would have been infeasible due to the need for dataset duplication for each noise level.

Future work will explore alternative dimensionality reduction techniques, such as Variational Autoencoder, to further optimize the feature representation.

Another important aspect to be addressed in future developments is the exploration of more data augmentation methods, such as statistically based undersampling and oversampling [74]. In the present work, our aim was to extend the PC class in order to construct the largest and most reliable dataset representation for evaluating our model. To this end, we employed a simple yet effective oversampling technique, consisting of horizontally flipping the global views of the PC class. This approach preserves the statistical properties of the original signals as it does not leverage on synthetic signal injections or noise distortions. On the other hand, undersampling could be considered in scenarios where training and evaluation are performed exclusively on real data. In this context, one could randomly select a representative subset from the majority classes (i.e., NTP, J, and AFP). However, while this approach may help balance the class distribution, it carries the risk of discarding informative examples crucial for capturing the diversity of non-planetary signals. This may, in turn, introduce bias and negatively affect the generalization capabilities of our model. Among the various augmentation techniques, we believe statistically based oversampling appears to be the most promising avenue for future exploration. Architectures such as VAEs and diffusion models offer the possibility to sample new data from a learned latent space, enabling the generation of realistic variations of planetary signals. These synthetic samples tend to preserve the underlying statistical distribution of the original data, potentially enhancing the robustness of our model without compromising data representation.

Additionally, we plan to extend the application of our model to upcoming transit surveys, including ESA's PLAnetary Transits and Oscillations of stars (PLATO) [75], to assess its generalization capabilities on new data.

Author Contributions: Conceptualization, S.F. and A.F.; methodology, S.F., A.F. and A.C.; software, S.F.; validation, S.F., A.F. and A.C.; formal analysis, S.F., A.F. and A.C.; investigation, S.F. and A.F.; resources, S.F., A.F. and A.C.; data curation, S.F., A.F. and A.C.; writing—original draft preparation, S.F.; writing—review and editing, S.F., A.F., A.C., L.I., G.C., M.G.O. and A.R.; visualization, S.F., A.F., A.C., L.I., M.G.O., G.C. and A.R.; supervision, A.F., A.C., L.I. and A.R.; project administration, A.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available in Mikulski Archive for Space Telescopes at http://archive.stsci.edu/missions/kepler/lightcurves/tarfiles/DOI_LINKS/Q0-17_LC+SC/ (Kepler data, accessed on 19 June 2020 for Kepler Q1–Q17 Data Release 24 and on 9 August 2024 for Kepler Q1–Q17 Data Release 25) and https://archive.stsci.edu/tess/bulk_downloads/bulk_downloads_ffi-tp-lc-dv.html#lc (TESS data, accessed on 20 May 2023), reference number [T98304] (Kepler) and reference number [t9-nmc8-f686] (TESS).

Acknowledgments: This research has made use of the NASA Exoplanet Archive, which is operated by the California Institute of Technology, under contract with the National Aeronautics and Space Administration under the Exoplanet Exploration Program. This paper includes data collected by the Kepler mission and obtained from the MAST data archive at the Space Telescope Science Institute (STScI). Funding for the Kepler mission is provided by the NASA Science Mission Directorate. STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS 5–26555. This paper includes data collected with the TESS mission, obtained from the MAST data archive at the Space Telescope Science Institute (STScI). Funding for the TESS mission is provided by the NASA Explorer Program. STScI is operated by the Association of Universities for Research in Astronomy, Inc., under NASA contract NAS 5–26555.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mayor, M.; Queloz, D. A Jupiter-mass companion to a solar-type star. *Nature* **1995**, *378*, 355–359. [CrossRef]
2. Giordano Orsini, M.; Ferone, A.; Inno, L.; Giacobbe, P.; Maratea, A.; Ciaramella, A.; Bonomo, A.S.; Rotundi, A. A data-driven approach for extracting exoplanetary atmospheric features. *Astron. Comput.* **2025**, *52*, 100964. [CrossRef]
3. Koch, D.G.; Borucki, W.J.; Basri, G.; Batalha, N.M.; Brown, T.M.; Caldwell, D.; Christensen-Dalsgaard, J.; Cochran, W.D.; DeVore, E.; Dunham, E.W.; et al. Kepler mission design, realized photometric performance, and early science. *Astrophys. J. Lett.* **2010**, *713*, L79. [CrossRef]
4. Ricker, G.R.; Winn, J.N.; Vanderspek, R.; Latham, D.W.; Bakos, G.Á.; Bean, J.L.; Berta-Thompson, Z.K.; Brown, T.M.; Buchhave, L.; Butler, N.R.; et al. Transiting exoplanet survey satellite. *J. Astron. Telesc. Instrum. Syst.* **2015**, *1*, 014003. [CrossRef]
5. Deeg, H.J.; Alonso, R. Transit photometry as an exoplanet discovery method. *arXiv* **2018**, arXiv:1803.07867.
6. Cacciapuoti, L.; Kostov, V.B.; Kuchner, M.; Quintana, E.V.; Colón, K.D.; Brande, J.; Mullally, S.E.; Chance, Q.; Christiansen, J.L.; Ahlers, J.P.; et al. The TESS Triple-9 Catalog: 999 uniformly vetted exoplanet candidates. *Mon. Not. R. Astron. Soc.* **2022**, *513*, 102–116. [CrossRef]
7. Magliano, C.; Kostov, V.; Cacciapuoti, L.; Covone, G.; Inno, L.; Fiscale, S.; Kuchner, M.; Quintana, E.V.; Salik, R.; Saggese, V.; et al. The TESS Triple-9 Catalog II: A new set of 999 uniformly vetted exoplanet candidates. *Mon. Not. R. Astron. Soc.* **2023**, *521*, 3749–3764. [CrossRef]
8. Kostov, V.B.; Kuchner, M.J.; Cacciapuoti, L.; Acharya, S.; Ahlers, J.P.; Andres-Carcasona, M.; Brande, J.; de Lima, L.T.; Di Fraia, M.Z.; Fornear, A.U.; et al. Planet Patrol: Vetting Transiting Exoplanet Candidates with Citizen Science. *Publ. Astron. Soc. Pac.* **2022**, *134*, 044401. [CrossRef]
9. Tenenbaum, P.; Jenkins, J.M. *TESS Science Data Products Description Document: EXP-TESS-ARC-ICD-0014 Rev D*; No. ARC-E-DAA-TN61810; NASA: Washington, DC, USA, 2018.
10. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
11. Dattilo, A.; Vanderburg, A.; Shallue, C.J.; Mayo, A.W.; Berlind, P.; Bieryla, A.; Calkins, M.L.; Esquerdo, G.A.; Everett, M.E.; Howell, S.B.; et al. Identifying exoplanets with deep learning. II. Two new super-Earths uncovered by a neural network in K2 data. *Astron. J.* **2019**, *157*, 169. [CrossRef]
12. Chaushev, A.; Raynard, L.; Goad, M.R.; Eigmüller, P.; Armstrong, D.J.; Briegal, J.T.; Burleigh, M.R.; Casewell, S.L.; Gill, S.; Jenkins, J.S.; et al. Classifying exoplanet candidates with convolutional neural networks: Application to the Next Generation Transit Survey. *Mon. Not. R. Astron. Soc.* **2019**, *488*, 5232–5250. [CrossRef]
13. Yu, L.; Vanderburg, A.; Huang, C.; Shallue, C.J.; Crossfield, I.J.; Gaudi, B.S.; Daylan, T.; Dattilo, A.; Armstrong, D.J.; Ricker, G.R.; et al. Identifying exoplanets with deep learning. III. Automated triage and vetting of TESS candidates. *Astron. J.* **2019**, *158*, 25. [CrossRef]
14. Osborn, H.P.; Ansdell, M.; Ioannou, Y.; Sasdelli, M.; Angerhausen, D.; Caldwell, D.; Jenkins, J.M.; Räissi, C.; Smith, J.C. Rapid classification of TESS planet candidates with convolutional neural networks. *Astron. Astrophys.* **2020**, *633*, A53. [CrossRef]

15. Fiscale, S.; Inno, L.; Ciaramella, A.; Ferone, A.; Rotundi, A.; De Luca, P.; Galletti, A.; Marcellino, L.; Covone, G. Identifying Exoplanets in TESS Data by Deep Learning. In *Applications of Artificial Intelligence and Neural Systems to Data Science*; Springer Nature: Singapore, 2023; pp. 127–135.
16. Shallue, C.J.; Vanderburg, A. Identifying exoplanets with deep learning: A five-planet resonant chain around Kepler-80 and an eighth planet around Kepler-90. *Astron. J.* **2018**, *155*, 94. [CrossRef]
17. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
18. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* **1989**, *2*, 303–314. [CrossRef]
19. Bishop, C.M. *Neural Networks for Pattern Recognition*; Oxford University Press: Oxford, UK, 1995.
20. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [CrossRef]
21. Valizadegan, H.; Martinho, M.J.; Wilkens, L.S.; Jenkins, J.M.; Smith, J.C.; Caldwell, D.A.; Twicken, J.D.; Gerum, P.C.; Walia, N.; Hausknecht, K.; et al. ExoMiner: A highly accurate and explainable deep learning classifier that validates 301 new exoplanets. *Astrophys. J.* **2022**, *926*, 120. [CrossRef]
22. Tey, E.; Moldovan, D.; Kunimoto, M.; Huang, C.X.; Shporer, A.; Daylan, T.; Muthukrishna, D.; Vanderburg, A.; Dattilo, A.; Ricker, G.R.; et al. Identifying exoplanets with deep learning. V. Improved light-curve classification for TESS full-frame image observations. *Astrophys. J.* **2023**, *165*, 95. [CrossRef]
23. Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
24. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
25. Friedman, J.; Hastie, T.; Tibshirani, R. Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors). *Ann. Stat.* **2000**, *28*, 337–407. [CrossRef]
26. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
27. Royden, H.L.; Fitzpatrick, P. *Real Analysis*; Macmillan: New York, NY, USA, 1968; Volume 2.
28. Shwartz-Ziv, R.; Armon, A. Tabular data: Deep learning is not all you need. *Inf. Fusion* **2022**, *81*, 84–90. [CrossRef]
29. McCauliff, S.D.; Jenkins, J.M.; Catanzarite, J.; Burke, C.J.; Coughlin, J.L.; Twicken, J.D.; Tenenbaum, P.; Seader, S.; Li, J.; Cote, M. Automatic classification of Kepler planetary transit candidates. *Astrophys. J.* **2015**, *806*, 6. [CrossRef]
30. Armstrong, D.J.; Günther, M.N.; McCormac, J.; Smith, A.M.; Bayliss, D.; Bouchy, F.; Burleigh, M.R.; Casewell, S.; Eigmüller, P.; Gillen, E.; et al. Automatic vetting of planet candidates from ground-based surveys: Machine learning with NGTS. *Mon. Not. R. Astron. Soc.* **2018**, *478*, 4225–4237. [CrossRef]
31. Caceres, G.A.; Feigelson, E.D.; Babu, G.J.; Bahamonde, N.; Christen, A.; Bertin, K.; Meza, C.; Curé, M. Autoregressive planet search: Application to the Kepler mission. *Astrophys. J.* **2019**, *158*, 58. [CrossRef]
32. Schanche, N.; Cameron, A.C.; Hébrard, G.; Nielsen, L.; Triaud, A.H.; Almenara, J.M.; Alsubai, K.A.; Anderson, D.R.; Armstrong, D.J.; Barros, S.C.; et al. Machine-learning approaches to exoplanet transit detection and candidate validation in wide-field ground-based surveys. *Mon. Not. R. Astron. Soc.* **2019**, *483*, 5534–5547. [CrossRef]
33. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
34. Wheatley, P.J.; West, R.G.; Goad, M.R.; Jenkins, J.S.; Pollacco, D.L.; Queloz, D.; Rauer, H.; Udry, S.; Watson, C.A.; Chazelas, B.; et al. The next generation transit survey (NGTS). *Mon. Not. R. Astron. Soc.* **2018**, *475*, 4476–4493. [CrossRef]
35. Pollacco, D.L.; Skillen, I.; Cameron, A.C.; Christian, D.J.; Hellier, C.; Irwin, J.; Lister, T.A.; Street, R.A.; West, R.G.; Anderson, D.; et al. The WASP project and the SuperWASP cameras. *Publ. Astron. Soc. Pac.* **2006**, *118*, 1407. [CrossRef]
36. Jolicoeur-Martineau, A.; Fatras, K.; Kachman, T. Generating and imputing tabular data via diffusion and flow-based gradient-boosted trees. In *International Conference on Artificial Intelligence and Statistics*; PMLR: Birmingham, UK, 2024; pp. 1288–1296.
37. Coughlin, J.L.; Mullally, F.; Thompson, S.E.; Rowe, J.F.; Burke, C.J.; Latham, D.W.; Batalha, N.M.; Ofir, A.; Quarles, B.L.; Henze, C.E.; et al. Planetary candidates observed by Kepler. VII. The first fully uniform catalog based on the entire 48-month data set (Q1–Q17 DR24). *Astrophys. J. Suppl. Ser.* **2016**, *224*, 12. [CrossRef]
38. Howell, S.B.; Sobeck, C.; Haas, M.; Still, M.; Barclay, T.; Mullally, F.; Troeltzsch, J.; Aigrain, S.; Bryson, S.T.; Caldwell, D.; et al. The K2 mission: Characterization and early results. *Publ. Astron. Soc. Pac.* **2014**, *126*, 398. [CrossRef]
39. Armstrong, D.J.; Pollacco, D.; Santerne, A. Transit shapes and self organising maps as a tool for ranking planetary candidates: Application to kepler and k2. *Mon. Not. R. Astron. Soc.* **2016**, *461*, 2461–2473.
40. Poleo, V.T.; Eisner, N.; Hogg, D.W. NotPlaNET: Removing False Positives from Planet Hunters TESS with Machine Learning. *Astron. J.* **2024**, *168*, 100. [CrossRef]
41. Ansdell, M.; Ioannou, Y.; Osborn, H.P.; Sasdelli, M.; Smith, J.C.; Caldwell, D.; Jenkins, J.M.; Räissi, C.; Angerhausen, D. Scientific domain knowledge improves exoplanet transit classification with deep learning. *Astrophys. J. Lett.* **2018**, *869*, L7. [CrossRef]
42. Salinas, H.; Pichara, K.; Brahm, R.; Pérez-Galarce, F.; Mery, D. Distinguishing a planetary transit from false positives: A Transformer-based classification for planetary transit signals. *Mon. Not. R. Astron. Soc.* **2023**, *522*, 3201–3216. [CrossRef]
43. Visser, K.; Bosma, B.; Postma, E. Exoplanet detection with Genesis. *J. Astron. Instrum.* **2022**, *11*, 2250011. [CrossRef]

44. Jenkins, J.M.; Caldwell, D.A.; Chandrasekaran, H.; Twicken, J.D.; Bryson, S.T.; Quintana, E.V.; Clarke, B.D.; Li, J.; Allen, C.; Tenenbaum, P.; et al. Overview of the Kepler science processing pipeline. *Astrophys. J. Lett.* **2010**, *713*, L87. [CrossRef]
45. Jenkins, J.M.; Twicken, J.D.; McCauliff, S.; Campbell, J.; Sanderfer, D.; Lung, D.; Mansouri-Samani, M.; Girouard, F.; Tenenbaum, P.; Klaus, T.; et al. The TESS science processing operations center. In *Software and Cyberinfrastructure for Astronomy IV*; SPIE: Bellingham, WA, USA, 2016; Volume 9913, pp. 1232–1251.
46. Jenkins, J.M.; Tenenbaum, P.; Seader, S.; Burke, C.J.; McCauliff, S.D.; Smith, J.C.; Twicken, J.D.; Chandrasekaran, H. *Kepler Data Processing Handbook: Transiting Planet Search*; Kepler Science Document KSCI-19081-002; NASA Ames Research Center: Mountain View, CA, USA, 2017; p. 9.
47. Kunimoto, M.; Huang, C.; Tey, E.; Fong, W.; Hesse, K.; Shporer, A.; Guerrero, N.; Fausnaugh, M.; Vanderspek, R.; Ricker, G. Quick-look pipeline lightcurves for 9.1 million stars observed over the first year of the TESS Extended Mission. *RNAAS* **2021**, *5*, 234. [CrossRef]
48. Kostov, V.B.; Mullally, S.E.; Quintana, E.V.; Coughlin, J.L.; Mullally, F.; Barclay, T.; Colón, K.D.; Schlieder, J.E.; Barentsen, G.; Burke, C.J. Discovery and Vetting of Exoplanets. I. Benchmarking K2 Vetting Tools. *Astron. J.* **2019**, *157*, 124. [CrossRef]
49. Catanzarite, J.H. *Autovetter Planet Candidate Catalog for Q1–Q17 Data Release 24*; NASA Ames Research Center: Mountain View, CA, USA, 2015.
50. Thompson, S.E.; Coughlin, J.L.; Hoffman, K.; Mullally, F.; Christiansen, J.L.; Burke, C.J.; Bryson, S.; Batalha, N.; Haas, M.R.; Catanzarite, J.; et al. Planetary candidates observed by Kepler. VIII. A fully automated catalog with measured completeness and reliability based on data release 25. *Astrophys. J. Suppl. Ser.* **2018**, *235*, 38. [CrossRef] [PubMed]
51. Fiscale, S.; De Luca, P.; Inno, L.; Marcellino, L.; Galletti, A.; Rotundi, A.; Ciaramella, A.; Covone, G.; Quintana, E. A GPU algorithm for outliers detection in TESS light curves. In *International Conference on Computational Science*; Springer: Cham, Switzerland, 2021; pp. 420–432.
52. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
53. Tenenbaum, J.B.; Silva, V.D.; Langford, J.C. A global geometric framework for nonlinear dimensionality reduction. *Science* **2000**, *290*, 2319–2323. [CrossRef] [PubMed]
54. Roweis, S.T.; Saul, L.K. Nonlinear dimensionality reduction by locally linear embedding. *Science* **2000**, *290*, 2323–2326. [CrossRef]
55. Belkin, M.; Niyogi, P. Using manifold stucture for partially labeled classification. *Adv. Neural Inf. Process. Syst.* **2002**, *15*, 1505–1512.
56. Hinton, G. Stochastic neighbor embedding. *Adv. Neural Inf. Process. Syst.* **2003**, *15*, 857–864.
57. Csiszár, I. I-divergence geometry of probability distributions and minimization problems. *Ann. Probab.* **1975**, *3*, 146–158. [CrossRef]
58. Jacobs, R.A. Increased rates of convergence through learning rate adaptation. *Neural Netw.* **1988**, *1*, 295–307. [CrossRef]
59. Zhang, C.; Liu, C.; Zhang, X.; Almpanidis, G. An up-to-date comparison of state-of-the-art classification algorithms. *Expert Syst. Appl.* **2017**, *82*, 128–150. [CrossRef]
60. Touzani, S.; Granderson, J.; Fernandes, S. Gradient boosting machine for modeling the energy consumption of commercial buildings. *Energy Build.* **2018**, *158*, 1533–1543. [CrossRef]
61. Machado, M.R.; Karray, S.; De Sousa, I.T. LightGBM: An effective decision tree gradient boosting method to predict customer loyalty in the finance industry. In Proceedings of the 2019 14th International Conference on Computer Science & Education (ICCSE), Toronto, ON, Canada, 19–21 August 2019; pp. 1111–1116.
62. Ma, B.; Meng, F.; Yan, G.; Yan, H.; Chai, B.; Song, F. Diagnostic classification of cancers using extreme gradient boosting algorithm and multi-omics data. *Comput. Biol. Med.* **2020**, *121*, 103761. [CrossRef] [PubMed]
63. Chen, T.; He, T.; Benesty, M.; Khotilovich, V.; Tang, Y.; Cho, H.; Chen, K.; Mitchell, R.; Cano, I.; Zhou, T. Xgboost: Extreme gradient boosting. In *R Package Version 0.4-2*; The R Project for Statistical Computing: Vienna, Austria, 2015; Volume 1, pp. 1–4.
64. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 2672–2680.
65. Li, A.C.; Prabhudesai, M.; Duggal, S.; Brown, E.; Pathak, D. Your diffusion model is secretly a zero-shot classifier. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 2206–2217.
66. Chen, R.T.; Rubanova, Y.; Bettencourt, J.; Duvenaud, D.K. Neural ordinary differential equations. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 6572–6583.
67. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
68. Lertnattee, V.; Theeramunkong, T. Analysis of inverse class frequency in centroid-based text classification. In Proceedings of the IEEE International Symposium on Communications and Information Technology, 2004. ISCIT 2004, Sapporo, Japan, 26–29 October 2004; Volume 2, pp. 1171–1176.
69. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, UK, 2016; Volume 1, No. 2.
70. Visser, K.; Bosma, B.; Postma, E. Size does matter: Exoplanet detection with a sparse convolutional neural network. *Astron. Comput.* **2022**, *41*, 100654. [CrossRef]

71. Gisbrecht, A.; Schulz, A.; Hammer, B. Parametric nonlinear dimensionality reduction using kernel t-SNE. *Neurocomputing* **2015**, *147*, 71–82. [CrossRef]
72. Guerrero, N.M.; Seager, S.; Huang, C.X.; Vanderburg, A.; Soto, A.G.; Mireles, I.; Hesse, K.; Fong, W.; Glidden, A.; Shporer, A.; et al. The TESS objects of interest catalog from the TESS prime mission. *Astrophys. J. Suppl. Ser.* **2021**, *254*, 39. [CrossRef]
73. Imambi, S.; Prakash, K.B.; Kanagachidambaresan, G.R. PyTorch. In *Programming with TensorFlow: Solution for Edge Computing Applications*; Springer Nature: Cham, Switzerland, 2021; pp. 87–104.
74. Braga, F.C.; Roman, N.T.; Falceta-Gonçalves, D. The Effects of Under and Over Sampling in Exoplanet Transit Identification with Low Signal-to-Noise Ratio Data. In *Brazilian Conference on Intelligent Systems*; Springer International Publishing: Cham, Switzerland, 2022; pp. 107–121.
75. Rauer, H.; Catala, C.; Aerts, C.; Appourchaux, T.; Benz, W.; Brandeker, A.; Christensen-Dalsgaard, J.; Deleuil, M.; Gizon, L.; Goupil, M.J.; et al. The PLATO 2.0 mission. *Exp. Astron.* **2014**, *38*, 249–330. [CrossRef]

Chapter 3

Cometary observations in support of the ESA/Comet Interceptor space mission

3.1 Introduction

A critical aspect of the ESA Comet Interceptor space mission is the target comet selection. To contribute to this mission phase, aiming at characterising the activity behaviour of long period comets, I conducted photometric observations at the Copernico Telescope in Asiago (Italy) and the Telescopio Nazionale Galileo in La Palma (Canary Islands, Spain). I also contributed to the corresponding data reduction process using a pipeline developed in the Interactive Data Language (IDL).

The rest of this chapter is organized as follows. Section 3.1 provides the scientific context of the Comet Interceptor mission. The aims of the work carried out in support of the mission are described in Section 3.2, and results are reported in Section 3.3. As this work culminated in the submission of a scientific paper (Bertini, Fulle, Mazzotta Epifani, Fiscale et al. 2025, Astronomy & Astrophysics, under review) [6], named Paper 5, reported in Section 3.3.1.

Comet Interceptor mission

Comets – cosmic snowballs of frozen gases, rock, and dust – are the most pristine planetesimals left from the time of planets formation [8, 22]. Passing most of their life in the coldest regions of planetary systems, cometary nuclei are likely composed of relatively unprocessed material, and could preserve pre-solar grains. For this reason,

their study through photometric observations and characterization represents the most valuable insights for improving our understanding of planet formation mechanisms.

Mission objective. ESA's Comet Interceptor mission builds upon the work carried out by previous missions such as Giotto [70] and Rosetta [31], with the aim of answering new questions about the primordial nature of comets.

Unlike previously studied comets, which have been processed due to multiple Perihelion passages, Comet Interceptor's primary target is a "dynamically new comet", i.e., a comet that leaving the Oort Cloud approaches the inner Solar System for the first time, or even an interstellar object, i.e. entering into the Solar System coming from the interstellar space.

The latter scenario would represents an unprecedented opportunity to study a poorly explored class of bodies, to date composed of only three members: 1I/'Oumuamua [61], 2I/Borisov [65], and 3I/ATLAS [74].

Such objects are the most pristine planetesimals that we can observe nowadays, providing the joining link between fully formed planetary systems, such as our, and protoplanetary disks.

The spacecraft. Comet Interceptor will fly-by its target employing a mother spacecraft (S/C A - developed by ESA) and two probes (S/C B1 - developed by JAXA and S/C B2 - developed by ESA) [44], as depicted in Figure 3.1.

Observation strategy. The observation method is innovative, as Comet Interceptor will be launched without a predefined target and will wait in a parking orbit at the Earth-Sun Lagrangian point L2 for up to three years. Once ground-based telescopes have identified a target in line with the primary objective, the probe will set off to intercept it.

Detection of the target. The ground-based telescope LSST at the Vera Rubin Observatory is one of those capable of identifying the target for Comet Interceptor first, thanks to its wide field of view, sensitivity, and speed of scanning the sky. LSST has a wide field of view of 9.6 square degrees achieved through a unique 3-mirror design, allowing it to scan the entire night sky every three days. This rapid and repeated scanning of the sky makes it much easier to identify new comets. The telescope is equipped with an 8.4-meter primary mirror and a 3.2-gigapixel camera, with which it will observe the sky in six different frequency bands (i.e. ugrizy) up to an average magnitude per image of 23.8. If no target suitable for the requirements is identified, Comet Interceptor will focus on a backup target taken from a list of known comets.

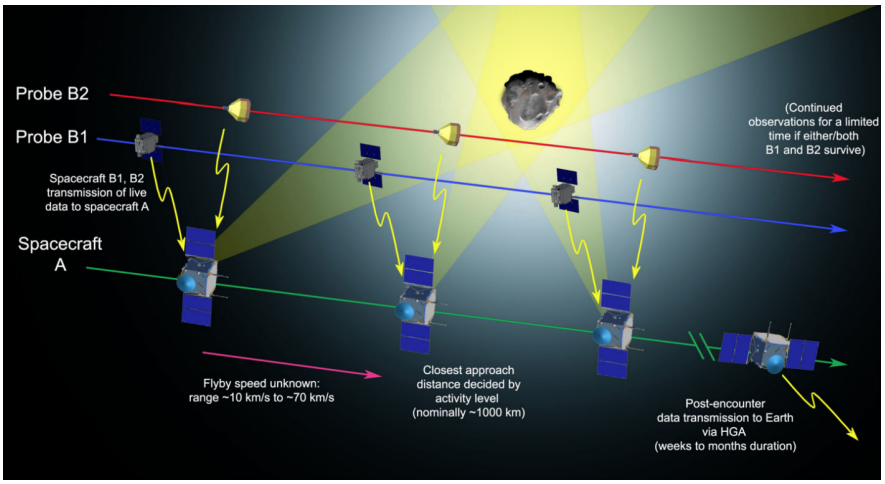

Figure 3.1: Schematic representation of the Comet Interceptor flyby trajectory (not to scale). Spacecraft A performs the most distant passage from the comet's nucleus, while Probes B1 and B2 follow closer paths. Both probes will transmit real-time data to Spacecraft A, where it will be collected and stored. Figure credit: Jones G.H. et al. (2024) [44].

My contribution to the mission. The projects supporting the mission I am part of required observing time at the Copernico and TNG telescopes, with the aim of investigating the processes driving the distant activity of dynamically new comets and modeling the pristine formation of comets in the protoplanetary disk. My role within these projects was to periodically visit the aforementioned telescopes to carry out photometric observations and, once sufficient data had been collected, I contributed with some members of these projects to the data reduction process.

These activities were conducted within the Working Package number 5200, "DISC Operation Environment" (Scientific responsible: Prof. Ivano Bertini), under the Agreement between Agenzia Spaziale Italiana (ASI) and Istituto Nazionale di Astrofisica (INAF) for supporting the Italian scientific team of the Comet Interceptor mission.

3.2 Aims

This section outlines the scientific objectives of the work carried out by our research group in support of the Comet Interceptor mission.

Currently, our understanding of dynamically new comets remains limited. Further studies are required to gain deeper insight into the evolution of these long-period comets (with orbital periods greater than 200 years [62]) coming from the Oort Cloud, often observed during their first passage in the Solar System. The main goal is to investigate how these comets evolve during their inbound journey from the outer regions of the

Solar System – where activity is dominated by sublimation of supervolatile ices – to the inner regions, where water-ice becomes the dominant driver of activity [30].

Our work focused on characterizing the dust activity of a sample of long-period comets active beyond the water-driven activity onset, i.e., at heliocentric distances greater than 4 au. We collected photometric observations of these comets over three years (2022-2024) as part of a long-term, multi-telescope observing program. The observations I conducted at the Copernico telescopes in Asiago and TNG in La Palma are described in detail in Section 3.3. The first two tables in Paper 5 list the comets observed and their respective orbital parameters.

Dust tails have been analyzed by a probabilistic cometary tail model based on a Monte Carlo integration of dust spherical particles along three-dimensional heliocentric orbits [30]. These particles exhibit a broad diversity in composition and structure, ranging from dense rocky grains formed in the inner Solar System [13] to extremely porous aggregates produced within the protosolar nebula [29].

By fitting the observed dust tails with the probabilistic model, we derived the main physical parameters governing cometary activity beyond Jupiter's orbit, including dust ejection velocity and particle size distribution.

3.3 Results for Comets

The results obtained for the work I performed dealing with comets are presented in the peer reviewed paper reported in Section 3.3.1 to which I contributed performing astronomical observations, data analyses and reduction (Section 3.3.2).

3.3.1 "Inbound evolution of twenty long period comets [6]"

The analysis of the collected and reduced data yielded the following results. The outcomes are fully consistent with the Water Enriched Blocks (WEB) model [27], which describes cometary nuclei as aggregates of water-poor pebbles containing localized inclusions of water ice. According to this model, the weak dust activity and low ejection velocities observed in our sample suggest that these comets are composed primarily of highly porous, water-depleted material. Such a composition implies that they would pose no dust-related hazard to the Comet Interceptor spacecraft during the flyby.

A detailed discussion of these results is provided in Sections 6 and 7 of Paper 5.

Astronomy & Astrophysics manuscript no. output
July 1, 2025

Inbound evolution of twenty long period comets⋆

I. Bertini[1,2], M. Fulle[3], E. Mazzotta Epifani[4], S. Fiscale[1,2], L. Inno[1,2], A. Rotundi[1,2], C. Tubiana[5], V. Della Corte[6], F. Ferrigno[1], C. Magliano[7], G. Portelli[1], M. Scuderi[1], L. Tonietti[1,2], A. Ferone[1], P. Ochner[8,9], E. Ammannito[10], C. Grappasonni[10], G. Sindoni[10], A. Aletti[11,12], P. Bacci[11,13], G. Baj[11,14], F. Bellini[11,12], E. Bryssinck[11,15], D. Carosati[11,16], M. di Grazia[11,13], M. Facchini[11,17], M. Feraco[11,18], E. Guido[11,19], F. Kugel[11,20], R. Ligustri[11,21], M. Maestripieri[11,13], G. Milani[11], J. Nicolas[11,20], L. Tinelli[11,22], D. Tirelli[11], A. Valvasori[11,19]

[1] Department of Science and Technology, Parthenope University of Naples, CDN-IC4 Napoli, Italy
e-mail: ivano.bertini@uniparthenope.it
[2] UNESCO Chair "Environment, Resources and Sustainable Development", Parthenope University of Naples, Italy
[3] INAF - Osservatorio Astronomico di Trieste, Via Tiepolo 11, 34143 Trieste, Italy
[4] INAF - Osservatorio Astronomico di Roma, Via Frascati 33, 00040 Monte Porzio Catone (RM), Italy
[5] INAF - Istituto di Astrofisica e Planetologia Spaziale, Via Fosso del Cavaliere 100, 00133 Roma, Italy
[6] INAF - Osservatorio Astronomico di Capodimonte, Salita Moiariello 16, 80131 Napoli, Italy
[7] Department of Physics "Ettore Pancini", University of Naples Federico II, Naples, Italy
[8] INAF - Osservatorio Astronomico di Padova, Vicolo dell'Osservatorio 5, I-35122 Padova, Italy
[9] Department of Physics and Astronomy, University of Padova, Via F. Marzolo 8, I-35131 Padova, Italy
[10] ASI - Agenzia Spaziale Italiana, Roma, Italy
[11] CARA Project, Osservatorio Astronomico di Trieste, Via Tiepolo 11, 34143 Trieste, Italy
[12] Società Astronomica Schiaparelli, Via Giovanni Borghi 7, I-21100 Varese, Italy
[13] GAMP, Osservatorio Astronomico Montagna Pistoiese, Pian dei Termini 1, 50128 San Marcello Pistoiese (PT), Italy
[14] Osservatorio M57, 21050 Saltrio, Varese, Italy (passed away on 16 May 2025)
[15] Brixiis Observatory, Kruibeke, Belgium
[16] EPT Observatories, 38780 Tijarafe, La Palma, Spain
[17] Osservatorio di Cavezzo, via Concordia 200, 40132 Cavezzo (MO), Italy
[18] Osservatorio BiAnto, San Crispino, 85044 Lauria, Italy
[19] ALMO Observatory (MPC G18), via Forlai 14/C, 40010 Sala Bolognese (BO), Italy
[20] Observatoire Chante-Perdrix, 04150 Dauban, France
[21] ITelescope.net and Skygems Observatories
[22] Osservatorio E. Castiglioni, via Garibaldi 12, 20865 Usmate-Velate (MB), Italy

Received Month Day, YYYY; accepted Month Day, YYYY

ABSTRACT

Aims. We aim to characterize the dust activity and nucleus properties of long-period comets during their inbound journey to the inner Solar System. This analysis supports the target selection and risk assessment for the ESA Comet Interceptor mission.
Methods. We observed and modeled the dust tails of twenty Oort cloud comets by means of the probabilistic tail model.
Results. About 60% of the dust tails are best fit by anisotropic dust ejection, due to nucleus inhomogeneity built-up by activity occurred before the comet ejection into the Oort cloud. The three free parameters of the probabilistic tail model are all correlated with the nucleus radius R_n, as predicted by the model. For about half of the sample, the trend of the coma photometry with respect to the heliocentric distance is available and always consistent with the predictions by the tail model. The comets drifting from supervolatile-driven to water-driven activity show a drop of coma photometry, suggesting a nucleus mainly composed of water-poor pebbles. The measured dust ejection velocity fits the R_n dependence predicted by dust coma thermophysical models. About 85% of the nuclei have $R_n < 1$ km, and about 25% of these are probably very elongated or contact-binary nuclei.
Conclusions. All the observed comets are consistent with the Water Enriched Blocks (WEB) model and are mainly composed of water-poor pebbles with $R_n < 4$ km, i.e., they would be safe for the ESA Comet Interceptor mission.

Key words. comets: general – comets: individual: C/2017 K2 PanSTARRS – comets: individual: C/2022 E3 ZTF – Oort cloud

1. Introduction

Comet Interceptor (CI) (Jones et al. 2024) will be the next effort in the field of cometary space science, designed and built by the European Space Agency (ESA) in cooperation with the Japan Aerospace Exploration Agency (JAXA).

The mission aims at investigating for the first time the environment of a possibly dynamically new comet coming from the Oort Cloud, or even a cometary object coming from

⋆ Based on observations made with the Italian Telescopio Nazionale Galileo (TNG) operated on the island of La Palma by the Fundacion Galileo Galilei of the INAF (Istituto Nazionale di Astrofisica) at the Spanish Observatorio del Roque de los Muchachos of the Instituto de Astrofisica de Canarias, and at Copernico 1.82-m telescope (Asiago Mount Ekar, Italy) INAF - Osservatorio Astronomico di Padova.

Table 1. Log of observations.

Comet	Observation date	r_h	Δ	I/O	$r_h - q$	α	Telescope	$Af\rho_1$	$Af\rho_2$	$Af\rho_3$
C/2016 Q2 (PanSTARRS)	2022 Jun 23	7.55	7.63	O	0.47	7.66	TNG	3.75	4.47	4.35
C/2017 K2 (PanSTARRS)	2022 Apr 30	3.32	2.84	I		17.0	Asiago	76.2	85.5	90.4
	2022 Jul 01	2.77	1.84	I		10.79	TNG	75.0	76.8	77.7
C/2017 U7 (PanSTARRS)	2022 May 29	9.12	8.17	O	2.70	2.32	TNG	3.15	5.21	7.28
	2022 Jul 01	9.26	8.46	O	2.84	4.02	TNG	2.07	3.66	
C/2019 O3 (Palomar)	2022 Apr 29	9.14	9.18	O	0.32	6.30	Asiago	21.9	36.2	45.5
	2022 Jun 23	9.23	9.20	O	0.41	6.32	TNG	37.8	48.1	48.4
C/2020 F2 (ATLAS)	2022 Apr 29	8.83	7.86	I		2.00	Asiago	17.5	27.8	33.0
	2022 May 24	8.82	8.05	O	0.005	4.46	TNG	16.2	21.4	20.8
	2023 Feb 13	8.90	8.30	O	0.08	5.20	Asiago	20.8	25.7	27.9
	2024 Apr 11	9.55	8.74	O	0.73	3.74	Asiago	5.7	9.3	12.1
C/2020 H6 (ATLAS)	2022 Mar 10	4.87	4.51	O	0.17	11.34	Asiago	7.9	10.7	
	2022 Jul 01	5.17	4.89	O	0.47	11.15	TNG	13.2	13.7	12.4
	2022 Aug 24	5.36	5.53	O	0.66	10.50	TNG	11.0	13.1	12.6
C/2020 K1 (PanSTARRS)	2022 May 24	4.51	3.87	I		10.77	TNG	32.7	34.7	
	2022 Jul 01	4.27	3.43	I		8.59	TNG	33.5	35.1	34.2
C/2020 K6 (Rankin)	2022 May 24	6.13	5.79	O	0.26	9.18	TNG	1.77	1.83	
C/2020 O2 (Amaral)	2022 May 24	5.29	4.69	O	0.43	9.42	TNG	7.8	7.8	
	2022 Jul 23	5.48	4.90	O	0.62	9.21	TNG	9.0	9.9	
	2022 Aug 24	5.60	5.27	O	0.74	10.09	TNG	8.1	10.2	
C/2020 V2 (ZTF)	2022 Feb 23	5.07	4.40	I		8.82	Asiago	50.0	56.1	55.0
	2022 Mar 10	4.95	4.30	I		9.41	Asiago	36.9	53.8	57.3
	2022 May 24	4.34	4.53	I		12.87	TNG	43.6	58.4	62.6
C/2021 K2 (MASTER)	2022 Jul 01	5.88	4.89	O	0.41	2.82	TNG	0.70	0.78	
C/2021 Q4 (Fuls)	2022 Jul 01	7.86	7.60	I		7.29	TNG	5.5	5.7	
	2022 Aug 24	7.78	7.16	I		6.14	TNG	6.9		
	2023 Oct 10	7.60	6.77	O	0.04	4.38	Asiago	4.7	6.8	7.3
C/2021 S1 (ATLAS)	2023 Dec 20	7.59	6.96	O	1.47	5.98	TNG	2.7	2.2	
C/2022 E2 (ATLAS)	2023 Dec 20	4.37	3.70	I		10.28	TNG	38.2	42.4	42.1
	2024 Jan 04	4.30	3.44	I		7.01	Asiago	38.1	41.9	41.1
	2024 Apr 11	3.92	3.96	I		14.60	Asiago	26.6	36.8	38.7
C/2022 E3 (ZTF)	2022 Jul 01	2.99	2.29	I		16.20	TNG	22.7		
	2022 Jul 23	2.74	2.13	I		19.35	TNG	23.5		
C/2022 L2 (ATLAS)	2022 Jul 01	6.39	5.80	I		7.80	TNG	7.0	7.2	
	2022 Oct 01	5.71	5.73	I		10.04	Asiago	3.6	5.4	5.7
C/2022 QE78 (ATLAS)	2023 Dec 19	7.11	6.32	I		5.01	TNG	11.1	19.5	26.5
	2024 Nov 07	5.92	5.36	I		8.32	Asiago	27.7	34.7	35.6
	2024 Dec 01	5.86	5.04	I		5.86	Asiago	31.6	36.9	37.5
C/2022 U1 (Leonard)	2023 Dec 20	4.28	4.09	I		13.26	TNG	0.7	1.1	
	2024 Jan 04	4.26	4.30	I		13.20	Asiago	2.4	2.5	
	2024 Sep 10	4.43	3.94	O	0.23	12.09	Asiago	4.7		
C/2022 U3 (Bok)	2023 Oct 10	5.33	4.83	I		9.77	Asiago	6.2		
	2023 Dec 19	5.13	4.18	I			TNG	3.6	4.5	3.7
	2024 Jan 03	5.09	4.22	I		5.68	Asiago	4.1	4.6	
	2024 Jan 04	5.09	4.22	I		5.85	Asiago	4.6	4.6	
	2024 Nov 07	4.89	4.60	O	0.06	11.28	Asiago	5.0	5.3	
C/2023 X2 (Lemmon)	2024 Dec 01	5.88	4.98	I		4.11	TNG	4.7	4.3	
C/2023 X7 (PanSTARRS)	2024 Dec 01	4.99	4.36	I		9.36	TNG	0.2	0.3	
C/2024 A1 (ATLAS)	2024 Dec 01	4.23	3.81	I		12.76	TNG	2.2	2.5	

Notes. Comets observed at TNG during AOT45 (April-September 2022), AOT48 (semester October 2023-March 2024), and AOT 50 (October 2024-March 2025), and at Asiago during LTP-01 (February 2022-January 2024) and LTP-02 (February 2024-January 2026). r_h and Δ, heliocentric and geocentric distances [au]. I/O, inbound/outbound orbital branch. α, phase angle [degrees]. $Af\rho$ [m] at the reference aperture radii $\rho = 5\ 10^3$ km ($Af\rho_1$), $\rho = 10^4$ km ($Af\rho_2$), and $\rho = 5\ 10^4$ km ($Af\rho_3$), normalised at $\alpha = 0$.

the outer interstellar space. The latter would represent an unprecedented opportunity to investigate an object of a still poorly explored class, up to now composed of only two members: 1I/Oumuamua (Meech et al. 2017; Micheli et al. 2018) and 2I/Borisov (Opitom et al. 2019; Cremonese et al. 2020; Mazzotta Epifani et al. 2021; Busarev et al. 2021; Prodan et al. 2024).

A novelty of the CI mission is that it is being designed and probably launched before its actual target will be identified. It will be parked around the Sun-Earth L2 point, where it will wait until a reachable target is found and its fly-by at heliocentric dis-

Table 2. Orbital parameters.

Comet	e	a_0	q [au]	t_q	i [°]
C/2016 Q2	0.99984	+0.83	7.08	2021 May 11.6	109.4
C/2017 K2	1.00053	+2.08	1.80	2022 Dec 19.8	87.6
C/2017 U7	1.00161	+1.00	6.42	2019 Sep 11.7	142.6
C/2019 O3	1.00356	+3.57	8.82	2021 Mar 08.9	89.8
C/2020 F2	1.00475	+1.92	8.82	2022 Jul 15.9	163.6
C/2020 H6	1.00057	+9.09	4.70	2021 Oct 01.1	80.0
C/2020 K1	0.99987	+0.29	3.07	2023 May 09.1	89.7
C/2020 K6	1.00320	+1.59	5.87	2021 Sep 15.6	103.6
C/2020 O2	1.00233	+3.23	4.86	2021 Aug 28.6	71.8
C/2020 V2	1.00103	+7.69	2.23	2023 May 08.5	131.6
C/2021 K2	1.00120	+0.46	5.47	2021 Sep 09.2	100.8
C/2021 Q4	1.00322	+0.83	7.56	2023 Jun 10.0	71.5
C/2021 S1	1.00096	+0.35	6.12	2022 Feb 28.9	52.1
C/2022 E2	1.00092	+1.41	3.67	2024 Sep 14.0	137.1
C/2022 E3	1.00030	+0.14	1.11	2023 Jan 12.8	109.2
C/2022 L2	1.00131	+10.0	2.69	2024 Mar 12.3	129.3
C/2022 QE78	1.00317	+2.70	5.48	2025 Sep 10.2	36.5
C/2022 U1	0.99983	+0.57	4.20	2024 Mar 25.6	128.1
C/2022 U3	1.00335	+4.55	4.83	2024 Jul 27.8	33.6
C/2023 X2	1.00182	+0.35	5.10	2025 Dec 26.1	77.0
C/2023 X7	1.00205	+0.36	4.82	2025 May 15.0	69.1
C/2024 A1	1.00142	+0.30	3.88	2025 Jun 13.0	94.5

Notes. a_0 [10^4 au] is the original semimajor axis in Nakano's Notes (http://www.oaa.gr.jp/~oaacs/nk.htm), determined after the integration backward one orbit, under the influence of the Sun and planets, Galactic tides and nongravitational forces. All aphelia lie in the Oort cloud.

tance $r_h \approx 1$ au will be programmed. The CI target will probably be identified at $r_h > 10$ au in the inbound segment of its orbit. This will give enough time to fix the target orbit and for the spacecraft to reach the flyby position, but it is not clear if it will also give enough time to predict the target activity levels at the flyby. Therefore, it is necessary to investigate a crucial and still poorly explored issue: how do long-period comets, coming from the Oort cloud and often on their first passage inside the Solar System, evolve while moving from the outer region, where activity is driven by the sublimation of supervolatile ices, to the inner region, where also the water ice comes in action, crossing the water-driven activity onset, probably occurring at $r_h \approx 3.8$ au (Fulle et al. 2020; Ciarniello et al. 2023) ?

This issue will be explored in the framework of the impact that the Legacy Survey of Space and Time (LSST) at the Vera Rubin Observatory will have on Solar System science (Vera C. Rubin Observatory LSST Solar System Science Collaboration et al. 2021). In the specific case, the discovery rate of distant Oort cloud comets will be increased and this will enable the study of key aspects of their evolution on a statistical basis. To this end, it becomes crucial to test the robustness and automation of the probabilistic tail model used to investigate the dust tail of the interstellar comet 2I/Borisov (Cremonese et al. 2020). We already applied this model to a first small data set of 5 distant Oort cloud comets observed in the range $4.8 < r_h < 9.2$ au, obtaining the important result that the three free parameters of the model are all correlated with the nucleus size, and suggesting that automation of the probabilistic tail model is feasible (Fulle et al. 2022) (hereafter Paper I).

In this paper, we apply the probabilistic tail model to an increased database of Oort cloud comets observed during 3 years (2022-2024) of our long-term, multi-telescope observing program of dust tails of distant Oort cloud comets active beyond the water-driven activity onset (Table 1). Orbital parameters of the observed comets are available in Table 2. Some comets were observed in the outbound orbital branch, however all these have the perihelion $q > 4$ au and were observed shortly after their perihelion, so that their activity cannot be distinguished from the inbound one. Observations performed in the heliocentric range $2.7 < r_h < 9.3$ au are described in Section 2. The coma photometry of the target comets is described in Section 3. Section 4 summarises the cometary activity model at $r_h > 3.8$ au and the probabilistic tail model (for further details of the models, we refer the reader to Paper I), while the tail fits and the overall results on the whole sample are described in Section 5 and compared to other observations in Section 6.

2. Observations

The images of the target comets were obtained at Telescopio Nazionale Galileo (TNG) and at the Copernico telescope operated by INAF Astronomical Observatory of Padova (Table 1).

TNG observations were performed in several nights along the semester AOT45 (April – September 2022), during the nights of 19 and 20 December 2023 (within the semester AOT48, October 2023 – March 2024), and during the night of 2 December 2024 (within the semester AOT50, October 2024 – March 2025). The Device Optimised for the LOw RESolution (DOLORES) instrument was used during all the observing slots, with a 2048 × 2048 px E2V 4240 thinned back-illuminated, deep-depleted, Astro-BB coated CCD detector with a pixel size of 13.5 μm, which yields a field of view of ~8.6 × 8.6 arcmin. The corresponding scale is 0.252 arcsec px^{-1}. The instrument was used with the broadband R filter of the Johnson-Cousins system.

Telescope Copernico observations were performed during two long-term dedicated programs approved for a total of four years, starting in February 2022. The Copernico images were acquired in the Sloan r band with the Asiago Faint Object Spectrograph and Camera (AFOSC) instrument. AFOSC is equipped with a 2048 × 2048 px CCD with a field of view of 8.85 × 8.85 arcmin, giving a scale of 0.26 arcsec px^{-1}.

All images were obtained with the telescopes tracking at the nonsidereal motion proper of each target, and were reduced using standard procedures (subtraction of masterbias, flat-field correction and proper background sky removal) by means of the use of IDL ASTROLIB routines (`https://asd.gsfc.nasa.gov/archive/idlastro/`).

In TNG images the absolute photometric calibration was performed by computing the photometric zero-points and atmospheric extinction coefficients for the R filter using the Bouguer method methodology with standard stars from Landolt (2013).

Copernico's images were instead calibrated performing differential photometry with catalogue stars identified in the comets frames, since atmospheric extinction was difficult to correct with classical methodologies. In order to do so, we identified stars from the Fourth U.S. Naval Observatory CCD Astrograph UCAC4 Catalogue (Zacharias et al. 2013) in the comets frames, using the University of Strasburg Vizier online service (https://vizier.cds.unistra.fr/viz-bin/VizieR). Since the UCAC4 catalogue contains stars magnitudes in Johnson B, Johnson V, and Sloan r magnitudes (Henden et al. 2016), we converted stars r magnitudes into the corresponding R magnitudes used by Landolt using the formula R = r – 0.108 (B–V) – 0.132 (Dymock & Miles 2018). We then derived appropriate correction factors between instrumental and catalogue R magnitudes which allowed the absolute flux calibration of cometary images.

For the purpose of this work, two kinds of targets have been identified and selected: a) Long Period Comets (LPCs) on their inbound orbital branch; b) LPCs on their outbound orbital branch, but having passed a perihelion beyond 4 au. The dust

Table 3. Dust and nucleus properties of the comets observed at TNG and Asiago observatories.

Comet	k	v_d (r_h = 10 au) m s^{-1}	σ	d_1	d_2	lat. range degrees	N 10^8	B_s	Figure	Q_d (r_h = 10 au) kg s^{-1}	R_n km
C/2016 Q2 (PanSTARRS)	1	1.0	0.7	0.058	0.053	−90 – +60	8	3	A.1	90	0.35
C/2017 K2 (PanSTARRS)	1,2	4.0	1.6	0.067	0.058	−60 – +90	5	3	2,A.2	4240	3.41
C/2017 U7 (PanSTARRS)	1	0.8	0.5	0.224	0.220	−90 – +60	10	2	A.3	70	0.31
C/2019 O3 (Palomar)	1	0.9	0.5	0.068	0.049	−60 – +90	8	3	3,A.4	840	1.07
C/2020 F2 (ATLAS)	2	0.9	0.5	0.129	0.097	−60 – +60	7	3	4,A.5	400	1.05
C/2020 H6 (ATLAS)	1	0.9	0.6	0.075	0.060	−60 – +90	10	3	5,A.6	215	0.54
C/2020 K1 (PanSTARRS)	1	3.0	1.8	0.200	0.165	−30 – +90	5	3	6,A.7	2070	1.69
C/2020 K6 (Rankin)	1	0.5	1.0	0.024	0.021	−90 – +60	8	3	A.8	20	0.17
C/2020 O2 (Amaral)	1	0.6	0.5	0.108	0.099	−60 – +90	10	3	A.9	110	0.39
C/2020 V2 (ZTF)	2	2.0	1.4	0.093	0.076	−60 – +90	7	3	7,A.10	1120	1.75
C/2021 K2 (MASTERS)	1	0.5	0.7	0.273	0.319	−60 – +60	9	2	A.11	10	0.12
C/2021 Q4 (Fuls)	1	0.6	1.0	0.118	0.117	−60 – +90	6	3	A.12	70	0.31
C/2022 E2 (ATLAS)	2	1.0	0.5	0.085	0.059	−90 – +60	10	3	8,A.13	340	0.97
C/2022 E3 (ZTF)	2	1.0	1.6	0.057	0.052	−90 – +30	7	3	1,9	110	0.55
C/2022 L2 (ATLAS)	2	1.0	1.0	0.520	0.570	−50 – +50	3	3	10,A.14	70	0.44
C/2022 QE78 (ATLAS)	3	1.0	1.6	0.066	0.073	−90 – +60	10	2	A.15	200	1.05
C/2022 U1 (Leonard)	1	1.0	1.6	0.244	0.240	−90 – +60	7	2	A.16	35	0.22
C/2023 X2 (Lemmon)	2	0.5	0.5	0.090	0.096	−60 – +90	7	3	A.17	25	0.26
C/2023 X7 (PanSTARRS)	1	0.8	1.0	0.073	0.082	−60 – +60	7	2	A.18	5	0.08
C/2024 A1 (ATLAS)	1	2.0	1.75	0.151	0.148	−60 – +90	6	2	A.19	100	0.37

Notes. For comet C/2017 K2, $k = 1$ for $r_h \leq 3.8$ au, $k = 2$ for $r_h \geq 3.8$ au. For comets C/2017 K2, C/2017 U7, C/2019 O3, C/2020 F2, C/2020 H6, C/2020 K1, C/2020 O2, C/2020 V2, C/2021 Q4, C/2022 E2, C/2022 E3 and C/2022 L2, the d_1, d_2, Q_d and R_n values are the averages from multiple observing spots (see Table 1). Q_d and R_n are computed using the $Af\rho$ value at $\rho = 10^4$ km, corrected at phase $\alpha = 0$ (Table 1). N is the Monte-Carlo number of particles. B_s is the brightness step factor between isophotes in Figure A.n.

environment of such targets is expected to mimic quite well the environment of the putative CI target when it will be discovered, e.g. when it will still not have experienced enough heating to trigger the more canonical water-driven activity.

3. Coma photometry

The dust environment of each comet was constrained by means of the $Af\rho$ value (A'Hearn et al. 1984), with the classical formulation in which A is the average grain albedo, f is the filling factor in the aperture field of view and ρ is the linear radius of the aperture at the comet. $Af\rho$ measures the dust coma brightness and is linked to the cometary dust production. $Af\rho$ values were derived measuring the calibrated flux in the correspondent photometric aperture and considering the solar flux in the R filter.

Since the data analyzed in this paper are referred to different observational geometries (i.e. different phase angles α), it is important to remove possible differential scattering effects when intensity-related parameters (i.e. $Af\rho$) data are considered together to derive temporal trends. In order to remove the intrinsic scattering properties of cometary dust from our $Af\rho$ measurement a phase function model for the dust particles in comae was taken into account (Bertini et al. 2025). This allowed translating initial $Af\rho(\alpha)$ measurements, coming from the images analysis, into $Af\rho(0)$ values, as all comets were observed at opposition in every considered date. Table 1 summarizes $Af\rho$ values obtained at TNG and, when applicable, at Asiago.

4. Activity and probabilistic tail models

The activity model of cometary nuclei, defining the gas coma parameters due to the sublimation of supervolatile ices (more volatile than water-ice), with its implications for the dust activity at large heliocentric distance, and the probabilistic dust tail model, applied to photometric images of distant Oort cloud comets, are described and discussed in complete details in Paper I. Here we schematically recall the main arguments of the issue.

4.1. Nucleus thermophysical model

The water-driven Water Enriched Blocks (WEB) model (Fulle et al. 2020; Ciarniello et al. 2023; Fornasier et al. 2023) has been extended to ices more volatile than water-ice. The WEBs are composed of water-rich pebbles and are embedded in a matrix of water-poor pebbles. At $3.8 \leq r_h \leq 85$ au, many supervolatile ices (CO, CO_2, O_2, CH_4, C_2H_6) present in the cometary nucleus dominate the gas drag that drives the dust ejection. The model assumes (i) that their fraction in other ices and clathrates is negligible; (ii) that they are present as pure ices in water-poor pebbles only; (iii) that the total volume fraction of all ices is lower than that of the refractories in the nucleus; and (iv) that the volume fraction of water ice in the water-poor pebbles is lower than that of each supervolatile ice. Then, the activity model computes different onset temperatures for the comet activity driven by each ice, at which the gas pressure overcomes the tensile strength bonding the dust to the nucleus. While the comet approaches inbound its perihelion, the dust ejection is sustained by the last activated ice and contributes to the dust loss rate Q_{dust} measured in the tail. The WEB model provides the nucleus erosion $E = E_{10}[r_{10}/r_h]$, where $r_{10} = 10$ au and $E_{10} = 25$ mm day^{-1}. The comparison between the free parameters of the probabilistic tail model and E allows us to constrain the lower limit of the nucleus radius R_n, because $Q_{dust} = A_d \rho_d E$, where A_d is the nucleus surface ejecting dust and $\rho_d = 800$ kg m^{-3} is the average dust density, so that $R_n = \sqrt{A_d/\pi}$. The higher the nucleus fraction of water-poor pebbles, actually containing all supervolatiles, the closer R_n to the actual value of the nucleus radius.

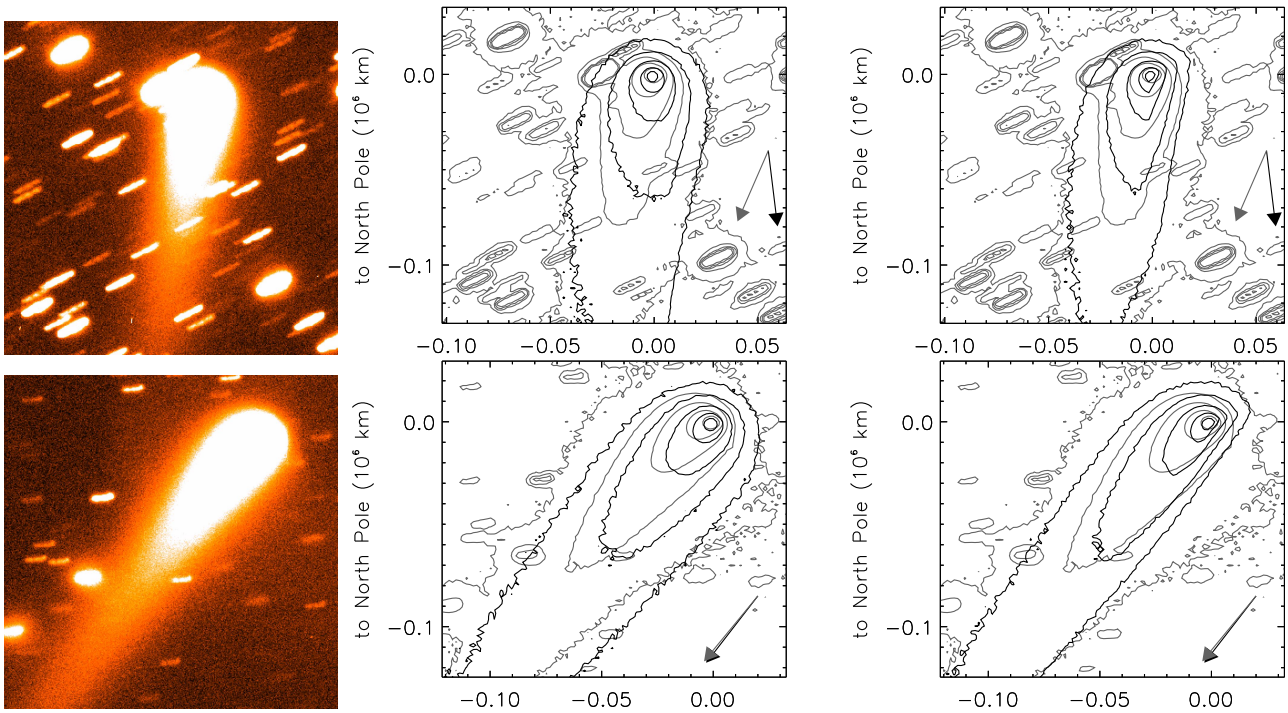

Fig. 1. Comet C/2022 E3 (ZTF) on 2022 July 1, at r_h = 2.99 au (upper left panel), and on 2022 July 23, at r_h = 2.74 au (lower left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panels) and anisotropic dust ejection (no dust ejection from nucleus latitude > +30°, right panels). Model with 7 10^8 particles, step between isophotes of a factor 3. Black arrow: antisolar direction; grey arrow: trailing orbit direction.

4.2. Free parameters of the probabilistic tail model

The motion of dust in the coma (under the gas drag) and in the tail (under the effect of the solar radiation pressure) depend on the dust bulk density, ρ_d, and size, s, both of which showed to cover a wide range of possible values in the many cometary dust environments investigated and modeled so far. Dust tails are best parametrised by the ratio of the solar radiation pressure to gravity forces, $\beta = C_{pr} Q_{pr} (\rho_d s)^{-1}$, where C_{pr} = 1.19 10^{-3} kg m^{-2} and $Q_{pr} \approx 1$ for the large absorbing particles usually observed in dust tails (Fulle et al. 2010). The values of β distribute on a log-normal probability function centred at β_0 = 5 10^{-4} with dispersion σ, which is the first free parameter of the probabilistic tail model. The higher σ, the closer the tail axis to the antisolar direction. The other two free model parameters are: the ejection velocity v_d of the dust particles at r_h = 10 au, $v_{dust} = v_d\ [r_{10}/r_h]$, and the k parameter describing the dependence of the dust loss rate Q_{dust} on the heliocentric distance, $Q_{dust} = Q_d\ [r_{10}/r_h]^k$, where $Q_d = C_q\ v_d\ A f\rho\ [r_{10}/r_h]^{1-k}$ and C_q = 20 kg m^{-2}. The higher v_d, the thicker the tail, and the higher k, the shorter the tail. The tail length depends slowly on k, so that it constrains integer values of k only. If k = 1, then $R_n = \sqrt{Q_d/(\pi\rho_d E_{10})}$. If $k > 1$, then a higher lower limit of the nucleus radius is computed at r_h = 5 au, so that $R_n = \sqrt{2^{k-1} Q_d/(\pi\rho_d E_{10})}$. According to the WEB model, the parameter k of a long period comet can change only crossing inbound $r_h \approx 3.8$ au, because Q_{dust} is proportional to A_d at $r_h > 3.8$ au, and to the nucleus surface area composed of water-rich pebbles A_r at $r_h < 3.8$ au. Crossing inbound $r_h \approx 3.8$ au, k increases if $A_d < A_r$, whereas k decreases if $A_d > A_r$.

4.3. Correlation with the nucleus radius

For each supervolatile considered in the activity model, and for each heliocentric distance, the extended WEB model computed the gas flux from the nucleus surface and the nucleus surface temperature, which were used as input in gas and dust coma models (Zakharov et al. 2018, 2021) to compute the gas and dust terminal velocities. The latter is best approximated by $v_{dust} = v_{10}\ [r_{10}/r_h]\ \sqrt{R_n/R_1}$, where v_{10} = 2 m s^{-1} and R_1 = 1 km. It follows that the statistics of many values of v_d provided by the tail probabilistic model must fit $v_{10}\sqrt{R_n/R_1}$. Also k and σ correlate to R_n. The higher $k > 1$, the steeper the inbound increase of Q_{dust} with respect to the nucleus erosion E, i.e., the larger the dust fallout, i.e., the higher the nucleus mass. The higher k, the larger the contribution of CO_2 to the gas drag of dust of sizes covering wider ranges than other supervolatiles, i.e., the higher σ. High σ values at small R_n imply a high ρ_d dispersion.

5. Tail fits and results

Model tails of comets observed at TNG were obtained by means of Monte Carlo integration of a large number of particles (Table 3), moving on 3D heliocentric orbits determined by the β parameter and by the dust ejection velocity vector. Figures 1 and A.1-A.19 show TNG R-images of the dust tails observed during AOT 45 (semester April-September 2022), AOT48 (semester October 2023 – March 2024), and AOT50 (semester October 2024 – March 2025), together with the observed and modeled isophotes, both in the case of isotropic and anisotropic dust emission from the nucleus. A dust tail must have its axis in the sky sector defined by the antisolar and the trailing orbit directions (black and gray arrows in Figures 1 and A.1-A.19). Comets C/2021 S1 and C/2022 U3 do not satisfy this constraint, preventing the application of any dust tail model.

Figures 2-10 allow us to compare the computed parameter k (dashed lines fitting the TNG-Loiano-Asiago data in the figures, values in Table 3) with the observed trend of $A f\rho$, when it covers a wide enough r_h range. To extend as much as possible such

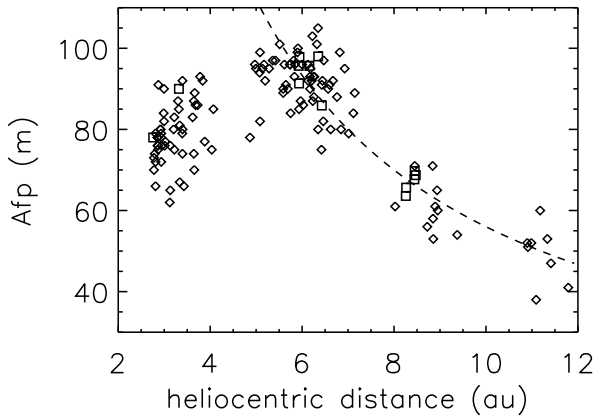

Fig. 2. $Af\rho$ data of comet C/2017 K2. Squares: TNG and Asiago data (Table 1) at $r_h < 5$ au, Loiano Obs. data at $r_h > 5$ au (Fulle et al. 2022). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the model parameter k (Table 3).

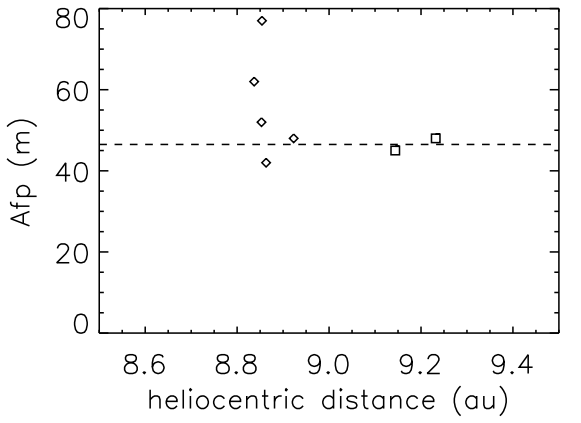

Fig. 3. $Af\rho$ data of comet C/2019 O3. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

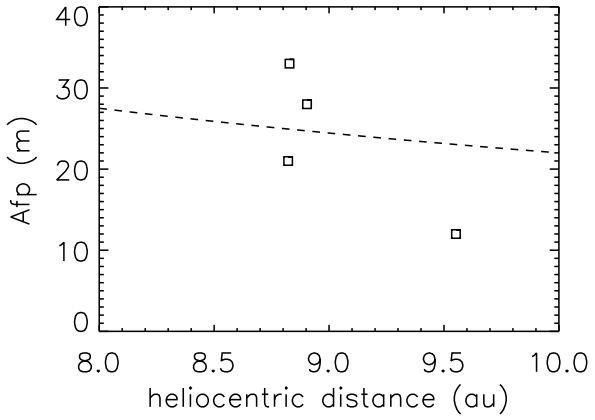

Fig. 4. $Af\rho$ data of comet C/2020 F2. Squares: TNG and Asiago data (Table 1). Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

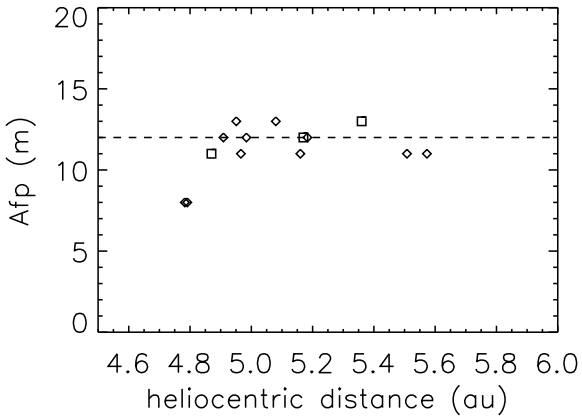

Fig. 5. $Af\rho$ data of comet C/2020 H6. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

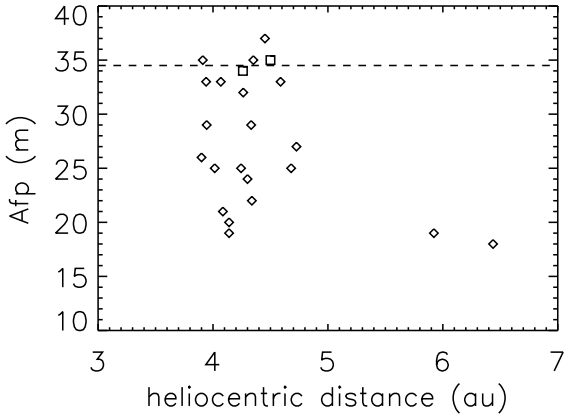

Fig. 6. $Af\rho$ data of comet C/2020 K1. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

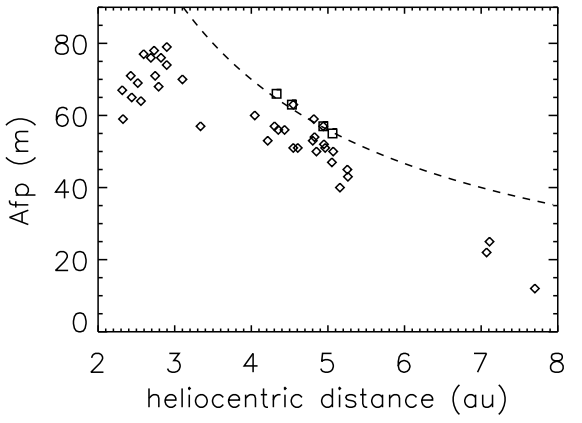

Fig. 7. $Af\rho$ data of comet C/2020 V2. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

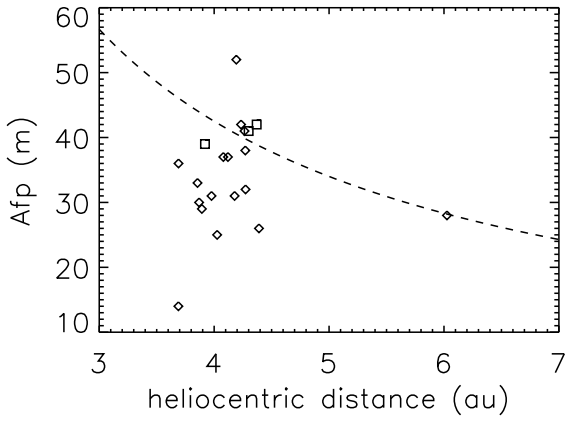

Fig. 8. $Af\rho$ data of comet C/2022 E2. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

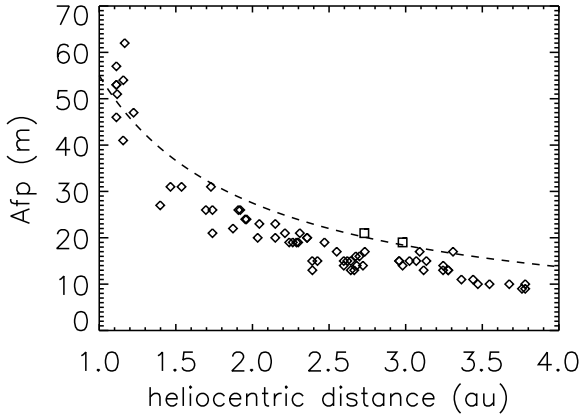

Fig. 9. $Af\rho$ data of comet C/2022 E3. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

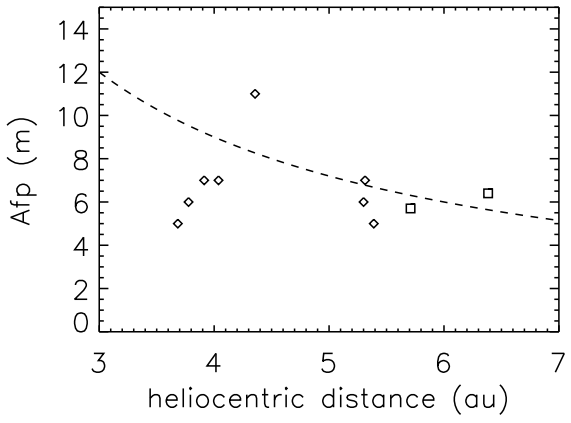

Fig. 10. $Af\rho$ data of comet C/2022 L2. Squares: TNG and Asiago data (Table 1). Diamonds: CARA database at $\rho = 2\ 10^4$ km. Dashed line: predicted heliocentric trend according to the parameter k (Table 3).

a range, our data have been integrated by the amateur CARA database (paper I). The agreement is good in all cases, apart C/2020 K1, where the significance of two CARA data at $r_h > 5$ au is unclear. Four comets (C/2017 K2, C/2020 V2, C/2022 E2 and C/2022 L2) cross inbound $r_h = 3.8$ au where the onset of water-driven activity is foreseen by the WEB activity model. All these four comets show a drop of activity at $r_h < 3.8$ au, suggesting $A_d \gg A_r$. This explains the R_n estimate at $r_h = 5$ au: closer to the sun, $Af\rho$ may not increase anymore. C/2022 E3, with $k = 2$ fitting both $Af\rho$ at $1 < r_h < 4$ au (Figure 9) and the dust tail at $3 < r_h < 85$ au (Figure 1), may be characterised by $A_d \approx A_r$.

Table 3 summarises the dust and nucleus properties of the comets observed at TNG. Model parameters v_d, σ, k are described in details in Section 4; d_1 and d_2 are the mean square errors of the fit in case of isotropic and anisotropic dust ejection, respectively, as defined in paper I. About 60% of the sample is best fit ($d_1 > d_2$) by latitudinal anisotropic ejection (assuming zero nucleus obliquity, paper I), confirming that most Oort cloud nuclei are characterised by latitudinal inhomogeneity necessarily due to activity occurred before the ejection from the outer disk into the Oort cloud (paper I). Data in Table 3 allow us to investigate the correlation between the free parameters of the probabilistic tail model and R_n (Figures 11, 12, and 13).

The free parameter v_d fits $v_{10}\sqrt{R_n/R_1}$ (Figure 11), excluded four data at $R_n \approx 1$ km and $v_d = 1$ m s^{-1}. In case of a very elongated nucleus, the radius of the sphere having the same nucleus volume is much smaller than $\sqrt{A_d/\pi}$. Moreover, the smaller the nucleus, the farther its probable shape from a sphere. We conclude that these four data refer to very elongated or contact-binary nuclei of volume equivalent $R_n < 1$ km. The agreement between data and model allows us also to conclude that the R_n values, actually lower limits of the nucleus radius, are very close to their actual values, i.e., that all the nuclei are mainly composed of water-poor pebbles (Section 4.1), consistent with the $Af\rho$ drop at the onset of water-driven activity of C/2017 K2, C/2020 V2, C/2022 E2 and C/2022 L2.

Figure 12 shows the correlation between the β dispersion σ and R_n (Table 3). The three data at $R_n \approx 1$ km and $\sigma = 0.5$ belong to the same group at $R_n \approx 1$ km and $v_d = 1$ m s^{-1}, i.e. they probably refer to very elongated or contact-binary nuclei of volume equivalent $R_n < 1$ km. Also the datum at $R_n \approx 1$ km and $\sigma \approx 1.5$ belongs to this group: together with the other data at $R_n < 1$ km and $\sigma > 1.5$, it suggests a high σ value not correlated to R_n and thus due to a high dispersion of the dust bulk density ρ_d (Section 4.3). After these considerations, the correlation between R_n and σ, predicted by the WEB model, is evident. The same occurs in Figure 13, taking into account the low sensitivity of the errors d_1 and d_2 to the parameter k, which allows us to sample integer k values only.

6. Discussion

6.1. C/2017 K2

Pre-discovery images of C/2017 K2 from 2013 to 2017 allowed Hui et al. (2018) to infer a dust loss rate of $\approx$ 240 kg s^{-1}, requiring an active surface area of 10–100 km^2 for sublimation of supervolatiles such as CO and CO_2, consistent with our results (Table 3). Observations at $r_h = 13.7$ au allowed Zhang et al. (2019) to estimate $1.2 \leq R_n \leq 2.5$ km for CO sublimation, and $3 \leq R_n \leq 180$ km for CO_2, again consistent with our results (Table 3). Jewitt et al. (2019) used the Hubble Space Telescope to study the comet inbound at $13.8 \leq r_h \leq 15.9$ au, with a dust loss rate estimated at $\approx$ 200 kg s^{-1}, probably sustained by CO subli-

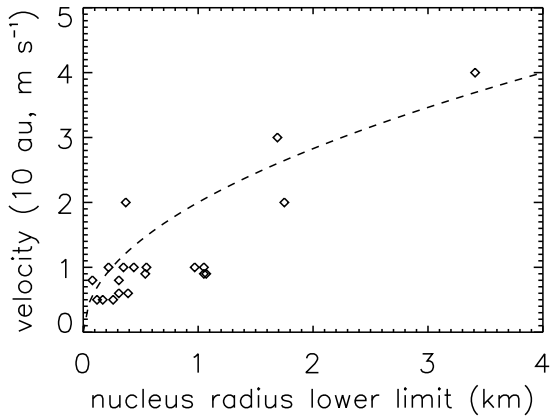

Fig. 11. Correlation between the nucleus radius R_n and the dust ejection velocity v_d (Table 3). Dashed line: Model prediction (Fulle et al. 2022).

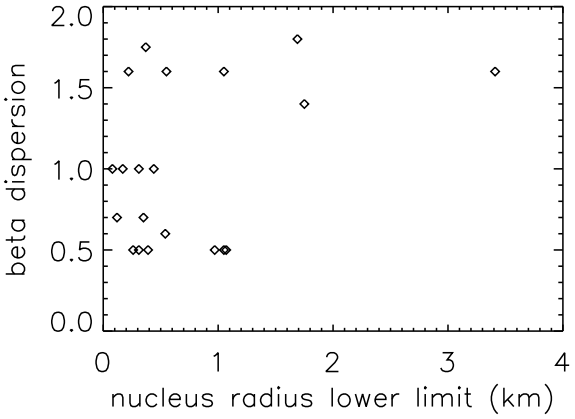

Fig. 12. Correlation between the nucleus radius R_n and the β-distribution dispersion σ (Table 3).

mation from a nucleus area ≤ 2 km^2. Adding inbound observations up to $r_h = 9$ au, Jewitt et al. (2021) estimated an activity onset at $r_h \approx 35$ au, and a dust loss rate at 10 au of $\approx 10^3$ kg s^{-1}, consistent with our results (Table 3). Observations with Hubble Space Telescope at $r_h = 6.8$ au (Zhang et al. 2022) appear consistent with a dust tail composed of millimeter-sized dust.

Polarimetric observations of C/2017 K2 revealed two discontinuous brightening events, namely at ≈ 6 au, possibly associated with changes of supervolatile activity, and at ≈ 2.9 au, possibly when water ice was activated (Kwon et al. 2024). The latter brightening was associated to changes in coma morphology and color, whose trends differ between the inner ($\approx 10^3$ km) and outer ($\approx 10^4$ km) coma. The heliocentric trend of $Af\rho$ (Kwon et al. 2024) and NEOWISE data in the thermal band W1 (3.4 μm) (Milewski et al. 2024) confirm the strong $Af\rho$ inbound drop at $r_h < 5$ au (Figure 2). Images at $2.3 \leq r_h \leq 2.6$ au revealed an active region embedded within an isotropic coma (Garcia et al. 2024), consistent with the anisotropic dust ejection required to best fit its dust tail (Table 3). According to these authors, the dust coma of C/2017 K2 was mainly composed of large dust particles emitted at a velocity of 180 m s^{-1}, higher than our results (Table 3). A JWST study of the comet estimated

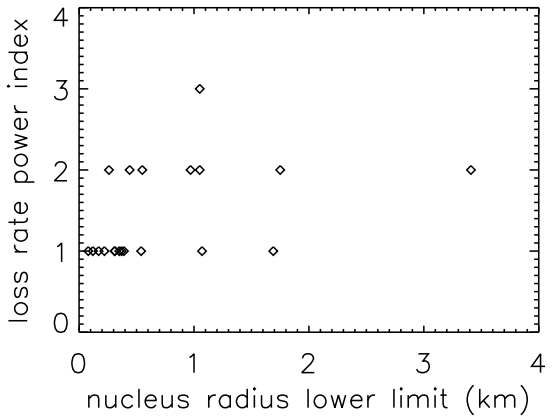

Fig. 13. Correlation between the nucleus radius R_n and the r_h power index k (Table 3).

$R_n \leq 4.2$ km (Woodward et al. 2025), again consistent with our results (Table 3).

6.2. C/2017 U7

Fernández & Sosa (2012), due to the high derived absolute magnitude of C/2017 U7, inferred that it could be one of the largest LPCs observed so far. However, the unresolved dust coma makes impossible to infer any nucleus size from the absolute magnitude. In fact, we obtain a C/2017 U7 nucleus much smaller than 1 km, sufficient to explain its activity. Evangelista-Santana et al. (2022) attributed the distant activity (beyond 7 au) of C/2017 U7 to dust release due to transition from amorphous to crystalline water ice, whereas its continuous activity requires sublimation from supervolatile ices, as assumed in this paper.

6.3. C/2019 O3

Hromakina et al. (2021) observed C/2019 O3 on July and August 2020 at $r_h = 8.92$ and 8.90 au, respectively. They attributed the noticeable differences obtained in the in the coma brightness profiles in different broadband filters to the predominance of subm- and m-sized particles in the coma, consistent with the low σ values provided by the probabilistic tail model (Table 3). Betzler (2024) measured the $Af\rho$ parameter at perihelion, and computed a minimum nucleus radius of 8 km, inconsistent with the coma photometry itself (Table 3, paper I).

6.4. C/2022 E3

Liu & Liu (2024) estimated a dust ejection velocity lower than 14 m s^{-1} for particles larger than 100 μm in the coma, consistent with our results (Table 3). According to the syndyne-synchrone analysis, October 2022 images suggest dust radii from 0.1 to 1 mm, and a dust loss rate increasing from 240 (July 2022) to 480 kg s^{-1} (October 2022). However, due to the size distribution, the loss rate of dust larger than 1 mm is orders of magnitude larger than this estimate (Table 3). Liu & Liu (2024) estimated a water production rate of 370 ± 70 kg s^{-1}, similar to the perihelion one of comet 67P/Churyumov-Gerasimenko, implying, according to the WEB model, a nucleus cross section of 5 km^2 (Fulle et al. 2020), smaller than the 8 km^2 inferred by Liu & Liu (2024). Both these estimates suggest a very elongated

or contact binary nucleus with a significant fraction of water-rich pebbles (Section 5). From narrow-band images obtained close to perihelion, C/2022 E3 showed two side-on jets with central axes aligned approximately north-south (Moulane et al. 2023; Bolin et al. 2024). From the coma morphology in enhanced narrowband CN images, Schleicher et al. (2023) confirmed the two jets in the inner coma, which allowed Knight et al. (2023) to measure a rotation period of 8.7 ± 0.1 hours. These observations are consistent with the highly latitudinal anisotropy of dust ejection required to best fit its dust tail (Table 3), and confirm that C/2022 E3 is probably characterised by a contact-binary nucleus with huge concavities, that are the sources of prominent jets (Shi et al. 2018).

7. Summary and conclusions

1. About 10% of the comet sample shows elongated comae that are not dust tails, so that were excluded from our analysis.
2. All the dust tails of the remaining twenty comets are consistent with the WEB model and the probabilistic tail model.
3. About 60% of the dust tails are best fit by a strong latitudinal anisotropy of dust ejection, due to nucleus inhomogeneity built-up by activity occurred before the comet ejection from the outer disk into the Oort cloud.
4. About 20% of the sample show a dispersion of the dust bulk density much larger than average.
5. The three free parameters of the probabilistic tail models are all correlated to the nucleus size, in agreement with the predictions by the WEB activity model.
6. For about half of the sample, a trend of the coma photometry with respect to the heliocentric distance r_h is available and always consistent with that predicted by the tail model.
7. The comets crossing inbound $r_h = 3.8$ au, i.e., drifting from supervolatile driven to water driven activity, show a drop of coma photometry, suggesting a nucleus mainly composed of water-poor pebbles.
8. The measured dust ejection velocity fits the values predicted by Fulle et al. (2022), suggesting that the estimated lower limits of the nucleus radii are very close to their actual values, i.e., that all the observed comets are mainly composed of water-poor pebbles.
9. About 85% of the nucleus radii are $R_n < 1$ km; About 25% of these are probably very elongated or contact-binary nuclei.
10. About 5% of the nuclei have $R_n > 2$ km and, if primarily composed of water-rich pebbles, at $r_h \approx 1$ au would exceed the 1P/Halley dust activity at the 1σ level (Fulle et al. 2023).
11. All the observed comets are mainly composed of water-poor pebbles and have $R_n < 4$ km, i.e., they would pose no dust-related hazard to Comet Interceptor (Fulle et al. 2023).

Acknowledgements. We thank the Italian Space Agency (ASI) within the ASI-INAF agreements 2020-4-HH.0 and 2023-14-HH.0. CARA network is supported by Unione Astrofili Italiani.

References

A'Hearn, M. F., Schleicher, D. G., Millis, R. L., Feldman, P. D., & Thompson, D. T. 1984, AJ, 89, 579
Bertini, I., Vincent, J.-B., Marschall, R., et al. 2025, Planetary and Space Science, in press
Betzler, A. S. 2024, Research in Astronomy and Astrophysics, 24, 095018
Bolin, B. T., Masci, F. J., Duev, D. A., et al. 2024, MNRAS, 527, L42
Busarev, V. V., Petrova, E. V., Shcherbina, M. P., et al. 2021, MNRAS, 502, 1882
Ciarniello, M., Fulle, M., Tosi, F., et al. 2023, MNRAS, 523, 5841
Cremonese, G., Fulle, M., Cambianica, P., et al. 2020, ApJ, 893, L12
Dymock, R. & Miles, R. 2018, Journal of the British Astronomical Association, 128, 347
Evangelista-Santana, M., Carvano, J. M., De Prá, M., et al. 2022, Icarus, 377, 114834
Fernández, J. A. & Sosa, A. 2012, MNRAS, 423, 1674
Fornasier, S., Hoang, H. V., Fulle, M., Quirico, E., & Ciarniello, M. 2023, A&A, 672, A136
Fulle, M., Blum, J., Rotundi, A., et al. 2020, MNRAS, 493, 4039
Fulle, M., Bockelée-Morvan, D., & La Forgia, F. 2023, Advances in Space Research, 71, 4424
Fulle, M., Colangeli, L., Agarwal, J., et al. 2010, A&A, 522, A63
Fulle, M., Lazzarin, M., La Forgia, F., et al. 2022, MNRAS, 513, 5377
Garcia, R. S., Fernández-Lajús, E., Di Sisto, R. P., & Gil-Hutton, R. A. 2024, Icarus, 422, 116267
Henden, A. A., Templeton, M., Terrell, D., et al. 2016, VizieR Online Data Catalog: AAVSO Photometric All Sky Survey (APASS) DR9 (Henden+, 2016), VizieR On-line Data Catalog: II/336. Originally published in: 2015AAS...22533616H
Hromakina, T., Belskaya, I., Krugly, Y., et al. 2021, A&A, 647, A71
Hui, M.-T., Jewitt, D., & Clark, D. 2018, AJ, 155, 25
Jewitt, D., Agarwal, J., Hui, M.-T., et al. 2019, AJ, 157, 65
Jewitt, D., Kim, Y., Mutchler, M., et al. 2021, AJ, 161, 188
Jones, G. H., Snodgrass, C., Tubiana, C., et al. 2024, Space Sci. Rev., 220, 9
Knight, M. M., Skiff, B. A., Schleicher, D. G., et al. 2023, in LPI Contributions, Vol. 2851, Asteroids, Comets, Meteors Conference, 2396
Kwon, Y. G., Bagnulo, S., Markkanen, J., et al. 2024, AJ, 168, 164
Landolt, A. U. 2013, AJ, 146, 131
Liu, B. & Liu, X. 2024, A&A, 683, A51
Mazzotta Epifani, E., Dotto, E., Perna, D., et al. 2021, Planet. Space Sci., 208, 105341
Meech, K. J., Weryk, R., Micheli, M., et al. 2017, Nature, 552, 378
Micheli, M., Farnocchia, D., Meech, K. J., et al. 2018, Nature, 559, 223
Milewski, D. G., Masiero, J. R., Pittichová, J., et al. 2024, AJ, 167, 99
Moulane, Y., Bodewits, D., Hmiddouch, S., et al. 2023, in LPI Contributions, Vol. 2851, Asteroids, Comets, Meteors Conference, 2047
Opitom, C., Fitzsimmons, A., Jehin, E., et al. 2019, A&A, 631, L8
Prodan, G. P., Popescu, M., Licandro, J., et al. 2024, MNRAS, 529, 3521
Schleicher, D. G., Knight, M. M., & Skiff, B. A. 2023, in LPI Contributions, Vol. 2851, Asteroids, Comets, Meteors Conference, 2532
Shi, X., Hu, X., Mottola, S., et al. 2018, Nature Astr., 2, 562
Vera C. Rubin Observatory LSST Solar System Science Collaboration, Jones, R. L., Bannister, M. T., et al. 2021, in Bulletin of the American Astronomical Society, Vol. 53, 236
Woodward, C. E., Bockélee-Morvan, D., Harker, D. E., et al. 2025, Planetary Science Journal, 6, 139
Zacharias, N., Finch, C. T., Girard, T. M., et al. 2013, AJ, 145, 44
Zakharov, V. V., Ivanovski, S. L., Crifo, J. F., et al. 2018, Icarus, 312, 121
Zakharov, V. V., Rotundi, A., Della Corte, V., et al. 2021, Icarus, 364, 114476
Zhang, Q., Kolokolova, L., Ye, Q., & Vissapragada, S. 2022, Planetary Science Journal, 3, 135
Zhang, X. L., Jewitt, D., & Hui, M. T. 2019, MNRAS, 487, 2919

Appendix A: Fits of dust tails

This appendix contains the images obtained at TNG for target comets, together with their fit with model isophotes.

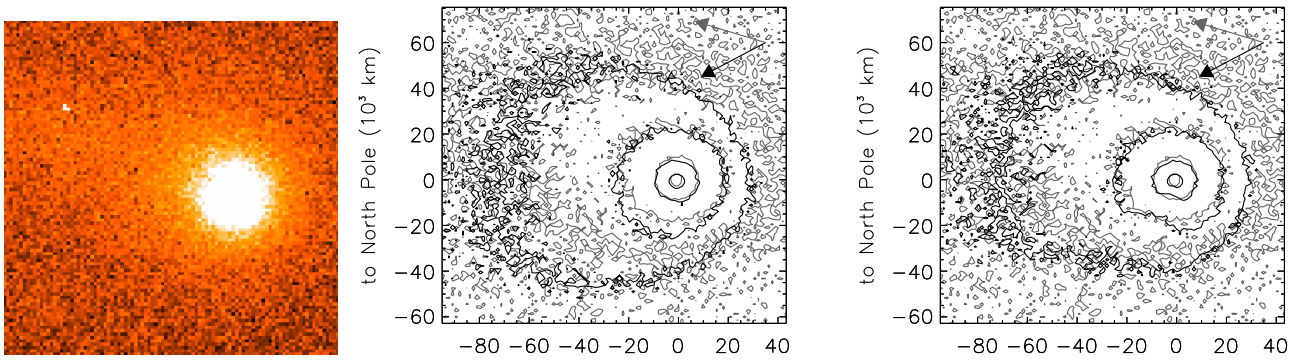

Fig. A.1. Comet C/2016 Q2 (PanSTARRS) on 2022 June, 23. TNG-R image at r_h = 7.55 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

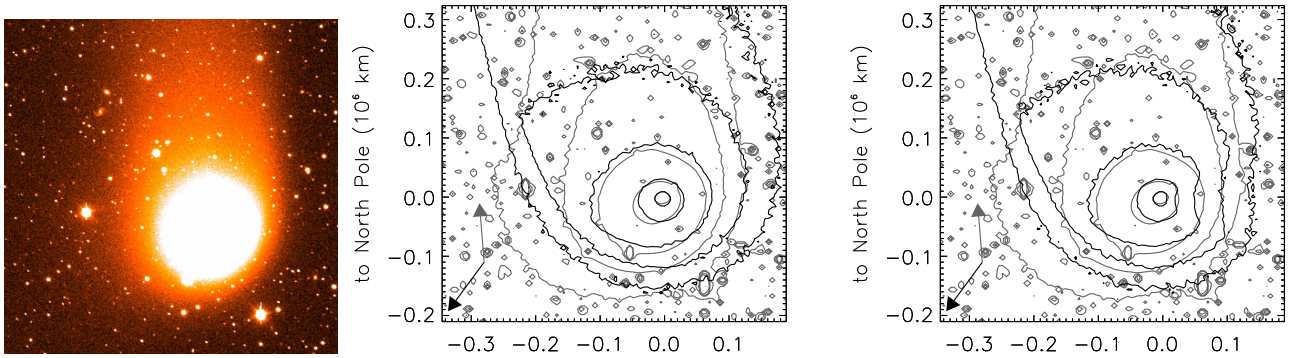

Fig. A.2. Comet C/2017 K2 (PanSTARRS) on 2022 July, 1. TNG-R image at r_h = 2.77 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

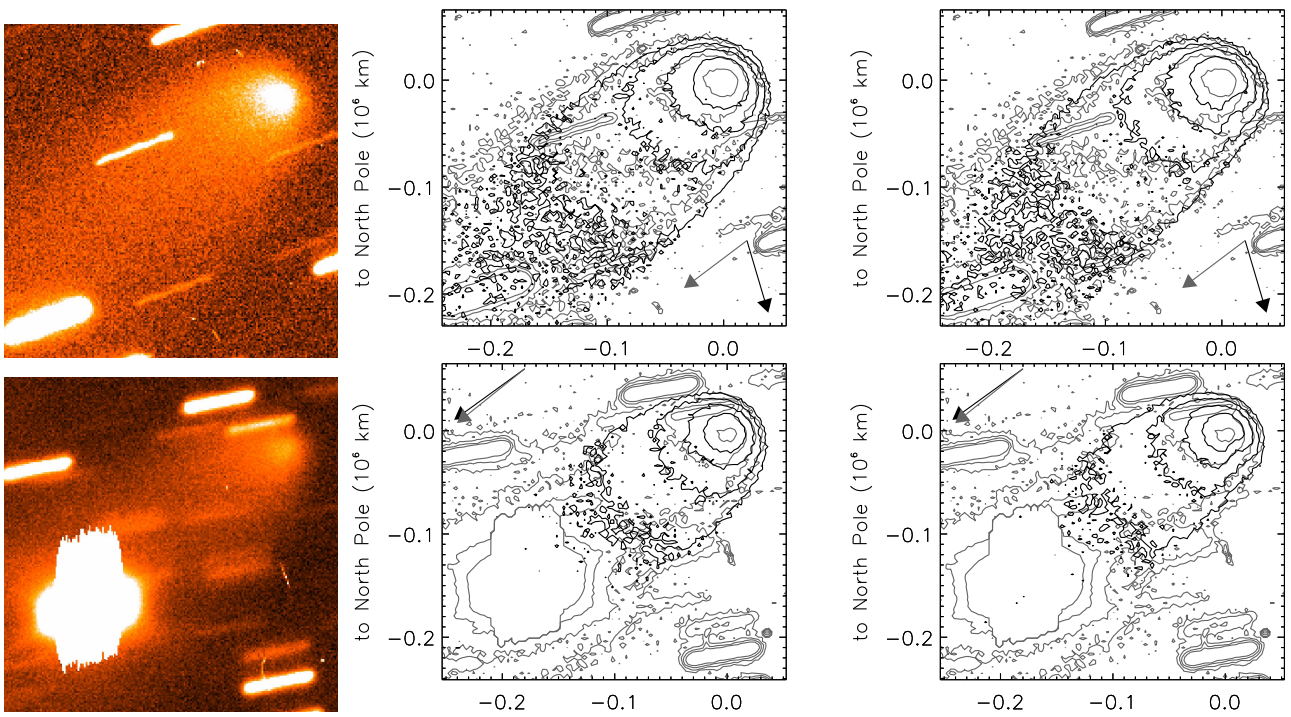

Fig. A.3. Comet C/2017 U7 (PanSTARRS) on 2022 May 29, at r_h = 9.12 au (upper left panel), and on 2022 July 1, at r_h = 9.26 au (lower left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panels) and anisotropic dust ejection (right panels). Further data in Figure 1 and Table 3.

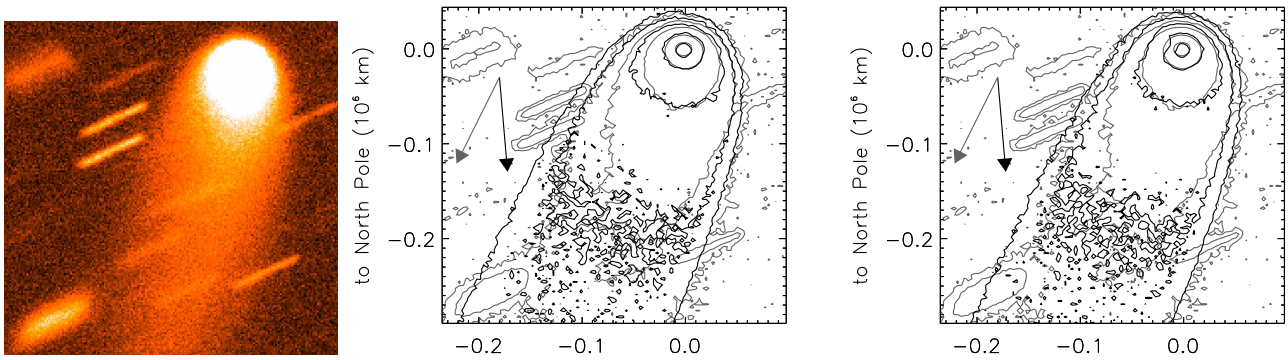

Fig. A.4. Comet C/2019 O3 (Palomar) on 2022 June, 23. TNG-R image at r_h = 9.23 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

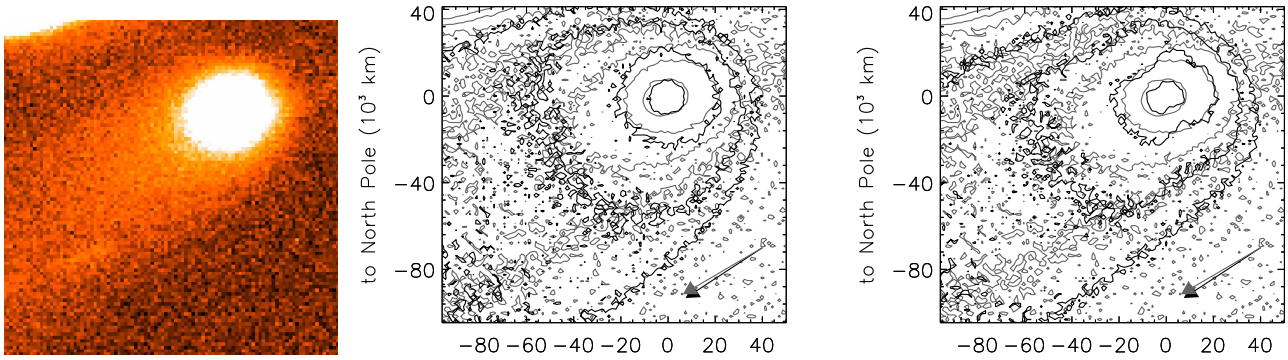

Fig. A.5. Comet C/2020 F2 (ATLAS) on 2022 May, 24. TNG-R image at r_h = 8.82 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

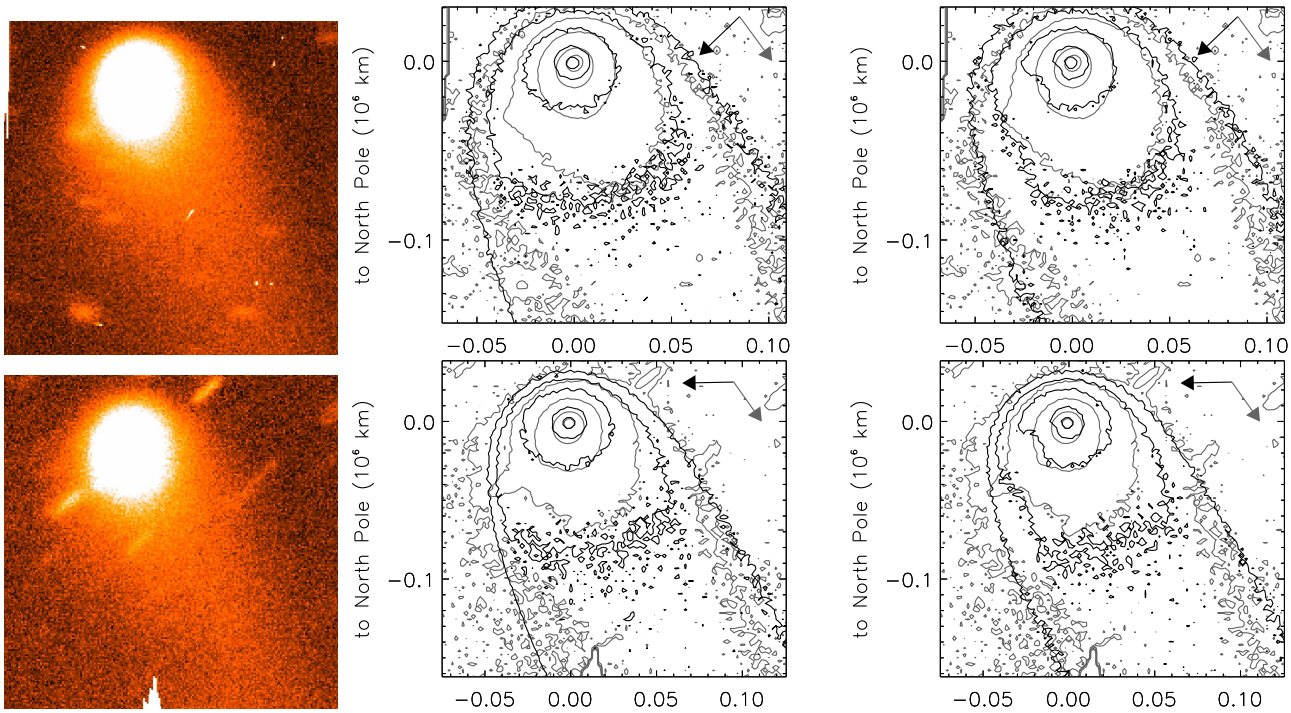

Fig. A.6. Comet C/2020 H6 (ATLAS) on 2022 July 1, at r_h = 5.17 au (upper left panel), and on 2022 August 24, at r_h = 5.36 au (lower left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panels) and anisotropic dust ejection (right panels). Further data in Figure 1 and Table 3.

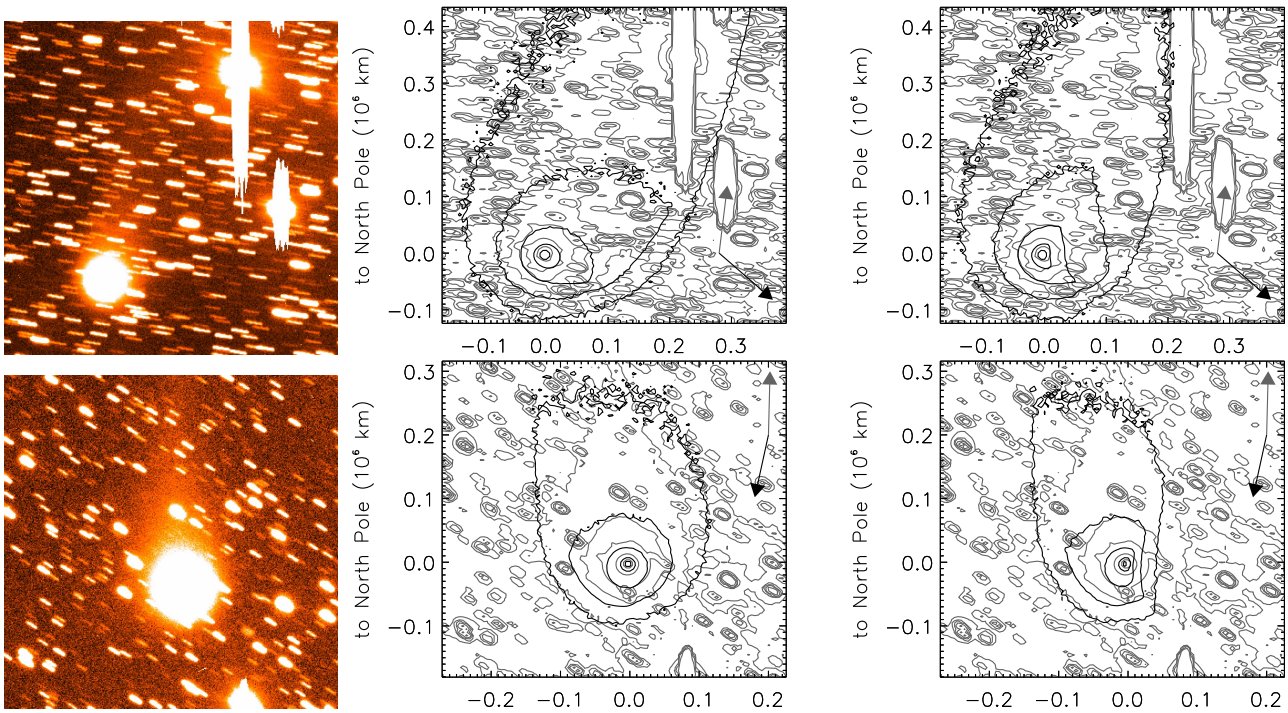

Fig. A.7. Comet C/2020 K1 (PanSTARRS) on 2022 May 24, at r_h = 4.51 au (upper left panel) and on 2022 July 1, at r_h = 4.27 au (lower left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panels) and anisotropic dust ejection (right panels). Further data in Figure 1 and Table 3.

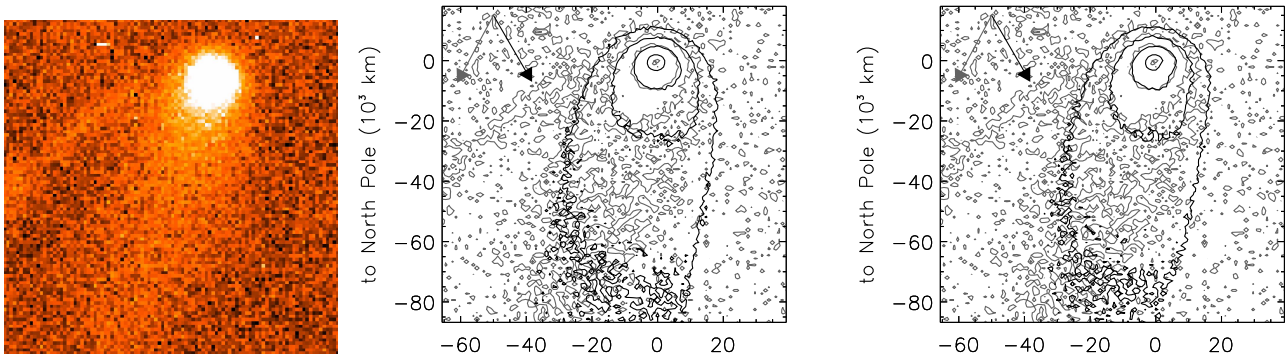

Fig. A.8. Comet C/2020 K6 (Rankin) on 2022 May, 24. TNG-R image at r_h = 6.13 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

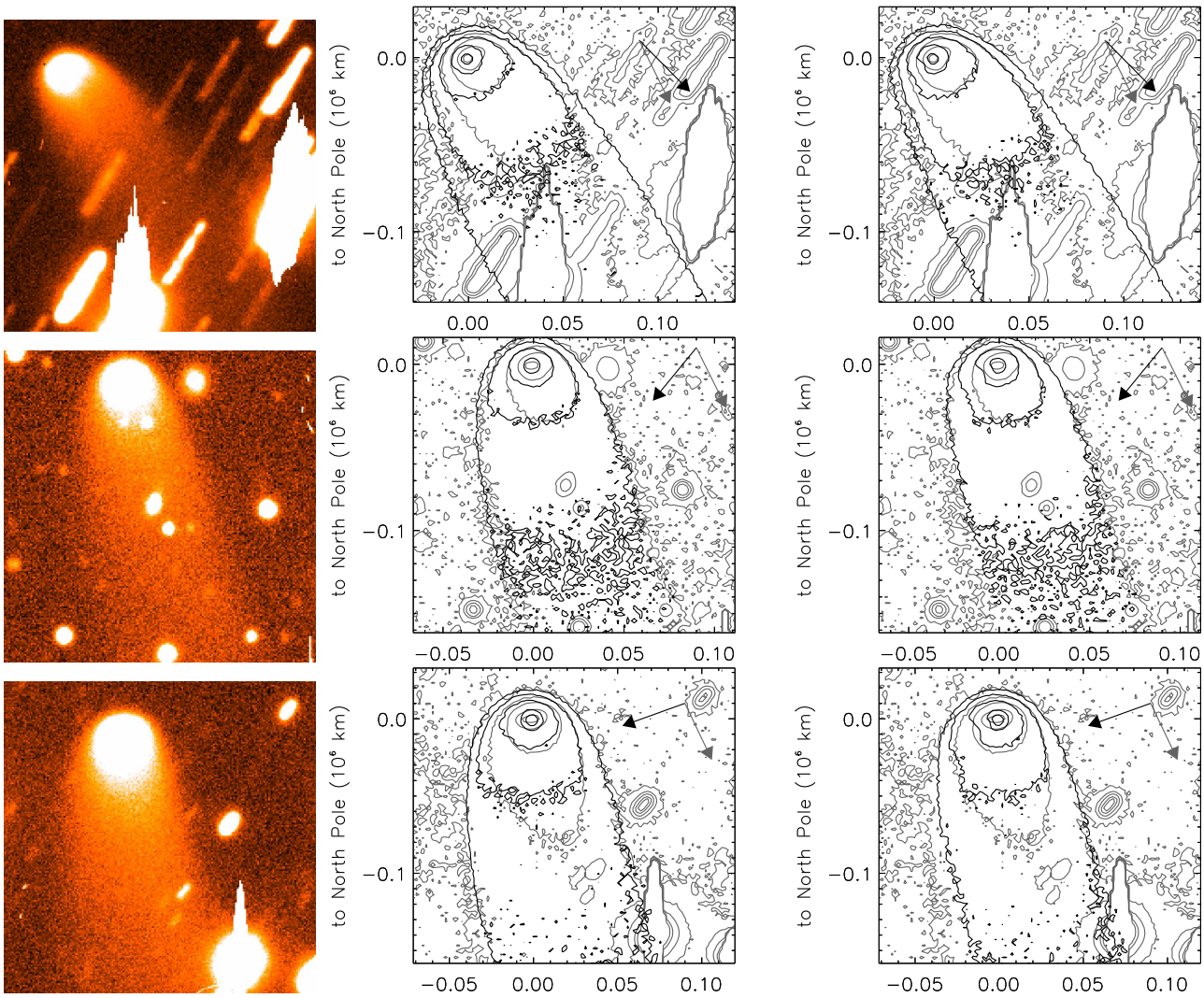

Fig. A.9. Comet C/2020 O2 (Amaral) on 2022 May 24, at r_h = 5.29 au (upper left panel), on 2022 July 23, at r_h = 5.48 au (middle left panel), and on 2022 August 24, at r_h = 5.60 au (lower left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (center panels) and anisotropic dust ejection (right panels). Further data in Figure 1 and Table 3.

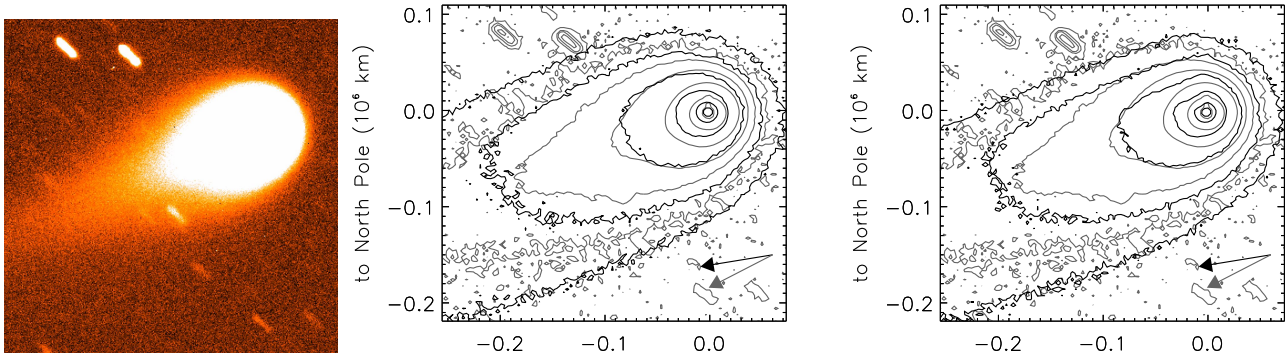

Fig. A.10. Comet C/2020 V2 (ZTF) on 2022 May, 24. TNG-R image at r_h = 4.34 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

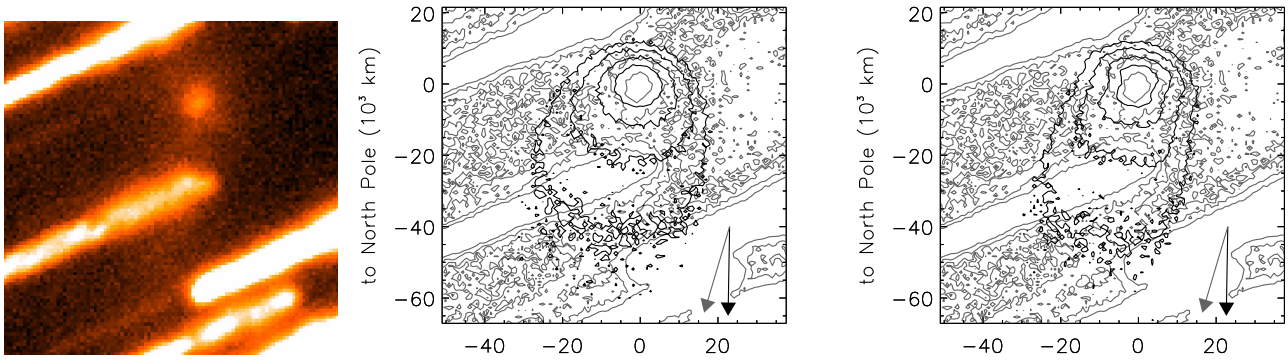

Fig. A.11. Comet C/2021 K2 (MASTER) on 2022 July, 1. TNG-R image at r_h = 5.88 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

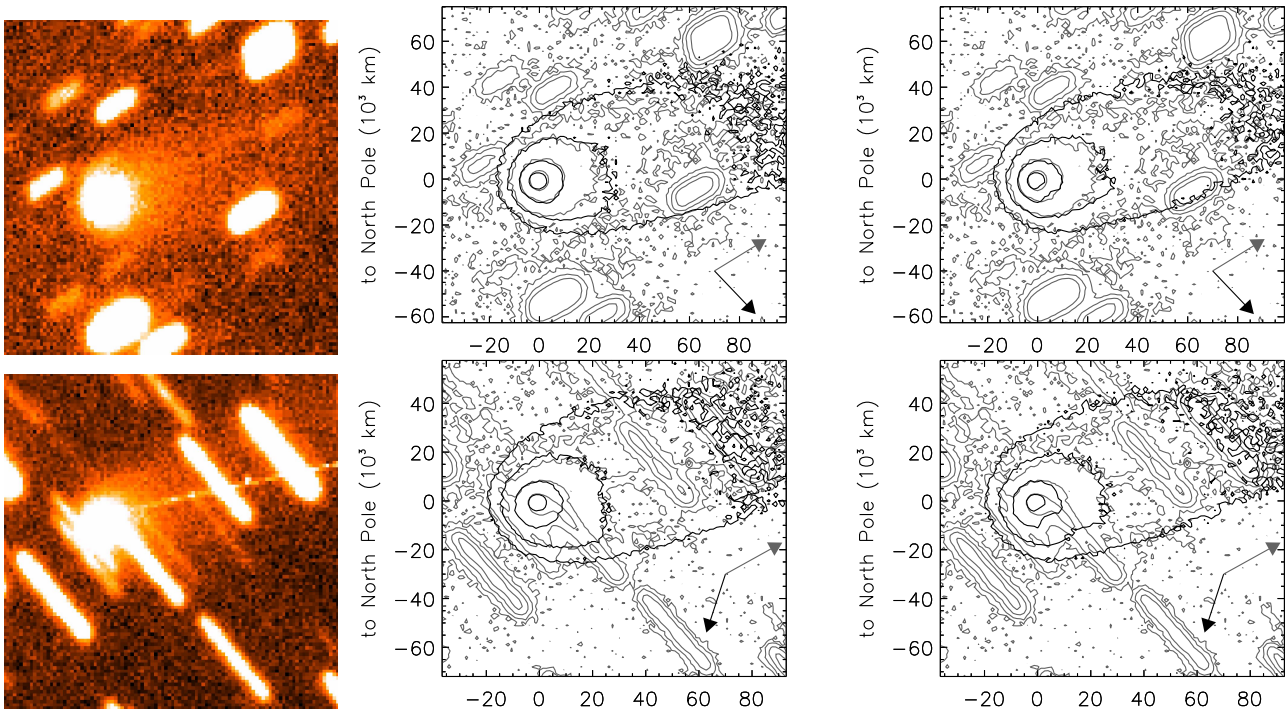

Fig. A.12. Comet C/2021 Q4 (Fuls) on 2022 July 1, at r_h = 7.86 au (upper left panel), and on 2022 August 24, at r_h = 7.78 au (lower left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panels) and anisotropic dust ejection (right panels). Further data in Figure 1 and Table 3.

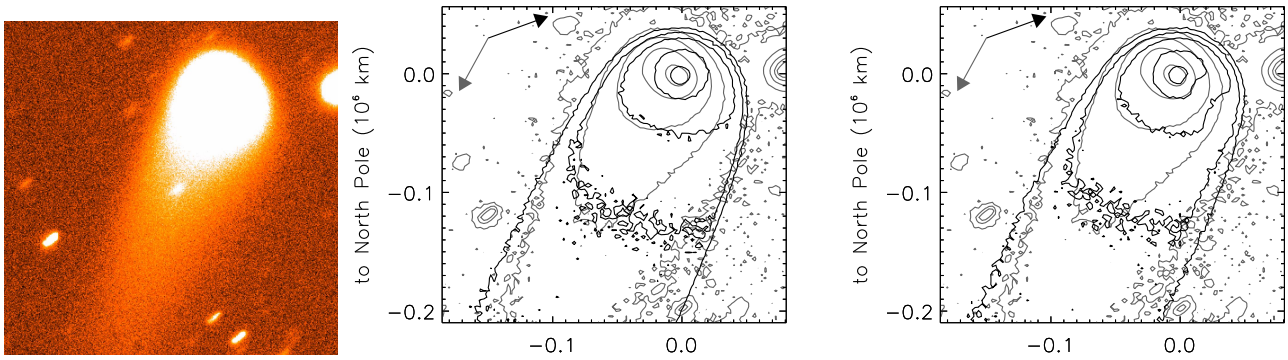

Fig. A.13. Comet C/2022 E2 (ATLAS) on 2023 December, 20. TNG-R image at r_h = 4.37 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

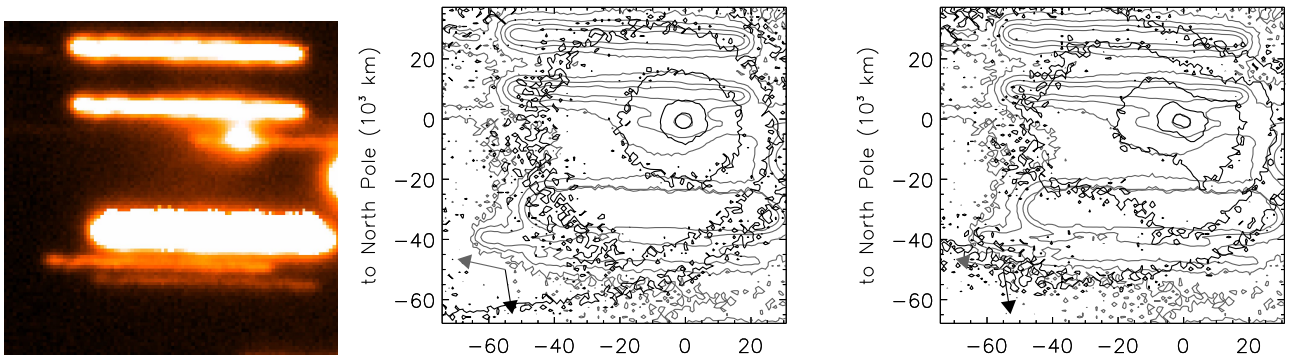

Fig. A.14. Comet C/2022 L2 (ATLAS) on 2022 July, 1. TNG-R image at r_h = 6.39 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

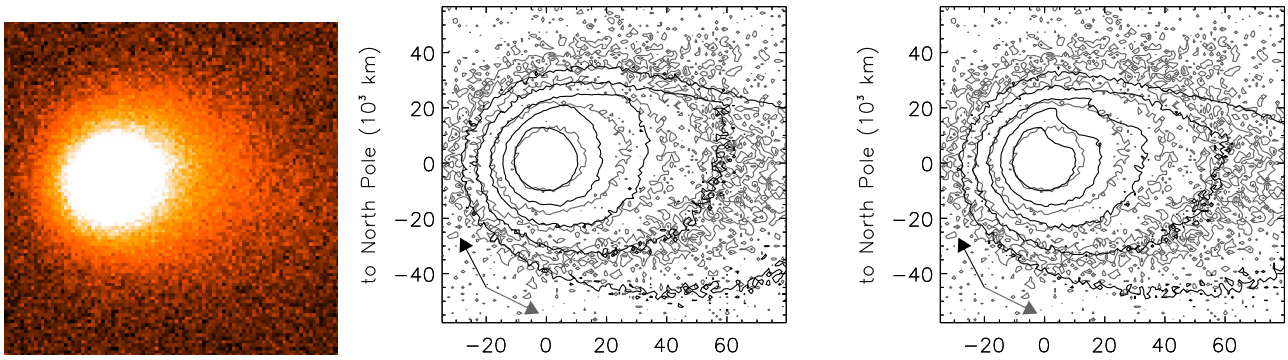

Fig. A.15. Comet C/2022 QE78 (ATLAS) on 2023 December, 19. TNG-R image at r_h = 7.11 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

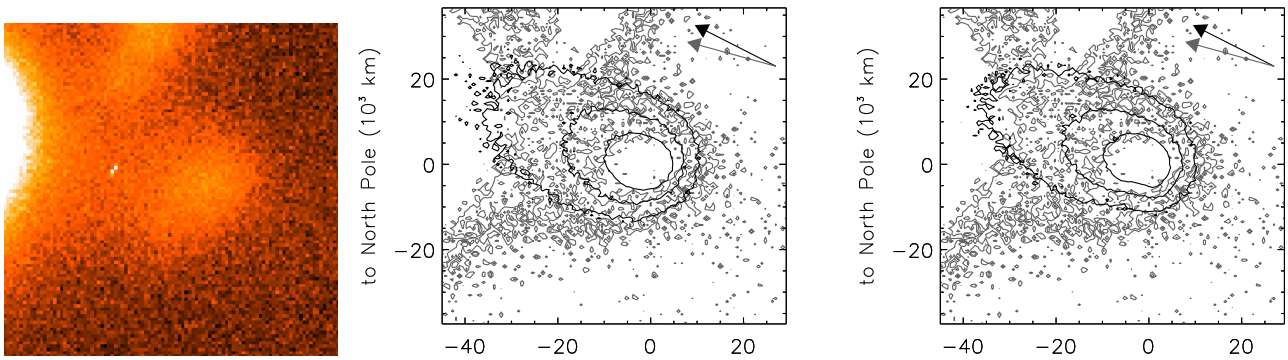

Fig. A.16. Comet C/2022 U1 (Leonard) on 2023 December, 20. TNG-R image at r_h = 4.28 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

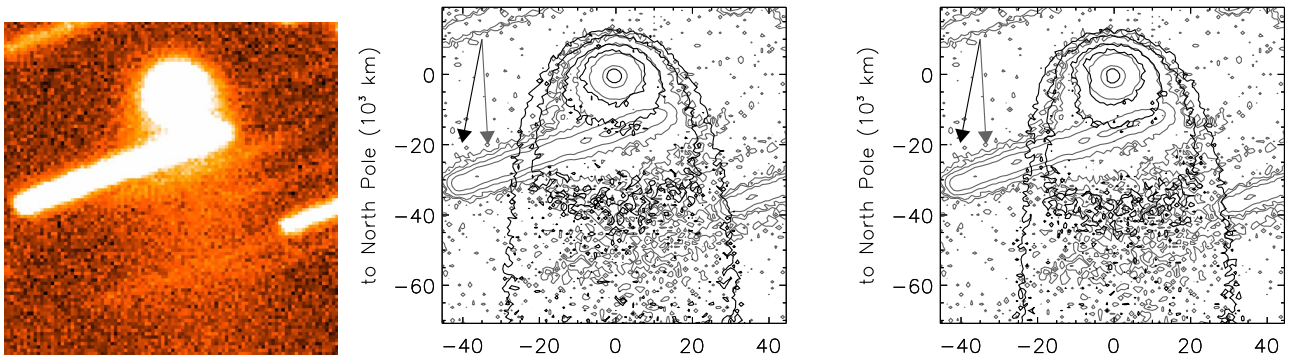

Fig. A.17. Comet C/2023 X2 (Lemmon) on 2024 December, 1. TNG-R image at r_h = 5.88 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

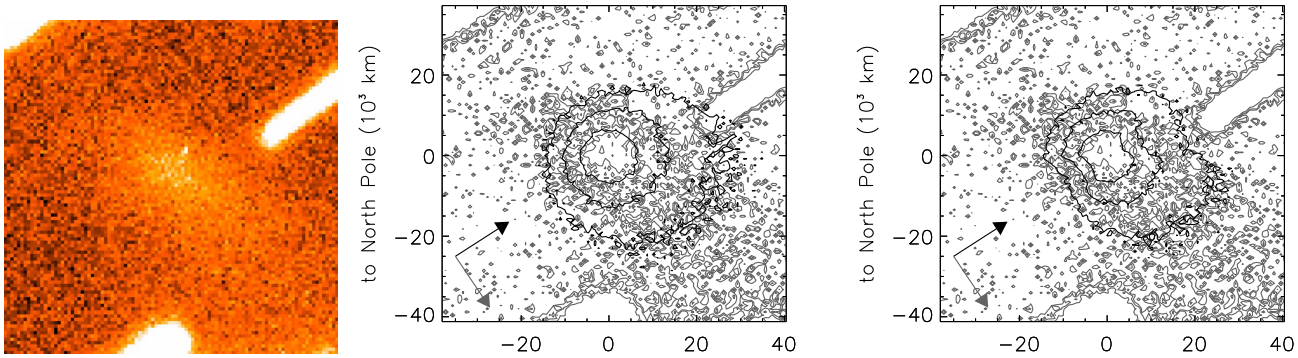

Fig. A.18. Comet C/2023 X7 (PanSTARRS) on 2024 December, 1. TNG-R image at r_h = 4.99 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

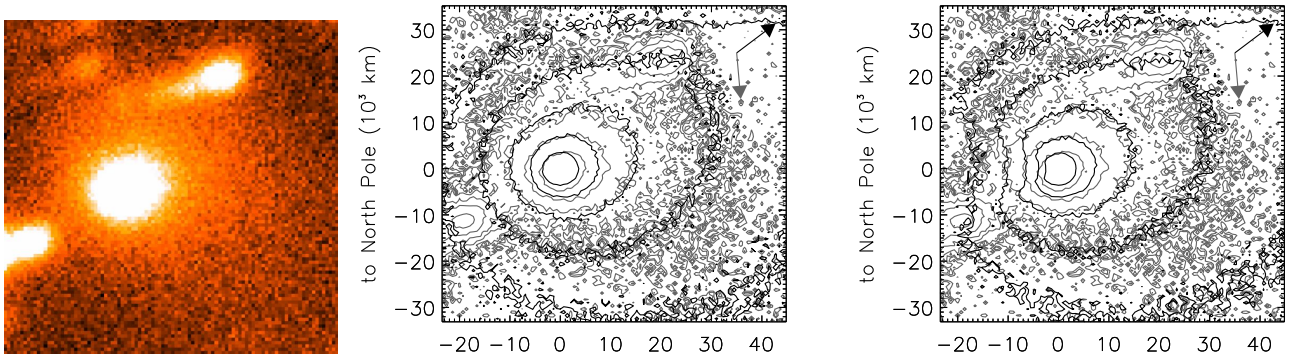

Fig. A.19. Comet C/2024 A1 (ATLAS) on 2024 December, 1. TNG-R image at r_h = 4.23 au (left panel, North is up, East to the left). Observed (grey isophotes) and modeled (black isophotes) dust tail, for isotropic (middle panel) and anisotropic dust ejection (right panel). Further data in Figure 1 and Table 3.

3.3.2 My contribution to the paper

My contribution to Paper 5 in terms of data collection and computation of $Af\rho$ values is detailed below.

Photometric Observations

Observations are carried out from the telescope control room. At Copernico, data can be acquired throughout the night, while at TNG, observations take place in a six-hour slot, which can cover either the first or second part of the night. The operations I carried out during the data collection phase at the Copernico and TNG telescopes are as follows:

1. **Instrumentation initialization.** I performed this operation exclusively at the Copernico telescope. At TNG, this procedure is not permitted for visitor-mode observers and is instead carried out by a technician assigned to assist the observer throughout the observing slot. The initialization operations performed at the Copernico telescope are listed below:

    - **Cooling of the CCD down to -95°C.** The first procedure to start is cooling the CCDs. They must be cooled to extremely low temperatures to minimize instrumental noise and thermal effects that may compromise data quality. At the Copernico telescope, CCDs works at -95°C, while at TNG the temperature is -120°C. Since this procedure takes a long time, it is carried out at dusk, before the conditions for acquiring the data occur;

    - **Initialization of the Asiago Faint Object Spectrograph and Camera (AFOSC).** AFOSC is a focal reducer instrument with a field of view of 8.8 × 8.8 arcmin. Among the wide set of operations it allows – such as low and medium resolution grism spectroscopy, polarimetry, and spectro-polarimetry observations – we use it for imaging. The instrument is equipped with sloan ugriz and BV Johnson-Bessel filters. The initialization of the instrument is performed through the AFOSC Control Software in the Init menu;

    - **Initialization of the guiding camera.** This camera is used to stabilize the telescope during differential tracking of the comet. It is pointed toward a target star, which I properly selected outside the field of view containing the comet to be observed. The target star must not be too faint – since the telescope might lose tracking – nor too bright, as it could saturate the counts recorded by the guiding camera;

    - **Focusing of the telescope and guiding camera.** To perform telescope

focus, it is necessary to point the telescope at a star field free of clouds and bright stars and capture an image for about 10 seconds, with no filter inserted. The exposure time depends on the brightness of the stars in the field. During exposure, the light captured from the stars enters a pyramidal prism located in the focal plane of the telescope. At this point, the light is divided into four beams, projected onto the CCD, and visible through the user interface. If the telescope is perfectly in focus, the four spots are symmetrical and of similar intensity. If, on the other hand, there is a focusing error (in the case of clouds or high humidity), the intensity distribution between the four spots changes. In case of massive differences in distribution, it is advisable to repeat the procedure as well as in case of wide rotations of the telescope or sensible temperature variation during the night. This operation is crucial for properly focusing the photometer to obtain high-resolution images. Since other observing campaigns may adopt different focusing configurations – such as exoplanet observations often using slightly defocused camera – it is mandatory to perform this step before acquiring any data. An incorrect focus setting may compromise the scientific success of the observing night.

- **Dome opening and tracking.** Before opening the dome, it is essential to check weather forecasts and the relative humidity parameter. In the presence of adverse weather conditions (e.g. strong winds, rain, or snow) or humidity exceeding 85%, the dome cannot be opened. Forcing the dome to open under high humidity conditions poses a serious risk of acquiring low-quality images. When water condenses into microdroplets, it can deposit onto cold optical surfaces (mirrors, lenses, filters, or CCD sensors), causing light scattering and contrast loss in the images, as water randomly disperses and refracts light. Additionally, opening the dome under such adverse conditions could lead to irreversible damage to the instrumentation, resulting in substantial economic loss.

2. **Acquisition of calibration frames: bias and flat field.** The bias frame is an image taken with the shutter closed and zero exposure time, used to measure and remove the CCD's intrinsic readout noise. The flat field is an image acquired on a uniformly illuminated region of the sky (or dome), used to correct for pixel-to-pixel sensitivity variations and vignetting effects visible near the edges of the CCD. Typically, flat fields can also be acquired with the dome closed, by pointing the telescope toward a uniformly illuminated white surface inside the dome. These calibration frames, taken in the same filter cometary data are acquired,

are subsequently used during the data reduction of cometary observations. An example of bias and flat field calibration files is shown in Figure 3.2.

3. **Data acquisition.** Once the initialization and calibration frames (bias and flat) are completed, the observation of the targets begins. During the preparation phase (approximately one week before the observing run), the project's Principal Investigator provides a target list containing up to seven comets and two standard fields. The number of observable comets ranges between 4 and 7, depending on the available nighttime hours. In summer, when nights are shorter, typically four comets are observed, while in winter, with longer observing time available, up to seven targets can be included.

   For each cometary target, the goal is to obtain at least one hour of total photometric exposure. Images are acquired using photometric filters r-sloan and BVRI Johnson-Bessel, which will be used to characterize the Comet Interceptor potential target. At the Copernico telescope, observations are performed using only the r-sloan filter.
   Before saving any scientific frame, a short-exposure test image (about 15 seconds) is taken to verify the comet's position in the field of view and to estimate the appropriate exposure time. The telescope is then set to differential tracking mode, allowing it to follow the comet's apparent motion across the sky. Typical exposure times range from 10 to 15 minutes per image, although this parameter is one of the most challenging to define during the observation itself. The exposure time may vary according to several factors, such as:

   - **Comet visual magnitude (V).** Faint comets ($V \geq 17$) require longer exposure times to increase the signal-to-noise ratio of the acquired image, with minimal risk of CCD saturation.

   - **Weather conditions.** Low-altitude clouds moving rapidly across the sky may partially or completely obscure the comet for an uncertain period. In such cases, it is advisable to close the shutter when near the end of an exposure. Otherwise, switch to a different cometary target located in a cloud-free region of the sky. Closing the shutter ensures that subsequent frames are dark, which can be subtracted during data reduction, preserving the valid portion of the exposure and avoiding noisy data caused by cloud contamination;

   - **Stellar field.** Comets are sometimes located in densely populated stellar fields (see Figure 3.3, right panel). Long exposure times may cause the comet's trajectory to intersect nearby stars, compromising the image quality.

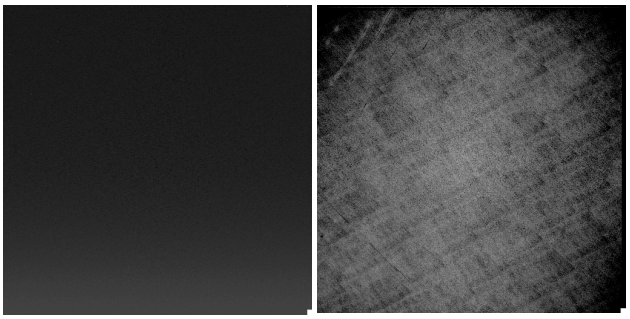

Figure 3.2: An example of bias (left panel) and flat field (right panel) calibration files acquired at TNG during an observing run.

Even in less crowded fields, the presence of a very bright star (Figure 3.3, central panel) requires shorter exposure times to prevent CCD saturation, which could damage the instrumentation and degrade image quality.

For standard fields, at least three to four measurements at different airmasses – from low to high – are acquired, interleaved with the comet observations, in order to cover the airmass range of the cometary targets. These standard fields are used during data reduction to calculate the atmospheric extinction coefficient and photometric zero point.

Since the observing program began in February 2022, photometric data have been acquired for a set of 42 comets. The typical observing schedule allocated time for an average of 6 targets per night. Within this dataset, the most monitored objects have been C/2023 R1 (PANSTARRS, 8 observations), C/2021 Q4 (Fuls, 7 observations), and C/2022 U1 (Leonard, 7 observations).

At the end of each observation session, I systematically drafted a nightly report, which I then shared with the research team. This documentation practice allows for detailed tracking of the observations made and provides all team members with a clear and immediate understanding of the data acquired, thus facilitating the subsequent stages of collective analysis. Each report summarizes the essential metadata associated with each FITS file, specifying the type of object observed, the exposure time used, and including relevant notes on the observation conditions or specific events that occurred during acquisition. A sample selection of these reports can be found in the Appendix B.

Data Reduction

The photometric reduction and calibration of the images acquired with Copernico and TNG telescopes have been performed with an IDL script developed by Prof. Ivano Bertini (IB), a member of these projects supporting the Comet Interceptor mission.

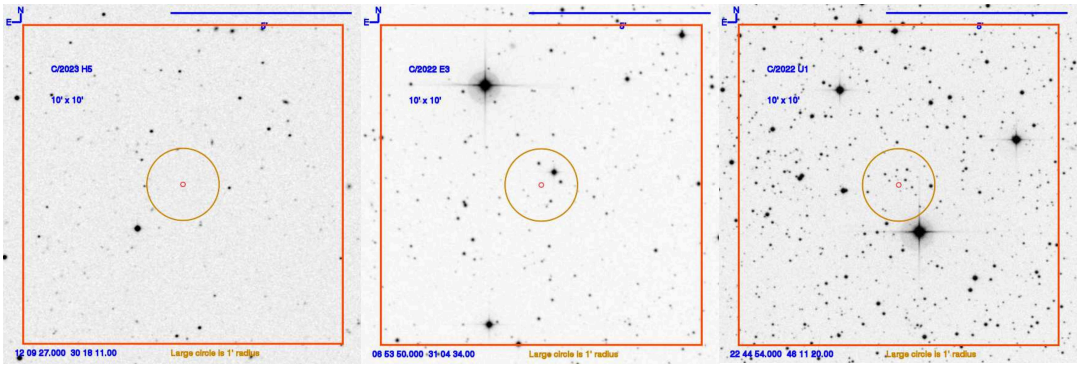

Figure 3.3: Assessing exposure times for capturing images of comets requires analysis of the relevant star fields. (Left panel) In regions of the sky free of star clusters and bright stars, the exposure time can be set to at least 10-15 minutes. (Center panel) With a very bright star in the upper left, the exposure time should not exceed 15 minutes due to the potential risk of image saturation and corruption of the portion of the image affected by the comet. (Right panel) If the comet is located in regions of the sky with many stars, some of which are very bright, it is imperative to set very low exposure times in the order of 5 minutes, at least during the first exposure.

For each cometary image, the process starts by reading the related FITS file to initialize the observational parameters:

- Airmass and exposure time: taken from the FITS file header;
- Atmospheric extinction coefficient and zero point: determined with specific calibrations by IB;
- Pixel scale: obtained by technical documentation for each telescope;

Once defined input metadata, the script continues with the subsequent operations for image processing:

1. **Image calibration in bias and flat**. The FITS file containing the raw cometary image ($\mathrm{img}_{\mathrm{raw}}$) is corrected from systematics related to (i: bias) electronic offset added to a CCD image during the readout process and (ii: flat) pixel-to-pixel differences in the photodetector sensitivity and distortions in the optical path.

   The median image is computed either for bias (μ_B) and flat (μ_F), from the five to ten calibration images acquired during the observations (Figure 3.4). The median bias is subtracted from the cometary image, and the result is divided by the flat median.

$$\mathrm{img}_{\mathrm{calib}_{BF}} = \frac{\mathrm{img}_{\mathrm{raw}} - \mu_B}{\mu_F} \tag{3.1}$$

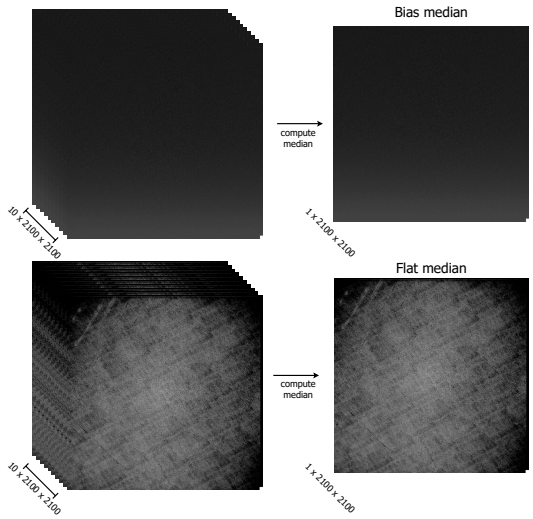

Figure 3.4: Computation of the median image for bias and flat field. The FITS files are loaded, obtaining a $n \times w \times h$ tensor. Trimming is applied to remove the regions at the edges of the image. Then, the median is computed at each pixel across the tensor with the IDL command `medarr`.

2. **Trimming**. The calibrated image $\text{img}_{\text{calib}_{BF}}$ is cut out with the same strategy to remove marginal regions not useful for calibration.

3. **Sky background subtraction and unit time normalization**. The sky background to be subtracted from the image (I_{sky}) is computed as average between two star-free areas close to the comet. Then, the image is normalized by dividing it for its exposure time (t_{exp}).

$$I_{\text{red}} = \frac{I_{\text{trim}} - I_{\text{sky}}}{t_{\text{exp}}} \tag{3.2}$$

   In IDL, the two star-free regions are visually identified through the software interface. The trimmed comet image is displayed using the `atv` command, and the saturation level is increased to enhance the visibility of the coma and tail extension. Subsequently, two suitable sky regions are selected for background estimation. The saturation is deliberately raised to ensure that the chosen areas are not contaminated by the comet's diffuse emission from the coma or tail. This process is illustrated in Figure 3.5.

4. **Image calibration in magnitude**. This operation transforms the image pixels count in visual magnitude values. Two alternative approaches exist for the computation:

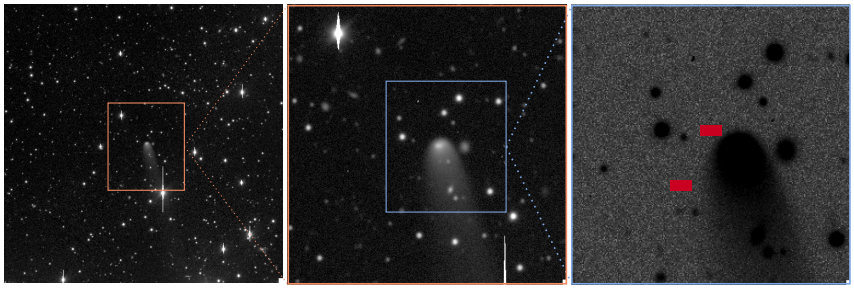

Figure 3.5: Determining the two starless areas for sky background removal. (Left panel) The image is opened in IDL and the area where the comet is located is identified (orange box). (Middle panel) A zoom in on the comet area is performed until a clear view of the sky arount it is achieved. (Right panel) The color of the image is inverted and the saturation is significantly increased in order to assess the maximum extent of the coma and tail. Finally, the two regions of interest in the sky are defined (red rectangles). Comet: C/2020 O2, image acquired at TNG on August 24, 2022.

$$I_{\mathrm{cal}} = I_{\mathrm{red}} \times 10^{-0.4\,(ZP - k\,\mathrm{Air})} \tag{3.3}$$

where I_{red} is the outcome from the previous step, ZP is the photometric zero point, k is the atmospheric extinction coefficient and Air is the airmass of the comet. Alternatively, calibration can be performed through differential photometry with respect to standard stars in the field [49]:

$$I_{\mathrm{cal}} = I_{\mathrm{red}} \times 10^{-0.4\,m_{\mathrm{diff}}} \tag{3.4}$$

where m_{diff} represents the average differential magnitude obtained with the comparison with the standard stars.

5. **Computing the Afρ curve**. Pixel values in magnitude are used to estimate the $Af\rho$ parameter, measuring the dust activity in cometary coma. The parameter A is the average grain albedo, f is the filling factor in the aperture field of view and ρ is the linear radius of the aperture at the comet. For a set of radius ρ (expressed in km), the script uses the magnitude values to compute $Af\rho$ as a function of the distance from the cometary nucleus, as:

$$Af\rho = \frac{(2\,\Delta\,r)^2\,10^{0.4\,(m_{\odot} - m_{\mathrm{com}})}}{\rho} \tag{3.5}$$

where r is the heliocentric distance, Δ is the geocentric distance, $m_{\odot}$ the solar magnitude in R filter and m_{com} the measured magnitude of the comet. In the end, the $Af\rho$ values are represented on a two-dimensional plot as a function of ρ

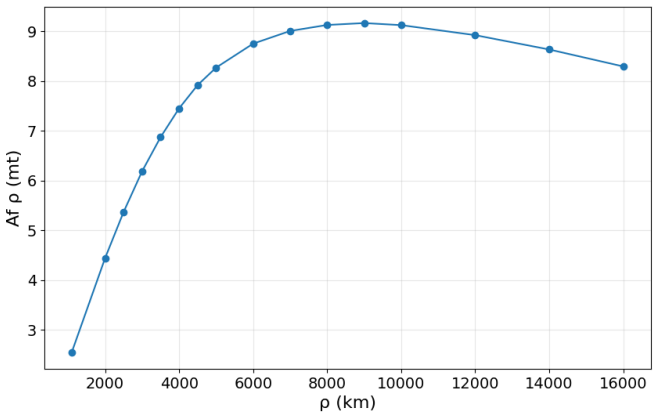

Figure 3.6: An example of $Af\rho$ curve computed for the comet C/2020 O2. The image of the comet was acquired at the TNG on July 23, 2022, with an exposure time of 480 seconds. The data reduction process was carried out on July 9, 2024.

and saved into a `CSV` file for subsequent analysis [30]. An example of $Af\rho$ curve computed for the comet C/2020 O2 is illustrated in Figure 3.6.

Chapter 4

Conclusions

The main goal of this thesis was to collect and analyze photometric data of moving and/or active objects in the Solar System and beyond, namely comets and exoplanets. In this Chapter, I report some conclusions on the work I performed for the development of Machine Learning models for the identification of exoplanet candidates (Section 4.1) and on the results obtained in the context of activities conducted in support of the ESA/Comet Interceptor space mission (Section 4.2).
A discussion of open issues related to exoplanet detection is provided in Section 4.1.1.

4.1 Conclusions on Exoplanet detection with Deep Learning

Regarding exoplanet candidates detection, state-of-the-art supervised Machine Learning approaches were mainly based on increasingly complex and overparameterized CNNs, often trained on relatively small datasets of labeled TCEs.

Research in this field focused on expanding network depth, width, and resolution, designing architectures capable of processing all features relevant for human classification of TCEs – without assessing the drawbacks of such choices (see Section 4.1.1). These assumptions likely resulted in an insufficient number of training samples to ensure proper generalization, thus increasing the risk of overfitting on the training manifold. This motivated the exploration of alternative approaches with the goal of minimizing CNNs architectural complexity.

In developing models to detect best candidates for further study, I drew on the results obtained by [80, 87, 88], which highlight the advantages of compact architectures. Having defined a basic network architecture such as Astronet – which represented

a major innovation in the field of exoplanet detection – I began to design models favoring architectural choices aimed at reducing the number of parameters. In Paper 2, I experimented with network architectures equipped with smaller convolutional filters and reduced pooling sizes [18], single TCE input representation, and a single fully-connected layer for classification.

The results I obtained have shown that the use of simple Deep Learning models – with minimal extension in depth, width, and resolution – for the classification of TCEs is preferable for a variety of reasons, including the possibility of extending their application to data acquired from future missions such as PLATO, without the need to design and train a new architecture from scratch (as this happened so far with the *Kepler*, K2 and TESS missions).

The simplification we adopted in resolution and width – processing only photometric data – ensures better portability of the model to different transiting surveys (as demonstrated in Paper 2 and Paper 3). From a technical point of view, this minimization has allowed us to train the model on a set of data from different missions (Paper 2) and to exploit pre-training and fine-tuning techniques (Paper 1). The advantages observed consist of a mitigation in overfitting risk and an improvement in the model's generalization. Table 4 and Figure 6 in Paper 2 illustrate that the predictive performance of the CNN improve when trained on the combined *Kepler* and TESS datasets of TCEs.

The minimization of complexity in depth was achieved by equipping our models with a single classification layer. This choice is consistent with the universal approximator property of neural networks, according to which even shallow architectures can represent complex functions. In this specific context, the applicability of such a property is further justified by the high degree of separability characterizing the feature vectors produced by the convolutional networks, which significantly simplifies the mapping for the final classification block. In practice, reducing the number of layers led to a quadratic reduction in the number of parameters and, consequently, a lower risk of overfitting training data.

A central question arising from these findings is why simpler architectures can compete with, or even outperform, significantly more complex models. The trick behind this effectiveness lies in the alignment between the model's representational capacity and the intrinsic dimensionality of the input signals: global views produced from TCEs are essentially low-dimensional features. While deep architectures are optimized for processing high-dimensional data, they often prove redundant for one-dimensional time-series, where excessive parameters are more likely to capture noise (e.g., instrumental artifacts or stellar variability) rather than the underlying transit shape. However, the

limitation of this simplicity becomes apparent when moving to more complex tasks, such as distinguishing between different eclipse types (e.g., planetary transits vs. eclipsing binaries). In such cases, the demand for expressivity increases, as the network must learn subtle morphological features – such as secondary eclipses [94] – which may require a tailored increase in resolution or depth to break the degeneracies between similar signal shapes.

Recent architectures like Watson-Net [21] and RAVEN [34] further validate the focus on efficiency and robustness explored here. By emphasizing specialized feature extraction and automated vetting, these works align with a community shift toward smarter, more generalizable models rather than simply deeper ones.

During the review process of Paper 2, the need arose to introduce an additional requirement into our models: to provide them with a degree of interpretability. The results obtained with CNNs so far were not explainable, i.e., it was difficult to understand the decision-making processes that guided the model in classification once the input signal was fed to the network. In Paper 3, we presented a new approach, oriented towards the interpretability of results. We used CNNs as feature extractors and projected the feature maps of the last convolutional layer into a two-dimensional space, with t-SNE algorithm, where it is possible to visualize the distribution of data. Finally, we delegated the classification to a diffusion model, i.e., a Conditional Flow Matching.
While the model proposed in Paper 3 demonstrated strong generalization capabilities, its main limitation lies in the dimensionality reduction step. The use of t-SNE introduced a computational bottleneck, as the algorithm does not learn an explicit mapping between the original and reduced feature spaces. Consequently, projecting new test data either requires re-running t-SNE – risking inconsistent embeddings – or jointly embedding training and test sets, which is not always feasible.
To overcome this issue, we are currently experimenting with alternative methods capable of learning a stable and generalizable mapping, such as a Multi-Layer Perceptron trained to reproduce the t-SNE mapping, and Uniform Manifold Approximation and Projection (UMAP) [60]. These approaches aim to obtain two- or three-dimensional embeddings that optimally distribute the feature vectors according to their membership class. Achieving such embeddings would enable the integration of less complex classifiers, such as Support Vector Machine (SVM). The outcomes and implementation details of these ongoing experiments are provided in Appendix A.1.
Visualizing both the learned feature distributions and the resulting SVM decision surfaces offers a more transparent understanding of the model's classification process, paving the way for future interpretable Deep Learning applications in exoplanet detection.

4.1.1 Open issues and future development

The open issues affecting this field, which undermine the predictive capabilities of the models, are discussed below.

Is this increase in model complexity strictly necessary?

The main motivation behind the development of network architectures inspired by Occam's razor lies in the unjustified growth of architectural complexity observed in recent years. Here, we highlight this issue by comparing the performance achieved by models as their size increases.

From the earliest CNNs such as Astronet to more advanced architectures like Exominer, the overall model size has grown substantially. We compare these network architectures in detail in Table 1 of Paper 3. While Astronet processed only phase-folded light curves, its successor Astronet-K2 incorporated additional stellar and planetary features. The model width was further expanded by the subsequent works Astronet-Triage [94] and Astronet-Triage-v2 [81]. In parallel, ExoMiner was designed to process an even broader range of input features.

Table 4 in Paper 2 quantifies the input dimensionality of the most relevant CNNs preceding our work. A key insight emerges when comparing the Input Size and F1-Score columns across model versions. For example, when classifying Kepler Q1–Q17 DR25 data, ExoMiner achieves an F1-score of 96% using an input size of 1×780, while ExoMiner++ [84] reaches 97% with a substantially larger input of $1 \times 19,560$. Such a marginal 1% improvement in accuracy does not justify the exponential increase in model complexity, computational costs, and training time. This trend mirrors the findings of [80], as illustrated in Figure 3 of their study. Similarly, Astronet-Triage-v2 improves its F1-score from 83% to 85% relative to Astronet-Triage, at the cost of increasing the input size from 1×262 to 1×5248.

These observations demonstrate that the continuous increase in model width and resolution yields diminishing returns in predictive performance. This reinforces the rationale behind our approach: pursuing simpler architectures that preserve high accuracy while enhancing efficiency, interpretability, and generalization on data from multiple domains – in full alignment with the heuristic principle of Occam's razor.

The problem of low signal-to-noise ratio

In this section, we discuss one of the most critical challenges in the field: the low signal-to-noise ratio that affects the photometric data, particularly those corresponding to TCEs originating from rocky planets orbiting very bright stars.

Supervised learning algorithms for TCEs classification have demonstrated good convergence capabilities, particularly in performing the triage operation – that is, distinguishing eclipse-like signals (e.g., planetary transits or eclipsing binaries) from non-eclipse signals caused by stellar variability or instrumental systematics.

These models are trained on datasets composed of pairs $(\vec{x}, y)$, where $\vec{x}$ represents the input feature vector describing the signal, and y is its corresponding label. Typically, $\vec{x}$ includes photometric data of the phase-folded light curve, possibly combined with auxiliary stellar and planetary parameters.

However, both $\vec{x}$ and y are affected by various sources of noise and uncertainty that can significantly impact model training and, consequently, the reliability of its predictions. For instance, photometric observations of a target star may be contaminated by nearby sources falling within the same pixel or pixel group used to extract the light curve. The TESS telescope represents a clear example: with its pixel scale of 21 arcseconds/pixel, the probability of flux contamination from background sources (up to magnitude 18) is high [36]. This phenomenon – known as TESS crowding[1] – is mitigated but not fully corrected by the TESS SPOC pipeline[2]. Details of the correction process are provided in Section 2.3.11 of [79].

The transit depth of a planet is proportional to the square of the planet-to-star radius ratio. Consequently, for small rocky planets orbiting bright stars, the observed flux decrement can be extremely shallow – often dominated by noise or contamination from nearby sources. During training, models exposed to such low signal-to-noise ratio inputs may learn to capture noise rather than meaningful transit features, given that current preprocessing pipelines are often unable to isolate clean representations of faint transit signals. This limitation contributes to the difficulty that Machine Learning models encounter when identifying Earth-sized planets, where the transit signal is dominated by the noise. During inference, planetary signals with low signal-to-noise ratio are more prone to be misclassified by the model, increasing the probability of missing true planet candidates.

Another factor emphasizing on the low signal-to-noise ratio problem is the orbital period. Earth-like planets generally have long orbital periods, which reduces the number of observed transits and the amount of data available for phase-folding – thus limiting the potential signal-to-noise ratio improvement. Detecting such planets often requires multi-year observations, ideally combined with higher-performance instrumentation. However, in most Machine Learning pipelines for exoplanet detection, long-baseline

[1] https://heasarc.gsfc.nasa.gov/docs/tess/UnderstandingCrowdingv2.html

[2] In our experiments, we used TESS SPOC light curves and therefore did not apply additional corrections beyond the preprocessing pipelines developed in each study to prepare model inputs.

light curves are seldom used due to uncertainties in the estimated period and transit duration of the TCEs (Figure 2 of Paper 2 and Section 6.2 of [81] provide details on this issue).

The low signal-to-noise ratio therefore remains a fundamental open problem in the automated classification of TCEs. It limits both overall classification accuracy and, more critically, the detection of terrestrial planets. While upcoming missions such as PLATO are expected to provide significant instrumental improvements, ongoing research focuses on developing methods to detect and characterize a broader variety of astrophysical and instrumental signals within light curves [36, 52].

At present, no end-to-end model exists that integrates all the contributions made across the *Kepler* and TESS light curve analysis pipeline. Future efforts should aim at designing models capable of handling the entire workflow – from noise suppression and light curve correction to modular, interpretable, and neural architectures for exoplanet detection working on data from any transiting survey.

4.2 Conclusions on Cometary observations in support of the ESA/Comet Interceptor space mission

Over the past four years, I have conducted photometric observations of long-period comets using the Copernico Telescope in Asiago (Italy) and the TNG in La Palma (Canary Islands, Spain). In total, observational data were collected for 42 comets. The acquired images were subsequently reduced using a dedicated pipeline developed by IB in the IDL programming language, with the goal of deriving the $Af\rho$ curves for each cometary target.

This data collection and analysis effort culminated in the submission of a scientific paper aimed at characterizing the dust activity and nucleus properties of these comets during their inbound journey toward the Sun (Paper 5). $Af\rho$ values were used to constrain the cometary dust environment, then the characterization was performed using a probabilistic dust tail model based on Markov Chain Monte Carlo method by [30].

This analysis provides a significant contribution to the target selection and risk assessment phases of the ESA/Comet Interceptor mission. More precisely, the results we achieved suggest that the nuclei of the observed comets are predominantly composed of water-poor pebbles, implying that their dust activity is intrinsically weak. Consequently, these comets pose no dust-related hazard to the Comet Interceptor spacecraft, neither in terms of particle impacts nor surface erosion during the flyby.

Chapter 5

Additional research activities

This chapter described the complementary activities that have enriched my doctoral studies, proving fundamental for the development of cross-disciplinary skills in scientific communication, international collaboration, and learning new technologies such as quantum computing. Listed below are the publications in which I appear as co-author, the presentations given at conferences and seminars, and the main scientific dissemination initiatives in which I have participated in the framework of my PhD.

5.1 List of co-authorship manuscripts

Exoplanets:

8. Barclay, T. et al. (2025). The transmission spectrum of the potentially rocky planet L 98-59 c. The Astronomical Journal, 169(5), 241.

7. Magliano, C. et al. (2024). Revisiting the conundrum of the sub-Jovian and Neptune desert. Astronomy & Astrophysics, 692, A162.

6. Magliano, C. et al. (2024). PLATO on the shoulders of TESS: analysing mono-transit planet candidates in TESS data as a prior knowledge for PLATO observations. Monthly Notices of the Royal Astronomical Society, 528(2), 2851-2870.

5. Barclay, T. et al. (2023). The transmission spectrum of the potentially rocky planet L 98-59 c. arXiv preprint arXiv:2301.10866.

4. Magliano, C. et al (2023). A systematic validation of hot Neptunes in TESS data. Monthly Notices of the Royal Astronomical Society, 519(1), 1562-1577.

3. Magliano, C. et al. (2023). The TESS Triple-9 Catalog II: a new set of 999 uniformly vetted exoplanet candidates. Monthly Notices of the Royal Astronomical Society, 521(3), 3749-3764.

2. Damiano, M. et al. (2022). A Transmission Spectrum of the Sub-Earth Planet L98-59 b in 1.1–1.7 μm. The Astronomical Journal, 164(5), 225.

1. Cacciapuoti, L. et al. (2022). TESS discovery of a super-Earth and two sub-Neptunes orbiting the bright, nearby, Sun-like star HD 22946. Astronomy and Astrophysics, vol. 668, 2022.

Comets:

4. Pastorello, A. et al. (2025). A long-lasting eruption heralds SN 2023ldh, a clone of SN 2009ip. Astronomy and Astrophysics, 701, A32.

3. Piccirillo, A. M. et al. (2025). A light dust shield, including aerogel, for DISC instrument on board ESA's Comet Interceptor mission: numerical simulations and Ballistic Limit Equation. CEAS Space Journal, 1-13.

2. Della Corte, V. et al. (2024). Performance assessment of an innovative light and compact dust shield for DISC onboard Comet Interceptor/ESA space probes. International Journal of Impact Engineering, 105146.

1. Della Corte, V. et al. (2023). DISC-the dust impact sensor and counter onboard Comet Interceptor: Characterization of the dust coma of a dynamically new comet. Advances in Space Research, 71(8), 3457-3467.

During two observation runs at the Copernico telescope, I had to temporarily interrupt the observation of cometary targets to conduct Target of Opportunity (ToO) observations following two alerts received. In both cases, the timely acquisition of photometric data provided an essential contribution to the characterization of the transients. Subsequent analyses, with results published on the Transient Name Server, allowed both events to be classified as supernovae. The classification certificates are shown below.

Objects classification

2. Fiscale, S., Ochner, P., Andreoli, V. Classification certificate for object 2024rkw

1. Pastorello, A., Reguitti, A., Fiscale, S. et al. Classification certificate for object 2023ldh

5.2 List of talks and posters

This section lists the talks and posters I presented as first author at national and international conferences to share my research findings.

Talks:

3. PLATO ESP2025, Planets throughout the Habitable Zone, Marseille, France (2025);
2. Exoplanets in Naples, First ExoplaNAts annual meeting, Naples, Italy (2023);
1. 109^o Congresso Nazionale, Società Italiana di Fisica, Salerno, Italy (2023).

Posters:

6. European Astronomical Society, in "Archives, Data Science & AI", virtual, Jul 2025.
5. Exoplanets in Italy: status and perspectives, Rome, Italy, Feb 26-28, 2025.
4. European Astrobiology Network Association, session "Exoplanets", Graz, Austria, Sept 03-06, 2024.
3. European Astronomical Society, session "The PLATO mission: Towards new horizons in Exoplanet and Stellar Science", virtual, Jul 2024.
2. ESO LSST workshop, Jan 22-26, 2024.
1. 4th Advanced School on Exoplanetary Science (ASES 4), Vietri sul Mare, Italy, May 22-26, 2023.

5.3 Collaborations

I have been part of the research group ExoplaNAts, founded in 2020 and made up of students and professors from Parthenope and Federico II universities of Naples. We held weekly meetings to share progress within our collaboration. Active participation in this group enhanced my presentation skills and broadened my multidisciplinary knowledge. Presenting the technical aspects of my Machine Learning models to colleagues with an astrophysics background required significant communicative adaptation, refining my ability to tailor the language to the audience. Furthermore, closely following the astrophysicists' work on exoplanet characterization [55] and validation [15] solidified my understanding of manual data analysis methodologies in this field.

Through this group, I had the opportunity to collaborate with international research groups working on exoplanets:

- PLATO Mission Consortium [(a)] (2025-now);
- NASA's Planet Patrol (2022-2024);

Collaboration with these research groups also represented a crucial learning opportunity, allowing me to contribute to the development of work based on manual vetting and atmospheric characterization. The scientific publications resulting from these collaborations are listed in Section 5.1.

[(a)] As member of the PLATO Mission Consortium, I will provide my contribution to the PLATO mission in the area of Core Science (PSM) - Working Group: Exoplanet Science; Leader: Prof. Don Pollaco (University of Warwick, UK).

5.4 Research stay at Laboratoire d'Astrophysique de Marseille

My doctoral program requires me to spend between three and eighteen months conducting research at a European research institution. I was hosted at the Laboratoire d'Astrophysique de Marseille (LAM), where I carried out my main research project from late May to early October 2025. I worked under the supervision of Professor Magali Deleuil (MD) and collaborated with her postdoc Hugo Vivien (HV).

Following a group discussion with my supervisors, we decided to set as the main objective of this collaboration the development of the model presented in Appendix A.1, and to apply it to the PLATO simulated data generated by Professor Deleuil.

I spent the first week at LAM studying the documentation relating to the PLATO simulated data and writing the code for reading the TCE catalogs and related light curves. I also wrote the code for pre-processing these light curves. Frequent interaction with MD and HV was useful in speeding up the process of learning about the nature of the data.

During the first month I have been involved in writing the Python code for the model. The code was developed following object-oriented programming paradigms, with the aim of ensuring modularity and maintainability. This approach facilitates future integrations and developments with a view to applications to PLATO data. The repository is open-source and accessible at the following link: `https://github.com/stefanofisc/exoplanets`. From the second half of June, I started

to coordinate a group of three peers to support the development and long-term maintenance of this code.

I spent most of July and September conducting experiments aimed at determining the sub-optimal configuration of the network architecture that maximizes the predictive performance of the model on the Kepler Q1-Q17 Data Release 25 benchmark dataset. We chose this benchmark dataset because it is the largest publicly available dataset, containing real TCEs with highly reliable labels.

In parallel with the code development and model selection phases, I held regular meetings with my supervisors to update them on both the general status of the work and the technical details of the model. Also, my time at LAM gave me the opportunity to participate in journal clubs organized by the research group Systèmes Planétaires. Another fundamental aspect of my growth as a researcher was the daily interaction with the astrophysics PhD students in the department.

5.5 Teaching experience

I had the opportunity to co-supervise two bachelor's theses in Computer Science focused on the development of Machine Learning models for the identification of exoplanets. This tutoring activity proved to be an important synergy for my main research project.

- Title: *A Random Forest classifier for vetting planet candidates in TESS data.*
  Student: Attilio Di Vicino.
- Title: *Detection di Esopianeti: Tecniche di Machine Learning a confronto.*
  Student: Luca Di Meglio.

With Di Meglio's thesis, I had the opportunity to delegate certain tests of my code to the student while I continued development. This allowed the student to apply and evaluate the theoretical knowledge learned in the context of Machine Learning, and to speed up the code development phase.

The collaboration with Attilio Di Vicino has allowed me to explore in parallel the development and performance of new classification models, such as Random Forest. This collaboration continues today in the development of codes based on models for exoplanet detection and quantum computing.

5.6 Scientific seminars

The list of lectures I gave at Parthenope University of Naples is provided below.

– Combined lecture on Astrobiology and Artificial Intelligence for the students from "Scienze e Tecnologie Spaziali" and "Astrobiologia" courses (May 2024);

– Lectures (3x) on exoplanets detection for the students from "Scienze e Tecnologie Spaziali" and "Astrobiologia" courses (May 2023, May 2025);

– Lectures (3x) on Quantum Computing for the master students from Physics and Quantum Computing course (Dec 2023, Apr 2025). In all the lessons I taught on quantum computing, I introduced the mathematical and linear algebra fundamentals underlying quantum processing. The theory was integrated with practical application, using IBM's Python Qiskit library with the aim of illustrating the evolution of the state of a qubit on the Bloch sphere following the application of quantum gates. The last seminar (April 2025) included an in-depth look at the current landscape of the sector, with a focus on:

  – **Research institutions involved**. An analysis of the leading companies, universities, and collaborations active in the development of quantum hardware and software globally.

  – **State of the art**. A review of the latest innovations, with examples such as the Majorana fermion-based processor developed by Microsoft.

  – **Economic context**. A discussion on economic investments in the field by the major countries.

  The aim of this in-depth study was to provide students with a contextualized view of the academic and industrial landscape, highlighting future career opportunities and major centers of innovation. I believe that this approach, which focuses on integrating theory, practice, and possible work applications, should be included in every course of study.

  The seminars held on Quantum Computing represent the culmination of a training course that began in 2020 thanks to my supervisor Prof. Alessandra Rotundi. In particular, two experiences were decisive in acquiring fundamental knowledge and curiosity in the field:

  – The seminar "Optics and Quantum Information", Sapienza University of Rome, Jan 2020;

  – The course "Formazione Avanzata di programmazione su Computer Quantistici IBM" held in collaboration between IBM and Fondazione CRUI.

  These activities allowed me to acquire the basic concepts of a constantly evolving

and extremely promising research topic.

5.7 Public Outreach

Communicating the concepts behind my studies to the general public is a subject I care about. In this context, I gave several talks on Exoplanets and Artificial Intelligence, at multiple organisations.

– Invited talk on Exoplanets detection with Artificial Intelligence for CISA (2025);
– Talk on Astrobiology and Artificial Intelligence for Pint of Science (2024);
– Talk on my doctoral project in Interdisciplinary young scholar meetings (2024);
– Talk on Exoplanets detection with Machine Learning - La Vita e lo spazio (2024);
– Scientific explainer at multiple editions of Futuro Remoto (2021-2024).

5.8 Additional Training

Here, I list the activities I carried out beside my curricular studies, following my interests in different subjects related to my research topic.

Nature Masterclasses online courses, Springer Nature:

– Finding Funding Opportunities;
– Persuasive Grant Writing;
– Networking for Researchers;
– Writing a Research Paper;
– Narrative Tools for Researchers;
– Advancing your Scientific Presentations.

Other online/in-person courses and schools:
– Basics of Quantum Information, a course by IBM;
– 4th Advanced School on Exoplanetary Science;
– Scientific Communication in Astronomy (SCIA) 2023;
– Corso pratico - Principi e Applicazioni della Microscopia Elettronica e Microanalisi;
– My Individual Development Plan, Science Careers (course completion);
– Quantum Computing and High Performance Computing 4th edition, *CINECA*.

Through these courses and conferences, I was able to test myself in stimulating and supportive environments that fostered my growth as a young researcher. I must give special mention to my participation in SCIA 2023. During that week, I was able to

learn crucial information and advice from experts in the field in the context of scientific writing, communication, and online self-promotion. In addition, I was tasked with improvising talks on scientific and non-scientific topics, writing and revising proposals, and carrying out group activities with other young researchers.

5.9 Reviewer activities

I have been invited to review articles on three different occasions:

– Invited for reviewing the manuscript "Identifying Exoplanets with Deep Learning: A CNN and RNN Classifier for Kepler DR25 and Candidate Vetting" submitted to Springer Nature (2025);
– Invited as reviewer for the workshop: Trustworthiness and Reliability in Neurosymbolic AI, at the 2025 International Joint Conference on Neural Networks (IJCNN 2025);
– EDAS Conference and Journal Management System (2024).

5.10 Prizes / Awards

– Deep Learning and High-Performance Computing for Exoplanet Detection, Characterization, and Astrobiological Studies, *CINECA* (2024).

As PI of the project, I was responsible for writing the proposal, which was subsequently revised and submitted together with the co-authors. The aim was to request computational resources from CINECA. I used the computational resources granted as part of the development of my doctoral project.

– Travel grant for *European Astrobiology Network Association (EANA) 2024*.

I applied for and won a travel grant, which allowed me to pay the registration fee for EANA 2024, where I had the opportunity to present a poster.

Bibliography

[1] Rachel Akeson and Jessie Christiansen. "TESS Follow-up Observing Program Working Group (TFOP WG): The ExoFOP-TESS Website". In: *American Astronomical Society Meeting Abstracts# 233*. Vol. 233. 2019, pp. 140–09.

[2] Megan Ansdell et al. "Scientific domain knowledge improves exoplanet transit classification with deep learning". In: *The Astrophysical journal letters* 869.1 (2018), p. L7.

[3] David J Armstrong, Don Pollacco, and Alexandre Santerne. "Transit shapes and self organising maps as a tool for ranking planetary candidates: Application to kepler and k2". In: *Monthly Notices of the Royal Astronomical Society* (2016), stw2881.

[4] David J Armstrong et al. "Automatic vetting of planet candidates from ground-based surveys: machine learning with NGTS". In: *Monthly Notices of the Royal Astronomical Society* 478.3 (2018), pp. 4225–4237.

[5] Yoshua Bengio and Olivier Delalleau. "On the expressive power of deep architectures". In: *International conference on algorithmic learning theory*. Springer. 2011, pp. 18–36.

[6] Ivano Bertini et al. "Inbound evolution of twenty long period comets". In: *Astronomy & Astrophysics* (under revision, 2025).

[7] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[8] Jürgen Blum et al. "Evidence for the formation of comet 67P/Churyumov-Gerasimenko through gravitational collapse of a bound clump of pebbles". In: *Monthly Notices of the Royal Astronomical Society* 469.Suppl_2 (2017), S755–S773.

[9] Marko Bohanec and Ivan Bratko. "Trading accuracy for simplicity in decision trees". In: *Machine Learning* 15.3 (1994), pp. 223–250.

[10] William J Borucki. "Kepler: A brief discussion of the mission and exoplanet results". In: *Proceedings of the American Philosophical Society* 161.1 (2017), pp. 38–65.

[11] William J Borucki et al. “Kepler-22b: a 2.4 Earth-radius planet in the habitable zone of a Sun-like star”. In: *The Astrophysical Journal* 745.2 (2012), p. 120.

[12] Leo Breiman. “Random forests”. In: *Machine learning* 45.1 (2001), pp. 5–32.

[13] Don Brownlee et al. “Comet 81P/Wild 2 under a microscope”. In: *science* 314.5806 (2006), pp. 1711–1716.

[14] Harry Buhrman and Ronald De Wolf. “Complexity measures and decision tree complexity: a survey”. In: *Theoretical Computer Science* 288.1 (2002), pp. 21–43.

[15] Luca Cacciapuoti et al. “TESS discovery of a super-Earth and two sub-Neptunes orbiting the bright, nearby, Sun-like star HD 22946”. In: *Astronomy & Astrophysics* 668 (2022), A85.

[16] Joseph H Catanzarite. “Autovetter Planet Candidate Catalog for Q1-Q17 Data Release 24”. In: *Astronomy & Astrophysics* (2015).

[17] David Charbonneau et al. “An Upper Limit on the Reflected Light from the Planet Orbiting the Star t-Bootis”. In: *The Astrophysical Journal* 522.2 (Aug. 1999), p. L145. DOI: 10.1086/312234. URL: https://doi.org/10.1086/312234.

[18] Dan C Ciresan et al. “Flexible, high performance convolutional neural networks for image classification”. In: *IJCAI proceedings-international joint conference on artificial intelligence*. Vol. 22. 1. Barcelona, Spain: 2011, p. 1237.

[19] Jeffrey L Coughlin et al. “Planetary candidates observed by Kepler. VII. The first fully uniform catalog based on the entire 48-month data set (Q1–Q17 DR24)”. In: *The Astrophysical Journal Supplement Series* 224.1 (2016), p. 12.

[20] Anne Dattilo et al. “Identifying exoplanets with deep learning. ii. two new super-earths uncovered by a neural network in k2 data”. In: *The Astronomical Journal* 157.5 (2019), p. 169.

[21] Martín Dévora-Pajares et al. “WATSON-Net: Vetting, Validation, and Analysis of Transits from Space Observations with Neural Networks”. In: *arXiv preprint arXiv:2511.08768* (2025).

[22] G Filacchione et al. “Comet nuclei composition and evolution”. In: *Comets III* (2024), p. 315.

[23] Stefano Fiscale et al. “DART-Vetter: A Deep LeARning Tool for automatic triage of exoplanet candidates”. In: *The Astronomical Journal* 170.2 (2025), p. 73.

[24] Stefano Fiscale et al. “Detection of Exoplanets in Transit Light Curves with Conditional Flow Matching and XGBoost”. In: *Electronics* 14.9 (2025), p. 1738.

[25] Stefano Fiscale et al. “Exploiting kepler’s heritage: a transfer learning approach for identifying exoplanets’ transits in TESS data”. In: *Research Notes of the AAS* 5.4 (2021), p. 91.

[26] Stefano Fiscale et al. “Identifying Exoplanets in TESS Data by Deep Learning”. In: *Applications of Artificial Intelligence and Neural Systems to Data Science*. Springer, 2023, pp. 127–135.

[27] S Fornasier et al. “Volatile exposures on the 67P/Churyumov-Gerasimenko nucleus”. In: *Astronomy & Astrophysics* 672 (2023), A136.

[28] Keinosuke Fukunaga. “Statistical pattern recognition”. In: *Handbook of pattern recognition and computer vision*. World Scientific, 1993, pp. 33–60.

[29] Marco Fulle and J Blum. *Fractal dust constrains the collisional history of comets*. 2017.

[30] Marco Fulle et al. “Comets beyond 4 au: How pristine are Oort nuclei?” In: *Monthly Notices of the Royal Astronomical Society* 513.4 (2022), pp. 5377–5386.

[31] Karl-Heinz Glassmeier et al. “The Rosetta mission: flying towards the origin of the solar system”. In: *Space Science Reviews* 128.1 (2007), pp. 1–21.

[32] Natalia M Guerrero et al. “The TESS objects of interest catalog from the TESS prime mission”. In: *The Astrophysical Journal Supplement Series* 254.2 (2021), p. 39.

[33] Isabelle Guyon and André Elisseeff. “An introduction to variable and feature selection”. In: *Journal of machine learning research* 3.Mar (2003), pp. 1157–1182.

[34] Andreas Hadjigeorghiou et al. “RAVEN: RAnking and Validation of ExoplaNets”. In: *arXiv preprint arXiv:2509.17645* (2025).

[35] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[36] Michael E Higgins and Keaton J Bell. “Localizing sources of variability in crowded TESS photometry”. In: *The Astronomical Journal* 165.4 (2023), p. 141.

[37] Michael Hippke and René Heller. “Optimized transit detection algorithm to search for periodic transits of small planets”. In: *Astronomy & Astrophysics* 623 (2019), A39.

[38] Alston S Householder. “A theory of steady-state activity in nerve-fiber networks: I. Definitions and preliminary lemmas”. In: *The bulletin of mathematical biophysics* 3.2 (1941), pp. 63–69.

[39] Steve B Howell et al. “The K2 mission: characterization and early results”. In: *Publications of the Astronomical Society of the Pacific* 126.938 (2014), p. 398.

[40] Xia Hu et al. “Model complexity of deep learning: A survey”. In: *Knowledge and Information Systems* 63.10 (2021), pp. 2585–2619.

[41] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *International conference on machine learning*. pmlr. 2015, pp. 448–456.

[42] Jon M Jenkins et al. “The TESS science processing operations center”. In: *Software and Cyberinfrastructure for Astronomy IV*. Vol. 9913. SPIE. 2016, pp. 1232–1251.

[43] Alexia Jolicoeur-Martineau, Kilian Fatras, and Tal Kachman. “Generating and imputing tabular data via diffusion and flow-based gradient-boosted trees”. In: *International conference on artificial intelligence and statistics*. PMLR. 2024, pp. 1288–1296.

[44] Geraint H Jones et al. “The comet interceptor mission”. In: *Space science reviews* 220.1 (2024), p. 9.

[45] Diederik P Kingma. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[46] David G Koch et al. “Kepler mission design, realized photometric performance, and early science”. In: *The Astrophysical Journal Letters* 713.2 (2010), p. L79.

[47] Teuvo Kohonen. “Self-organized formation of topologically correct feature maps”. In: *Biological cybernetics* 43.1 (1982), pp. 59–69.

[48] Geza Kovács, Shay Zucker, and Tsevi Mazeh. “A box-fitting algorithm in the search for periodic transits”. In: *Astronomy & Astrophysics* 391.1 (2002), pp. 369–377.

[49] Arlo U Landolt. “UBVRI photometric standard stars around the celestial equator: updates and additions”. In: *The Astronomical Journal* 137.5 (2009), p. 4186.

[50] Yann LeCun et al. “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* 1.4 (1989), pp. 541–551.

[51] Alexander C Li et al. “Your diffusion model is secretly a zero-shot classifier”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2206–2217.

[52] Min Li et al. “Stellar flare detection methods in TESS data: application and performance study”. In: *Astronomical Techniques and Instruments* (2025).

[53] Tengyuan Liang et al. “Fisher-rao metric, geometry, and complexity of neural networks”. In: *The 22nd international conference on artificial intelligence and statistics*. PMLR. 2019, pp. 888–896.

[54] Laurens van der Maaten and Geoffrey Hinton. “Visualizing data using t-SNE”. In: *Journal of machine learning research* 9.Nov (2008), pp. 2579–2605.

[55] Christian Magliano et al. “The TESS Triple-9 Catalog II: a new set of 999 uniformly vetted exoplanet candidates”. In: *Monthly Notices of the Royal Astronomical Society* 521.3 (2023), pp. 3749–3764.

[56] Shude Mao and Bohdan Paczynski. “Gravitational microlensing by double stars and planetary systems”. In: *Astrophysical Journal, Part 2-Letters (ISSN 0004-637X), vol. 374, June 20, 1991, p. L37-L40.* 374 (1991), pp. L37–L40.

[57] Christian Marois et al. “Direct imaging of multiple planets orbiting the star HR 8799”. In: *science* 322.5906 (2008), pp. 1348–1352.

[58] Michel Mayor and Didier Queloz. “A Jupiter-mass companion to a solar-type star”. In: *nature* 378.6555 (1995), pp. 355–359.

[59] Sean D McCauliff et al. “Automatic classification of Kepler planetary transit candidates”. In: *The Astrophysical Journal* 806.1 (2015), p. 6.

[60] Leland McInnes, John Healy, and James Melville. “Umap: Uniform manifold approximation and projection for dimension reduction”. In: *arXiv preprint arXiv:1802.03426* (2018).

[61] Karen J Meech et al. “A brief visit from a red and extremely elongated interstellar asteroid”. In: *Nature* 552.7685 (2017), pp. 378–381.

[62] Alessandro Morbidelli. “Origin and dynamical evolution of comets and their reservoirs”. In: *arXiv preprint astro-ph/0512256* (2005).

[63] Frederick Mosteller, John W Tukey, et al. “Data analysis, including statistics”. In: *Handbook of social psychology* 2 (1968), pp. 80–203.

[64] Andrew Ng and Michael Jordan. “On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes”. In: *Advances in neural information processing systems* 14 (2001).

[65] Cyrielle Opitom et al. “2I/Borisov: A C2-depleted interstellar comet”. In: *Astronomy & Astrophysics* 631 (2019), p. L8.

[66] Hugh P Osborn et al. “Rapid classification of TESS planet candidates with convolutional neural networks”. In: *Astronomy & Astrophysics* 633 (2020), A53.

[67] Valentina Tardugno Poleo, Nora Eisner, and David W Hogg. “NotPlaNET: Removing False Positives from Planet Hunters TESS with Machine Learning”. In: *The Astronomical Journal* 168.3 (2024), p. 100.

[68] Maithra Raghu et al. “On the expressive power of deep neural networks”. In: *international conference on machine learning*. PMLR. 2017, pp. 2847–2854.

[69] Heike Rauer et al. “The PLATO mission”. In: *Experimental Astronomy* 59.3 (2025), p. 26.

[70] Rüdeger Reinhard and Bruce Battrick. *The Giotto mission: ITS scientific investigations*. Tech. rep. Eur. Space Agency, 1986.

[71] George R Ricker et al. “Transiting exoplanet survey satellite”. In: *Journal of Astronomical Telescopes, Instruments, and Systems* 1.1 (2015), pp. 014003–014003.

[72] Helem Salinas et al. “Distinguishing a planetary transit from false positives: a Transformer-based classification for planetary transit signals”. In: *Monthly Notices of the Royal Astronomical Society* 522.3 (2023), pp. 3201–3216.

[73] Helem Salinas et al. “Exoplanet transit candidate identification in TESS full-frame images via a transformer-based algorithm”. In: *Monthly Notices of the Royal Astronomical Society* 538.3 (2025), pp. 2031–2049.

[74] Darryl Z Seligman et al. “Discovery and preliminary characterization of a third interstellar object: 3I/ATLAS”. In: *The Astrophysical Journal Letters* 989.2 (2025), p. L36.

[75] Christopher J Shallue and Andrew Vanderburg. “Identifying exoplanets with deep learning: A five-planet resonant chain around kepler-80 and an eighth planet around kepler-90”. In: *The Astronomical Journal* 155.2 (2018), p. 94.

[76] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[77] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[78] Mervyn Stone. “Cross-validatory choice and assessment of statistical predictions”. In: *Journal of the royal statistical society: Series B (Methodological)* 36.2 (1974), pp. 111–133.

[79] Martin C Stumpe et al. “Kepler presearch data conditioning I—Architecture and algorithms for error correction in Kepler light curves”. In: *Publications of the Astronomical Society of the Pacific* 124.919 (2012), p. 985.

[80] Mingxing Tan and Quoc Le. “Efficientnet: Rethinking model scaling for convolutional neural networks”. In: *International conference on machine learning*. PMLR. 2019, pp. 6105–6114.

[81] Evan Tey et al. “Identifying exoplanets with deep learning. v. improved light-curve classification for Tess full-frame image observations”. In: *The Astronomical Journal* 165.3 (2023), p. 95.

[82] Susan E Thompson et al. “Planetary candidates observed by Kepler. VIII. A fully automated catalog with measured completeness and reliability based on data release 25”. In: *The Astrophysical Journal Supplement Series* 235.2 (2018), p. 38.

[83] Hamed Valizadegan et al. “ExoMiner: A highly accurate and explainable deep learning classifier that validates 301 new exoplanets”. In: *The Astrophysical Journal* 926.2 (2022), p. 120.

[84] Hamed Valizadegan et al. “ExoMiner++ on TESS with Transfer Learning from Kepler: Transit Classification and Vetting Catalog for 2-min Data”. In: *arXiv e-prints* (2025), arXiv–2502.

[85] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[86] Dan Ventura and Sean Warnick. “A theoretical foundation for inductive transfer”. In: *Brigham Young University, College of Physical and Mathematical Sciences* 19 (2007).

[87] Koko Visser, Bas Bosma, and Eric Postma. “Exoplanet detection with Genesis”. In: *Journal of Astronomical Instrumentation* 11.03 (2022), p. 2250011.

[88] Koko Visser, Bas Bosma, and Eric Postma. “Size does matter: Exoplanet detection with a sparse convolutional neural network”. In: *Astronomy and Computing* 41 (2022), p. 100654.

[89] Long Wen et al. “A new transfer learning based on VGG-19 network for fault diagnosis”. In: *2019 IEEE 23rd international conference on computer supported cooperative work in design (CSCWD)*. IEEE. 2019, pp. 205–209.

[90] Peter J Wheatley et al. “The next generation transit survey (NGTS)”. In: *Monthly Notices of the Royal Astronomical Society* 475.4 (2018), pp. 4476–4493.

[91] Alex Wolszczan. “Discovery of pulsar planets”. In: *New Astronomy Reviews* 56.1 (2012), pp. 2–8.

[92] Sanghyun Woo et al. “Convnext v2: Co-designing and scaling convnets with masked autoencoders”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 16133–16142.

[93] Andrew Chi-Chih Yao. “Decision tree complexity and Betti numbers”. In: *Proceedings of the twenty-sixth annual ACM symposium on Theory of Computing*. 1994, pp. 615–624.

[94] Liang Yu et al. “Identifying exoplanets with deep learning. III. Automated triage and vetting of TESS candidates”. In: *The Astronomical Journal* 158.1 (2019), p. 25.

[95] T Zingales et al. “A joint effort to discover and characterize two resonant mini-Neptunes around TOI-1803 with TESS, HARPS-N, and CHEOPS”. In: *Astronomy & Astrophysics* 695 (2025), A273.

Appendix A

Exoplanet detection with Deep Learning

A.1 Detection of Exoplanets in Transit Light Curves (ongoing work)

Introduction. The model presented in Paper 3 demonstrated excellent generalization. However, the dimensionality reduction module exhibits a processing bottleneck. The t-SNE algorithm does not learn any explicit mapping function, which compels us either to run a new projection for the test data or to jointly project both training and test samples before separating them again in the lower-dimensional space. In the former case, there is a risk that t-SNE may project the test data into subspace regions different from those occupied by the training samples. As a result, the classifier performance on the test set degrades, since it is exposed to data distributed in regions that were not encountered during training. In the latter case, the projection requires access to the original training set, which is not always feasible.

The primary goal of this work, here after Paper 4, is to replace t-SNE with a method capable of learning an explicit mapping from the high-dimensional feature space – where the CNN-derived feature vectors reside – to the embedding space where classification occurs. Following the principle of minimizing model complexity, we also implement changes to the feature extraction and classification modules. Specifically, we experiment with ResNet-34 and ResNet-18 architectures [35] as alternatives to VGG-19, and replace the Conditional Flow Matching approach with simpler classifiers such as Support Vector Machine, Linear Discriminant Analysis, and Quadratic Discriminant Analysis, enabling a direct visualization of the resulting decision boundaries.

We selected the Kepler Q1–Q17 Data Release 25 as our benchmark dataset, as it represents the largest and most reliable publicly available collection of TCEs with high-quality labels. Details about this dataset are provided in Table 2 of Paper 3.

All experiments were performed on a MacBook Air M1 (8-core CPU, integrated GPU, 8 GB RAM). Despite the very limited hardware resources, the models were trained and tested within reasonable time, highlighting the computational efficiency of the proposed pipeline.

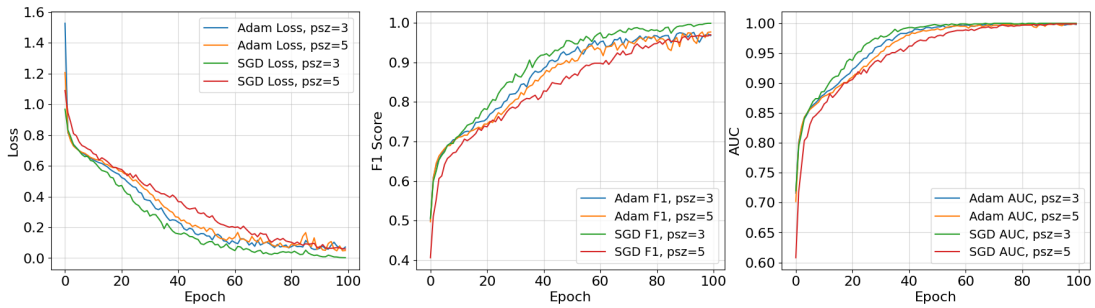

Figure A.1: Training curves of the loss function, F1-score, and AUC obtained by training VGG-19 for 100 epochs with pooling size (psz) set to 3 and 5.

The methodological innovations introduced in this study represent an additional step toward the development of modular and interpretable neural architectures for exoplanet candidates detection.

The rest of this section presents the experiments we made on each module.

Module: Feature Extraction

Experiment 1: Improving classes separation by increasing the pooling size of VGG-19. Since our goal is to reduce the overall model complexity while preserving predictive performance, we evaluate the feature extraction capability of VGG-19 under different pooling configurations. Specifically, we compare the class distributions in the two-dimensional embedding obtained by setting the pooling size to 3 and 5.

To ensure the robustness of the analysis, we report results over a grid of training epochs of $\{40, 60, 100\}$ and consider two different optimization algorithms: Adam [45] and Stochastic Gradient Descent (SGD). The choice of such a specific grid of epochs was guided by the inspection of the training dynamics; which are also very similar between the two optimization algorithms, as depicted by the evolution of the loss, AUC, and F1-score of Figure A.1. The corresponding two-dimensional distributions of the training set classes are displayed in Figure A.2 (Adam) and Figure A.3 (SGD).

Inspection of Figure A.2 shows that the class separation achieved after 60 and 100 training epochs is slightly improved when the pooling size is set to 5. This finding is consistent with the aim of this project in demonstrating that better performance can be achieved with progressive architectural simplifications.

On the other hand, Figure A.3 shows that VGG-19 struggles in extracting relevant patterns for maximizing separation between classes when trained with SGD. This could be the reason explaining the widespread use of the Adam optimizer for VGG-like architectures such as Astronet and subsequent versions [94, 81].

The second and third columns of Table A.1 summarize the architectural differences of VGG-19 under the two pooling configurations. In addition to the modest improvement in class separation in the two-dimensional embedding, increasing the pooling size provides the following advantages:

– **Feature extraction block.** The number of parameters remains unchanged, but the number of multiply–accumulate operations (MACs) decreases by approximately 12 million (M), resulting in a lower computational cost.

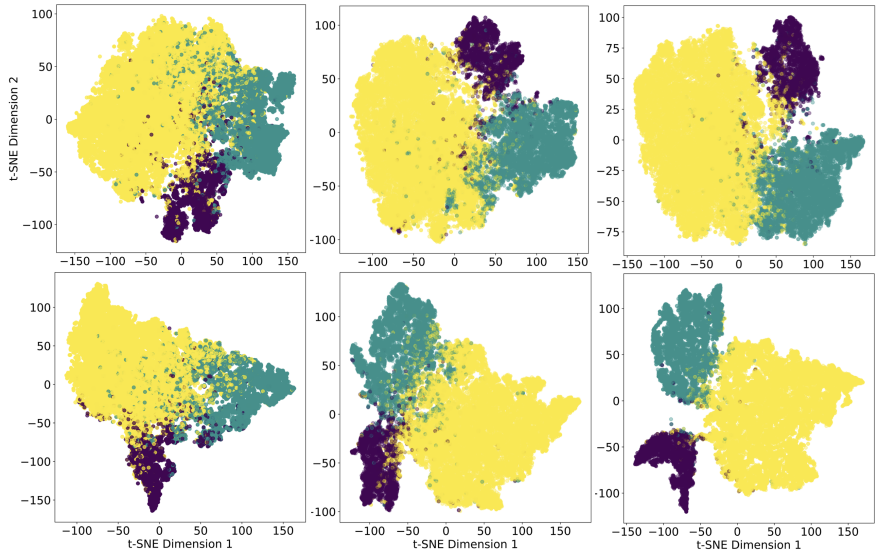

Figure A.2: Two-dimensional distributions of the feature vectors obtained by training VGG-19 with Adam, as a function of pooling size and training epochs. The top row shows results with pooling size 3 after 40, 60, and 100 epochs (from left to right). The bottom row shows the corresponding distributions for pooling size 5. Purple points indicate samples belonging to the EB class, green points represent samples from the PC class, and yellow points correspond to the J class.

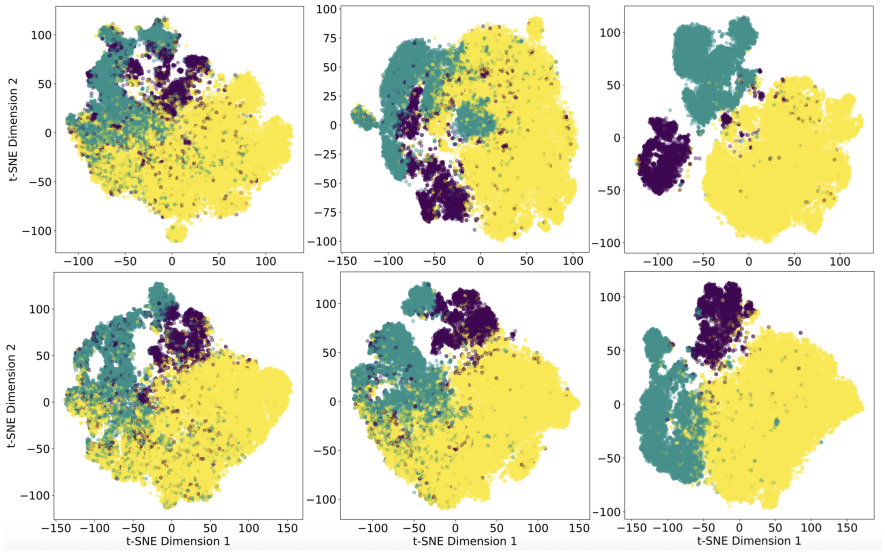

Figure A.3: Two-dimensional distributions of the feature vectors obtained by training VGG-19 with SGD, as a function of pooling size and training epochs. The top row shows results with pooling size 3 after 40, 60, and 100 epochs (from left to right). The bottom row shows the corresponding distributions for pooling size 5. Purple points indicate samples belonging to the EB class, green points represent samples from the PC class, and yellow points correspond to the J class.

Component	**VGG-19 (psz=3)**	**VGG-19 (psz=5)**	**ResNet-34**	**ResNet-18**
Fully-Connected input [neurons]	2,560	1,536	**512**	**512**
Number of parameters [M]				
Feature Extraction	6.28	6.28	7.22	**3.84**
Fully-Connected	6.56	2.37	**0.01**	**0.01**
Total	12.85	8.65	7.23	**3.85**
Multiply–Accumulate operations [M]				
Feature Extraction	125.14	112.97	75.23	**37.14**
Fully-Connected	6.56	2.37	**0.01**	**0.01**
Total	131.91	115.53	75.24	**37.15**
Training time [s/epoch]	∼33	∼30	∼29	∼**20**

Table A.1: Comparison of VGG-19 (with two different pooling sizes (psz) of 5 and 3), Resnet-34 and Resnet-18 architectures. The best values are highlighted in boldface.

- **Fully connected block.** The number of parameters decreases by about 4.2M, together with the corresponding MACs. This reduction follows from the smaller spatial dimension of the feature maps when a larger pooling size is applied. Consequently, the single fully connected layer we use only requires 1536 input neurons, compared to 2560 when pooling size is set to 3.

Overall, increasing the pooling size from 3 to 5 reduces the total number of VGG-19 parameters from 12.85M to 8.65M, and the MACs from 131.91M to 115.53M. Training time per epoch decreases, from ∼33 s/epoch to ∼30 s/epoch. Also, the training curves do not show any substantial degradation, as shown in Figure A.1. These results demonstrate that the reduction in model complexity is achieved without compromising – and in fact slightly improving – the separation between classes.

Module: Feature Extraction
Experiment 2: Further minimizing model complexity and improving classes separation with Resnet. As the next step, we evaluated ResNet as alternative feature extractor. Compared to VGG-19, ResNet is less complex in terms of both the number of trainable parameters and the number of MACs. This reduction in complexity provides several advantages: a lower risk of overfitting, as well as shorter training times.

Following the same experimental protocol used for VGG-19, we trained two different architectures, ResNet-34 and ResNet-18, and compared their performance as feature extractors. The evaluation considered different optimizers (Adam and SGD) and a grid of training epochs identical to that used for VGG-19, that is {40, 60, 80, 100}.

The results revealed that the two configurations providing the best classes separation on the training set were: ResNet-34 trained with Adam for 100 epochs, and ResNet-18 trained with Adam for 80 epochs.

We then compared how the two-dimensional distributions of the three classes varied across the four

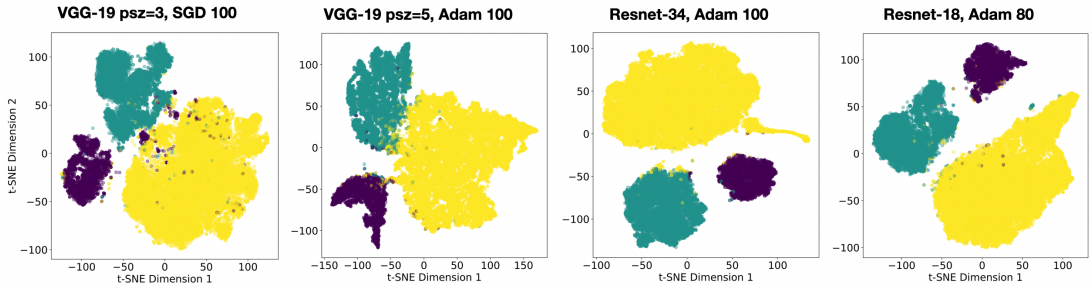

Figure A.4: Two-dimensional distributions of the training set feature vectors obtained with our best four network configurations. VGG-19 with pooling sizes 3 and 5 (first two panels), ResNet-34 (third panel), and ResNet-18 (fourth panel). These distributions were produced using the t-Stochastic Neighbor Embedding algorithm. Purple points indicate samples belonging to the EB class, green points represent samples from the PC class, and yellow points correspond to the J class.

best-performing models identified so far: VGG-19 with pooling size 3, VGG-19 with pooling size 5, ResNet-34, and ResNet-18.

As shown by the results in Figure A.4, the best separation between the three classes is achieved with ResNet-18, which is also the least complex model, as reported in Table A.1. We therefore define ResNet-18, trained for 80 epochs with Adam, as the reference model in the feature extraction block. This configuration will serve as the baseline for the experiments conducted on the dimensionality reduction block.

Module: Dimensionality Reduction

Experiment 3: Learning a dynamical mapping from high- to low-dimensional spaces. The t-SNE algorithm allowed us to observe how the features extracted from CNNs distribute in two-dimensional spaces, highlighting that ResNet-18 is the network maximizing classes separation.

However, t-SNE cannot be used directly in the inference phase as mentioned earlier in this section. The algorithm does not provide an explicit mapping from high-dimensional data to those in reduced space, and each new dataset requires a new fitting, with high computational costs. Furthermore, test data projected with t-SNE does not necessarily fall into the same regions occupied by the training data, unless additional constraints are placed on the reduction, with the risk of compromising classification. In our previous work, we circumvented this problem by projecting training and test sets simultaneously into the two-dimensional space, and deriving the representations of the test data from there.
But this approach is computationally expensive and unrealistic in an inference context, where only test data are available.

To overcome this limitation, we trained a Multi-Layer Perceptron (MLP) to learn the t-SNE two-dimensional mapping. The MLP architecture consists of 512 input neurons and three hidden layers of 128, 32, and 8 neurons, with a progressive reduction of units by powers of four. The output consists of 2 neurons, corresponding to the coordinates in the two-dimensional space. The MLP was trained with Adam (learning rate 10^{-3}, batch size 128) and Mean Squared Error (MSE) loss function, which compares the t-SNE coordinates with those estimated by the MLP. To stabilize convergence, we normalized the data produced by t-SNE before training the MLP.

As shown in the first two panels of Figure A.5, the MLP faithfully reconstructs the t-SNE distribution

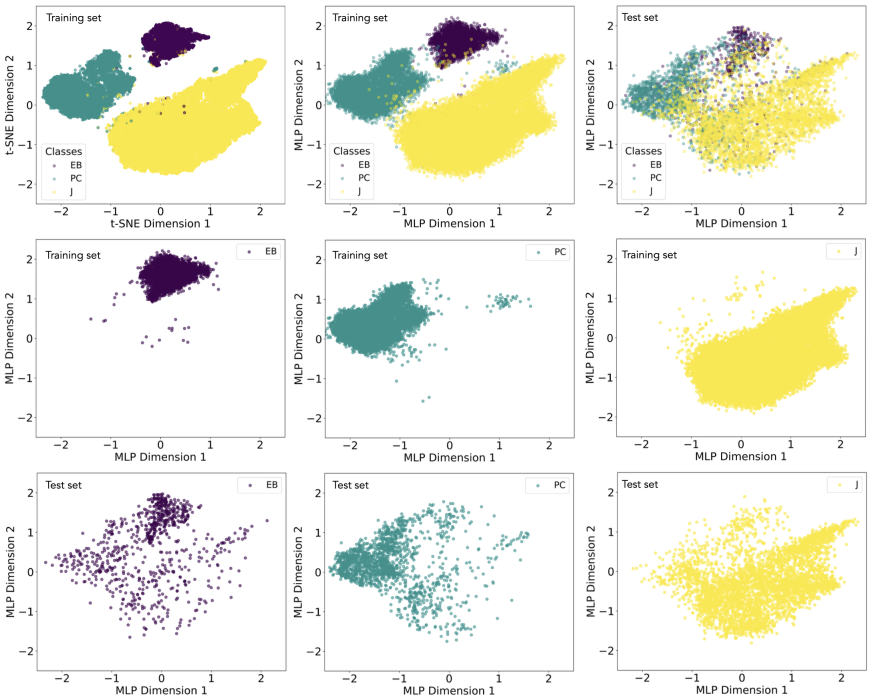

Figure A.5: Two-dimensional distributions of training and test set features. (Top row) The first panel shows the normalized data projected with t-SNE, while the second shows the relative mapping learned by the MLP on the training set. The last panel shows the test set features projected with the MLP. (Middle row) The three panels show the mapping learned by the MLP for each class on the training set. (Bottom row) Distribution of test data when projected with the MLP.

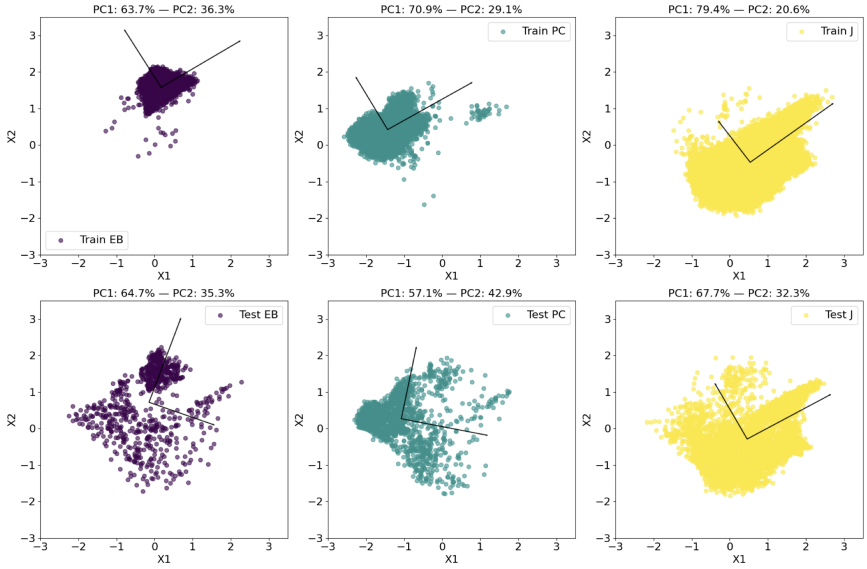

Figure A.6: Two-dimensional distributions of training and test set features produced by the MLP, with the first two principal components provided for each class.

Class	Precision	Recall	F1	TP	TN	FP	FN	Misclass. Rate
EB	0.581	0.457	0.511	358	5533	258	426	0.104
PC	0.722	0.729	0.725	1197	4472	460	446	0.138
J	0.863	0.895	0.878	3711	1836	591	437	0.156

Table A.2: Classification metrics obtained with a Support Vector Machine, radial basis function kernel, applied on the two-dimensional test data.

on the training set. In the test set, however, the distributions are more sparse and the three classes partially overlap (last panel of the top row). Visually examining these classes separately (last row of the figure), we observe that the minority classes (EB and PC) tend to assume a pattern similar to that of the majority class (J), which numerically dominates the training set.

To quantitatively describe this phenomenon, we estimated the Principal Component Analysis (PCA) on the training and test distributions of each class. Figure A.6 shows the first two principal components (PC1 and PC2) and their variance. In the training set, PC1 is marked for all classes (minimum 63.7% for EB), while we observe that as the cardinality of the classes decrease, there is an increase in variance for PC2. In the test set, class J maintains component directions very similar to those in training, a sign that the model has consistently learned its relevant patterns. The EB and PC classes, on the other hand, show a rotation of the principal components in the direction of PC1 of class J, suggesting potential overfitting of the model on the majority class.

Module: Classification

Experiment 4: Classification with Support Vector Machine. We then quantified the predictive performance of the entire pipeline after the first model selection cycle. Starting from the training set distributions shown in Figure A.5, second panel in the first row, we trained a Support Vector Machine with a radial basis function kernel. This non-linear kernel allows the classifier to define complex

separation surfaces, adapting better to the data structure than a linear model.

Next, we used the trained model to classify the test set data (Figure A.5, third panel at the top). From the results shown in Table A.2, it can be seen that the majority class (J) achieves higher performance in terms of precision, recall, and F1-score than the two minority classes (EB and PC). This latter result is consistent with the observations we made on distributions: the stability of the representation of class J in the transition from training to test set facilitates its correct classification, while classes EB and PC suffer more from generalization errors.

Module: Dimensionality Reduction

Experiment 5: Is the MLP overfitting on the majority class? The predictive performance of the model reported in Table A.2 are not optimal. To assess whether the problem is due to overfitting of the MLP, we trained this network on a grid of 5, 15, and 30 epochs. In all cases, the results on the test set projections remained qualitatively similar, indicating that overfitting related to the number of epochs does not seem to be the main issue. We also reduced the architectural complexity by switching from three to a single hidden layer of 512 neurons, without finding any benefits on generalization between majority and minority classes.

Another possible explanation for the partial overlap among classes (as shown in the top right panel of Figure A.5) concerns the nature of t-SNE. By design, the algorithm emphasizes local relationships rather than global ones, sometimes producing representations that are less faithful to the overall structures. The MLP, learning from these representations, could therefore inherit some of these distortions.

Summarizing the findings after the first cycle of model selection. To sum up, the experiments conducted highlighted the strengths and limitations of the different approaches. t-SNE proved to be a useful tool for preliminary exploration, allowing the identification of the convolutional network most effective in separating classes. However, its non-parametric nature makes it unsuitable for inference applications, as it is unable to learn a stable mapping from the data.

The MLP, trained to mimic the mapping of t-SNE, showed the ability to faithfully reproduce the training set distributions. In the test set, however, the minority classes (EB and PC) showed a tendency to follow the pattern of the majority class (J), suggesting a form of bias induced by the imbalance of the dataset and the type of representation learned.

To address the issues related to class imbalance and the limitations of t-SNE, we applied undersampling to class J and employed Uniform Manifold Approximation and Projection (UMAP) [60] as a replacement, respectively. We tested these changes sequentially. In the first test, we evaluated the same pipeline on the undersampled dataset, while in the second test we replaced t-SNE with UMAP.

Experiment 6. Test 1. Running the pipeline on the undersampled dataset. We generated a dataset with an equal number of samples for each class by undersampling class J and removing half of the PC samples obtained with horizontal reflection of global views [1]. The resulting dataset of global views has 4108 PCs, 3917 EBs, and 4000 Js.

We extracted features from global views using Resnet-18 trained with Adam for 40 epochs. The learning rate is 10^{-3} and the batch size is 128.

[1] In Paper 2 and Paper 3 we doubled the number of samples of the PC class to deal with class imbalance. Details on this data augmentation strategy are provided in Paper 2 (Section 4) and Paper 3 (Section 3.1.3)

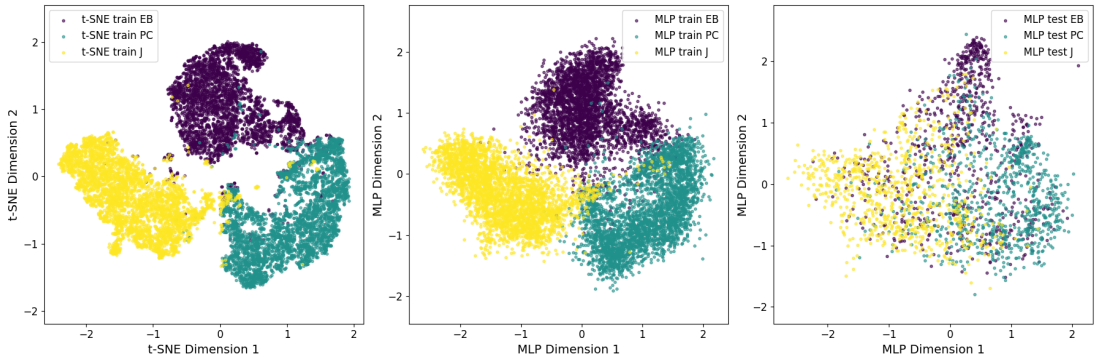

Figure A.7: Two-dimensional distributions of the training set feature vectors within the embeddings generated by t-SNE (left panel) and by the MLP (middle panel). The right panel shows the feature vectors of the test set produced by the MLP.

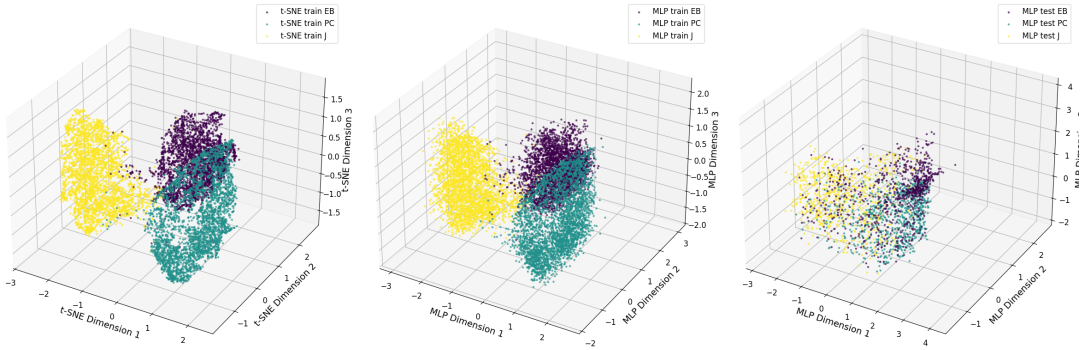

Figure A.8: Three-dimensional distributions of the training set feature vectors within the embeddings generated by t-SNE (left panel) and by the MLP (middle panel). The right panel shows the feature vectors of the test set produced by the MLP

The feature vectors were projected in both two and three dimensions using t-SNE, and two MLPs were trained to learn the mappings, respectively. In addition to the original MLP architecture (with three hidden layers of 128, 32, and 8 neurons), we also tested an architecture with a single hidden layer of 512 neurons, obtaining similar results. The two- and three-dimensional MLP projections on the training and test sets are shown in Figure A.7 and Figure A.8.

Both figures show that the MLP faithfully learns the mapping on the training set but fails to separate the test data in the same way. The distributions of the classes in the test set remain highly overlapping in both the embeddings.

We then questioned the quality of the features produced by Resnet-18 on the test set. In fact, these features may not be optimal for projection into subspaces since they are produced by a model trained for classification purposes. This is because the Resnet-18 architecture we use has a fully-connected final layer for classification, and the model parameters are optimized after feed-forward processing of the feature vectors through this layer. We used Resnet-18 in inference mode to quantify its classification accuracy on the test set. Table A.3 shows the best performance on the test set are obtained by training the model for 60 epochs, but an F1-score of 69% on the PC class is not an optimal result.

Epochs	Class	Precision	Recall	F1	TP	TN	FP	FN	Misclass. Rate
40	EB	0.5406	0.6373	0.5850	499	1198	424	284	0.2944
	PC	0.6688	0.6338	0.6508	521	1325	258	301	0.2324
	J	0.7283	0.6400	0.6813	512	1414	191	288	0.1992
50	EB	0.6065	0.5926	0.5995	464	1321	301	319	0.2578
	PC	0.6592	0.6776	0.6683	557	1295	288	265	0.2299
	J	0.6893	0.6850	0.6871	548	1358	247	252	0.2075
60	EB	0.6352	0.6271	**0.6311**	491	1340	282	292	**0.2387**
	PC	0.6974	0.6983	**0.6979**	574	1334	249	248	**0.2067**
	J	0.7046	0.7125	**0.7085**	570	1366	239	230	**0.1950**

Table A.3: Resnet-18 in inference mode. Evaluation metrics on test set by varying training epochs on the grid {40, 50, 60}.

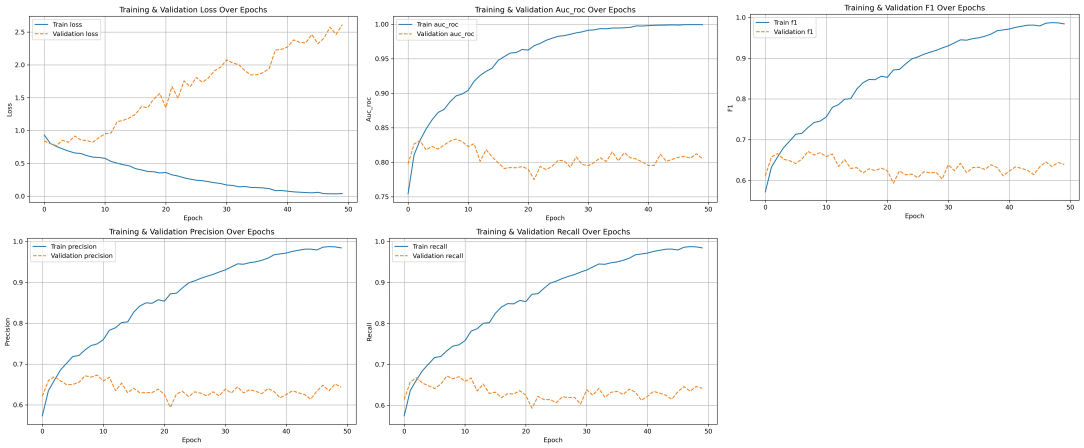

Figure A.9: Training and validation metrics when training Resnet-18 on the balanced dataset.

We introduced a validation set into the training and found that performance does not improve on unseen data, while the metrics on the training data grow as expected (Figure A.9). The same results were obtained by varying the validation split in terms of size (10% and 20% drawn from the original dataset) and samples (through random shuffles performed a-priori on the original dataset). Even replacing Adam with SGD did not lead to improvements.

These findings suggest that Resnet-18, although it learns discriminative patterns on the training data, is unable to produce optimal test set feature vectors for either classification or subspaces projection.

Experiment 6. Test 2. Parametric UMAP. We decided to consider the UMAP algorithm as an alternative to t-SNE. This algorithm is able to construct a global graph of relationships between data points, resulting in a more faithful reconstruction of the intrinsic structure of the actual data distributions. Since UMAP can also learn an explicit mapping from the data, there is no need to train an MLP as in the case of t-SNE.

Based on the results obtained in the previous experiment on the undersampled dataset, we also decided to make a change to the Resnet-18 architecture. Our goal is to prevent the Resnet-18 parameters from being optimized according to the loss function computed after processing the data through the final fully-connected layer. For this reason, we keep the feature extraction block of Resnet-18 and cut the last

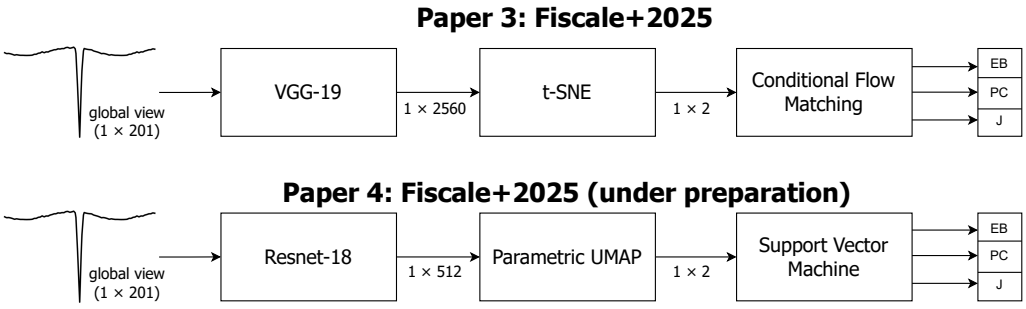

Figure A.10: Comparison of the network architectures presented in Paper 3 [24] and Paper 4 (Fiscale et al. 2025, under preparation).

layer responsible for classification. We use the resulting architecture as an encoder for the Parametric UMAP algorithm. This algorithm differs from its original version by learning the relationships between the input data and the embedding using an encoder, rather than learning the embedding directly. The use of Resnet-18 as encoder allows to train the model to minimize UMAP loss instead of classification loss.

We are currently fine-tuning both the architecture and parameters of Parametric UMAP, then no partial outcomes are available yet.

Summarizing the novelties from Paper 3 to Paper 4. The architectural changes we applied from Paper 3 to Paper 4 are summarized below:

- Reduction of the model trainable parameters. Table A.1 shows a reduction of ∼70 in model trainable parameters when replacing VGG-19 (psz=3) with Resnet-18. Consequently, both MACs operations and training times decreases.
- Better separation among classes in the embedding. Although it is the simplest model and trained for fewer epochs, Resnet-18 guarantees the best separation by class of training data (Figure A.5).
- Learning the mapping from high-to low-dimensional space. We are working on the replacement of t-SNE with a MLP or a Parametric UMAP. This allows us to obtain a model for dimensionality reduction able to learn the parameters useful to automatically project data into the embedding.
- Visualization of the decision surfaces. With the integration of classifiers such as SVM, Linear, and Quadratic Discriminant Analysis, we are able to visualize the decision surfaces defined by the classifiers in the embedding.

Figure A.10 shows the two network architectures we presented in Paper 3 and Paper 4 for comparisons.

Appendix B

Night report examples

B.1 Date: 2025-01-03. Telescope: Copernico.

Table B.1: Seeing quality: ∼4. Humidity (%): [80,100]

Filename	**Object**	**Filter_texp (sec)**	**Comments**
AF9319(01-05)	Dome flat	R10	Acquired on 2025-01-02
AF9319(06-10)	Bias	R0	Acquired on 2025-01-02
Observing night 2025-01-03			
AF9319(24-28)	SA26269	R30	Time 18:27 UT. Airmass ∼1.55
AF931929	C2024 N3	R600	
AF931930	C2023 X2	R600	Humidity 97%
AF9319(31-32)	C2023 X2	R900	
AF931933	C2023 X2	R1200	
AF9319(34-38)	SA26269	R45	Time 22:55 UT. Airmass ∼1.02
AF931939	C2023 X7	R600	Seeing 4. Paolo suggests to change target.
AF931940	C2022 U3	R600	
AF9319(41-44)	C2022 U3	R900	
AF931945	C2023 H5	R600	
AF931946	C2023 H5	R900	
AF931947	C2023 H5	R1200	
AF931948	C2023 H5	R900	
AF9319(49-53)	SA26269	R45	Time 2:20 UT. Airmass ∼1.20

B.2 Date: 2024-12-01. Telescope: TNG

Table B.2: Information about the contents of each FITS file. The prefix of each of the following files is LRS.2024-12-03.

Filename	Object	Filter_texp	Comments
T00-04-20.003	C2023 X2	R10	
T00-10-13.811		R300	Max counts 5825
T00-26-06.889		R600	
T00-42-00.101		R900	
T01-12-53.374		R1800	
T01-22-37.250	PG0918+029	V2	First airmass point
T01-23-30.869		V5	
T01-24-42.988		R2	
T01-25-39.595		R5	
T01-27-37.716		I2	
T01-28-17.343		I5	
T01-29-32.476		I5	Duplicate acquired in I5 by mistake
T01-39-12.332	C2023 X7	R10	
T01-45-14.170		R300	
T01-56-35.156		R600	
T02-12-57.315		R900	
T02-45-26.074		R1800	
T02-57-06.468	PG0918+029	V2	Second airmass point
T02-58-02.097		V5	
T02-59-08.222		R2	
T02-59-59.834		R5	
T03-02-08.455		I2	
T03-03-06.075		I5	
T03-14-07.995	C2024 A1	V10	
T03-20-33.701		V300	
T03-31-46.705		V600	
T03-33-34.848		R10	
T03-39-33.684		R300	
T03-51-10.913		R600	
T04-11-58.243		R1200	
T04-13-46.392		I10	
T04-19-40.179		I300	
T04-31-06.241		I600	
T05-49-43.933		V2	See note
T05-50-41.552		V5	See note
T05-51-44.200		R2	See note
T05-52-59.308		R5	See note
T06-07-21.775	PG0918+029	V2	Third airmass point
T06-08-12.393		V5	
T06-09-27.532		R2	
T06-10-28.154		R5	
T06-12-10.293		I2	

Filename	Object	Filter_texp	Comments
T06-13-28.408		I5	

Calibration file. During the first night, the staff astronomer on duty (Avet) took 10 bias frames for me, which should be available in the CALIB program.
Regarding the flats, as I reported yesterday, due to a malfunction of an instrument called M3, all telescope initialization procedures were delayed. For this reason, after specifying the filters (V, R, I) to the support technician "Gianni" (I assume Giovanni Mainella, checking the TNG staff page), he assured me that he would take them tonight (December 3, 2024).

NOTE. These 4 images of C2024A1 were mistakenly acquired with standard field filters and exposure times by the operator.

B.3 Date: 2024-11-07. Telescope: Copernico

Table B.3: Information about the contents of each FITS file.

Filename	Object	Filter_texp	Comments
AF9108(56-65)	bias r	r_0	**Error: binning 2×2**
AF9108(66-75)	flat r sky	r_1; r_2	**Error: binning 2×2**
AF9108(76-85)	bias r	r_0	
AF9108(86-90)	flat r sky	r_2	
AF9108(91-95)	flat r sky	r_5	
AF9109(33-72)	C2023A3	r_10	Images in [AF910896–AF910932] to be rejected. Telescope out of focus.
AF9109(73-74)	C2022U1	r_600	
AF910975	C2022U1	r_900	
AF9109(76-78)	C2022U1	r_600	
AF9109(79-83)	PG2331	r_20	Time 19:42 UT. Airmass ($a \sim 1.30$)
AF9109(84-88)	PG2331	r_30	
AF9109(89-92)	C2024R4	r_900	
AF9109(93-96)	C2024N4	r_900	
AF9109(97-99)	PG2331	r_60	Time 22:23 UT. $1.55 < a < 1.74$
AF9110(00-01)	PG2331	r_60	
AF9110(02-05)	C2024N4	r_900	
AF9110(06-10)	PG2331	r_60	Time 23:36 UT. $a \sim 2.36$
AF9110(11-25)	C2022QE78	r_300	Comet begins to approach the star, then
AF9110(26-30)	C2022QE78	r_120	t_exp is reduced from 300 to 120.
AF9110(31-35)	SA26269	r_10	Time 01:20 UT. $a \sim 1.04$
AF9110(36-40)	SA26269	r_15	
AF911041	C2022U3	r_600	
AF9110(42-48)	C2022U3	r_900	
AF911049	C2023H5	r_600	
AF9110(50-52)	C2023H5	r_900	
AF911053	C2023H5	r_600	

B.4 Date: 2024-10-30. Telescope: Copernico

Table B.5 shows the main information for each FITS file collected during the night of October 30, 2024. For each file, I indicate the name, the type of object (i.e., whether bias, flat, comet, or standard), the filter and exposure time, and any comments. When filenames of the type AFXXXX(a-b) appear in the table, it means that all files in the range [AFXXXa-AFXXXb] have the information shown in the `Object` and `Filter_texp` columns in common. For example, AF9036(44-48) indicates that files AF903644, AF903645, AF903646, AF903647, and AF903648 are related to flat fields taken in the sky, with filter `r` and exposure time 1.

1 This problem had also occurred earlier in the night, when Paolo was present. In both cases, it was resolved by temporarily disabling dome tracking and moving the dome using the manual controls. I reported the problem in the telescope's night report.

Object	Exposures (n×s)	Total (s)	Comments
C2022U1	5×600; 1×900;	3900	
C2024R4	4×900;	3600	
C2024N4	8×900;	7200	
C2022QE78	15×300; 5×120;	5100	
C2022U3	1×600; 7×900;	6900	
C2023H5	2×600; 3×900;	3900	
PG2331	5×20; 5×30; 10×60;		3 different airmasses.
SA26269	5×10; 5×15;		Fourth airmass point

Table B.4: Summary of data collected during the night in terms of number of exposures and duration of each.

Table B.6 shows, for each target observed (comets and standard fields), the number of exposures and the duration of each exposure (column `Exposures`, expressed in seconds as $n \times s$ where n is the number of exposures and s is the duration of each exposure). The `Total` column provides the total exposure time for each object in seconds, obtained by adding up the individual exposure times. The `Comments` column includes additional information about the standard fields.

Table B.5: Information about the contents of each FITS file.

Filename	**Object**	**Filter_texp**	**Comments**
AF9036(44-48)	flat sky	r_1	
AF9036(59-63)	bias	r_0	
AF9037(08-11)	gamma ray burst		
AF9037(12-16)	SA41634	r_20	
AF9037(17-18)	C2022U1	r_600	
AF903719	C2022U1	r_900	
AF903720	C2022U1	r_300	To be discarded. Comet overlapped with a star.
AF9037(21-25)	SA41634	r_30	
AF903726	C2022U1	r_600	
AF903727	C2022U1	r_600	To be evaluated. Comet partially overlapping a star.
AF903728	C2023V1	r_600	
AF903729	C2023V1	r_900	Comet about to overlap with a star. Target changed.
AF903730	C2021Q4	r_900	
AF9037(31-33)	C2021Q4	r_1200	Aircraft trail in frame 33. Comet intact.
AF9037(34-38)	SA41643	r_25	
AF9037(39-41)	C2023R1	r_1200	
AF9037(42-44)	C2023R1	r_900	
AF903745	C2024N4	r_600	10-min exposure acquired to evaluate the influence of bright stars.
AF9037(46-49)	C2024N4	r_900	Between exposures 46–47: alarm "dome not connected"[1].
AF9037(50-54)	SA23441	r_10	
AF9037(55-59)	SA23441	r_5	

Object	**Exposures (n×s)**	**Total (s)**	**Comments**
SA41634	5×20; 5×30; 5×25		airmass 1,2,3.
C2022U1	4×600; 1×900;	3300	
C2023V1	1×600; 1×900;	1500	
C2021Q4	1×900; 3×1200;	4500	
C2023R1	2×1200; 3×900;	5100	
C2024N4	1×600; 4×900;	4200	
SA23441	5×10; 5×5;		airmass 4.

Table B.6: Summary of data collected during the night in terms of number of exposures and duration of each.