

# FEATURE EXTRACTION FOR MACHINE LEARNING-BASED SINGLE PHOTON LIGHT CURVE CLASSIFICATION

Nadine M. Trummer<sup>(1)</sup>, Amit Reza<sup>(1)</sup>, Michael A. Steindorfer<sup>(1)</sup>, Christiane Helling<sup>(1)</sup>

<sup>(1)</sup>Space Research Institute, Austrian Academy of Sciences, Lustbühelstraße 46, A-8042 Graz, Austria  
Email: nadinemaria.trummer@oeaw.ac.at

## ABSTRACT

Space Debris Light Curves (LCs) are temporal variations of object brightness and allow to retrieve characterizing information about the target by e.g., applying methods of Machine Learning (ML). So far, only LCs obtained from non-resolved, CCD-based images have been analysed using ML. In this paper, we successfully classify LCs from IWF SPARC, a LC catalogue captured on single photon basis, thereby showing this type of LC is an alternative to its CCD-based counterpart. A key finding of our experiments is that a pre-processing step called feature extraction can significantly boost the performance of ML classifiers (e.g., Random Forest). This insight can be used to tackle tasks for which a limited amount of training data is available.

## 1 INTRODUCTION

The ever-growing number of space debris [1] in Earth's orbit is source of concern for ongoing as well as future satellite missions due to risk of collision. Characterizing debris (i.e., inferring information such as attitude motion, shape and material [2]) is of high interest to mitigate this threat, however, the retrieval of object properties is challenging. Due to their small size paired with atmospheric effects, not all space debris are optically resolvable, making the analysis of unknown targets difficult.

Unresolved optical measurements, however, still allow the retrieval of object brightness (i.e., the fraction of sunlight reflected from the target towards the observer). Light Curves (LCs) describe the variations of object brightness over time and have been shown to hold key information such as object shape [3], attitude [4], and rotational state [5]. LCs are either obtained through extracting pixel brightness from a sequence of CCD based telescope images (e.g., [4, 5]) or counting the incoming photons directly using a Single Photon Avalanche Diode (SPAD) detector [6]. SPADs in particular permit high sampling frequencies (e.g., 100 Hz), while file sizes remain relatively low compared to images (a few kB per measurement).

With a growing number of LCs available, Machine Learning (ML) has been identified as a possible technique for space debris characterization. So far, promising results have been achieved by applying Neural

Networks (NNs) (e.g., [7]) using LCs from CCD-based catalogues as input. NN-architectures, however, might comprise millions of parameters, and hence require significant amounts of data for training, which can be difficult to provide. Obtaining ground truth labels for space debris remains a core challenge of the field [8], since prior knowledge about objects is necessary. Efforts have been made to address this issue by including simulated LCs in the training process [7, 8, 9].

In this paper, we successfully classify space debris LCs, using deep as well as traditional ML models. Our contribution to the field is two-fold: first, rather than relying on CCD-based LCs for our experiments, we train our models using LCs captured on single photon basis, thereby showing this type of LC can be an alternative type of input for ML models. Second, we found that a pre-processing step called feature extraction can significantly boost the performance of classifiers such as Random Forests.

The rest of the paper is structured as follows: Section 2 will outline our approach. We introduce the study dataset and explain the ML framework. Section 3 will provide details on the feature extraction step. In Section 4, we will present our results and document details on implementation. This is followed by a discussion in Section 5. Section 6 concludes the paper.

## 2 APPROACH

The objective of this work is to classify space debris light curves captured on single photon basis using machine learning, thereby showing that this type of data is an alternative to CCD based measurements. This section provides information on the study dataset, explains the logic of the employed labelling schemes and gives details on the ML framework.

### 2.1 Study Dataset

The dataset used in this paper is a subset of the **IWF SPARC** [10], a catalogue of space debris light curves curated by the Space Research Institute (IWF) Graz, Austria. IWF SPARC includes more than 6500 LCs representing 700 individual targets. The number of LCs available per object varies between 1 and 243, with 60% of objects being represented by 3 or less LCs. Space objects in this catalogue can be broadly categorized into active satellites (e.g., geodetic satellites [11]), defunct

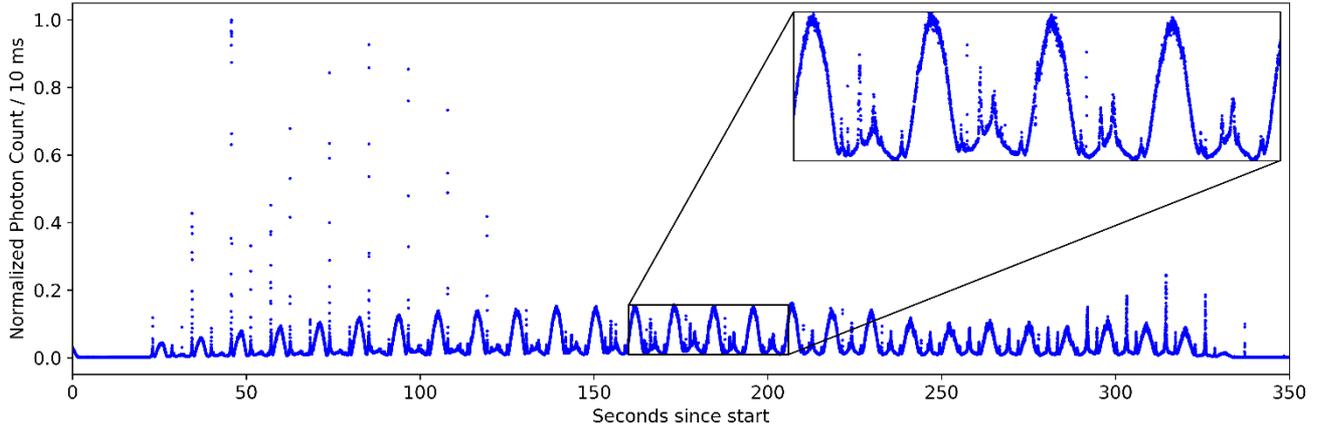


Figure 1. A light curve of TOPEX/Poseidon (NORAD 22076) on 2015/07/18. The first third of the measurement (up to 120 s) shows so-called glint patterns, which are narrow spikes in photon count. They correspond to moments of specular reflection. The pattern in the zoomed-in panel (length 46 s) depicts 4 full rotations.

satellites (e.g., ENVISAT), and debris (e.g., rocket bodies). Fig. 1 depicts a LC measurement of TOPEX/Poseidon (NORAD 22076), which is a defunct satellite with a known spin period of 10.73 s (2016) [12]. In addition to the pattern consisting of a primary and secondary peak, we observe what we call glints. Glints are narrow, steep spikes in photon count and only comprise a couple of datapoints. We interpret these to be moments of specular reflection, which occurs on smooth, shiny surfaces (e.g., solar panels) for the case when the light source and observer angles are equal relative to the surface normal. Lastly, we observe a change in overall signal amplitude (in this case, a continuous increase towards the 230 s mark, and a decrease after that). These changes are related to observation geometry.

The observed variation in object brightness is a function of the target’s shape, attitude and rotational behaviour as well as the observation geometry. It follows that LCs of rotating objects are of special interest, since they hold more relevant information. With a spin period of around 10 s, TOPEX/Poseidon is one of the fastest rotating objects in the catalogue. Most objects rotate significantly slower (e.g., GLONASS satellites in our catalogue have been associated with rotation periods between 8 s and 400 s [13]). This has implications for the desired measurement length (e.g., [14] found a measurement length of at least six times the rotation period to be necessary for proper spectral analysis). The mean duration of LC measurements in the IWF SPARC is 269 s, the median duration is 187 s.

For this study, we used a well-behaved subset of the IWF SPARC. Based on visual quality inspection, we selected 852 LCs representing 150 individual objects. We focused on targets that show clear rotational behaviour and included LCs with a measurement length of at least 100 s. ML algorithms generally require the input data to be of uniform shape (i.e., same length). For this reason, we further sliced the data into segments of 100 s, extracting

multiple slices from one LC where possible. This method, also referred to as window slicing, is considered a form of data augmentation [15]. This leaves us with a total of 1141 LCs available for our experiments.

## 2.2 Light Curve Categories

Supervised ML models require a labelled dataset for training, meaning each input must be associated with a target variable. In this study, we adopt three labelling schemes, which are summarized in Tab. 1.

Table 1. Summary of the three data subsets employed in this study. All subsets draw from the initial set of 1141 LCs selected for this study.

| Subset   | Classes | Unique Objects | Total LCs |
|----------|---------|----------------|-----------|
| Uniques  | 8       | 8              | 675       |
| Families | 5       | 22             | 752       |
| Types    | 3       | 97             | 510       |

The LCs are either labelled according to the *unique* object they represent (e.g., TOPEX/Poseidon or Jason-2), object *families* they are associated with (e.g., satellites from the GLONASS constellation, or ATLAS 5 rocket bodies), or their general *type* (rocket bodies, geodetic satellites, or active satellites). This results in three classification tasks that aim to represent our data at different levels of abstraction. The notable difference between the schemes lies in the number of unique objects contributing to classes. The *Uniques* task is characterized by highly homogenous classes (i.e., LCs within a class stem from the same object) at the cost of having a relatively low number of samples per class available (29 LCs for the smallest, 214 LCs for the largest class). The *Types* set on the other hand is more heterogenous (i.e., LCs of 97 unique objects are assigned to just 3 classes). The *Families* set provides a midpoint between the other two

representations. It allows us to include objects into training that have not been considered for the *Uniques* set due to a low number of LCs available while still preserving information about *what* makes objects in a class similar to each other (in this case, shape).

In the following, we refer to the process of classifying one of the three subsets (*Uniques*, *Families*, *Types*) as a *task*.

### 2.3 Machine Learning Framework

The standard procedure of ML involves the selection, training, validation, and testing of models in order to classify data. For this, the available data is partitioned into subsets used for training, validation and testing, typically using a predefined ratio for splitting (e.g., 80% training, 10% validation, 10% testing). The model is first trained, which involves the optimization of parameters with the goal to minimize a chosen loss function. The validation set is used to evaluate performance during training and tune model hyperparameters, which are model parameters that cannot change during training. The final performance is assessed through a test set. In case of a classification task, the performance can be quantified through an accuracy score.

Before training can begin, a suitable ML model needs to be selected and build. For space debris classification, promising results have been achieved using Neural Networks (NNs). This study, however, has a limited amount of data available, which might not allow a NN to unfold its full potential. We therefore decided to employ an ensemble-based ML method (Random Forest Classifier) in addition to a deep learning method (Convolutional Neural Network).

A Random Forest Classifier (RDF) [16] is an ensemble learning method consisting of individual decision trees. A decision tree classifies the data by recursively splitting it into subsets with the aim to minimize impurity. Each tree is trained using a random subset of data and input features; hence, individual trees may produce different predictions. The final prediction of the RDF is obtained through a majority voting scheme.

A Convolutional Neural Network (CNN) [17] is a deep learning model that is especially suited for classifying structured data (e.g., images). CNNs consist of multiple layers that have varying functions. Convolutional layers aim to learn spatial hierarchies of the input by capturing local patterns (e.g., edges, textures) by applying filters. Pooling layers are generally employed after convolutional layers and help extract dominant features as well as reduce special dimensions. Predictions are made at so-called fully connected layers, which integrate the features learned from convolution and make the final prediction.

To assess classifier performance, we use a method called

$k$ -fold cross validation. The training data is split into  $k$  subsets (folds). One subset is held back for validation while the model is trained on the remaining data. Repeating this process  $k$  times allows to retrieve an average accuracy score, which is more robust than a score achieved on a single validation set. Further, we utilize the confusion matrix (CM) representation, which is a visual tool for understanding prediction outcomes. In case of multi-class classification with  $p$  classes, each row and column of a  $C_{p \times p}$  confusion matrix represents the actual and predicted classes, respectively. A matrix cell ( $C_{ij}$ ) represents the number of instances of class  $i$  that were predicted as class  $j$ . A well-performing classifier should yield high values across the matrix diagonal ( $C_{i=j}$ ), indicating correct classification, and low values on the off-diagonal ( $C_{i \neq j}$ ) (i.e., predicted and actual class do not agree).

## 3 FEATURE EXTRACTION

LC classification has seen promising results using CNNs. However, algorithms such as RDFs can obtain similar accuracy by passing feature-engineered data as input instead of training on the raw data. Feature extraction is a well-established technique within the domain of time series classification [18]. The approach aims to capture relevant characteristics of data and hence find a new representation by selecting its most representative features. Trivial examples for features are the mean or variance of a time series. For a time series consisting of  $n$  ordered, real-valued datapoints, a feature vector can be constructed by applying a feature mapping. Feature vectors can then be used as training input for classifiers (e.g., RDF). The approach described above is general, it becomes problem-specific with the choice of features.

This work utilizes the TSFresh feature extraction pipeline, which is a python package designed for automated extraction of statistical features from time series data [19]. TSFresh presents its features in three groups: features from summary statistics (e.g., mean or variance), characteristics of sample distribution (e.g., number of datapoints above mean), and features derived from observed dynamics (e.g., Fourier transformation coefficient).

## 4 RESULTS

For computational reasons, we downsampled the LC data through photon count binning (hereafter referred to as decimated data). Tests showed that classifier performance begins to suffer significantly for sampling rates of 1 Hz or lower, hence, we chose a sampling rate of 10 Hz for our decimated data. Next, we extracted features from the decimated data using the TSFresh feature extraction pipeline, which provided us with 342 important features.

As our first step, we trained a RDF classifier on the

decimated data to obtain a baseline performance. Next, we trained a RDF using the feature vectors as input for training (Features+RDF) instead of the decimated data. We repeated our experiments for all three data subsets (*Uniques*, *Families*, *Types*), which results in 6 possible model-task combinations. We evaluated the performance of all classifiers using 10-fold-cross validation. The average accuracy scores for the baseline RDF are compared to the Features+RDF approach are listed in Tab. 2. Further, Fig. 3 and Fig. 4 depict the aggregated (i.e., averaged across all 10 runs) confusion matrices (CM) on the *Families* dataset.

Table 2. Average 10-fold cross validation accuracy scores on the RDF-related model-task combinations.

| Classifier     | Uniques | Families | Types |
|----------------|---------|----------|-------|
| RDF            | 65 %    | 69 %     | 67 %  |
| Features + RDF | 88 %    | 91 %     | 87 %  |

Next, we investigated the performance of CNN. Fig. 2 represents the architecture of the network build for this study. We use a fixed kernel size of 64, a filter size of 3 and the tanh activation function for the convolutional layers. The output layer uses the softmax activation function. Batch normalization is used between the convolution and max pooling layers.

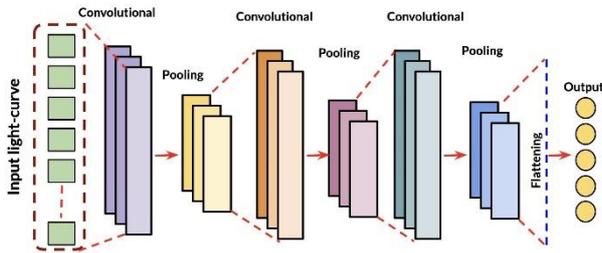


Figure 2. Schematic of the CNN architecture used in this study. Three convolutional and max-pooling layers are used between the input and output layers [10].

For our CNN experiments, we focused on the *Families* subset and tested three possible cases: classification on original resolution LCs, training on decimated data, and training on features obtained through TSFresh. For these cases, we achieve accuracies of 80%, 83%, and 88% respectively. To allow comparison with Features+RDF, Fig. 5 depicts the CM results using TSFresh features as input.



Figure 3. Aggregated CM for the RDF classifier on the *Families* subset.



Figure 4. Aggregated CM for the Features+RDF classifier on the *Families* subset [10].

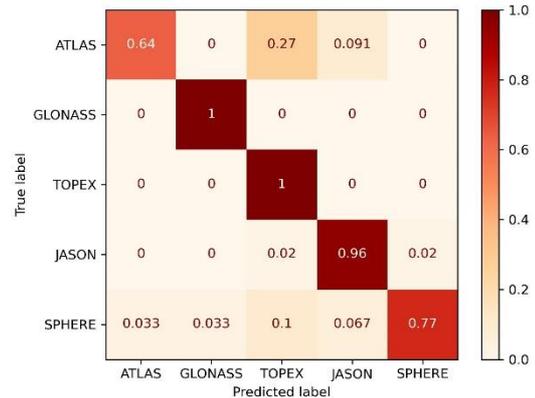


Figure 5. Aggregated CM for the CNN trained on features extracted using TSFresh [10].

## 5 DISCUSSION

The RDF classifier trained on decimated data yields baseline accuracies between 65% and 69% across the three proposed tasks. The CM in Fig. 3 shows that the baseline RDF model has trouble differentiating between the majority classes JASON and TOPEX. Further, the classes SPHERE (geodetic satellites) and ATLAS (rocket bodies) are frequently misclassified. Lastly, the model tends to confuse SPHERE and GLONASS.

When introducing a feature extraction step, however, the performance of the RDF is boosted significantly. The Features+RDF model achieves accuracies between 87-91%, with the highest result being achieved on the *Families* subset.

In our experiments, we observed that a CNN trained on features extracted from TSFresh performs better than a CNN trained on raw data or decimated data. CNN thereby outperforms RDF on the *Families* subset, but could not beat the Features+RDF approach. CNN extracts features automatically using convolutional filters; the lower performance indicates that CNN struggles to capture important key aspects of the input data. Imbalanced classes combined with a lack of training data are likely the reason why CNN could not outperform Features+RDF in this specific experiment.

As the CMs show, both Features+RDF (Fig. 4) and CNN (Fig. 5) do not suffer from most of the inaccuracies observed in the baseline RDF model (Fig. 3). The difficulties between classes ATLAS and GLONASS, however, remain. ATLAS is a minority class (contributing only 7.6% of samples to the *Families* subset), and visual inspection has shown that LCs in this class are very similar to those in the GLONASS class. It is possible that the features provided by TSFresh might not be enough to capture the more subtle differences between the LCs of these classes. The visual similarity paired with the low number of training samples available for ATLAS could be the reason this specific type of misclassification occurs.

## 6 CONCLUSION

This work presents the classification of single photon space debris LCs using multiple ML approaches. We employ three different schemes for labelling our data in order to allow comparability with existing studies and classify the sets using an ensemble-learning (RDF) as well as a deep learning (CNN) method. Combining RDF with a feature extraction step, we achieve high classification accuracies across all three data subsets (88-91%). Our results compare well to results reported in related literature and show that space debris LCs captured on single photon basis are suitable as input for ML models.

Further, our experiments indicate that a feature extraction

step can significantly boost the performance of classifiers. By using features extracted from LCs for training, we were able to boost the accuracy of a RDF classifier by ~21% compared to a baseline model trained on raw data. Further, the classifier trained on LC features was able to outperform a CNN, which was likely not able to unfold its full potential due to limited amounts of data available. From this we conclude that methods such as RDF, if combined with feature extraction, can be an alternative to NNs for cases in which well-labelled, high-quality training data is not readily available and therefore scarce.

We recognize that this paper has two limitations. First, we did not perform experiments on data with original resolution (100 Hz), thereby not fully capitalizing on potential advantages of the high sampling frequencies. Second, we conducted our experiments on a well-behaved subset of LCs. However, we have to anticipate cases in which we might not be able to capture a high-quality LC of an unknown object. Therefore, training models to work with less-than-ideal data is desirable. Future work will explore the utilization of full-resolution LCs as well as the inclusion of a larger data quantities from IWF SPARC.

## 7 REFERENCES

1. Schildknecht, T. (2007). Optical surveys for space debris. *Astron. Astrophys. Rev.* **14**, 41-111.
2. Fan, S. & Frueh, C. (2020). A direct light curve inversion scheme in the presence of measurement noise. *J. Astronaut. Sci.* **67**, 740-761.
3. Šilha, J., Zigo, M., Hrobár, T., Jevčák, P. & Verešvárska, M. (2021). Light curves application to space debris characterization and classification. In *Proc. 8th European Conference on Space Debris*, Darmstadt, Germany, 20–23 April 2021.
4. Kucharski, D., Kirchner, G., Jah, M. K., Bennett, J. C., Koidl, F., Steindorfer, M. A. & Wang, P. (2021). Full attitude state reconstruction of tumbling space debris TOPEX/Poseidon via light-curve inversion with Quanta Photogrammetry. *Acta Astronaut.* **187**, 115-122.
5. Šilha, J., Pittet, J. N., Hamara, M. & Schildknecht, T. (2018). Apparent rotation properties of space debris extracted from photometric measurements. *Adv. Space Res.* **61**(3), 844-861.
6. Steindorfer, M. A., Kirchner, G., Koidl, F. & Wang, P. (2015). Light curve measurements with single photon counters at Graz SLR. In *Proc. ILRS Technical Workshop*, Matera, Italy, October 26-30, 2015.

7. Linares, R., Furfaro, R. & Reddy, V. (2020). Space objects classification via light-curve measurements using deep convolutional neural networks. *J. Astronaut. Sci.* **67**(3), 1063-1091.
8. Allworth, J., Windrim, L., Bennett, J. & Bryson, M. (2021). A transfer learning approach to space debris classification using observational light curve data. *Acta Astronaut.* **181**, 301-315.
9. Furfaro, R., Linares, R. & Reddy, V. (2018). Space objects classification via light-curve measurements: deep convolutional neural networks and model-based transfer learning. In *19th Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS 2018)*, 11-14 September 2018, Maui, Hawaii, USA.
10. Trummer, N. M., Reza, A., Steindorfer, M. A. & Helling, C. (2025). Machine learning-based classification for Single Photon Space Debris Light Curves. *Acta Astronaut.* **226**, 542-554.
11. Pearlman, M., Arnold, D., Davis, M. et al. (2019). Laser geodetic satellites: a high-accuracy scientific tool. *J. Geod.* **93**, 2181–2194.
12. Kucharski, D. et al. (2017). Photon pressure force on space debris TOPEX/Poseidon measured by satellite laser ranging. *Earth Space Sci.* **4**(10), 661-668.
13. Kirchner, G. et al. (2017). Determination of attitude and attitude motion of space debris, using laser ranging and single-photon light curve data. In *Proc. of the 7th European Conference on Space Debris*, Darmstadt, Germany.
14. Linder, E., Silha, J., Schildknecht, T. & Hager, M. (2015). Extraction of spin periods of space debris from optical light curves. In *Proc. of 66th International Astronautical Congress*.
15. Le Guennec, A., Malinowski, S. & Tavenard, R. (2016). Data augmentation for time series classification using convolutional neural networks. In *ECML/PKDD workshop on advanced analytics and learning on temporal data*.
16. Pal, M. (2005). Random forest classifier for remote sensing classification. *Int. J. Remote Sens.* **26**(1), 217-222.
17. Albawi, S., Mohammed, T. A. & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, Antalya, Turkey, 2017, pp1-6
18. Abanda, A., Mori, U., & Lozano, J. A. (2019). A review on distance based time series classification. *Data Min. Knowl. Discov.* **33**(2), 378-412.
19. Christ, M., Kempa-Liehr, A. W., & Feindt, M. (2016). Distributed and parallel time series feature

extraction for industrial big data applications. *arXiv preprint arXiv:1610.07717*.