Automatic Recognition of Low Earth Orbit Objects from Image Sequences

Florin Oniga, Melania Miron, Radu Danescu, and Sergiu Nedevschi
Computer Science Department
Technical University of Cluj-Napoca, Romania
florin.oniga@cs.utcluj.ro

Abstract — **This paper presents an approach for automatic recognition of interest objects from the low earth orbit that are visible in astronomical images. The main interest objects are satellites, but various objects, such as planes, stars, can be identified. The proposed technique starts with background estimation and removal. Then, potential objects are identified by labeling the background free image. Relevant features are computed for each individual object and used for classification, which is performed with a decision tree. Real and relevant data was used to evaluate the performance of this methodology and to determine its main strengths and weaknesses.**

I. INTRODUCTION

In the context of increasing the number of artificial satellites that are launched every year, detecting satellites coordinates becomes more and more important, in order to avoid collisions and to keep the satellites on their orbits. Due to external forces, they tend to change their predicted trajectory, therefore they must be regularly monitored and have their orbital parameters updated. A series of images acquired by the Astronomical Observatories can be used for the detection of relevant objects.

The first step towards measuring the distance, and thus the orbit of Low Earth Orbit (LEO) objects is to detect their signature in the digital image. A defining characteristic of these objects is the linear nature of their image trace, which discriminates them from the point-like stars. The line segment trajectory is caused by the speed of the object, combined with the high exposure time of the imaging device.

The astronomical images contain a multitude of objects like stars, planes, outliers that might make it harder to observe satellite streaks, as well as to distinguish between satellites and the other astronomical objects. To perform the recognition task accurately, a proper approach would be to detect and classify all the above-mentioned objects into their corresponding classes.

The issue of detecting and classifying objects in astronomical images is addressed in this paper.

II. RELATED WORK

The task of detecting satellites streaks in astronomical images is quite new, and as a consequence, little amount of bibliography describes this area of research.

A first approach is the one proposed by Brad Wallace in [2]. The paper starts with describing the sensor characteristics in order to know precisely the main features of an astronomical image. An algorithm for streaks detection is proposed. Image-plane artifacts and large scale background are removed. A noise floor is determined for each image pixel, and objects above this floor are identified. Image moments are used to classify objects as point-like or streak-like.

Levesque et al [3] have published some of the most representative work. The approach is based on a series of steps, the most important ones being background estimation and removal, star detection and removal, followed by an iterative matched filter used to detect streaks.

III. OVERVIEW OF CONTRIBUTIONS AND PROPOSED ALGORITHM

Detecting and classifying astronomical objects (LEO) is the main goal addressed in this paper. The main difference between our approach and the existing methods are the background removal technique and the classification step, which is able to cope with more object types.

The proposed algorithm consists of the following steps:
1. Background detection and removal – an exponential moving average technique,
2. Objects detection – based on labeling and feature computation,
3. Classification – using a decision tree.

Our proposal for background removal is based on estimating the background image with an exponential moving average technique. The local statistics background removal algorithm proposed by Levesque [3] will be used as reference for comparison. Our proposed method provides similar results regarding streaks detection (even better regarding the removal of stars) but has a lower computational complexity.

The classification part represents another contribution of this algorithm. A decision tree, trained with manually labeled data, is used for the classification of objects based on their features. The star/satellite discrimination presented by current methods is not enough because astronomical images can contain streak-like objects that are not satellites: planes, condensation trails, clouds etc. Our method provides a more general output that deals with more object types.

IV. BACKGROUND REMOVAL

Background estimation and removal is the first step of the algorithm. Thus, it directly influences the input for the ones that follow and, consequently, the correctness of the obtained results.

When looking for the best approach for background removal, both the accuracy and the speed of the technique should be taken into account. Another important aspect is the influence of the Earth's rotation on the background. Considering that the exposure time is about 5 seconds, stars will also present a certain displacement (several pixels) between two frames. However, if the camera is mounted on an automated star tracker device then the stars will have no motion between frames. Our test images were mixed. Some of the sequences were acquired with a tracker device, others without.

The exponential moving average technique proposed by us performs optimally for images acquired with a tracker device (stars will be removed also), but it also works with standard images (because stars do not move significantly between two frames).

We will compare our approach with the background estimation and removal proposed by Levesque, referred later in text as the local statistics approach. The local statistics approach represents an adjustment of Levesque's local statistics background removal algorithm that has been presented in [3]. The method described by Levesque consists of 5 iterations. At each of them, the background is preserved, while the peaks are not saved as being part of the background image.

The following formula is used to implement all five iterations:

$$<B_i> = \text{local mean } (I_{i-1})$$
$$I_i = \text{minimum } (I_{i-1}, <B_i> + 2\sigma_n),$$

where i is in the range 1 to 5, I_0 is the initial image and σ_n is the local standard deviation.

The local mean and local standard deviation values are used for estimating the background image. The images used in our application contain enlarged stars, with dimensions of about 130% of the ones in Levesque's work. Consequently, the window size must be in accordance with these sizes. The optimum value for the window size has been determined experimentally to be 29. Using a 29x29 pixels window, saturated stars should be covered by the window.

After the background is subtracted from the current image, a binarization needs to be performed. The threshold is chosen experimentally, in order to ensure that faint satellite streaks are still visible. The smaller the threshold is, the better the results are, but the time for processing the image increases significantly. A proper threshold value that satisfies both time and quality is around seven gray levels.

The approach that we propose for background removal gives similar results in terms of satellite streaks. It provides an improved result if a tracker device is used, by removing almost all of the stars. If the images are acquired in a sequence by a tracker device-mounted camera, then the background and the stars do not move. During an entire sequence of images acquired by a tracker device-mounted camera, the only moving objects are satellites, planes, and some outliers (like clouds, condensation trails), making the object detection and classification steps much easier. If the camera is not mounted on a tracker device, the moving-average background technique also emphasizes stars and some parts of them are detected as outliers.
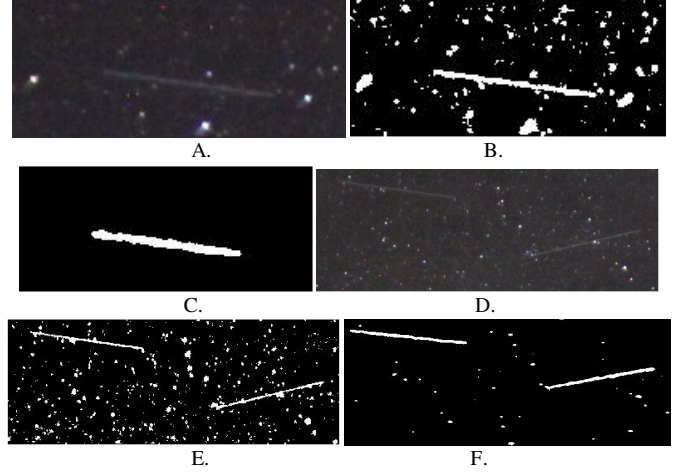

Figure 1. Background-free images A. Original image (in a "tracker" sequence) B. Background-free image obtained with the local-statistics based approach C. Background-free image obtained with the moving-average technique D. Original image (in a "non-tracker" sequence) E. Background-free image obtained with the local statistics based approach F. Background-free image obtained with the moving-average technique

Two additional pre-processing steps are performed. In order to avoid an increased number of small false objects due to image noise, each image is smoothed with a Gaussian convolution after the acquisition. Secondly, in order to compensate changes in the atmospheric lighting and the variation of the auto camera exposure, the mean of each image is computed and subtracted from each pixel.

When the first image is acquired, it is used to initialize the background image B_0. Afterwards, starting from the second image in the sequence, a weighted average is performed between the values of the pixels in the previous background B_{i-1} and the current image I_i.

$$B_i = \alpha B_{i-1} + (1-\alpha)I_i \qquad (1)$$

In order to reduce the number of detected stars or other objects due to the moving background (for images acquired without the tracker device), we want to give the current frame a higher weight. The background weight α is set to 0.1.

After the difference between the current frame I_i and the background image B_{i-1} is computed, the resulting image should contain positive non-zero values only where moving objects are present, as the other areas are supposed to be identical. A low threshold is applied in order to extract the relevant areas that will be processed further. Similarly to the local statistics background removal method, a value for the threshold that satisfies both time and quality is 7.

Figure 1 presents the background-free images resulted after processing images 1.A and, respectively, 1.D, in both of the approaches previously described. It can be easily observed that the results are similar, except the fact that the moving average

approach removes most of the stars, as they are considered part of the background. Also, for the image in the "tracker" sequence (with the tracker device), the background-free image contains very few stars (none visible around the satellite).

V. OBJECT DETECTION

A. Object Labeling

The algorithm used for labeling [5] is based on the following four steps: run-length encode the input image; scan the runs, assigning preliminary labels and recording label equivalences in a local equivalence table; resolve the equivalence classes; re-label the runs based on the resolved equivalence classes.

This algorithm works on the images obtained with both of the background removal approaches presented. Figure 2 shows the images obtained after the labeling algorithm is applied on background-free images. Each binary object has a specific color assigned after running the labeling algorithm.

B. Object Feature Computation

The labeled binary objects detected in the previous step must be classified, so that we can decide whether they are satellite streaks, planes or other objects. In order to perform that classification, we will approximate every binary object by an ellipse, and we'll compute the geometrical properties of this ellipse.
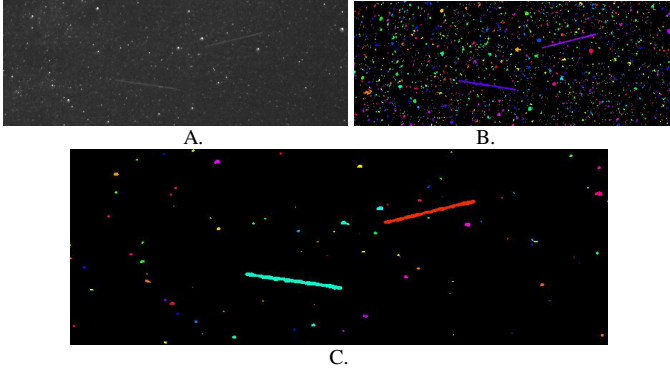

Figure 2. A. The original image (aquired without a tracker device) containing two satelite streaks, B. Labeled objects on the background-free image obtained with the local statistics approach, C. Labeled objects on the background-free image with the moving average background removal approach.

The initially considered properties for the binary objects were: Area, Major Axis Length, Minor Axis Length, Eccentricity, Equivalent Diameter, Perimeter, Solidity and standard deviation of the pixel values. Some of these properties will not be used in the final classification, since they do not determine a clear separation between the classes, as we will show in the following section.

The remaining properties for the final classifier are: Area, Major Axis Length (L_{MAX}), Minor Axis Length (L_{MIN}) and Eccentricity (e). These can be computed using the following equations:

$$S_{xx} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2 \qquad (2)$$

$$S_{yy} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \overline{y})^2 \qquad (3)$$

$$S_{xy} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})(y_i - \overline{y}) \qquad (4)$$

$$L_{MAX}{}^2 = 8(S_{xx} + S_{yy} + C) \qquad (5)$$

$$L_{MIN}{}^2 = 8(S_{xx} + S_{yy} - C) \qquad (6)$$

$$e = \sqrt{1 - \frac{L_{MIN}{}^2}{L_{MAX}{}^2}} \qquad (7)$$

$$C = \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}{}^2} \qquad (8)$$

In the above equations, N stands for the number of points of an object (object area in pixels), x_i and y_i are the row and column coordinates of the object's points, and $\overline{x}$ and $\overline{y}$ are the coordinates of the center of mass.

VI. TRAINING OF THE DECISION TREE

In order to build a decision tree, a training and testing database needs to be created by manually labeling each object. The result is a database containing all the relevant objects and their classes.

Next, a classifier was automatically generated by Weka [4], using J48 trees.

The imbalance in class distribution needs to be taken into account [6]. Most classification techniques, including decision trees, assume that the training samples are evenly distributed among different classes. Hence, the classifiers are affected by the prevalent classes and tend to ignore or treat the small classes as noise.

For our application, 229 images have been used. Each of them contributed with an average of five relevant objects to the training set. This was necessary in order to establish a balance between the satellites, planes and other objects (like remaining stars and outliers). For the moving average background technique, if the camera is not on a tracker device, there will be a large amount of stars that imbalance the training process. A similar situation happens also in the case of local statistics background removal.

A constraint is needed to decrease the number of stars in the training set, especially small and medium size ones that cannot be confused with satellite streaks or with planes. It has been observed experimentally that the area of satellite streaks and planes is greater than 85 pixels. Therefore, the constraint was that the objects' area is greater than this value.

VII. RESULTS

In this subchapter, the classification results for both the local statistics and our proposal for background removal will be presented.

Considering the analysis performed by Weka, the only selected attributes that need to be computed are area, eccentricity, minor and major axes length.

For the local statistics-based approach, the confusion matrix was the following:

Satellite	other	plane	Classified as
148	2	3	Satellite
3	1175	0	Other
1	0	5	Plane

The decision tree is the following (the area related condition was added to the result provided by Weka):

```
Area <=85: other
Area > 85
|  Eccentricity <= 0.99: other
|  Eccentricity > 0.99
|  |  MinorAxisLength <= 11.106: satellite
|  |  MinorAxisLength > 11.106: plane
```

For the moving-average approach, the following decision tree was generated by Weka (the area related condition was added to the Weka result):

```
Area <=85: other
Area > 85
|  Eccentricity <= 0.99
|  |  Eccentricity <= 0.935: other
|  |  Eccentricity > 0.935
|  |  |  MajorAxisLength <= 47.543: other
|  |  |  MajorAxisLength > 47.543: satellite
|  Eccentricity > 0.99
|  |  MajorAxisLength <= 200.337: satellite
|  |  MajorAxisLength > 200.337
|  |  |  Eccentricity <= 0.998: plane
|  |  |  Eccentricity > 0.998
|  |  |  |  MajorAxisLength <= 294.676: satellite
|  |  |  |  MajorAxisLength > 294.676: plane
```

The confusion matrix is the following:

Satellite	other	plane	Classified as
146	8	1	Satellite
0	523	0	Other
2	2	9	Plane

As observed, the number of objects detected in the two approaches differs significantly. This behavior is caused by the fact that, for images acquired with tracker device-mounted cameras, the stars are considered background, and when tracker devices are not used, there are only parts of stars remaining. Those parts, being smaller, are not taken into account when building the training dataset, because of the imposed area constraint.

Sample results are given in figure 3 and 4.

For both of the background removal approaches, there are situations when faint satellites are detected in an image, but not in the previous or following ones. Such an example is presented in figure 5.

VIII. CONCLUSIONS AND FUTURE WORK

The previously presented confusion matrix proves that the classifications are good, with a weighted ROC Area of 0.987 for the local statistics-based approach, and of 0.965 for the sequence-based one. The classification performance is slightly lower when the moving average technique is used, but this is compensated by the much lower computational complexity of the method. Furthermore, it manages to provide no false positives (stars classified as satellites) on the available image set, and we are confident that the results would improve if only sequences with tracker devices are used. Future evaluation will be performed in this direction.
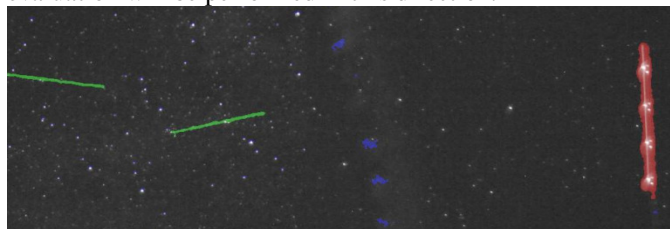

Figure 3.   Example of LEO objects classification: green – satellite, blue – outlier (other), red – plane
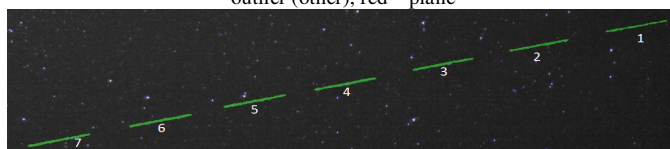

Figure 4.   Example of satellite detected in a sequence of images (the detection is overimposed on the first image)
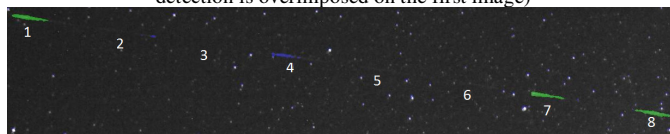

Figure 5.   Satellite streaks detection in eight successive frames: very faint or not visible streaks are not detected

Further improvement will be performed, based on temporal validation of the classification. It relies on the fact that satellites position will be on the same direction as in the previous and following images, so misclassifications can be corrected.

REFERENCES

[1] "Low Earth Orbit", online resource at http://en.wikipedia.org/wiki/Low_Earth_orbit

[2] Brad Wallace, Robert (Lauchie) Scott, Aaron Spaans "The DRDC Ottawa Space Surveillance Observatory"

[3] Martin P. Levesque, "Automatic Reacquisition of Satellite Positions by Detecting Their Expected Streaks In Astronomical Images", Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference, Hawaii, September 1-4, 2009, Ed.: S. Ryan.

[4] http://en.wikipedia.org/wiki/Weka_(machine_learning)

[5] Haralick, Robert M., and Linda G. Shapiro, "Computer and Robot Vision, Volume I", Addison-Wesley, 1992, pp. 28-48.

[6] Giang Hoang Nguyen, Abdesselam Bouzerdoum, Son Lam Phung, "Learning Pattern Classification Tasks with Imbalanced Data Sets", book chapter in "Pattern Recognition", 2009, publisher: InTech, ISBN 978-953-307-014-8, pp: 193-208.