IMPROVED SPACE DEBRIS ASTROMETRY WITH TEMPLATE MATCHING

Junju Du, Shaoming Hu, Xu Chen, and Difu Guo

School of Space Science and Physics, Institute of Space Sciences,
Shandong Provincial Key Laboratory of Optical Astronomy and Solar-Terrestrial Environment,
Shandong University, Weihai,
Weihai 264209, China, Email: dujunju@126.com

ABSTRACT

Images with trails can be obtained when observing space debris, no matter the telescope tracks on sidereal or target's speed. Low astrometric accuracy of trailed sources remains one of the most essential sources of the orbital data uncertainty for space debris. When trails are long, faint or distorted by tracking error, the widely used point spread function (PSF) fitting techniques tend to fail. We present a fresh source extraction pipeline based on Template Matching, which is a method for searching and finding the location of a template image in a larger image. The principle and implementation of Template Matching are described in detail. The performance of Template Matching was tested on thousands of synthetic and real observation images. Compared to thresholding, Template Matching is more accurate and robust. Template Matching is more suitable for faint and long trails extraction.

Keywords: Space debris; Source Extraction; Template Matching.

1. INTRODUCTION

In a modern era, ground-based optical telescopes are one of major observation methods to survey the space debris since its effective and feasible [12, 16]. In optical surveys, space debris appear as fast moving objects with high angular velocities respecting to the stellar background [11]. Tracking the target space debris is a reasonable imaging strategy to get a higher SNR [7]. In this mode, space debris in astronomical images is point-like and field stars appear as trails. Figure 1 shows a typical star-trailed image obtained by 1m telescope of Weihai Observatory of Shandong University[1]. In differential astrometry, it is necessary to get the accurate position of trails. A problem with trails detection is the 'loss' of SNR as the stars become trailed [7]. There are two ways to think of this effect: (1) the peak signal per pixel decrease as 1/length while the per pixel noise remains roughly constant or (2) the total signal in the trail remains constant as the noise increase because there are more pixel under the trail. On the other hand, the profiles of trails change frame by frame due to influence of the brightness of the star, tracking accuracy of the telescope, atmosphere turbulence and extinction fluctuation [6, 14] (Figure 2). All the above factors lead to the detection of trailed sources is a common difficulty in astronomical image analysis, whether space debris or near earth objects [15].

Figure 1. A typical observation image of space debris obtained by 1m telescope of Weihai Observatory of Shandong University. Target space debris was tracked and background stars were trailed. The image is of size 2048×2048. *With 2 second exposure time, the trails are about 70 pixels in length.*

A lot of efforts have been devoted to the processing of trailed sources in recent years. Some methods have been developed such as thresholding, edge detection and some techniques based on point-spread-function (PSF) fitting [6, 5]. Thresholding is a simple and effective method for image segmentation, which can be used to decide which regions (connected pixels) are considered as objects and which are background [9]. The barycenter of each object region serves as its center. But defining an appropriate threshold is not easy due to noise, background variations or diffuse edges of the objects. Any

[1] http://astro.wh.sdu.edu.cn/

Proc. 1st NEO and Debris Detection Conference, Darmstadt, Germany, 22-24 January 2019, published by the ESA Space Safety Programme Office
Ed. T. Flohrer, R. Jehn, F. Schmitz (http://neo-sst-conference.sdo.esoc.esa.int, January 2019)

chosen threshold may result in some true objects being overlooked (false negatives) and some spurious objects being considered as real (false positives). Techniques based on the edge detection are much more accurate. However, they tend to fail at low-SNRs also due to brightness variations along the trail [6]. The widely used point spread function (PSF) fitting technique seems to be the most robust and versatile way to obtain the accurate trail positions. Nevertheless, some specific issues may make PSF fitting fail or accuracy decreases, such as (1) trails overlapping; (2) trails under-sampled; (3) trails low-SNRs or (4) trails surrounded by a rapidly changing background [13]. Furthermore, PSF fitting method is more time consuming, which limits its application in real-time image processing.

After examining our observations carefully, we found, as illustrated by Figure 2, same trails in continuous observations takes on different shapes. The shapes are far from the PSF profile proposed by some researchers [6, 14, 5] , and there is no apparent pattern of shape change with time. But within a single frame, all trails share similar shapes. It could be that for a telescope with small field of view, the atmosphere turbulence, extinction fluctuation and telescope jitter have same effect on all stars. Inspired by this situation, we adopt Template Matching to gain the center of trails, even the trails are seriously distorted.

Template Matching is a technique in digital image processing for finding small parts of an image which match a template image [10]. In this method, we cut a part of the actual observation image containing a bright trail as a template. Then, Template Matching was performed to identify parts the actual image that match the predefined template. Normalized cross-correlation was taken as the similarity measurement. Positions of local maximum value stronger than specified threshold value in correlation image represents the occurrences of template within the initial image. Meanwhile, the trail in template image was centered with thresholding. The position of the trail in the template and the position of the template in the original image determine the position of the trail in the original image. A large number of simulation and measured images were tested, and the result show the performance of Template Matching is good.

This short paper is organized as follows: Section 2 redefine the trailing model with considering the tracking error of telescope. Section 3 introduces the concepts behind an Template Matching which the reader should be familiar with. Section 4 show some experiments result. Hundreds of real and synthetic images were processed with Template Matching and thresholding. More discussion and a short conclusion are presented in Section 5.

2. TRAILING MODEL

An object moving with a apparent angular rate of motion in a CCD image in an exposure of T leaves a trail. Vere P. approximate the trail as the convolution of an axisymmetric Gaussian PSF with σ moving at a constant rate in a direction [14]. But as we can see from Figure 2, the target's motion is not constant. The mechanical instability of the telescope or tracking error should be considered. In this section, we redefine trailing model.

Within the exposure time T, the center of the star move in CCD image of size $M \times N$ at a given speed and a random error. Assuming the initial time and position is t_0 and (x_0, y_0), respectively. In a short time interval Δt, the position of star image moves from (x_{n-1}, y_{n-1}) to (x_n, y_n). The relation between two points is given:

$$x_n = x_{n-1} + \Delta x + e_x \quad (1)$$

$$y_n = y_{n-1} + \Delta y + e_y \quad (2)$$

where Δx and Δy is the displacement along x and y direction in time duration Δt. The e_x and e_y are tracking error along x and y direction. In this work, we assume e_x and e_y are random variable that obeys a normal distribution.

$$e_x \sim N(0, \sigma_x^2) \quad (3)$$

$$e_y \sim N(0, \sigma_y^2) \quad (4)$$

Within Δt, the flux of star is given by A. Without considering the atmosphere turbulence, we assign A to a single pixel $([x_n], [y_n])$. $[*]$ represents the integer part of $*$. The trail within exposure time T can be represented by a set of points:

$$trail = \{([x_n], [y_n]) \mid n = 1, 2 \ldots N, N = \frac{T}{\Delta t}\} \quad (5)$$

The flux from star in each pixel is:

$$S(x, y) = n \cdot A \quad (6)$$

where $x = 1, 2 \ldots M; y = 1, 2 \ldots N$. and n is the number of times the coordinate $([x_n], [y_n])$ falls into the coordinate (x, y). Because of the wobble of the atmosphere, flux in trail were diffused by 2-D Gaussian function $g(x, y)$. Let's say the sky background is $B(x, y)$, the image represented by:

$$I(x, y) = g(x, y) * S(x, y) + B(x, y) \quad (7)$$

$$g(x, y) = \frac{1}{2\pi\sigma^2} exp[-\frac{x^2 + y^2}{2\sigma^2}] \quad (8)$$

In most astronomical images the background is dominated by the sky and over a small region near a trail. In the absence of Poisson noise the flux at a given pixel is

$$I^*(x, y) = I(x, y) + rand(1)\sqrt{I(x, y)} \quad (9)$$

where $rand(1)$ is a random number from the normal distribution with centroid 0 and width 1.

3. TEMPLATE MATCHING

Template Matching is an general-purpose technique of object localization, which allows to identify parts of an

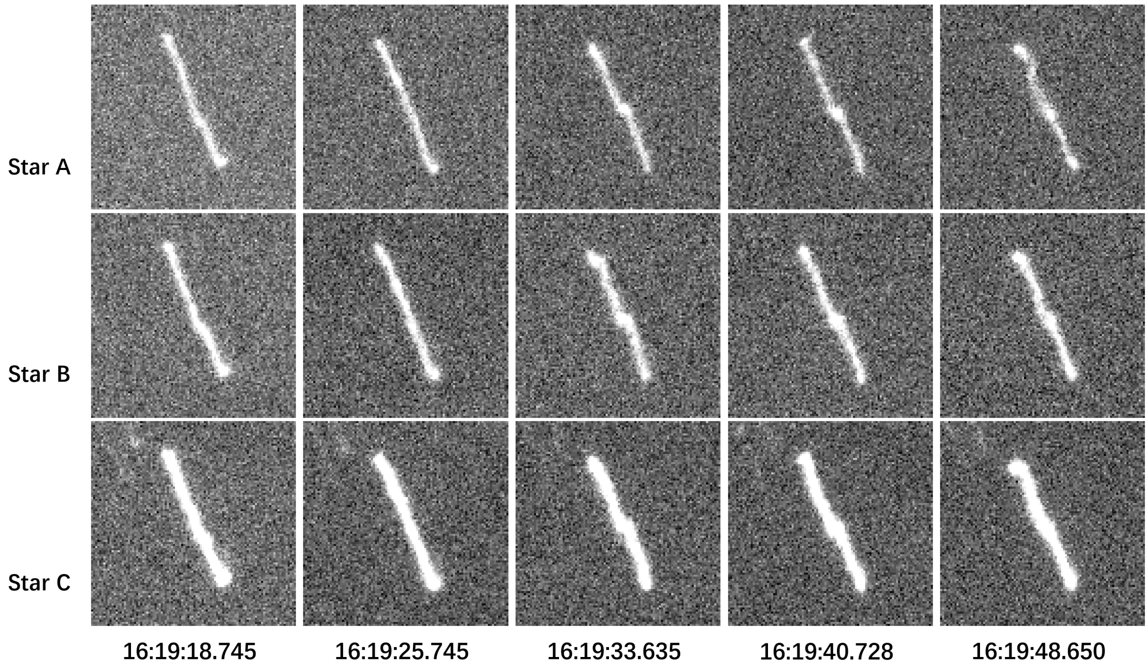

Figure 2. Screenshots of three trailed background stars in five consecutive images. All five images were obtained by 1m telescope of Weihai Observatory of Shandong University on 31, May, 2018. Exposure time is 2 second. Timestamps of mid-exposure were showed corresponds to every image.

image that match, under some criterion of similarity, an arbitrarily chosen image template. It is widely used in manufacturing as a part of quality control [1], a way to navigate a mobile robot [8], or as a way to detect edges in images [3]. As illustrated by Figure 3, the matching problem can be summarized as follows: considering the source image I and the reference image R, find the offset (u, v) within the search region S such that the similarity between the shifted reference image $R_{u,v}$ and the corresponding sub-image of I is the maximum [4]. In general,

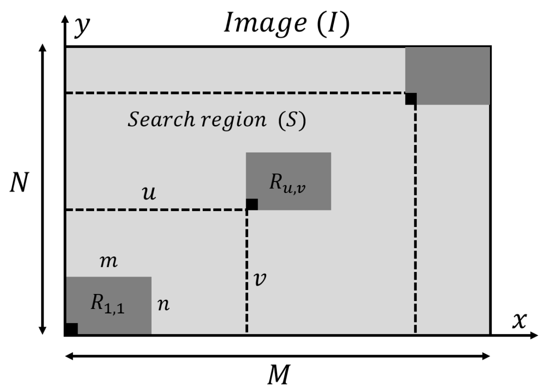

Figure 3. Template Matching geometry: the reference image R is shifted across the search image I by an offset (u, v) by using the origins of the two images as reference points. The dimensions of the source image $(M \times N)$ and the reference image $(m \times n)$ determine the maximal search region (S) for the comparison.

Template Matching involves two critical points: the similarity measurement and the search strategy. But based on our particular research objectives, template creation, matches identify should be considered in detail. Each step is described as following.

3.1. Template Creation

A well-prepared template is the key to successful Template Matching. The term $template$ is used in everyday language, but recalling the definitions more closely related to their technical meaning is useful. A template is a typical model or representative instance you want to find in an image. In this work, a template should be a rectangle image region containing a typical trail. The rectangle is parallel to the original image in the X and Y directions. Although the rectangle's size increases significantly when the trail away from the horizontal or vertical direction, rectangle is the simplest representation. Even so, the rectangle should be as small as possible to reduce the computation. Template Matching is time consuming, so the pixels within template out of trail can be set to 0, which can reduce the computation and avoid the surrounding targets. On the other hand, the trail should be bright, unsaturated and isolated, which means overlapping or contaminate with other trails, target or cosmic ray should be avoided. The template can be created through thresholding method to realize automatic processing (Figure 4).

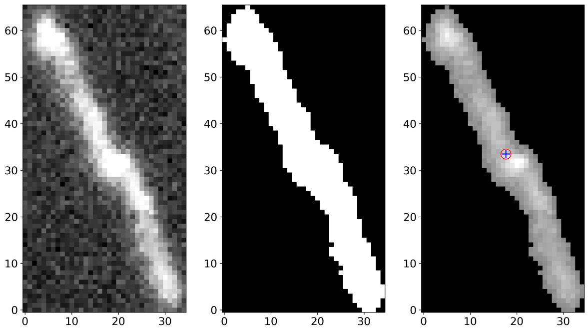

Figure 4. Screenshots of template. Left: a region cut from input image containing a typical trail. Center:mask image generated with thresholding and Right: a template we want.

3.2. Similarity measurement

The term $matching$ means comparing in respect of similarity or to examine the likeness or difference of two objects. The most important part of Template Matching is the specific measure of image similarity that will be used to evaluate possible matches. Several metrics have been proposed to evaluate the matching between two images, the most important are: sum of absolute differences (SAD), sum of squared differences (SSD) and the normalized cross-correlation (NCC). The most used matching criterion is the NCC coefficient which is invariant to linear changes in image brightness or image contrast. The NCC value between a given image I of size $M \times N$ and a template image R of size $m \times n$, at the displacement (u, v), is given by:

$$\gamma_{u,v} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}[I_{u,v}(i,j) - \bar{I}_{u,v}][R(i,j) - \bar{R}]}{\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}[I_{u,v}(i,j) - \bar{I}_{u,v}]^2}\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}[R(i,j) - \bar{R}]^2}} \tag{10}$$

where $\bar{I}_{u,v}$ is the greyscale average intensity of the source image for the coincident region of template image R whereas $\bar{R}$ is the greyscale average intensity of the template image. These values are defined as follows:

$$\bar{I}_{u,v} = \frac{1}{m \cdot n}\sum_{i=1}^{m}\sum_{j=1}^{n} I(u+i, v+j) \tag{11}$$

$$\bar{R} = \frac{1}{m \cdot n}\sum_{i=1}^{m}\sum_{j=1}^{n} R(i,j) \tag{12}$$

A example correlation image is shown as Figure 5.

3.3. Matches Identify

If we are supposed to find the template occurrences, we need to specify what does it mean that a template occurs at some position in an image. The point (u, v) that

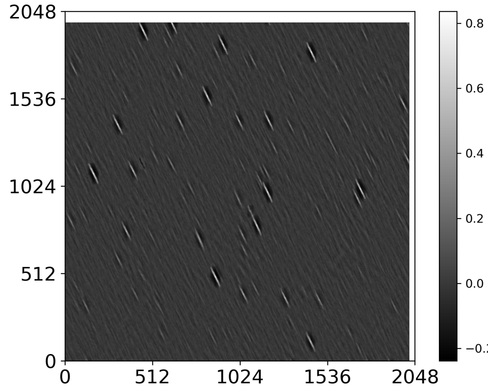

Figure 5. An example of correlation image. This image is generated by Template Matching between source image (Figure 1) and tempalte image (Figure 4).

presents the best possible resemblance between R and I is thus defined as follows:

$$(u, v)_{max} = \arg\max_{(u,v)\in S} \gamma_{u,v} \tag{13}$$

where $S = \{(u, v) \mid 1 \leq u \leq M - m, 1 \leq v \leq N - n\}$. Suppose you are searching for an object which has multiple occurrences, Equation. 13 won't give you all the locations. In that case, we will use local maximum. The points $(u, v)_{max}$ presents the possible resemblance between R and I, which given by:

$$(u, v)_{max} = \arg\max_{(u,v)\in S^*} \gamma_{u,v} \tag{14}$$

where S^* is the the local region in which to find the maximum.

4. RESULT

As illustrated by Figure 2, the trails were distorted seriously. The techniques based on PSF fitting are not applicable. Therefore, we measured the performance of the Template Matching compared to thresholding and the barycenter of trails were taken as the metric. The principle and implementation of thresholding are described in detail by Bertin E. [2]. In this work, the thresholding was done by Photutils[2], which is an open source (BSD licensed) Python package. Photutils includes a general-use function to detect sources (both point-like and extended) in an image using a process called image segmentation in the computer vision field. After detecting sources using image segmentation, we can then measure their photometry, centroids, and morphological properties by using additional tools in Photutils. On the other hand, Template Matching was carried out by OpenCv-Python[3], which combines the best qualities of OpenCV C++ API and

[2] https://photutils.readthedocs.io/en/stable/
[3] https://opencv-python-tutroals.readthedocs.io/en/latest/#

Python language. OpenCV-Python comes with a function cv2.matchTemplate() for Template Matching.

To compare the two methods roughly, several real observation images were proceeded with thresholding and Template Matching. An effective bias subtraction and flat correction have been performed. All parameters of the two methods were fine-tuned to achieve the best result. Threshold levels were carefully selected to balance the two types of error, false negatives and false positives. A screenshot of result is shown in Figure 6. All bright trails were detected and showed by red circle. However, some trails split into several parts. The result of Template Matching were showed by blue plus. Template Matching seems to get more and fainter trails. Since the real location of trails are not known, it is difficult to evaluate the accuracy of the both methods.

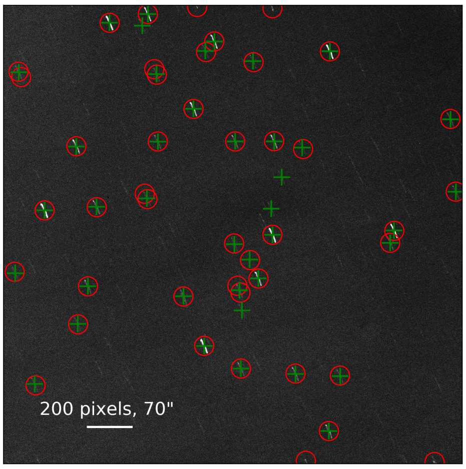

Figure 6. Template Matching result (blue plus) and threholding result (red circle). Template Matching seems to get more and fainter trails.

The synthetic trails were created using trailing model described in Section 2 to mimic the real trails, which allow us to control the exact flux, position and length of the trails. Then, we can determine how well the thresholding and Template Matching procedure reproduce the generated values. Sixty trails of the same shape and different flux compose a set, which corresponding to a real observation image. The parameters of simulation are list in Table 1. Three synthetic trails within same set are shown in Figure 7, where the only difference between them is their SNR. The faintest trail at SNR = 10 is only visible as a faint smudge and the eye is guided to it because we know that the trail lies at the center of that image.

In the following we quantify the Template Matching algorithms performance as a function of SNR of the trails. The signal S is the integrated flux of the source as measured within 3σ around the trail. The background B is the integrated total background flux within the same region including readout noise, dark current, sky, and other sources of flux that are not due to the trailed object.

Table 1. The parameters of simulated images.

Parameters	Value
Size of image	128×128
Step length in x direction (pixel)	0.5
Step length in y direction (pixel)	0.5
Step length error in x direction (pixel)	0.5
Step length error in y direction (pixel)	0.5
Flux from star in time interval (ADU)	$10 \sim 600$
Tracking step number	60, 90, 120
Background level (ADU)	300
Sigma of PSF (pixel)	2
Noise type	Poisson

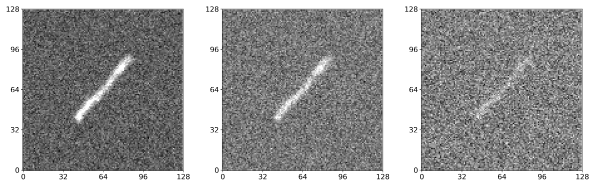

Figure 7. Three synthetic trails with same shape. From left to right, the SNR of trails are 50, 30, 10.

We measured the Template Matching performance compare to thresholding on 1000 set synthetic image at three different step number 60, 90 and 120. The corresponding length of trails are about 40, 60 and 80 pixel. Figure 8 show that both astrometric error and uncertainly become smaller as SNR increase. The thresholding yields higher astrometric errors and uncertainly than Template Matching. And thresholding is seem to breaks down for low SNR sooner than Template Matching. Increasing the length of trails will significantly increase the astrometric error and uncertainly of theresholding, but has little effect on Template Matching.

5. DISCUSSION AND CONCLUSION

We provide the analytic form for a space debris trailing model assuming that a symmetrical Gaussian PSF is moving at a given constant speed plus a random error, which could be due to tracking error, atmosphere turbulence or extinction fluctuation. These trails are no longer suitable for PSF fitting methods. We adopted a technique based on Template Matching and tested its performance compared to thresholding. The results are surprising. Template Matching is more accurate and robust than thresholding, which means that Template Matching is more suitable for fainter and longer trails extraction. Nevertheless, more trails with different distortion, different orientation or surrounded by other objects should be tested in further. Better template generation method and more effective search strategy of Template Matching should be studied in future.

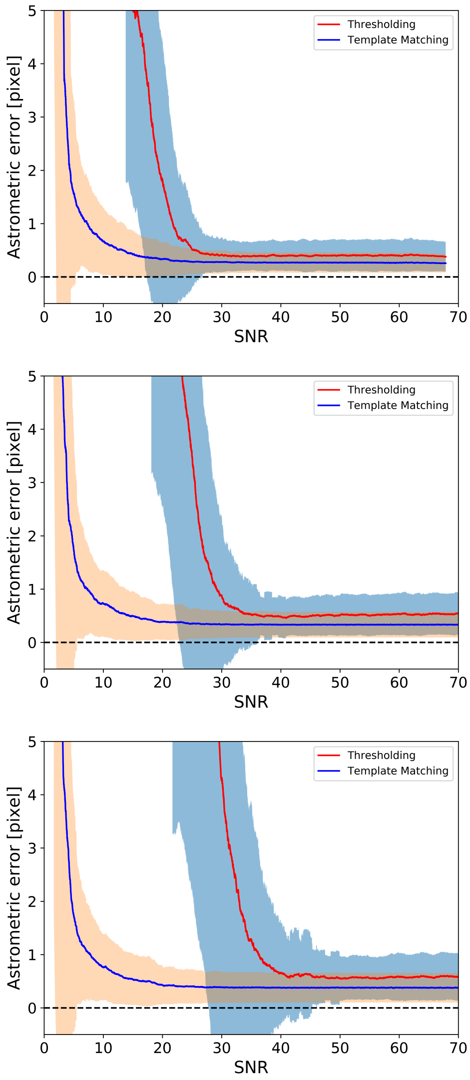

Figure 8. Moving average of the astrometric error for Template Matching and thresholding as a function of the trail's SNR. SNR were calculated within 3σ around the trail. The shadows are the standard error on the mean. Top, middle and bottom figure are three different simulation step number: 60, 90 and 120, corresponding trail length about 40, 60 and 80 pixel.

ACKNOWLEDGMENTS

This work is supported by the Natural Science Foundation of Shandong province (No.JQ201702) and the Young Scholars Program of Shandong University (No.20820162003)

REFERENCES

1. Aksoy M. S., Torkul O., Cedimoglu I. H. (2004). An Industrial Visual Inspection System That Uses Inductive Learning. Journal of Intelligent Manufacturing, 15(4), 569–574.
2. Bertin E., Arnouts S.,(1996). SExtractor: Software for Source Extraction, Astronomy & Astrophysics Supplement Series, 117(2): 393-404.
3. Ching Yang Wang,(1985). Edge Detection Using Template Matching (Image Processing, Threshold Logic, Analysis, Filters), Duke University, 1985.
4. Erik C., Valentín O., Diego O.,(2017). Evolutionary Computation Techniques: A Comparative Perspective, Springer.
5. Fraser W., Alexandersen M., Schwamb M. E., et al., (2016). TRIPPy: Trailed Image Photometry in Python, The Astronomical Journal, 151(6), 158.
6. Kouprianov, V.,(2008). Distinguishing Features of Ccd Astrometry of Faint Geo Objects, Advances in Space Research, 41(7),1029-1038.
7. Kouprianov V.,(2010). Advanced Image Processing Techniques for Automatic Reduction of Geo Survey Data, 8th Us-Russian Space Surveillance Workshop.
8. Kyriacou T., Bugmann G., Lauria S.,(2005). Vision-Based Urban Navigation Procedures for Verbally Instructed Robots, Robotics and Autonomous Systems, 51(1),69-80.
9. Masias M., Freixenet J., Lladó X., et al., (2012). a Review of Source Detection Approaches in Astronomical Images, Monthly Notices of the Royal Astronomical Society, 422(2), 1674–1689.
10. Roberto B.,(2009). Template Matching Techniques in Computer Vision, Wiley.
11. Schildknecht T.,(2007). Optical Surveys for Space Debris, Astronomy and Astrophysics Review, 14(1), 41-111.
12. Seitzer P., Smith R., Africano J., et al.,(2004). Modest Observations of Space Debris at Geosynchronous Orbit, Advances in Space Research, 34(5), 1139–1142.
13. Sun, R. and Jia P.,(2017). Image Restoration Based on Statistic Psf Modeling for Improving the Astrometry of Space Debris, Publications of the Astronomical Society of the Pacific, 129(974).
14. Vereš P., Jedicke R., Denneau L., et al, (2012). Improved Asteroid Astrometry and Photometry With Trail Fitting, Publications of the Astronomical Society of the Pacific, 124(921), 1197–1207.
15. Virtanen J., Poikonen J., Säntti T., et al.,(2016). Streak Detection and Analysis Pipeline for Space-Debris Optical Images, Advances in Space Research, 57(8), 1607–1623.
16. Chris Zingarelli J., Pearce E., Lambour R., et al., (2014). Improving the Space Surveillance Telescope's Performance Using Multi-Hypothesis Testing, The Astronomical Journal, 147(5), 111.