SYNTHETIC TRACKING IN SST AND NEO SEARCHES

Paweł Białobłocki((1,2)), **Krzysztof Kamiński**((2)), **and Tomasz Kluwak**((1,3))

(1)*ITTI Sp. z o.o., ul. Rubież 46, 61-612 Poznań, Poland , Email:* `space@itti.com.pl`
(2)*Adam Mickiewicz University, Faculty of Physics, Umultowska 85, 61-614 Poznań, Poland , Email:* `obserwatorium.astro@amu.edu.pl`
(3)*Lusówko Platanus Observatory, 62-080 Lusówko, Poland , Email:* `tomasz.kluwak@platanus.pl`

ABSTRACT

In this paper, we present our improvements to the synthetic tracking algorithm. We aim to extend its use to detect not only point-like objects but also streaks, taking into account their potential non-linear movement. Moreover, we include multiple usage scenarios, such as tracking known objects, following up on objects with uncertain ephemeris, and surveying unknown objects. We present the initial results of our approach using real observational images of artificial satellites. For every stage of the algorithm, we describe the implementation techniques used, along with ideas for future development.

Keywords: Synthetic Tracking; SST; NEO; Optical Observations; Signal-to-noise ratio, Image stacking, Astrometry, Photometry.

1. INTRODUCTION

Synthetic tracking (ST) is a relatively new technique, based on well-established concepts, that was originally implemented for detection of faint asteroids [1]. The method acquires multiple short-exposure images to prevent target elongation due to proper motion. These images are subsequently aligned and stacked using shifts based on an assumed velocity vector, which significantly improves the target's signal-to-noise ratio (SNR) while reducing SNR for all objects not co-moving with the target. This technique replicates numerically the benefits of long-exposure imaging with a non-sidereal tracking telescope for observations of known objects, at the penalty of accumulating more read noise in the stacked image. With modern low-noise cameras this is usually acceptable and often does not increase the overall noise by much. Synthetic tracking can also be used for surveys by multiple trials and errors of various proper motions. This approach requires much higher computational power ant therefore is often limited to relatively low number of images and restricted parameter space.

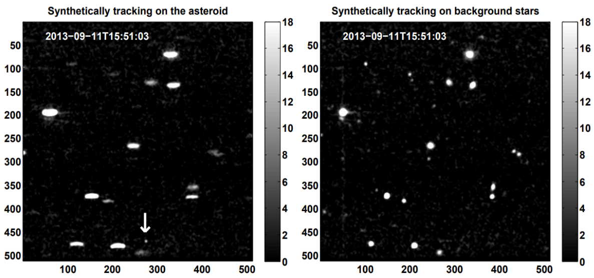

Figure 1. Example of synthetic tracking result for a faint asteroid (left) and corresponding long exposure sidereal tracking image (right). [2]

2. SYNTHETIC TRACKING IN THE SST DOMAIN

Adaptation of synthetic tracking technique to optical Space Surveillance and Tracking (SST) observations is not trivial. In the case of tracking observations of Low Earth Orbit (LEO) targets the proper motions are typically in the range of 0.2-1.5 deg/s which requires a very short exposures to record non-trailed images. This limits the number of reference stars and may result in an excessive amount of imaging data to process. In the case of survey observations it is worth noting that the parameter space to search might be large because of the vide range of proper motions and the fact that for observations with field of view (FoV) larger than about 1 deg the nonlinearity of satellite's motion (both: direction changes and angular velocity changes) is becoming detectable.

In this paper we present our updated implementation of synthetic tracking algorithm that addresses challenges inherent to satellite observations. In contrast to earlier implementations that assumed the target is recorded as a non-elongated point source our approach effectively processes both point-like and elongated images assuming that the length of the trail does not change too much. This enables much more flexible observation planning without the need to use exposure time corresponding to the highest expected target's angular speed. Additionally our algorithm is able to correct for changes in the direction of motion by rotation images before stacking. This enables to combine images even from distant locations in the sky. Moreover in the case of survey observations we implemented a dual stage approach. Initially we search the parameter space for different constant velocities and directions only. After detecting a target we relax the parameter space slightly to include also a non-linear and accelerated motion. This second stage is speed up by cropping images to the regions where and objects is found and using only limited range of possible parameters. It also serves as an attempt to improve the quality of the final stack. By using smaller than initially steps in velocity and direction we search for a maximum SNR in the stacked image.

In our approach to the ST survey we also utilize the knowledge about whether an object would be recorded as a trail or point at a particular assumed velocity. When target is trailed we use larger steps, because during initial search we only need parts of the trail to overlap for the target to be detected.

In our ST algorithm we also implemented astrometry, photometry and uncertainty estimation routine. Astrometry relies on centroid in the stacked image and WCS (World Coordinate System) in the input images. Photometry is made by summing the background subtracted pixels of a target. The extend of pixels is estimated from the stacked image but the measurements are implemented in each individual image. Uncertainty is estimated using bootstrapping technique [3] by repeating the last stage of stacking and statistical analysis of the scatter of the results. In the following sections we describe details of our approach and present initial results of several of them.

3. OVERVIEW OF THE ALGORITHM

We are currently working on the first version of the algorithm which focuses on basic handling functionalities and is largely dependent on the input files provided by the user. At current stage, the system assumes that the data supplied are of sufficient quality, already reduced and plate-solved, ready for ST analysis. Processing is carried out using built-in algorithms that operate in distinct modes, as described below.

The proposed extended synthetic tracking routine is composed of six main stages, which are described in detail below. Each stage is designed to be repeatable, with most relying solely on the output from the previous stage and on user inputs such as a configuration file or ephemeris data.

The algorithm already has two operating modes implemented and one mode planned for future extension:

1. **Tracking** — single stack of images when exact object ephemeris is known
2. **Follow-up** — stack the set of images around uncertain ephemeris (not implemented in the current version)

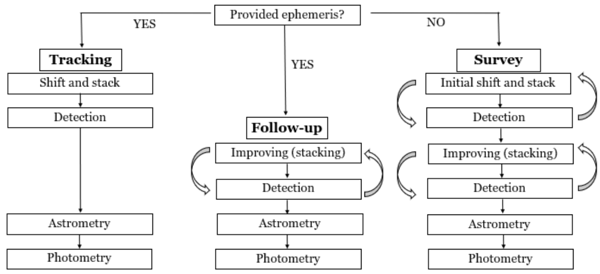

Figure 2. Algorithm working scheme for every method

3. **Survey** — stack the set of the images with multiple different parameters of sky motion

Tracking mode can be used for example in maintaining orbital catalogue of very faint known targets. It also finds its application when a constant shift of observed object position with respect to the ephemeris is present. Follow-up mode is intended for situations where an orbit is uncertain but similar to the ephemeris. It can be applied for detection of new debris after satellite fragmentation or after orbital maneuver. Survey mode can be used to detect unknown objects for example in GEO belt searches.

Since plate solving is not the primary focus of this initial version of the algorithm, it is assumed that the user supplies input FITS (Flexible Image Transport System) files with at least a first-order WCS solution already determined and stored in each file's header; higher-order solutions may optionally be included using SIP (Simple Imaging Polynomials) functions. A constant exposure time across all files is also required, and it is assumed that the target's FWHM (full width at half maximum) — whether trailed or point-like—remains relatively unchanged during the observations. While FITS images used for tracking and follow-up analysis may cover different areas of the celestial sphere, survey analysis requires all images to cover the same region with only minimal drift between them. Moreover, in survey analysis, the target's trajectory can be assumed to deviate slightly from linear motion. This is important for field of view larger than about 1 deg, when non-linearity in artificial satellite's motion becomes observable.

4. DETAILED DESCRIPTION OF ALGORITHM

4.1. Data Format Requirements

The frames provided by the user must meet two basic conditions:

1. They must be in FITS format, a widely used standard for storing astronomical data, including images and associated metadata [4].

2. They must include a WCS header, which defines standard methods to map pixel coordinates onto various astronomical reference frames [5].

For the synthetic tracking algorithm to function correctly, each FITS frame must satisfy the following requirements:

- `NAXIS`: The dimension of each image must be constant
- `BITPIX`: The pixel data type must be uniform (commonly `16` for 16-bit integer or `-32` for 32-bit floating-point).
- `EXPOSURE`: The exposure time must be identical for all frames to ensure consistent stacking.

In addition, accurate WCS keywords must be present in each FITS header to ensure proper astrometric calibration and alignment. They include `CRPIXn` (reference pixel), `CRVALn` (world coordinate value at the reference pixel), `CDELTn` (scale per pixel), and `CTYPE` (coordinate type and projection). This mapping is critical for aligning astronomical images, overlaying catalogs, and performing precise astrometric calibrations. Higher-order corrections, such as those provided by the Simple Imaging Polynomial (SIP) convention, can be included to correct for optical distortions. Additionally, the CD matrix (coordinate description matrix) offers an alternative to using `CDELTn` and a separate rotation matrix; it directly encodes the linear transformation—encompassing scale, rotation, and skew—between pixel and intermediate world coordinates [5].

```
WCSAXES =                    2
CTYPE1  = 'RA---TAN-SIP'
CTYPE2  = 'DEC--TAN-SIP'
EQUINOX =               2000.0
LONPOLE =                180.0
LATPOLE =                  0.0
CRVAL1  =        311.547159631
CRVAL2  =        17.0390731477
CRPIX1  =        1218.21746826
CRPIX2  =        1221.39876302
CUNIT1  = 'deg     '
CUNIT2  = 'deg     '
CD1_1   =   -0.000610514278009
CD1_2   =     0.00105407941364
CD2_1   =    -0.00105449878325
CD2_2   =   -0.000610467790349
```

4.2. Stage 1: Shift and Stack Algorithm

Each operating mode within the shift and stack algorithm employs a distinct image stacking strategy, primarily determined by whether the ephemeris of the target objects in the captured frames is known. When this ephemeris data is available, the object's proper motion can be calculated directly from the WCS header embedded in the FITS files. This enables the use of **Tracking Mode**, where the stacking process follows the known trajectory of the object across multiple frames. Conversely, in the absence of prior information—such as when the FITS images are obtained during wide-field surveys without predefined targets—the only applicable method is the **Survey Mode**, which performs a blind search for moving objects.

Regardless of the selected mode, several core procedures remain consistent. Only one operational mode—tracking or survey - is active during processing. Image shifts are computed in pixel coordinates (X, Y) rather than in celestial coordinates (RA, Dec), a choice driven by the practicality of pixel-based operations in Survey Mode and the ease of implementation following image rectification.

The calculated shifts are applied by modifying the reference pixel positions in the WCS header (`CRPIX1`, `CRPIX2`). Any required image rotations are introduced through alterations to the CD matrix

$$\begin{bmatrix} CD1\_1 & CD1\_2 \\ CD2\_1 & CD2\_2 \end{bmatrix}$$

which governs the image orientation in the WCS. Depending on the analysis type, the software prepares pixel-space trajectories for potential objects, using parameters such as the FITS mid-exposure time, object coordinates $(x_{\text{obj}}, y_{\text{obj}})$, velocities $(v_{x,\text{obj}}, v_{y,\text{obj}})$, and angular direction (θ_{obj}). These trajectories form the basis for guiding the image alignment and stacking operations across the dataset.

Crucial to this process are several parameters. One key parameter is `quant_prob`, a pixel averaging quantile value ranging from 0.0 to 1.0 (with 0.5 corresponding to the median), which determines the representative pixel value during the stacking procedure. Thanks to these measures of central tendency, the correct combination of the frames keeps the trail clearly visible while erasing unwanted background stars. In addition, the configuration file specifies the pixel interpolation method, allowing a choice among `nearest-neighbor`, `bilinear`, `biquadratic`, `cubic`, `adaptive`, and `exact` [6]. It is very important to interpolate the combined pixel values because, as the pixels nearly ever always overlay perfectly. The selection of methods used to determine the value of the combined pixels allow to balance between efficiency and accuracy of the transformations.

Example of the raw image of the satellite with a 4-second exposure creating the characteristic trail due to its movement relative to the stars, which are tracked by the telescope using the sidereal rate:

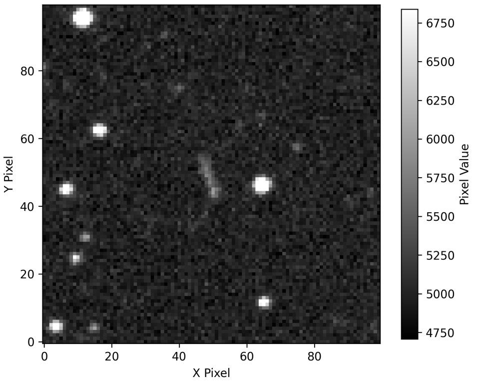

Figure 3. Example of the raw streak (part of full frame)

The stacked image is produced using the median procedure and biquadratic pixel interpolation. In this processed image, the satellite is clearly visible, while the noise around it appears approximately flat and the stars are no longer visible.

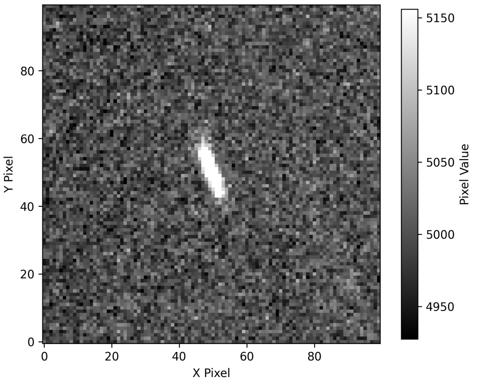

Figure 4. Example of stacked streak (part of full frame)

Tracking Mode

Tracking mode is designed for objects with well-known ephemerides. By using precise orbital data, the algorithm predicts the object's position in subsequent images. This enables the algorithm to shift and rotate multiple frames into a single composite image, enhancing the signal-to-noise ratio of the target object. In addition to verifying ephemeris positions, the algorithm also measures small deviations over time, ensuring continuous and updated monitoring, which is crucial for Space Situational Awareness (SSA).

It is important to note that when an object's position is initially unknown, Tracking mode cannot be applied. However, once the ephemeris is determined through alternative methods or external sources, the system can continuously update and refine the object's tracking. Compared to Survey mode, Tracking mode is significantly faster and produces more current and accurate ephemerides.

Moreover, the procedure can be performed only one time because of the well-known trajectory of the object, so after the stacking procedure the detection process can be applied immediately.

Survey Mode

In Survey mode, the primary goal is to identify objects in images without any prior information about their existence or location. This approach is highly demanding on memory resources because it involves performing thousands of frame manipulations to search for the optimal alignment. The strength of this exhaustive search lies in its ability to detect objects that were not anticipated or cataloged.

The main goal is to optimize the calculation time and precision of the stacked images. Before the process begins, the user must provide several key parameters: minimum and maximum velocity [pix/s], minimum and maximum angle [deg], and the granulation of both speed and angle.

The use of granulation, as opposed to fixed steps, is particularly important in cases where the minimum velocity is significantly smaller than the maximum velocity ($v_{\text{min}} \ll v_{\text{max}}$). Using a constant step size under such conditions is inefficient: for small velocities, the step size corresponds to a large fractional change, while for higher velocities, it becomes negligible and has little effect. Granulation addresses this by maintaining a consistent fractional increment between successive velocity and angular values during the stacking procedure, ensuring both efficiency and precision.

4.3. Stage 2: Detection

After the frames have been successfully and accurately aligned and stacked, the detection stage begins. The user either selects the specific stacked frames to be analyzed or uses all the stacked images, accompanied by text files containing essential information about the assumed object's trajectory used during stacking. This trajectory data is crucial for generating a synthetic satellite trace, which the detection algorithm will use to identify and characterize potential satellite streaks.

To accomplish this detection, a 2D correlation is performed between each stacked image and a synthetically generated target's trail. The width of this trail corresponds to the FWHM of PSF in the image. The length of the trail is calculated based on the exposure time and assumed target's velocity. This way, the correlation operation highlights any signal that closely resembles the anticipated shape, allowing the algorithm to robustly detect and characterize the presence of satellite streaks. The trail is assumed to have constant brightness throughout its length and 2D Gaussian profile at its ends.

With the input data and trajectory information prepared, the first step is to estimate and subtract the background on the stacked images. This is accomplished by applying a median filter, using an appropriately sized footprint matrix, to isolate and remove any residual background signal. Once this background has been removed, the next task is to perform a correlation using the synthetic trail as a template.

Following the correlation, the median and the scatter of the background are determined for the entire correlation map. This involves sampling a large number of randomly distributed points across the image—while discarding the most extreme one percent of values—to obtain a reliable estimate of the background's sigma. Subtracting the previously calculated median and dividing by sigma transforms the correlation map into an SNR (signal-to-noise ratio) map, which can be easily used for thresholding and target's detection.

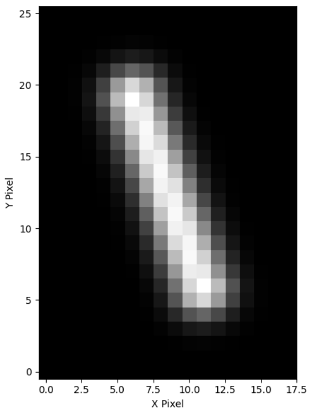

Figure 5. Example of synthetic streak used for correlation process.

Once the SNR map has been created pixels above the selected threshold are chosen, effectively singling out the significant detections from the noise.

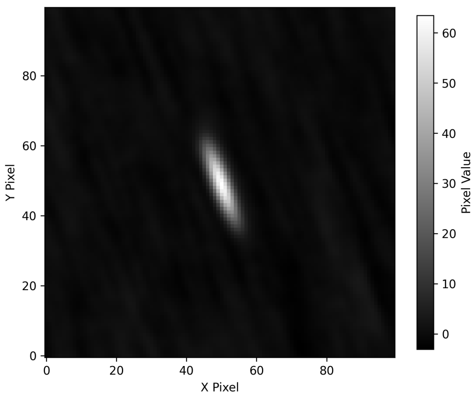

Figure 6. Example fragment of SNR map with clearly visible single trail detection.

Along with the transformed correlation map, a text file containing the relevant parameters and candidate information is also provided. Additionally, for every detected candidate, the raw streak is cropped and saved as an independent FITS frame. Along with the transformed correlation map, a text file containing the relevant parameters and candidate information is also provided. An example of its contents is shown below:

```
velocity: 3.25
angle: 109.75
mid_time_min: ’2024-10-17T18:44:43.941’
mid_time_max: ’2024-10-17T18:46:14.538’
CRPIX1: 1799.0548970760033
CRPIX2: 1095.6171446189983

config_data:
    correlation:
        correlation_threshold: 10
        FWHM: 2.6
        min_connected_pixels: 25
        max_connected_pixels: 1000

candidates:
-   number_of_pixels: 114
    average_pixel_value:
    20.66995497017395
    max_pixel_correlation_value:
    36.2345397094228
    x_init: 1514
    y_init: 507
    x_peak: 1513.7451626795864
    y_peak: 507.26978869476767
    file_name: candidates_1.fits
    label: 1
```

All high SNR pixels are segmented and each group is analysed as a potential detection. Centroid of the trail (x_peak, y_peak) is calculated using a center of mass (COF) algorithm, which estimates the location of the streak by averaging pixel positions weighted with their intensity. Specifically, for a set of pixels within the trail, if I_i is the intensity of the i-th pixel, and (x_i, y_i) its coordinates, then:

$$x_{\mathrm{peak}} = \frac{\sum(I_i \cdot x_i)}{\sum I_i}, \quad y_{\mathrm{peak}} = \frac{\sum(I_i \cdot y_i)}{\sum I_i}.$$

This measure of the streak's geometric center, accounting for pixel brightness variations across the trail, is essential for accurate final results and plays a critical role in subsequent algorithm steps.

4.4. Stage 3: Improving Method – Final Stacking:

This stage is intended specifically for the Survey and Follow-up mode. For objects with large parameter space to search, the initial velocity and direction step size is usually as high as possible to reduce calculation time at the risk of imperfect stacking result. To improve the results another round of shift and stack is performed during the Final Stacking stage. During this stage the steps sizes are reduced, parameter space is constrained and optionally additional parameters are allowed. These include targets acceleration or trajectory bend. Additionally images are cropped to significantly increase computation speed.

This procedure extends the techniques used in the initial stages. It involves creating a large number of image configurations by varying only the regions where potential objects were identified earlier. This focused approach allows for even more precise matching and optimization of the results.

Below is a set of key parameters considered in the stacking procedure, many of which represent higher-order derivatives of an object's position with respect to time:

- Velocity (min, max, step) [pix/s]
- Acceleration (min, max, step) [pix/s^2]
- Jerk (min, max, step) [pix/s^3]
- Angle change (min, max, step) [°]
- Trajectory shape change (min, max, step) [pix/s^2]

After each stacking process concludes, the detection stage is executed again. Eventually the case with highest SNR is selected as a results and used for subsequent stages.

4.5. Stage 4 & 5: Uncertainty & Astrometry

Calculating uncertainties is crucial, especially in astrometric analyses. This step is dedicated to estimating the positional uncertainties in the x and y directions. It relies on repeating the stacking process and determining the $(x_{\text{peak}}, y_{\text{peak}})$ coordinates multiple times using randomly selected subsets of FITS images, following a bootstrap resampling approach [3].

The bootstrap method allows the algorithm to assess the stability and variability of the centroid estimation using slightly different image combinations. It captures uncertainties related to centroiding accuracy and image registration, and also partially accounts for secondary effects such as variable aberration (as targets move across the field of view) and variations in the WCS solution due to different reference star selections across the FITS images. However, it does not account for systematic errors or astrometric uncertainties inherent in the WCS headers; independent astrometric calibration could be considered in future algorithm upgrades.

Uncertainties in each coordinate are estimated from the distribution of peak positions across bootstrap iterations using a selected percentile (e.g., the 68% percentile for a 1-σ interval):

$$x_{\text{unc}} = \text{selected\_percentile}(x - x_{\text{peak}}) \quad [\text{pix}]$$
$$y_{\text{unc}} = \text{selected\_percentile}(y - y_{\text{peak}}) \quad [\text{pix}]$$

In the astrometry stage, the algorithm calculates the object's position within the reference frame. It identifies the center of the object and converts its pixel coordinates into equatorial coordinates (RA and Dec). The process involves iterating through all candidate positions identified during detection, determining the peak positions for each raw frame used in the final stacking, and finally converting the pixel positions into astronomical coordinates. In Fig. 7 we present the initial astrometry results of the Lageos 2 satellite using survey mode.

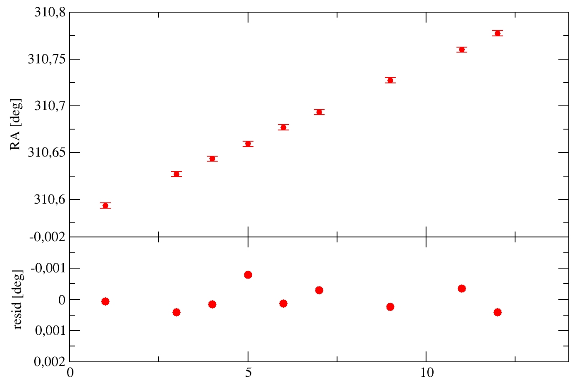

Figure 7. Right ascension measurements and residuals for Lageos 2 observation. The x axis is the frame number.

4.6. Stage 6: Photometry Reduction

The photometry process is used to calculate the brightness of astronomical objects from the FITS images and reference parameters. It calculates instrumental magnitude and optionally uses user provided reference point for the magnitude scale. The algorithm sums background-subtracted values of all pixels belonging to target's image, and then applies a method based on Pogson's formula to derive the instrumental magnitude. For each detected object, relevant details such as the FITS file name, exposure time, mid-exposure time, object sum, and the calculated instrumental magnitude are recorded in a text file. In Fig. 8 the example of instrumental photometry of the Lageos 2 satellite is presented.

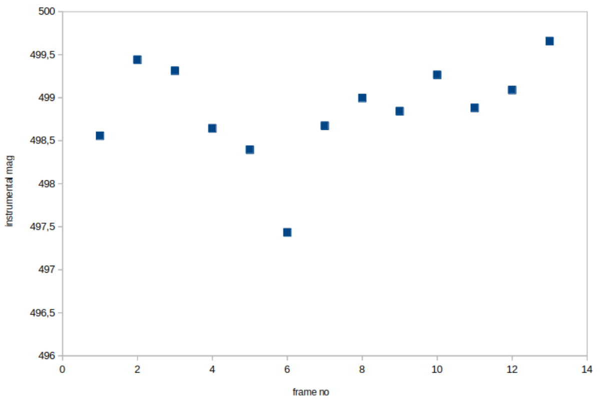

Figure 8. Instrumental photometry for Lageos 2 satellite.

5. FURTHER DEVELOPMENT

The current implementation is a proof-of-concept that does not prioritize computational efficiency. Future versions will be supplemented with Follow-up mode and optimized for production applications by implementing user friendly GUI and API for pipeline integration. Additional preprocessing steps, which have not yet been implemented, will be incorporated to address this gap. These preprocessing routines will correct numerous optical defects that are critical for achieving high-precision stacking and perform independent plate solving. Such enhancements will improve both the reliability and accuracy of the final results. Optimization efforts will also focus on reducing computation time and enhancing scalability. These improvements are essential for meeting the rigorous demands of scientific and industrial applications.

6. CONCLUSION

This project has demonstrated significant potential in the application of the ST method within the SST and very fast NEO domain. In our algorithm we are not limited to the point-like images of the target as well as we include non-linearity in its motion. The development of a technologically mature system could represent a transformative breakthrough in the observation of small, fast-moving, and low-brightness objects. By enhancing both detection sensitivity and overall observational efficiency the overall accessibility for smaller telescopes can be increased enabling a larger number of telescopes to participate in coordinated observations. Continued refinement and optimization will be essential to fully realize these benefits in a production environment.
The project was supported by European Space Agency OSIP grant number: 4000144301.

REFERENCES

1. M. Shao, B. Nemati, C. Zhai, S. G. Turyshev, J. Sandhu, G. W. Hallinan, and L. K. Harding, ”Finding very small near-Earth asteroids using synthetic tracking,” *The Astrophysical Journal*, vol. 782, no. 1, p. 1, 2014, doi:10.1088/0004-637X/782/1/1.
2. C. Zhai, M. Shao, B. Nemati, T. Werne, H. Zhou, S. G. Turyshev, J. Sandhu, G. Hallinan, and L. K. Harding, ”Detection of a faint fast-moving near-Earth asteroid using the synthetic tracking technique,” *The Astrophysical Journal*, vol. 792, no. 1, p. 60, Aug. 2014, doi:10.1088/0004-637X/792/1/60.
3. B. Efron, ”Bootstrap Methods: Another Look at the Jackknife,” *Annals of Statistics*, vol. 7, no. 1, pp. 1–26, January 1979, doi:10.1214/aos/1176344552.
4. NASA FITS and WCS Standards, `https://fits.gsfc.nasa.gov/fits_wcs.html`.
5. M. R. Calabretta and E. W. Greisen, ”Representations of celestial coordinates in FITS,” *Astronomy & Astrophysics*, vol. 395, no. 3, pp. 1077–1122, Dec. 2002, doi:10.1051/0004-6361:20021327 (Received: 24 July 2002; Accepted: 9 September 2002; Published online: 18 November 2002).
6. C. E. Deforest, ”On Re-sampling of Solar Images,” *Solar Physics*, vol. 219, no. 1, pp. 3–23, Jan. 2004, doi:10.1023/B:SOLA.0000021743.24248.b0.