Available online at www.sciencedirect.com

ScienceDirect

Advances in Space Research 64 (2019) 733–746

ADVANCES IN SPACE RESEARCH (a COSPAR publication)

www.elsevier.com/locate/asr

Robust foreground segmentation and image registration for optical detection of GEO objects

Huan N. Do [a], Tat-Jun Chin [a,*], Nicholas Moretti [b], Moriba K. Jah [c], Matthew Tetlow [b]

[a] School of Computer Science, The University of Adelaide, Australia
[b] Inovor Technologies, Australia
[c] Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, United States

Received 8 July 2018; received in revised form 6 March 2019; accepted 9 March 2019
Available online 19 March 2019

Abstract

With the rapid growth in space utilisation, the probability of collisions between space assets and orbital debris also increases substantially. To support the safe utilisation of space and prevent disruptions to satellite-based services, maintaining space situational awareness (SSA) is crucial. A vital first step in achieving SSA is detecting the man-made objects in orbit, such as space-crafts and debris. We focus on the surveillance of Geo-stationary (GEO) orbital band, due to the prevalence of major assets in GEO. Detecting objects in GEO is challenging, due to the objects being significantly distant (hence fainter) and slow moving relative to the observer (e.g., a ground station or an observing satellite). In this paper, we introduce a new detection technique called GP-ICP to detect GEO objects using optical sensors that is applicable for both ground and space-based observations. Our technique is based on mathematically principled methods from computer vision (robust point set registration and line fitting) and machine learning (Gaussian process regression). We demonstrate the superior performance of our technique in detecting objects in GEO.

Keywords: SSA; Space object detection; Geostationary orbit; Gaussian process regression; Robust point set registration; Robust line fitting

1. Introduction

Virtually all major public and private assets, such as transportation hubs, commercial buildings, power stations and cultural monuments, are protected by extensive surveillance networks. Unfortunately the same cannot be said of space assets, such as communications satellites and space stations. Currently there are more than a thousand operating satellites and space installations, amounting to Trillions of dollars worth of investments. Protecting these assets from interference and destruction is of utmost importance. A major risk is collision with other resident space objects (RSO), including debris and satellites, both operational and disused. For instance, the collision between Iridium 33 (then an operational satellite) and Cosmos 2251 introduced over 2000 unregistered pieces of debris into the orbit. Crucially, the collision generated even more debris, which eventually drifted all over the globe due to celestial mechanics (Schildknecht, 2007).

Ground-based observations have been the primary tools for SSA. The U.S. Strategic Command (USSTRATCOM) maintains the Space Surveillance Network (SSN), which is a global network of facilities (mostly ground-based detectors) for space surveillance (Schildknecht, 2007). A core facility is the Space Surveillance Telescope (SST), which is DARPA's most advanced ground-based telescope for

* Corresponding author.
E-mail addresses: huan.do@adelaide.edu.au (H.N. Do), tat-jun.chin@adelaide.edu.au (T.-J. Chin), nick.m@inovor.com (N. Moretti), moriba@utexas.edu (M.K. Jah), matt.t@inovor.com (M. Tetlow).
URLs: http://www.inovor.com.au (N. Moretti), http://www.inovor.com.au (M. Tetlow).

https://doi.org/10.1016/j.asr.2019.03.008
0273-1177/

Table 1
Existing space object detection techniques.

Point-like (object tracking)	Streak-like (sidereal tracking)
Yanagisawa et al. (2012, 2005), Šára et al. (2013), Kouprianov (2008), Oda et al. (2014), Oda et al. (2014), Rutten et al. (2005), Bertin and Arnouts (1996)	Dawson et al. (2016), Levesque and Buteau (2007), Vananti et al. (2015), Virtanen et al. (2016), Danescu et al. (2014, 2012), Zimmer et al. (2013), Šára and Cvrcek (2017), Tagawa et al. (2016)

SSA, to be located in Western Australia under the US-Australia SSA Initiative. Furthermore, in 2006, the Japanese Aerospace Exploration Agency (JAXA) opened a new observational facility at Mt. Nyukasa, Nagano in an effort to search for un-cataloged GEO objects and orbit determination (Yanagisawa et al., 2009).

The efficacy of ground-based observations is affected by uncontrollable factors such as atmospheric effects, weather, and night-time-only observational constraints. Therefore, space-based SSA has been considered as a promising alternative. However, due to the much higher establishment cost (Flohrer et al., 2005), there is a much smaller number of fully functional space-based SSA systems up to date. Lauched in 1996, the Midcourse Space Experiment (MSX) satellite with a Space-based Visible (SBV) sensor on board has been feeding observation data to the SSN as a Contributing Sensor (Stokes et al., 1998).

Since maintaining a fleet of observatory-class spacecraft like the MSX satellite is financially prohibitive, developing nano-satellites, e.g., CubeSat, with equivalent observation capacity has been an active research area. For instance, a mission named Space-based Telescopes for Actionable Refinement of Ephemeris (STARE) was proposed to overcome the lack of tracking accuracy and high number of daily false alarms in the SSN (Riot et al., 2013).

2. Survey of optical space object detection techniques

In both ground-based and space-based SSA frameworks, a critical element is *object detection*, i.e., identifying potential man-made objects in a given set of measurements taken of the target region in space. In this section, we discuss state-of-the-art techniques for object detection using *optical sensors*, i.e., the measurements are a set of images.

In general, there are two main operational modes for optical telescopes: sidereal tracking (the telescope is re-oriented continuously to be fixed to the stars, thus, objects appear as *streak-like* regions in the images) and object tracking (the telescope is re-oriented continuously to be fixed to objects, thus, they appear as *point-like* regions). Based on the two operational modes, recent space object detection methods can be grouped as shown in Table 1.

2.1. Point-like object detection

Our paper focuses on this setting. While it may seem odd that one should still perform object detection given that the object can already be tracked by a telescope, in reality, "object tracking" merely refers to fixating on a *target region* in near space. Since the relative speed between objects in the region and the camera is usually small, the objects take a point-like form.

For point-like object detection, a common procedure is: (1) capture multiple images consecutively (i.e., an image sequence), (2) detect potential objects, which usually are referred to as "candidates", and finally (3) removing spurious ones by utilizing the fact that the objects move in different pattern with respect to the background stars (Yanagisawa et al., 2012; Šára et al., 2013).

The candidate detection process is essentially a matter of thresholding segmentation per frame where the threshold value is treated as a tunable parameter. The main difference between the methods is the quantity being thresholded. Thresholding directly on raw image intensity is simplistic and prone to include a high number of spurious candidates even after image reduction. Therefore, an intermediate processing step to suppress noise and to intensify the true object's signal is required. For instance, Šára et al. (2013) calculate a quantity named *event score*, which shows how likely an object pixel is an outlier given a temporal statistical model for the background pixel. Another common practice is to threshold on the correlation map of a point-source PSF and the input image (Kouprianov, 2008). The intensified object signal, however, still depends on its original signal-to-noise ratio (SNR) to some extent, thus, the detectable apparent magnitude of this family of methods is generally limited by the observable magnitude of the hardware's CCD sensor (Uetsuhara et al., 2010) leading to its incapability in detecting faint objects.

In order to deal with objects of low visual magnitudes, there is another family of methods that is called track-before-detect (TBD). The central principle in all TBD detection methods is to accumulate the "likelihood"[1] of an object's existence given the measurements without any assumption about the object's shape or its trajectory. In other words, given a sufficiently long period of observation, different measurements of a weak signal can be gathered up to the level of being detectable.

In the context of point-like detection, particle filter (PF) is a natural TBD solution framework (Salmond and Birch, 2001), in which the "accumulation" effect is accomplished through the particle re-sampling step where the probability of choosing a particle, i.e., its normalized weight, is set to be the evaluated value of the likelihood function. As a result, the particles are gradually driven toward the region

[1] The word "likelihood" here is used with a general meaning, which is not necessarily likelihood in the statistical sense.

with high likelihood value throughout the iterations. The pivotal factor in PF-based methods is the design of the likelihood function such that it is able to assign higher response to the object than the background noise after each iteration, even by a small amount. For example, the likelihood function can be a hand-crafted function of image intensity within a Region of Interest (ROI) (Uetsuhara and Ikoma, 2014). Alternatively, a more complete treatment can be achieved by conditioning the likelihood on an additional variable that indicates existence of object on each particle (Rutten et al., 2005). Apart from PF, a more "direct" approach was developed by Yanagisawa et al. (2005), which is called Stacking Image. In brief, star-registered frames are stacked and a cropping sub-window is defined for each frame. The sub-window's location in one frame is set to be shifted from the previous frame's sub-window by a shift value, and one shift value is shared by the whole stack. Finally, a median image is obtained from the shifted sub-windows. The process is repeated for different shift values. Fundamentally, each shift value can be interpreted as one hypothesis about the movement of the object and the correct shift shall have all the object's observations in the same location across sub-images yielding a high value in the final median image.

2.2. Streak-like object detection

For streak-like object detection, matched filters are widely applied as primary tools to detect objects (Dawson et al., 2016; Levesque and Buteau, 2007; Vananti et al., 2015; Pohlig, 1989). Filters are often constructed by a convolution between the Point Spread Function (PSF) and a source with pre-defined shape such as a line segment. An object then can be detected as the image region that shows high response to the filter. Additionally, the TBD framework can also be applied to find streak-like objects. A common strategy in the literature is to accumulate a quantity, e.g., pixel intensity value, along the direction of the streak to intensify the final response. Therefore, the problem essentially amounts to efficiently searching for the streak's direction. Tagawa et al. (2016) proposed a pipeline that consists of horizontal image shearing with different angles followed by a vertical compression step. Thus, the shearing angle that aligns with the streak's direction will yield the highest response in the compressed image. Alternatively, Šára and Cvrcek (2017) proposed a TBD-based single-frame streak detection method using Bayesian inference on image spatial domain instead of time domain. The authors design two parametric statistical models of pixel values of an image: with and without the existence of a streak. The models are joint distributions of pixel values, the shearing angle ϕ, and other line-descriptive parameters. The merit of the method arises from the aspect that instead of doing exhaustive search for ϕ, it uses a branch-and-bound scheme to find the optimal ϕ that maximizes its joint distribution with the pixel values. The method yields remarkably high detection accuracy (97%) even for faint and short simulated streaks on images in the TAOS data set.

2.3. Our contributions

In this work, we develop a novel technique named GP-ICP to detect point-like GEO objects under object tracking mode (see Section 2.1). Though our method is designed with a foresight to be integrated into a vision processing unit on a CubeSat, it is applicable to both ground-based and space-based situations. We focus on two major requirements when designing the detection algorithm: (1) high recall and precision rates and (2) low computational cost.

We describe our pipeline in detail in Section 3 and validate it on a number of real and synthetic data sets in Section 7. The closest techniques to our work are Stacking Image (Yanagisawa et al., 2012) and RANSACing Image Sequence (Šára et al., 2013), which we will compare in Section 7.

3. Overall pipeline of proposed detection algorithm

The input to our detection pipeline is a consecutive set of images $\{I_t\}_{t=1}^{T}$ that were captured by focussing on a small GEO region under object tracking mode. Relative motion exists between RSOs and the background stars, regardless of observational modes (sidereal or object tracking). Furthermore, due to short observing intervals and hyper-velocity movement of space objects, the rotational motion caused by either the Earth's rotation or the RSO's changes of course in the images is negligible, leaving only translational motion, i.e., the positions of a RSO in $\{I_t\}_{t=1}^{T}$ form a line.

Our overall pipeline, summarised in Fig. 1, employs the strategy of first detecting candidates by foreground/background (FG/BG) segmentation, then eliminating spurious candidates by using the fact that genuine ones lie close to a line. Details of the important modules in our method are described in the following sections. We keep description of the algorithm generic and not tied to a target sensor/telescope specification. Later in Section 7, we will provide the details of our actual imaging setup.

4. FG/BG segmentation using GPR

In our pipeline, each image I_t in the sequence $\{I_t\}_{t=1}^{T}$ is first subject to our novel FG/BG segmentation algorithm based on Gaussian process regression (GPR) (Rasmussen and Williams, 2006).

4.1. Why is FG/BG segmentation difficult?

The challenge in segmenting the foreground pixels (belonging to stars and objects) from the background pixels (belonging to the space void) in a star field image is due to

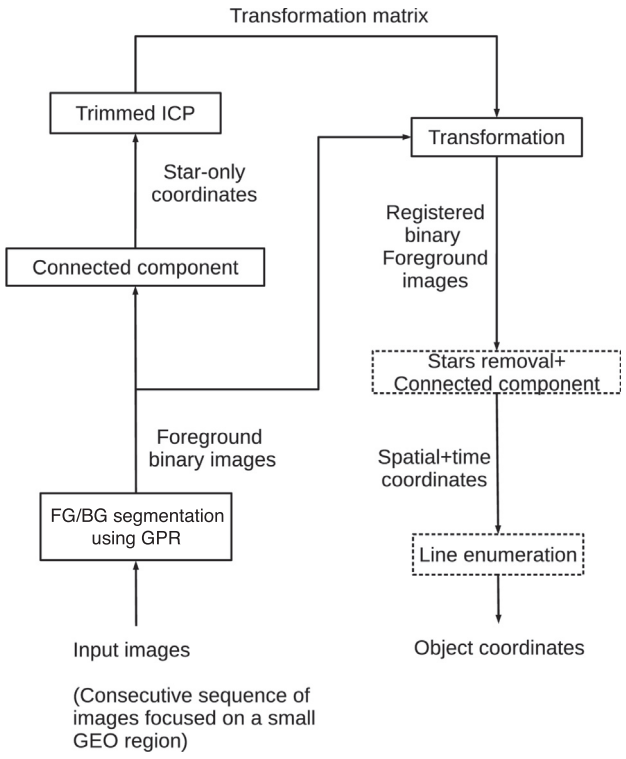


Fig. 1. GP-ICP overall pipeline: image-wise processing steps are shown in solid rectangles and processing steps that are applied on stack of images or a collapsed image are shown in dashed rectangles.

the faintness of the object pixels relative to the noise artefacts in the image, which could arise due to unexpected celestial phenomena or inherent flaws in the imaging pipeline. Fig. 2(a) shows an image from the TAOS dataset (to be described in Section 7.1) that has been used in the SSA literature, while Fig. 2(b) plots in scaled colours the subimage outlined by the green rectangle in the input image. A simple thresholding on the image intensity gives the FG/BG segmentation in Fig. 2(c), which is evidently noisy, especially near the bottom left corner of the subimage. By examining the intensities along a cross section of the image (indicated by the red line) as shown in Fig. 2(d), the reason behind the undesirable outcomes of direct thresholding is clear; the background intensities are not only noisy, they also vary as a function of the location in the image. In fact, this example shows that no single threshold will cleanly separate the FG/BG, since there are background pixels that are as bright as the target object along the cross section (indicated by the single peak in Fig. 2(d)).

In the literature, simple foreground object models such as PSF (Pohlig, 1989) or local statistical approaches such as median filtering (Bertin and Arnouts, 1996), have been used for FG/BG segmentation — these do not fully examine the structure of the object relative to the background distribution. On the other hand, Yanagisawa et al. (2012) and Šára et al. (2013) use relatively complex hand-crafted foreground models to perform the FG/BG segmentation. We develop a non-parametric technique based on GPR (Rasmussen and Williams, 2006) that is more capable of adapting to the data.

Another idea is to increase the detectability/magnitude of faint RSOs via image deconvolution (Nunez et al., 2015). We see this as complementary to our work, since deconvolution can be performed on our images first prior to FG/BG segmentation.

4.2. Problem formulation

Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$ indicate a 2D image grid, i.e., each $\mathbf{x}_i$ is a pixel location in the input image I_t. The image can be interpreted as a function y that gives the observed (noisy) intensity value $y(\mathbf{x}_i)$ over each $\mathbf{x}_i \in \mathcal{X}$. Define y^* as the noiseless image, where

$$y(\mathbf{x}_i) = y^*(\mathbf{x}_i) + \epsilon_i, \tag{1}$$

and $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$ is i.i.d. Normally distributed noise, for all $i = 1, \ldots, N$. Define m^* as the *ideal foreground mask*, where

$$m^*(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ corresponds to a foreground object;} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Our overarching aim is to estimate m^* given y.

4.3. GPR basics

Our method begins by using GPR to reconstruct y^*. Fundamentally, we impose a *Gaussian process prior* over y^*. Focussing on our problem context, this implies that the vector of noiseless intensity values

$$\mathbf{y}^* = [\, y^*(\mathbf{x}_1) \quad y^*(\mathbf{x}_2) \quad \cdots \quad y^*(\mathbf{x}_N) \,]^T \in \mathbb{R}^N \tag{3}$$

distributes according to the multivariate Gaussian distribution

$$\mathbf{y}^* \sim \mathcal{N}(\mathbf{0}, \mathbf{K}), \tag{4}$$

where the covariance matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ is defined as

$$\mathbf{K}_{i,j} = \kappa(\mathbf{x}_i, \mathbf{x}_j). \tag{5}$$

Here, κ is called the *kernel function* (hence, $\mathbf{K}$ is called the *kernel matrix* in the machine learning literature). Roughly, $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ computes the inner product between $\mathbf{x}_i$ and $\mathbf{x}_j$ in a higher-dimensional embedding space (defined by the form of κ; more details later) (Shawe-Taylor and Cristianini, 2004).

Conceptually, the Gaussian process prior (4) induces a *distribution of functions* over the domain $\mathcal{X}$, where the specific form of κ and the setting of its internal parameters define this distribution of functions. Combining (1) and (4), the marginal distribution of the observed intensities

$$\mathbf{y} = [\, y(\mathbf{x}_1) \quad y(\mathbf{x}_2) \quad \cdots \quad y(\mathbf{x}_N) \,]^T \in \mathbb{R}^N \tag{6}$$

can be established as

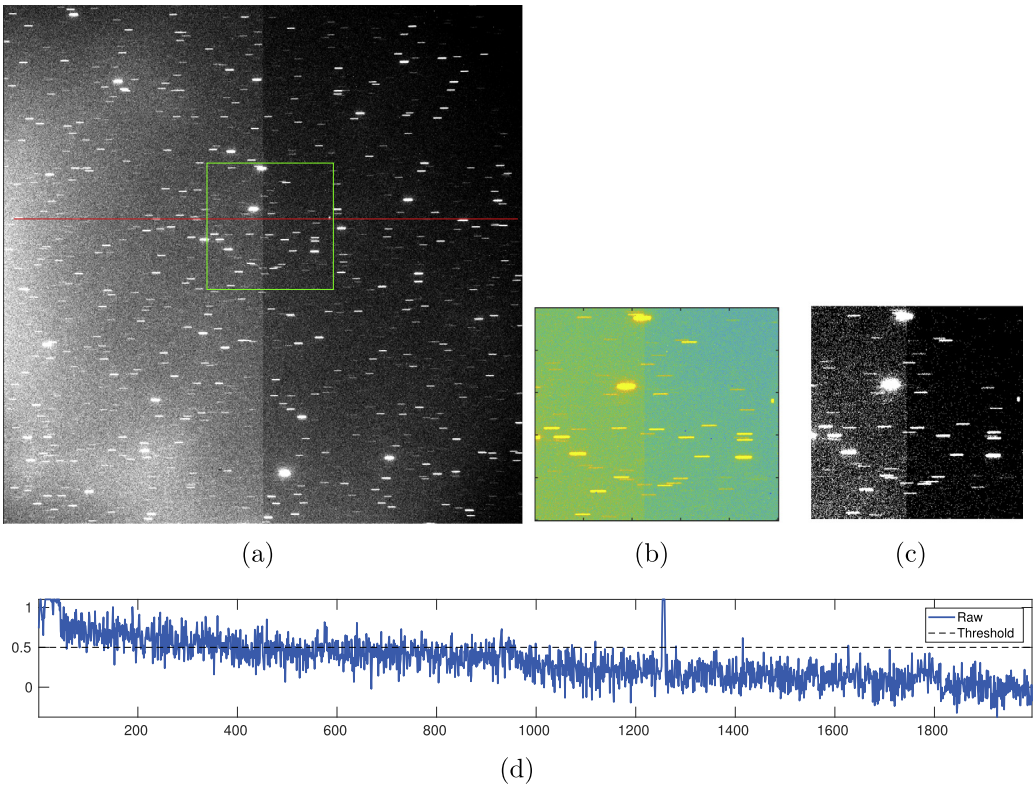


Fig. 2. (a) An image from the TAOS data set (see Section 7.1). (b) Subimage from the green rectangle, plotted in scaled colours. (c) FG/BG segmentation obtained by directly thresholding the intensities in the subimage. (d) Normalised intensities along the cross section indicated by the red line. The peak between horizontal coordinates 1200 and 1400 corresponds to a target object. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$$\mathbf{y} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K} + \sigma_n^2 \mathbf{I}\right); \tag{7}$$

see Section 2.2 in Rasmussen and Williams (2006) for detailed derivations.

Now, let $\mathbf{x}_\star$ be an image coordinate for which we wish to "predict" the value of $y^*(\mathbf{x}_\star)$, given (1) and (4). It can be established that the distribution of the extended vector

$$\begin{bmatrix} \mathbf{y} & y(\mathbf{x}_\star) \end{bmatrix}^T \in \mathbb{R}^{N+1} \tag{8}$$

is again a Gaussian of the form

$$\begin{bmatrix} \mathbf{y} \\ y(\mathbf{x}_\star) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K} + \sigma_n^2 \mathbf{I} & \mathbf{k}(\mathbf{x}_\star) \\ \mathbf{k}(\mathbf{x}_\star)^T & \kappa(\mathbf{x}_\star, \mathbf{x}_\star) + \sigma_n^2 \end{bmatrix}\right), \tag{9}$$

where $\mathbf{k}(\mathbf{x}_\star)$ is the vector of kernel evaluations

$$\mathbf{k}(\mathbf{x}_\star) = \begin{bmatrix} \kappa(\mathbf{x}_1, \mathbf{x}_\star) & \kappa(\mathbf{x}_2, \mathbf{x}_\star) & \cdots & \kappa(\mathbf{x}_N, \mathbf{x}_\star) \end{bmatrix}^T \in \mathbb{R}^N. \tag{10}$$

Since we observed $\mathbf{y}$ and wish to predict $y^*(\mathbf{x}_\star)$, we obtain the posterior distribution

$$p(y(\mathbf{x}_\star)|\mathbf{y}), \tag{11}$$

which is a univariate Gaussian with mean and variance

$$\mu(\mathbf{x}_\star) = \mathbf{k}(\mathbf{x}_\star)^T (\mathbf{K} + \sigma_n \mathbf{I})^{-1} \mathbf{y}, \tag{12}$$

$$\sigma^2(\mathbf{x}_\star) = \kappa(\mathbf{x}_\star, \mathbf{x}_\star) + \sigma_n^2 - \mathbf{k}(\mathbf{x}_\star)^T \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{k}(\mathbf{x}_\star). \tag{13}$$

Again, we refer the interested reader to the Section 2.2 in Rasmussen and Williams (2006) for detailed derivations.

Of immediate practical interest here is that the *maximum a posteriori (MAP)* estimate for $y^*(\mathbf{x}_\star)$ is $\mu(\mathbf{x}_\star)$, i.e., $\mu(\mathbf{x}_\star)$ is the most probable guess for the unknown function value $y^*(\mathbf{x}_\star)$. Further, in our application, $\mathbf{x}_\star$ is taken from the original grid $\mathcal{X}$, i.e., $\mathbf{x}_\star$ is always a pixel location, thus $\mu(\mathbf{x}_\star)$ for all $\mathbf{x}_\star \in \mathcal{X}$ is actually a reconstruction of y^* over the grid $\mathcal{X}$.

Fig. 3(a) and (b) show (in scaled colours) two GPR reconstructions μ_{BG} and μ_{FG} over the subimage in Fig. 2 (b) (for reasons that will be explained in the next section, the reconstructions are named "foreground/FG" and "background/BG"). Both cases employed the *squared exponential (SE)* kernel

$$\kappa(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Lambda^{-2} (\mathbf{x} - \mathbf{x}')\right), \tag{14}$$

where σ_f is the scale factor, and $\Lambda = \mathtt{diag}\left(\sigma_1^2, \sigma_2^2\right)$ with σ_1, σ_2 being the spatial bandwidths (Section 4.5 will discuss other choices of kernel functions). The reconstructions μ_{BG} and μ_{FG} in Fig. 3 use the same noise σ_n and scale σ_f magnitudes, but different spatial bandwidths σ_1, σ_2 (the bandwidths for μ_{BG} are larger than the bandwidths for μ_{FG}). Observe that in both results, the noise in the raw data is suppressed (though with different strengths). Section 4.5

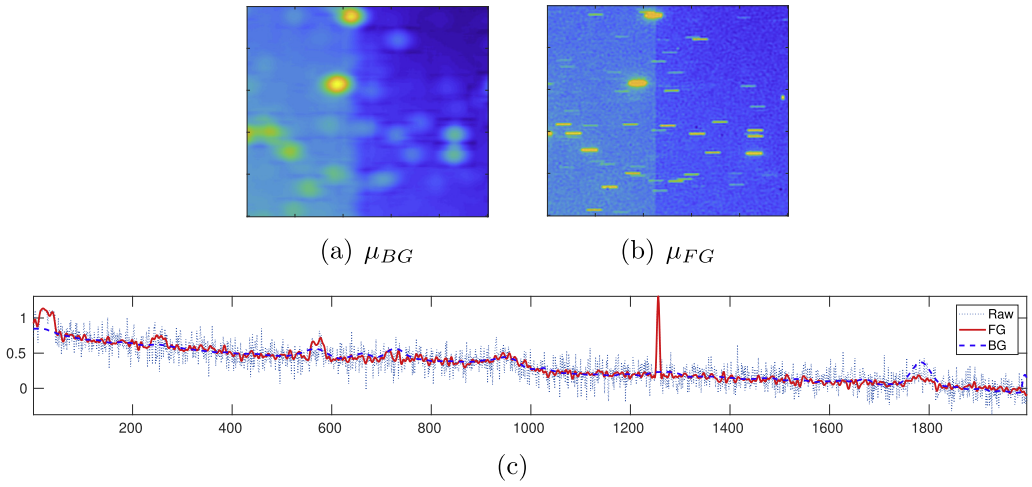


Fig. 3. (a)(b) Two different GPR reconstructions $\mu_{BG}(\mathbf{x}_\star)$ and $\mu_{FG}(\mathbf{x}_\star)$ over the subimage in Fig. 2(b). (c) The GPR reconstructions along the cross section in Fig. 2(d).

will discuss the tuning of the *hyperparameters* $\sigma_n, \sigma_f, \sigma_1$, and σ_2.

4.4. Estimating the foreground mask

Recall from Section 4.2 that our aim is to approximate the ideal foreground mask m^*. Using GPR, we estimate the mask as

$$m(\mathbf{x}_i) = \mu_{FG}(\mathbf{x}_i) - \mu_{BG}(\mathbf{x}_i), \tag{15}$$

where μ_{FG} and μ_{BG} are "foreground" and "background" GPR reconstructions; see Fig. 3. Intuitively, while both reconstructions perform denoising of the original signal, μ_{FG} is more sensitive to foreground intensities (e.g., small spikes from faint target objects), while μ_{BG} adapts more strongly to general trend of the background intensities. Section 4.5 will describe hyperparameter tuning for μ_{FG} and μ_{BG} to achieve these effects.

Fig. 4(a)(c) shows the estimated foreground mask $m(\mathbf{x}_i)$ over the subimage and cross section for the input in Fig. 2 based on the μ_{FG} and μ_{BG} in Fig. 3. Evidently $m(\mathbf{x}_i)$ is more amenable to direct thresholding for FG/BG segmentation, since the effects of noise and non-stationary background intensities have been suppressed, while the faint target object remains apparent in $m(\mathbf{x}_i)$. Fig. 4(b) shows the output of thresholding $m(\mathbf{x}_i)$, which is much cleaner than the result in Fig. 2(c). In Section 7 we will compare our technique against previous methods.

4.5. Kernel selection and hyperparameter tuning

As mentioned in Section 4.3, the kernel function corresponds to an inner product in a higher-dimensional embedding space of $\boldsymbol{\mathcal{X}} = \{\mathbf{x}_i\}_{i=1}^N$. This requires $\mathbf{K}$ to be positive semidefinite for any $\boldsymbol{\mathcal{X}}$, which imposes certain conditions on κ. It is beyond the scope of this paper to discuss these

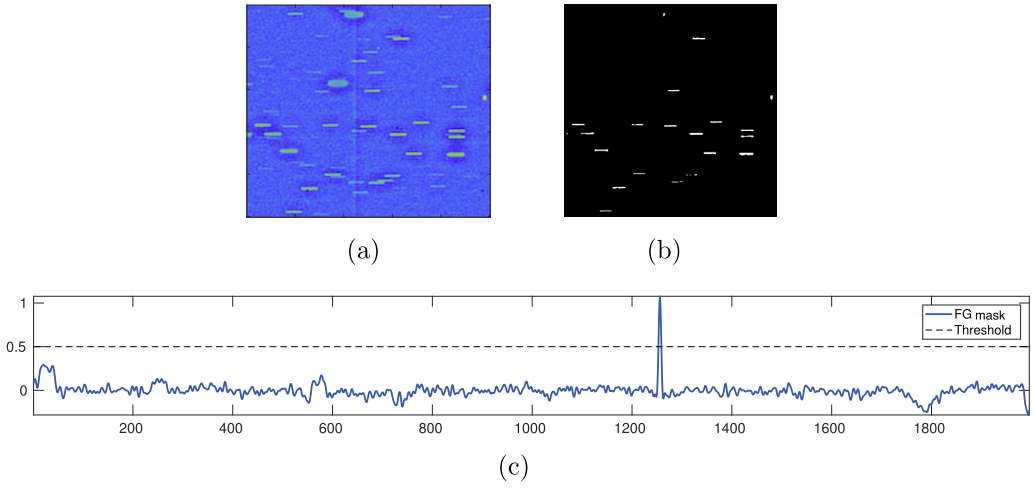
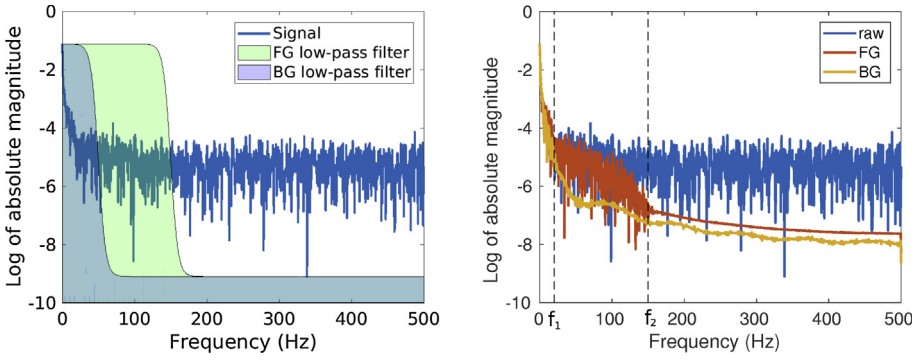


Fig. 4. (a)(c) Estimated foreground mask $m(\mathbf{x}_i) = \mu_{FG}(\mathbf{x}_i) - \mu_{BG}(\mathbf{x}_i)$ over the subimage and cross section in the input in Fig. 2, using the μ_{FG} and μ_{BG} in Fig. 3. (b) FG/BG segmentation obtained by thresholding the estimated foreground mask.



(a) (b)

Fig. 5. (a) The dark blue curve is the FFT (magnitude only) of the intensities along the cross section shown in Fig. 2(d). To build the foreground mask (15), we would like to construct two low-pass filters with respectively low and medium cut-off frequencies. (b) Superimposed on the original signal are the FFT of the μ_{BG} and μ_{FG} estimated from the data, as shown in Fig. 3. The vertical dashed lines indicate the cut-off frequencies. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

conditions, and the interested reader is referred to Rasmussen and Williams (2006).

In our problem, $\mathcal{X}$ is always a uniform 2D grid, thus it makes sense to use *homogenous kernels*, such as the SE kernel (14), where the value $\kappa(\mathbf{x},\mathbf{x}')$ is a function of only the distance $\|\mathbf{x}-\mathbf{x}'\|$. Other homogeneous kernels include the radial basis function (RBF) kernel

$$\kappa(\mathbf{x},\mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\sigma_1^2}\right) \tag{16}$$

and the Ornstein-Uhlenbeck (OU) kernel

$$\kappa(\mathbf{x},\mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|}{\sigma_1}\right). \tag{17}$$

A main distinguishing factor between the RBF and OU kernels and the SE kernel, is that the former two are *isotropic*. This implies that RBF and OU assume that the rate of intensity change in the horizontal and vertical directions to be the same, which may not be hold in practical settings. For this reason, we mainly use the SE kernel.

As mentioned in Section 4.3, the hyperparameters for the SE kernel are $\theta = \left[\sigma_n \sigma_f \sigma_1 \sigma_2\right]^T$. Generally speaking, the GPR reconstruction behaves as a low-pass filter that denoises the image. To achieve the specific effects required for estimating the foreground mask (15), we would like μ_{BG} to allow only low frequency (background) signals to pass through, while μ_{FG} to admit both low (background) and medium (stars and RSOs) frequency signals.

To illustrate the above requirements in the frequency domain, we take the FFT of the intensities along the cross section of the sample image, as shown in Fig. 2(d) (assuming the sampling frequency to be 1000). The FFT (magnitude only) is plotted in Fig. 5(a), where we superimpose two low-pass filters that we would like to construct. Fig. 5(b) shows the FFT of the GPR reconstructions μ_{BG} and μ_{FG} in Fig. 3. Evidently, both reconstructions have attenuated much of the noise, while μ_{FG} still retains some medium frequency components, including the target object signal.

We developed a procedure to find the hyperparameters θ_{BG} and θ_{FG}, respectively for μ_{BG} and μ_{FG}, that can achieve the above effects for a given input image. Note that in practice, the tuning procedure need only be carried for prototypical images in a data set, and the optimised hyperparameters can then be reused for other images that were captured in the same setting.

Define the loss function

$$\mathcal{L}(\theta_{FG},\theta_{BG}) = \begin{cases} \left(\mathcal{F}_{FG}(f)-\mathcal{F}_{BG}(f)\right)^2 & \text{for } 0 \leqslant f < f_1 \text{ (Hz)}, \\ -\left(\mathcal{F}_{FG}(f)-\mathcal{F}_{BG}(f)\right)^2 & \text{for } f_1 \leqslant f < f_2 \text{ (Hz)}, \\ \mathcal{F}_{FG}^2(f)+\mathcal{F}_{BG}^2(f) & \text{for } f > f_2 \text{(Hz)}, \end{cases} \tag{18}$$

where $\mathcal{F}_{BG}(f) = FFT(\mu_{BG}(\mathbf{x}))$ is the FFT of GPR reconstruction μ_{BG} using the hyperparameters θ_{BG} (similarly for $\mathcal{F}_{FG}(f)$), and f_1 and f_2 are cut-off frequencies that are assumed known in $\mathcal{L}(\theta_{FG},\theta_{BG})$ (to be described later). The desired hyperparameters are then optimised as

$$\theta_{FG}^*, \theta_{BG}^* = \underset{\theta_{FG},\theta_{BG}}{\operatorname{argmin}} \mathcal{L}(\theta_{FG},\theta_{BG}). \tag{19}$$

Intuitively, problem (19) minimises the difference between μ_{BG} and μ_{FG} in $[0, f_1)$ (preserving the background signals), maximises the difference between μ_{BG} and μ_{FG} in $[f_1, f_2)$ (separating background signal from object signal), and minimises both μ_{BG} and μ_{FG} in (f_2, ∞) (eliminating noise).

The setting of f_1 and f_2 depends on the appearance of objects of interest (e.g., RSOs, stars) in an image. For example, in our data sets, the sizes of foreground objects range from 6 to 100 pixels,[2] which correspond to $f_1 = 10$ Hz and $f_2 = 150$ Hz, using an FFT sampling frequency of 1000 Hz.

To solve problem (19), a simple grid search over the domain θ_{BG} and θ_{FG}. As alluded to above, since hyperparameter tuning is done only on a few prototypical images from a dataset, this procedure is sufficient.

[2] Note that this information is also derivable from the metadata of any data set, such as field of view or exposure time.

4.6. Computational considerations

In the posterior mean (12), since the domain $\mathcal{X}$ is the same 2D grid as long as the input image is of the same size, the matrix $(\mathbf{K}+\sigma_n\mathbf{I})$ can be pre-computed and pre-inverted before calculating the reconstruction $\mu(\mathbf{x}_\star)$. Moreover, across different $\mathbf{x}_\star$, the input intensity values $\mathbf{y}$ does not change, hence $(\mathbf{K}+\sigma_n\mathbf{I})^{-1}\mathbf{y}$ can also be precomputed. The main effort is thus to calculate $\mathbf{k}(\mathbf{x}_\star)$ (linear in the number of pixels) and multiply it with $(\mathbf{K}+\sigma_n\mathbf{I})^{-1}\mathbf{y}$ (quadratic in the number of pixels). This unfortunately still represents a costly procedure. To speed up computations, we take the simple approach of subdividing the input image into 100 pixel by 100 pixel blocks, and perform the GPR reconstruction on each subimage individually. This speeds up the reconstruction process without sacrificing accuracy, as we will demonstrate in Section 7.

5. Image registration

Once we have converted the input image sequence $\{I_t\}_{t=1}^T$ into foreground images $\left\{\widehat{I}_t\right\}_{t=1}^T$, we estimate *planar perspective transforms* or *homographies* (Szeliski, 2005) that bring the images into geometric alignment in a common image coordinate frame. In the common coordinate frame, different observations of the same star will converge to the same pixels, while the observations of an RSO will form a line (see Section 3 for the rationale).

First, a reference frame $\widehat{I}_r$ is chosen, where typically $r=\text{round}(T/2)$. For each $t\neq r$, we estimate the 2D homography, which is defined by the 3×3 matrix $\mathbf{H}_t^r$, that warps $\widehat{I}_t$ to align with $\widehat{I}_r$. The warping of an image point $\mathbf{p}$ (a 2D coordinate) in $\widehat{I}_t$ is calculated as

$$w\left(\mathbf{p}|\mathbf{H}_t^r\right)=\frac{\mathbf{H}_t^r[1:2]\begin{bmatrix}\mathbf{p}\\1\end{bmatrix}}{\mathbf{H}_t^r[3]\begin{bmatrix}\mathbf{p}\\1\end{bmatrix}} \tag{20}$$

where $\mathbf{H}_t^r[1:2]$ is the first-2 rows of $\mathbf{H}_t^r$, and $\mathbf{H}_t^r[3]$ is the 3rd row of $\mathbf{H}_t^r$. See Szeliski (2005) for the rationale of using a homography for image alignment.

To estimate $\mathbf{H}_t^r$, we first reduce $\widehat{I}_t$ and $\widehat{I}_r$ into discrete point sets $\{\mathbf{p}_i\}_{i=1}^{N_t}$ and $\{\mathbf{q}_j\}_{j=1}^{N_r}$, by using a standard connected component analysis to extract centroids of the connected foreground regions in $\widehat{I}_t$ and $\widehat{I}_r$. If the two point sets fully overlap, i.e., each $\mathbf{p}_i$ has a genuine matching point in $\mathbf{q}_j$ (implying also that $N_t=N_r$), the standard iterative closest points (ICP) algorithm (Besl and McKay, 1992) will be effective in estimating $\mathbf{H}_t^r$. However, in our setting, the point sets will only partially overlap due to the apparent motion of the background stars. This produces non-matching points, which act as outliers that can bias ICP.

To deal with outliers, we utilize a robust registration algorithm called Trimmed Iterative Closest Point (T-ICP) (Chetverikov et al., 2002). To set the scene, given a candidate $\mathbf{H}_t^r$, define

$$d_i\left(\mathbf{H}_t^r\right)=\min_j\left\|w\left(\mathbf{p}_i|\mathbf{H}_t^r\right)-\mathbf{q}_j\right\|_2 \tag{21}$$

as the *residual* of the i-th point $\mathbf{p}_i$. Intuitively, $d_i\left(\mathbf{H}_t^r\right)$ is the distance of the point in $\{\mathbf{q}_j\}_{j=1}^{N_r}$ that is closest to the warped version of $\mathbf{p}_i$. Also, let

$$d_{(i)}\left(\mathbf{H}_t^r\right) \tag{22}$$

indicate the i-th largest residual amongst $d_1\left(\mathbf{H}_t^r\right),d_2\left(\mathbf{H}_t^r\right),\ldots,d_{N_t}\left(\mathbf{H}_t^r\right)$ for the candidate homography $\mathbf{H}_t^r$. The goal of T-ICP is to minimise the following *trimmed* sum of squared residuals

$$\sum_{i=1}^{\xi}d_{(i)}\left(\mathbf{H}_t^r\right) \tag{23}$$

over the unknown homography $\mathbf{H}_t^r$, where $1\leqslant\xi\leqslant N_t$. The conventional ICP method (Besl and McKay, 1992) is obtained by setting $\xi=N_t$, i.e., all the closest point distances are considered. In contrast, T-ICP only minimises the $\xi<N_t$ smallest residuals, which enables outliers (non-matching points) to be ignored. To minimise the T-ICP cost, an alternating point-to-point assignment and transformation estimation (using only the ξ-best assignments) technique is used, as summarised in Algorithm 1.

Algorithm 1. T-ICP for image registration.

Require: Point sets $\{\mathbf{p}_i\}_{i=1}^{N_t}$ and $\{\mathbf{q}_j\}_{j=1}^{N_r}$, trimming parameter ξ with $1\leqslant\xi\leqslant N_t$, convergence threshold ϵ.
$\mathbf{H}\leftarrow\mathbf{I}$ (identity matrix).
$d_{old}\leftarrow\infty$.
while true **do**
  for $i=1,\ldots,N_t$ **do**
    $c_i\leftarrow\text{argmin}_j\left\|w(\mathbf{p}_i|\mathbf{H})-\mathbf{q}_j\right\|_2$.
    $d_i(\mathbf{H})\leftarrow\left\|w(\mathbf{p}_i|\mathbf{H})-\mathbf{q}_{c_i}\right\|_2$. /*Calculate current residuals.*/
  end for
  if $|d_{(\xi)}-d_{old}|\leqslant\epsilon$ **do**
    Break./*Stop if residuals have converged.*/
  end if
  $\mathcal{I}\leftarrow\left\{i\in\{1,\ldots,N_t\}|d_i(\mathbf{H})\leqslant d_{(\xi)}(\mathbf{H})\right\}$.
  $\mathcal{C}\leftarrow\left\{(\mathbf{p}_i,\mathbf{q}_{c_i})\right\}_{i\in\mathcal{C}}$./*Create ξ-best correspondence set.*/
  $\mathbf{H}\leftarrow$ Estimate homography from $\mathcal{C}$./*See Section 4.1 of Hartley and Zisserman (2004).*/
  $d_{old}\leftarrow d_{(\xi)(\mathbf{H})}$.
end while
$\mathbf{H}_t^r\leftarrow\mathbf{H}$.
return $\mathbf{H}_t^r$.

Fig. 6 compares ICP and T-ICP on point sets obtained by our FG/BG segmentation technique on two overlapping star field images. True and false correspondences are marked by green and red lines, respectively. Observe that the non-matching points have biased the ICP result; on the other hand, T-ICP successfully ignored the non-matching points. In the same figure, we also show the final registered images that were obtained by applying the estimated homographies and composited using the `max` operator. Observe that the background star streaks are much sharper in the T-ICP composite, due to much more accurate registration.

6. Track detection

Once the homographies $\{\mathbf{H}_t^r\}_{t\neq r}^{t=1,\ldots,T}$ are estimated, we use them to register the foreground images $\left\{\widehat{I}_t\right\}_{t\neq r}$ to the foreground reference frame $\widehat{I}_r$. In the common coordinate frame, we remove background stars by eliminating any overlapping binary regions over the temporal dimension. After this step, the foreground images contain only RSOs and false foreground pixels. We collapse the stack by the `max` operator. Finally, we extract the centroids of the binary connected regions to obtain a set of spatial coordinates; see Fig. 13 for a sample outcome. Also, each coordinate is associated with the index t that it belongs to before the collapsing step.

We utilize a simple and straightforward procedure to detect tracks:

1. Choose an arbitrary initial pair of two coordinates with different time frame.
2. Create a line that passes through the two coordinates.
3. Find hypothetical spatial coordinates and time frames of other points on the proposal track.

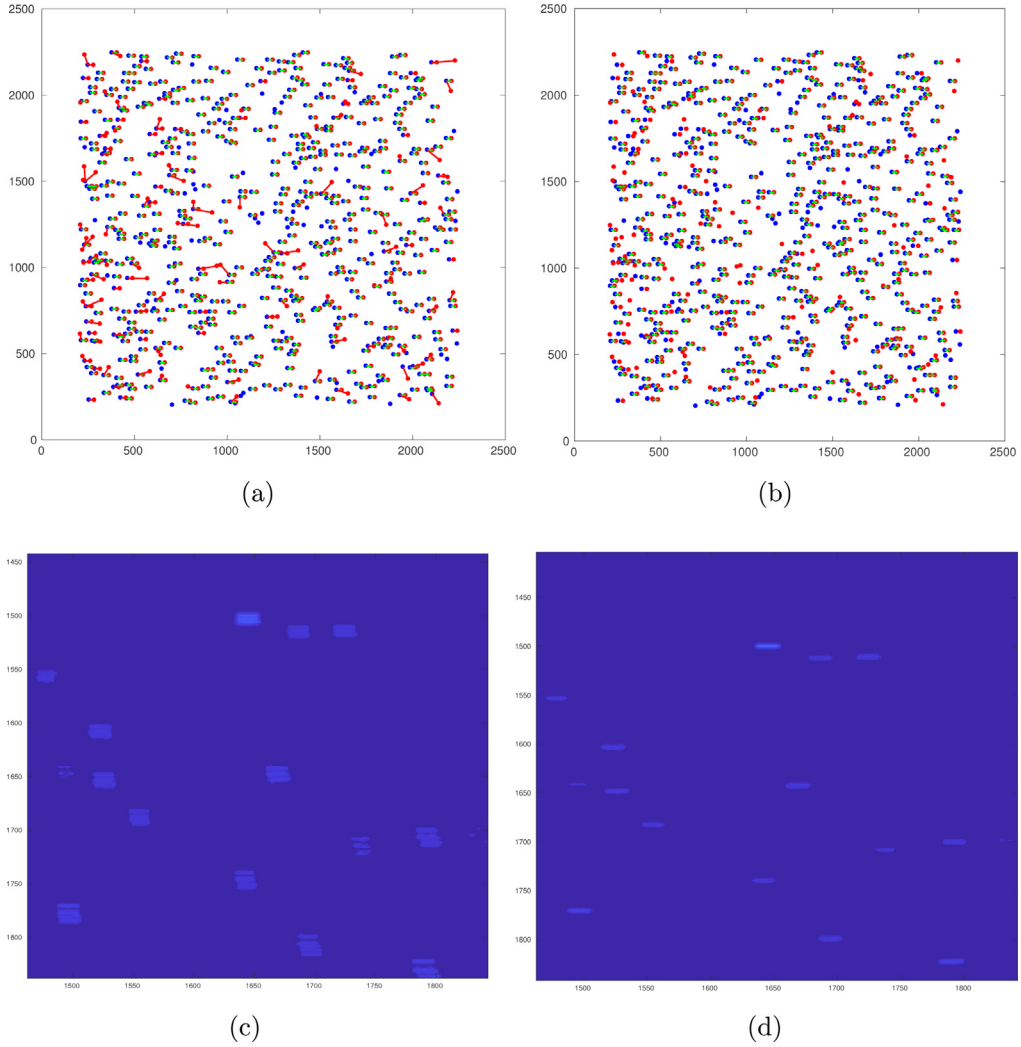


Fig. 6. This figure is best viewed in color. **First row:** Example of image registration process using (a) regular ICP and (b) Trimmed ICP to register two point clouds (red and blue). True and false correspondences are marked by green and red lines, respectively. Note that the Trimmed ICP yields no false correspondences in this particular example. **Second row:** The resulting registered 5 images using (c) regular ICP and (d) Trimmed ICP, composited using the `max` operator. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4. Count the number of supports and return qualified tracks.

We define the *supports* of a track as follows. When there is an actual centroid exists within the vicinity of a proposal coordinate with the same time frame, we count it as one support for this proposal track. The eventual detected tracks are determined by thresholding the proposal tracks in terms of their support size; see Fig. 7 for an illustration.

7. Experimental study

The closest work to ours is RANSACing Image Sequence (Šára et al., 2013) and Stacking Image (Yanagisawa et al., 2012). Unfortunately, Stacking Image is feasible on an FPGA platform only, hence, an in-depth comparison was done only with RANSACing Image Sequence. Nonetheless, we also provide a final "pipeline-to-pipeline" comparison against Stacking Image.

Since the full source code of Šára et al. (2013) was not available, we implemented (parts of) the method ourselves. Two datasets were used in our experiments: the TAOS dataset (Lehner et al., 2006) and our own dataset which was provided by Defence Science and Technology (DST) Group of Australia.

7.1. TAOS data set

The images were captured in the 2nd day of a 3-day campaign to observe the 1968-081E fragments at the Lulin observatory in Taiwan. The FOV of the telescope was $1.3^\circ \times 1.3^\circ$, image size was 2049×2047, and the data pixel was 16-bit gray-scale. There were 66 observations, where each observation contains 29 images captured at 5.9s exposure time and 2.9s between two shots. The pointing direction of the telescope was fixed during each observation (object-tracking mode).

To our knowledge, the two best performing methods for RSO detection on the TAOS dataset are Yanagisawa et al. (2012) and Šára et al. (2013).

7.1.1. Image registration comparison

In this section, we compared our image registration module with the one developed by Šára et al. (2013). In brief, their image registration is done as follows: from raw images, stars coordinates are found by a procedure of 2-D Gaussian convolution, thresholding, and standard connected component analysis. For each coordinate of a bright star, a descriptive feature is formed as two 2-D vectors pointing from that star to the two nearest stars of lower relative magnitudes. Based on this descriptor, stars in the fixed and moving images are matched using nearest neighborhood algorithm. Since this matching is prone to contain outliers, Random Sample Consensus (RANSAC) algorithm is used to find inliers using homography warping and Sampson registration error. Finally, the output homography warping is estimated based on the inlier set using a standard gradient descent minimization.

We designed an experiment to compare this registration scheme with ours as follows: we fed the same pair of binary foreground images, I^f and I^m, to both methods. Then, we applied the resulting homography transformation matrices T_m^f to warp binary image I^m into I^f. The registration error is defined as:

$$\texttt{err} = \frac{\texttt{SUM}\left(\texttt{XOR}\left(I^f, T_m^f I^m\right)\right)}{\texttt{SUM}\left(I^f\right) + \texttt{SUM}\left(T_m^f I^m\right)}, \tag{24}$$

where XOR is the exclusive-OR operator and SUM is the numerical pixel-wise summation of a binary image. err ranges from 0 (the two images are perfectly aligned) to 1 (the two images are completely misaligned). Essentially the better estimated T_m^f makes the binary star blobs overlap more, hence yields smaller error. We ignored the contribu-

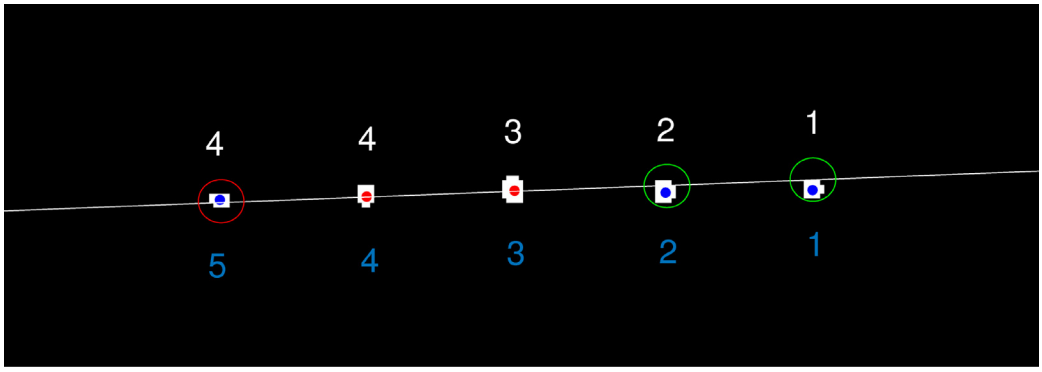


Fig. 7. Example of a detected track: The image shows a close-up look of 5 observations of an object in the registered foreground images collapsed by the max operator. The frame indices of actual observations and proposed coordinates are shown in white and blue, respectively. In this particular example, coordinates from frames 3 and 4 are chosen as the initial pair (marked with red dots). A track proposal is formed by the line connecting the two points, and the neighbourhoods of locations of the remaining observations are marked with green (validly activated) and red (invalidly activated) circles. This particular proposal track has a support of 4. (This figure should be viewed in color.). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

tion of objects and noise in the error since their appearances and areas are negligible with respect to the stars.

We tested the two methods on a set of raw image pairs obtained from the whole data set. For each pair of raw images, we obtained different FG binary images using various thresholding values. Fig. 8 shows the performance of the two methods on different thresholding values. For each thresholding value, we computed the error for the two methods over 66 observations and show the statistics (mean and standard deviation). As shown in Fig. 8(a), our method yielded lower registration error and more stable performance (smaller standard deviation across observations). In addition, our method was faster by an order of magnitude, as shown in Fig. 8(b).

7.1.2. FG/BG segmentation comparison

In Šára et al. (2013), FG/BG segmentation (also called "primitive event detection") is done as follows: once the raw images are registered, they are stacked to create a 3D data set. Next, they are convoluted by a 3D Gaussian kernel function, then normalized by local variance to form what is called a *feature map*. Finally, each pixel is assigned with an *event score*, which indicates how likely its feature map value is outlying over the temporal dimension. Finally, the binary foreground image is determined by thresholding the event score map.

We compared our FG/BG segmentation technique with the one in Šára et al. (2013) in terms of their ability to retain objects with different signal strengths in the output foreground images. In particular, we added synthesized objects with different Signal-to-Noise Ratio (SNR) (range from 19 dB to 26 dB) to the original images and fed the images to the respective FG/BG segmentation modules. The synthesized objects are formed as follows: first, we created a layer image with zero-valued background and point source pixels that have intensities corresponding to the assigned SNRs. The locations of the point sources are placed randomly in the image. Then, we applied Gaussian smoothing to this layer image, before adding it to a real astronomical image.

The true positives are counted as the number of connected regions that contain coordinates of synthesized objects in the final binary foreground images, and the false positives are the remaining otherwise. Note that we would prefer a high Recall rate for the FG/BG segmentation since even the Precision rate is low, the false positives shall be eliminated by the next track detection step.

We vary the threshold values (thresholding on the foreground mask (15) calculated using GPR reconstructions, and on the event score map in Šára et al. (2013)) and plot the Precision-Recall curve as shown in Fig. 9. The Precision and Recall rates are defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{25}$$

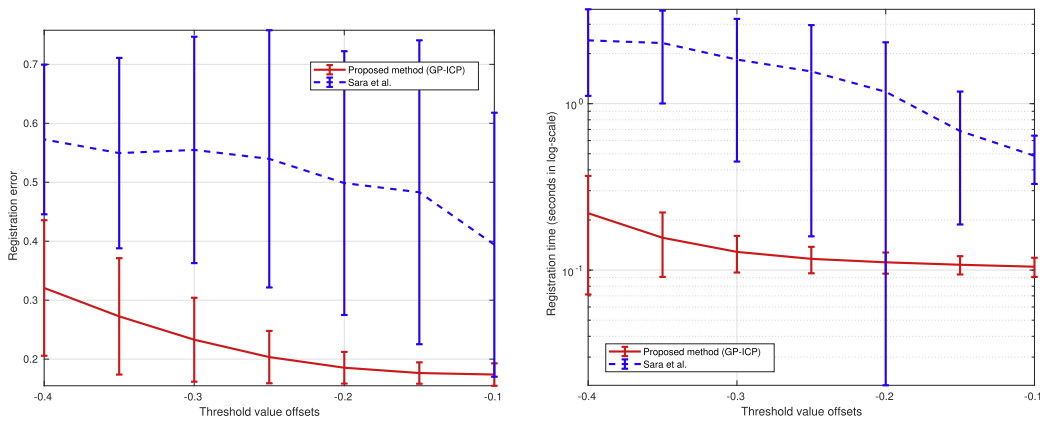
$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{26}$$

where TP, FP, and FN stand for True Positive, False Positive, and False Negative, respectively. When raising the threshold, both methods show the same level of increase in Precision rate, however, our method yields better Recall rate when the threshold value is decreased. Hence, compared to Šára et al. (2013), our FG/BG segmentation module is more capable of detecting dimmer objects.

7.1.3. RSO detection results

We also report our final detection results for RSOs in the TAOS data set. Fig. 10 shows the final detection results of our method, along with the ones reported in Šára et al. (2013) and Yanagisawa et al. (2012). In summary, our method detected 41 verified tracks, while Šára et al. (2013) and Yanagisawa et al. (2012) detected 42 and 51 tracks respectively (see Fig. 11).

Overall we detected almost the same objects that were found by Šára et al. (2013). While both GP-ICP and RANSACing Image Sequence seem to be less accurate/sensitive



(a) (b)

Fig. 8. Comparison of registration modules in the proposed pipeline and Šára et al. (2013) in terms of (a) registration error and (b) registration run time. Note that this result is based on our own reimplementation of Šára et al. (2013).

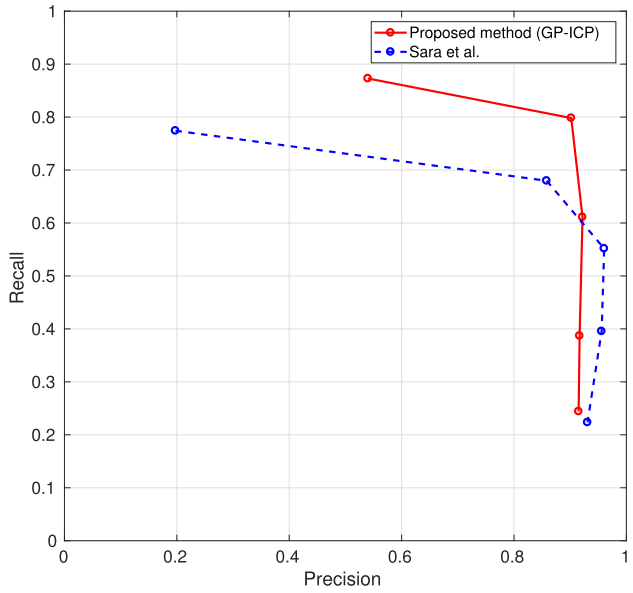


Fig. 9. FG/BG segmentation module comparison: the Precision-Recall curve.

than Image Stacking based on the final result in Fig. 10, this difference needs to be put into context:

- It was difficult to pinpoint the reasons behind the lower accuracy of GP-ICP and RANSACing Image Sequence, since the result of Image Stacking was available in the form of Fig. 10 only, i.e., for an object that was missed, it was difficult to identify the source image and pixel coordinates from which the object was detected.
- Image Stacking is feasible only on an FPGA, whereas GP-ICP and RANSACing Image Sequence are feasible on a general purpose processor.

7.1.4. Runtime information

We implemented GP-ICP in `MATLAB` on a standard core i7, 8G RAM PC. On average, our method requires a processing time of 1.638 seconds/image, which is less than Šára et al. (2013) (3.406 seconds/image).

Fig. 11. A Officina Stellare RH200 telescope.

7.2. Adelaide-DST data set

This is a homegrown dataset that consists of two observation campaigns done in two different dates, hence the names `150826` and `160403`. `150826` has 10 observations and `160403` has 15 observations.

The electro-optical sensor is a pair of an Officina Stellare RH200 telescope and an FLI Proline PL4240 camera. This sensor operates in the visible wavelength and has a field of view of approximately 2.6 degrees with 4.6 arcseconds per pixel. The sensor is mounted to a Software Bisque Paramount MEII robotic mount, inside an Aphelion Domes 7ft clamshell dome.

Each observation contains 5 images, captured at 5s exposure time, 10s between two shots. The image is 16-bit and has the size of 2048×2048. Since the image sequence contains only 5 frames, detection would be more challenging than the TAOS data set. The ground truth objects are manually labels by an individual who was not involved in this research.

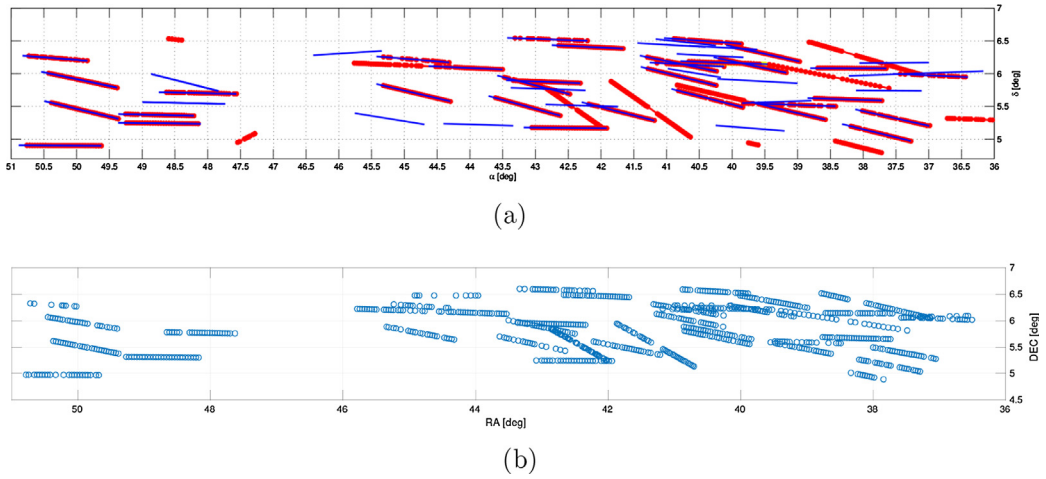


Fig. 10. RSO detection results (displayed using object attitude) for TAOS data set: (a) Detection results of Šára et al. (2013) (red dots) and Yanagisawa et al. (2012) (blue lines) (the image is taken from Šára et al. (2013)). (b) Detection results of our method GP-ICP. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

As part of our contributions, we will publicly release this dataset with ground truth detection coordinates to the community. Please visit the homepage of Dr. Tat-Jun Chin for further information.

7.2.1. RSO detection results

We achieved good accuracty on this data set as shown in the Precision-Recall curve in Fig. 12. The maximum Recall rates for `150826` and `160403` are 98% and 93%, respectively. Fig. 13 shows an example of detection result from `150826` data set. The image is obtained by applying a `max` operator, which returns the maximum value of a pixel over the temporal dimension, on registered binary foreground images. A false positive can be seen near the left lower corner where three spurious FG regions line up. Additionally, two instances of an object are undetected near the left edge due to the fact that the other instances are occluded by stars, hence being eliminated during the star removal procedure. Our track detection automatically excludes proposal tracks with supports fewer than 3, since any two points can form a line. Therefore, inevitably our method would miss any object that has fewer than 3 appearances in the image sequence (because of being occluded by stars or moving out of sight). Nevertheless, our pipeline does not have this shortcoming with long image sequence data sets such as TAOS.

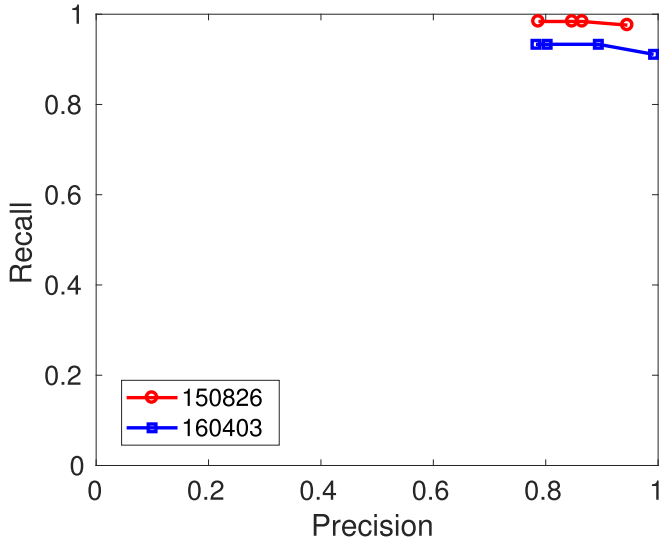


Fig. 12. Precision-Recall curves for RSO detection on the Adelaide-DST dataset.

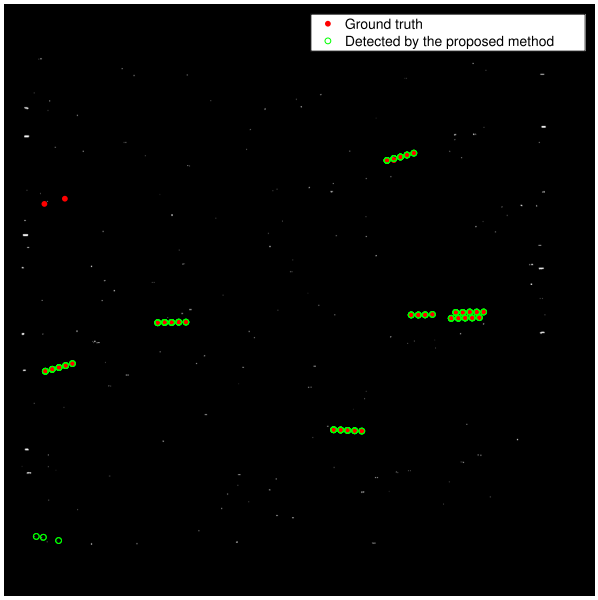


Fig. 13. Example of detected objects in the first observation of `150826` data set: 5 registered binary FG frames (with stars removed) are collapsed by a `max` operator. The ground truth and detected coordinates are marked in red dots and green circles, respectively (Only the ground truth coordinates that are not occluded by stars are shown). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

8. Conclusion

We introduced GP-ICP, a novel pipeline for optical detection of RSOs in the GEO band. We tested our method on several realistic data sets, where we showed that our method outperformed RANSACing Image Sequence (Šára et al., 2013) in two critical modules: image registration and FG/BG segmentation. On the whole GP-ICP yields similar accuracy as Šára et al. (2013) while requiring less computational resource. Overall, our approach is accurate (above 90% recall and precision) on real-world data sets.

Acknowledgments

This work was mainly supported by ARC project LP160100495. The authors would like to express their sincere thanks to Dr. Yanagisawa of JAXA and Dr. Šára of Center for Machine Perception, Czech Technical University for making the TAOS data set available to us and helping us with the data analysis. We also would like to thank the Defence Science and Technology (DST) Group and Inovor Technologies for providing their data sets and their continued support on this research.

References

Bertin, E., Arnouts, S., 1996. Sextractor: software for source extraction. Astron. Astrophys. Suppl. Series 117, 393–404.
Besl, P.J., McKay, N.D., 1992. A method for registration of 3-d shapes. IEEE Trans. Pattern Anal. Machine Intell. 14, 239–256.
Chetverikov, D., Svirko, D., Stepanov, D., Krsek, P., 2002. The trimmed iterative closest point algorithm. In: Proceedings of 16th International Conference on Pattern Recognition. IEEE, pp. 545–548.
Danescu, R., Ciurte, A., Turcu, V., 2014. A low cost automatic detection and ranging system for space surveillance in the medium earth orbit region and beyond. Sensors 14, 2703–2731.
Danescu, R., Oniga, F., Turcu, V., Cristea, O., 2012. Long baseline stereovision for automatic detection and ranging of moving objects in the night sky. Sensors 12, 12940–12963.

Dawson, W.A., Schneider, M.D., Kamath, C., 2016. Blind detection of ultra-faint streaks with a Maximum Likelihood Method. arXiv preprint arXiv:1609.07158.

Flohrer, T., Peltonen, J., Kramer, A., Eronen, T., Kuusela, J., Riihonen, E., Schildknecht, T., Stöveken, E., Valtonen, E., Wokke, F., et al., 2005. Space-based optical observations of space debris. In: 4th European Conference on Space Debris, p. 165.

Hartley, R., Zisserman, A., 2004. Multiple View Geometry in Computer Vision. 2nd ed. Cambridge University Press.

Kouprianov, V., 2008. Distinguishing features of CCD astrometry of faint GEO objects. Adv. Space Res. 41, 1029–1038.

Lehner, M.J., Alcock, C., Axelrod, T., Bianco, F., Byun, Y., Chen, W., Cook, K.H., Dave, R., de Pater, I., Giammarco, J., King, S., Lee, T., Lissauer, J., Marshall, S.L., Mondal, S., Nihei, T., Rice, J., Schwamb, M., Wang, A., Wang, S., Wen, C., Zhang, Z., 2006. TAOS - the Taiwanese-American occultation survey. Astron. Nachr. 327, 814–817 http://taos.asiaa.sinica.edu.tw/.

Levesque, M.P., Buteau, S., 2007. Image processing technique for automatic detection of satellite streaks. Technical Report. Defense research and development Canada Valcartier (Quebec).

Nunez, J., Nunez, A., Montojo, F.J., Condominas, M., 2015. Improving space debris detection in GEO ring using image deconvolution. Adv. Space Res. 56, 218–228.

Oda, H., Yanagisawa, T., Kurosaka, H., Tagawa, M., 2014. Optical observation, image-processing, and detection of space debris in geosynchronous earth orbit. COSPAR Scientific Assembly, Moscow, PEDAS 1, 0004–14.

Pohlig, S.C., 1989. An algorithm for detection of moving optical targets. IEEE Trans. Aerospace Electron. Syst. 25, 56–63.

Rasmussen, C.E., Williams, C.K., 2006. Gaussian Processes for Machine Learning, vol. 1. MIT Press, Cambridge.

Riot, V., de Vries, W., Simms, L., Bauman, B., Carter, D., Phillion, D., Olivier, S., 2013. The space-based telescopes for actionable refinement of ephemeris (STARE) mission. In: Proceedings of 2013 Small Satellite Conference.

Rutten, M.G., Gordon, N.J., Maskell, S., 2005. Recursive track-before-detect with target amplitude fluctuations. IEE Proceedings-Radar, Sonar and Navigation 152, 345–352.

Salmond, D., Birch, H., 2001. A particle filter for track-before-detect. In: Proceedings of the 2001 American Control Conference. IEEE, pp. 3755–3760.

Šára, R., Cvrcek, V., 2017. Faint streak detection with certificate by adaptive multi-level Bayesian Inference. In: Proceedings of the 7th European Conference on Space Debris.

Šára, R., Matoušek, M., Franc, V., 2013. RANSACing optical image sequences for GEO and near-GEO Objects. In: Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference.

Schildknecht, T., 2007. Optical surveys for space debris. Astron. Astrophys. Rev. 14, 41–111.

Shawe-Taylor, J., Cristianini, N., 2004. Kernel Methods for Pattern Analysis. Cambridge University Press.

Stokes, G.H., Von Braun, C., Sridharan, R., Harrison, D., Sharma, J., 1998. The space-based visible program. Lincoln Laboratory J. 11, 205–238.

Szeliski, R., 2005. Image alignment and stitching: a tutorial. Technical Report MSR-TR-2004-92. Microsoft Research.

Tagawa, M., Yanagisawa, T., Kurosaki, H., Oda, H., Hanada, T., 2016. Orbital objects detection algorithm using faint streaks. Adv. Space Res. 57, 929–937.

Uetsuhara, M., Hanada, T., Yamaoka, H., Fujiwara, T., Yanagisawa, T., Kurosaki, H., Kitazawa, Y., 2010. Detection of faint GEO objects using population and motion prediction. In: The 11th Annual Advanced Maui Optical and Space Surveillance Technologies Conference, Maui, HI, USA.

Uetsuhara, M., Ikoma, N., 2014. Faint debris detection by particle based track-before-detect method. In: Proceedings of the 2014 Advanced Maui Optical and Space Surveillance Technologies Conference.

Vananti, A., Schild, K., Schildknecht, T., 2015. Streak detection algorithm for space debris detection on optical images. In: Proceedings of AMOS Conference, Maui, Hawaii.

Virtanen, J., Poikonen, J., Säntti, T., Komulainen, T., Torppa, J., Granvik, M., Muinonen, K., Pentikäinen, H., Martikainen, J., Näränen, J., et al., 2016. Streak detection and analysis pipeline for space-debris optical images. Adv. Space Res. 57, 1607–1623.

Yanagisawa, T., Kurosaki, H., Banno, H., Kitazawa, Y., Uetsuhara, M., Hanada, T., 2012. Comparison between four detection algorithms for GEO objects. In: Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference, p. 9197.

Yanagisawa, T., Kurosaki, H., Nakajima, A., 2009. Activities of JAXA's innovative technology center on space debris observation. In: Advanced Maui Optical and Space Surveillance Technologies Conference, Poster presentations.

Yanagisawa, T., Nakajima, A., Kadota, K., Kurosaki, H., Nakamura, T., Yoshida, F., Dermawan, B., Sato, Y., 2005. Automatic detection algorithm for small moving objects. Publ. Astronom. Soc. Jpn. 57, 399–408.

Zimmer, P.C., Ackermann, M.R., McGraw, J.T., 2013. GPU-accelerated faint streak detection for uncued surveillance of LEO. In: Proceedings of the 2013 AMOS Technical Conference.