Performance Assessment of Track-Before-Detect Algorithms and Post-Search Refinement Methods

Sophia Matthews
University of Maryland
Tam Nguyen
University of Maryland

ABSTRACT

Track-before-detect (TBD) is a method of extending an optical system's sensitivity by adding sequential images along an object's motion to increase the effective integration time. TBD enables detection and tracking of faint space objects, supporting space domain awareness, astronomy and planetary science. Typically, TBD is performed with the nearest pixel shift-and-add algorithm, which can be computationally intensive, especially when used for large-scale blind searches. The fast discrete X-ray transform (FaXT) is an alternative TBD algorithm with slightly less accuracy than nearest pixel shift-and-add but can perform the same search orders of magnitude faster. The time complexities of the algorithms are well understood, but the differences in astrometric and photometric accuracy of the two algorithms have not been well characterized. In this work we examine both algorithms to analyze differences in detection performance. Simulated images are generated and tested for both algorithms with metrics including photometric accuracy, astrometric accuracy, and runtimes. Finally, we use a detection refinement method to improve the accuracy of detections made with both FaXT and nearest pixel shift-and-add. Results show that although FaXT suffers more photometric and astrometric losses than nearest pixel shift-and-add, FaXT's significant runtime savings make the algorithm an appealing option for large-scale blind searches.

1. INTRODUCTION

Detection and tracking of faint space objects with limited prior state knowledge can advance space domain awareness (SDA), astronomy, and planetary science. The sensitivity of an optical system dictates whether an object can be detected with traditional methods [4, 6, 7]. The sensitivity of an optical system is limited by the integration time, the physical properties of the optics and focal plane array, the noise properties of the focal plane array, and background conditions when observations are made [4]. An optical system's sensitivity can be extended with track-before-detect (TBD) [2, 4, 6, 7, 8].

Track-before-detect (also referred to as synthetic tracking or velocity-matched filtering) algorithms add sequential images along an object's motion to improve its signal-to-noise ratio (SNR), enabling the detection of objects too faint to detect in a single image [2, 4, 6, 7, 8]. Demonstrations of TBD methods have successfully detected near-earth and main-belt asteroids [2, 4, 6, 7, 8]. For faint objects with known state information, TBD is performed by adding along an object's velocity to improve SNR [4, 8]. Often, TBD is used when state information is unknown, to search for faint objects in sequential images, referred to as a blind search [2, 6, 7]. Blind searches involve adding images along multiple motion hypotheses to find objects [4, 6, 7] and are typically performed with the nearest pixel shift-and-add algorithm [7], which can be computationally intensive.

The fast discrete X-ray transform (FaXT) is a recently developed shift-and-add algorithm that is significantly faster than traditional TBD methods, enabling large-scale blind searches [4]. The runtime and detection rates of both algorithms have been compared in literature [4], but there has been little quantification of the astrometric and photometric losses associated with both algorithms.

In this work, we evaluate three metrics to quantify and compare the performance of nearest pixel shift-and-add and FaXT: astrometric accuracy, photometric accuracy, and runtimes. Astrometric accuracy refers to the accuracy of the detected position and motion, which we evaluate in terms of RMS error in pixels. Photometric accuracy is the ratio of the signal detected to the total signal available in a series of sequential images. In sections 2 and 3, we introduce the equations that govern both shift-and-add algorithms and perform a deterministic comparison of their performance.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

Sections 4 and 5 are an overview of the image simulation methods and detection refinement algorithm used in this analysis. In section 6, we evaluate the performance of both algorithms by performing blind searches on simulated image data. In addition to evaluating the performance of the TBD algorithms, we investigate the refinement of the detections produced by both algorithms to improve accuracy.

2. TRACK-BEFORE-DETECT ALGORITHMS

TBD algorithms add successive images along an object's motion to improve the object's SNR. Theoretically, shift-and-add can be used to test any motion hypothesis, but blind searches are typically limited to linear motion. Limiting the search to linear motion reduces the complexity of the search space, which in turn minimizes the runtime [2]. In this work, we test linear velocities associated with integer displacements over the images, consistent with the typical search space of a blind search.

In theory, the SNR of a detection can be improved by a factor of $\sqrt{N}$, where N is the number of co-added images [4, 7]. This is shown in Eq. 1 where S is the per-frame signal, and σ_n is the per-frame noise of the image [4, 7]. Integer shift-and-add methods suffer from path quantization loss, i.e., the photometric loss associated with shifting images by integer pixel quantities. Therefore, integer shift-and-add algorithms generally cannot achieve the TBD theoretical SNR improvement. Subpixel shift-and-add is possible but the algorithm requires a Fourier Transform, making it even more computationally intensive than nearest pixel shift-and-add [7]. For this reason, it is not typically used for blind searches and is not used as a point of comparison in this work. Both nearest pixel shift-and-add and FaXT are integer shift-and-add methods and suffer from path quantization loss.

$$SNR_{TBD} = \frac{S\sqrt{N}}{\sigma_n} \tag{1}$$

The difference between nearest pixel shift-and-add and FaXT lies in how the algorithms quantize the hypothesis motion. The nearest pixel shift-and-add algorithm solves for integer shifts by rounding hypothesis motion to the nearest pixel at each frame [7]. FaXT solves for the integer shift in each frame by recursively dividing the hypothesis motion in half. For certain hypotheses, FaXT shifts images along a path further from the hypothesis track than nearest pixel shift-and-add, which may result in greater photometric and astrometric loss. However, FaXT enables a significant runtime reduction for large scale blind searches. The following sections expand on the shifting methods and time complexities of the algorithms.

2.1 Nearest Pixel Shift-and-Add

The nearest pixel shift-and-add algorithm (also known as integer pixel shift-and-add [7]) is an integer shift TBD algorithm. Integer shifts for a displacement hypothesis are determined by rounding the subpixel location of a signal in each frame to the nearest pixel. Eq. 2 describes the shifting path of the nearest pixel shift-and-add method for any given number of frames and displacement or velocity hypothesis, where n is the frame number, d is the total displacement over the images, and N is the total number of frames. Eq. 2 assumes linear motion and integer displacements.

for $n \in \{1,2,...,N\}$ and $d \in \{0,1,...,N-1\}$, provided by Zhai et al. [7]

$$f_{N,Nearest}(n,d) = \left[(n-1)\frac{d}{N-1}\right] \tag{2}$$

The time complexity of nearest pixel shift-and-add in a blind search is $\mathcal{O}(N^5)$, assuming the number of frames, N, is approximately equal to the dimensions of the images in the stack and the dimensions of the velocity search space (i.e. $N \approx N_x \approx N_y \approx N_{v_x} \approx N_{v_y}$) [4].

2.2 Fast Discrete X-ray Transform

FaXT is also an integer shift TBD algorithm that utilizes a divide and conquer algorithm to achieve a significantly lower computational intensity than traditional TBD algorithms, such as nearest pixel shift-and-add. To accomplish this, FaXT quantizes the hypothesis track by recursively dividing the displacement in half as described in Eqs. 3-5, where n is the frame number, d is the total displacement over the images, and N is the total number of frames. The

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

structure of this algorithm requires the number of frames to be a power of two (i.e. $N = 2^m$).

$f_{N,FaXT}(k,d)$ for $d \in \{0,1,...,N-1\}$, provided by Nguyen et al. [4]

$$\begin{aligned} f_{N,FaXT}(1,d) &= 0 \\ f_{N,FaXT}(N,d) &= d \end{aligned} \tag{3}$$

for $k = 2^p$ where $p \in \{m-1, m-2, ..., 1\}$, provided by Nguyen et al. [4]

$$\begin{aligned} f_{N,FaXT}(k,d) &= \left\lfloor \frac{f_{N,FaXT}(2k,d)}{2} \right\rfloor \\ f_{N,FaXT}(k+1,d) &= \left\lceil \frac{f_{N,FaXT}(2k,d)}{2} \right\rceil \end{aligned} \tag{4}$$

All remaining values of f_N can be filled in for $k = 2^p$ where $p \in \{3, ..., m-1, m\}$ for $i = 2, 3, ... \frac{k}{2} - 1$

$$f_{N,FaXT}\left(\frac{k}{2} + i, d\right) = f_{N,FaXT}(i,d) + f_{N,FaXT}\left(\frac{k}{2} + 1, d\right) \tag{5}$$

The time complexity of FaXT is $\mathcal{O}(N^4)$ for a blind search where the number of frames, N, is approximately equal to the dimensions of the images in the stack and the dimensions of the velocity search space [4]. FaXT was designed to improve the computational intensity of large-scale, blind searches. The algorithm restricts the hypotheses to linear, integer displacements ranging from 0 to $N-1$ over the images, consistent with the typical search space of a blind search [2, 4]. While the structure of FaXT's shifting algorithm enables low runtimes, the algorithm can also result in more astrometric and photometric loss than nearest pixel shift-and-add. The source of these losses are explored in the following sections.

3. DETERMINISTIC COMPARISON OF TBD ALGORITHMS

FaXT and nearest pixel shift-and-add both suffer from path quantization loss as a result of adding images along their integer shifting paths. Fig. 1 shows two examples of the shifting paths used by each algorithm generated with Eqs. 2 and 3-5, demonstrating that the algorithms will use the same or different shifting path to co-add images depending on the displacement hypothesis. For example, in the 13 pixel displacement hypothesis over 64 images, the algorithms use different shifting paths; on the other hand, in the 32 pixel displacement hypothesis, the algorithms use the same shifting path. FaXT shifts for the 13 pixel displacement hypothesis are further from the true track than those of nearest pixel shift-and-add. FaXT error from the hypothesis track is always greater than or equal to nearest pixel shift-and-add's error because nearest pixel shift-and-add uses the rounding operator, resulting in shifts as close to the hypothesis displacement as possible without performing subpixel shifting.

The shifting path errors from the displacement hypothesis result in astrometric and photometric losses. When there is sufficient error between the shifting path and a true track in a frame, the signal in that frame will not be added. The magnitude of error that will affect the photometric accuracy of a detection is determined by the sampling of the signal and the signal aperture used to make detections. In other words, if the size of the signal aperture is greater than the RMS error between the shifting and true tracks, photometric losses will be mitigated. When performing TBD, detections are made when the SNR in a co-added image is greater than a given detection threshold. The photometric accuracy of the shifting path determines the detections made by each algorithm.

In the following subsections, we calculate the RMS error from the shifting paths, $f_{N,Nearest}$ and $f_{N,FaXT}$ described by Eqs. 2 and 3-5, and the associated displacement hypothesis tracks over N images. We then calculate the deterministic photometric loss associated with a displacement hypothesis given a signal and signal aperture. Both metrics are calculated to predict the performances of both algorithms.

3.1 Shifting Path Error from Displacement Hypothesis Track

The error between the integer shifting path for both algorithms and the displacement hypothesis is calculated for all integer displacement hypothesis given the number of frames (i.e. For a 64 frame image stack, there are 64 possible

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

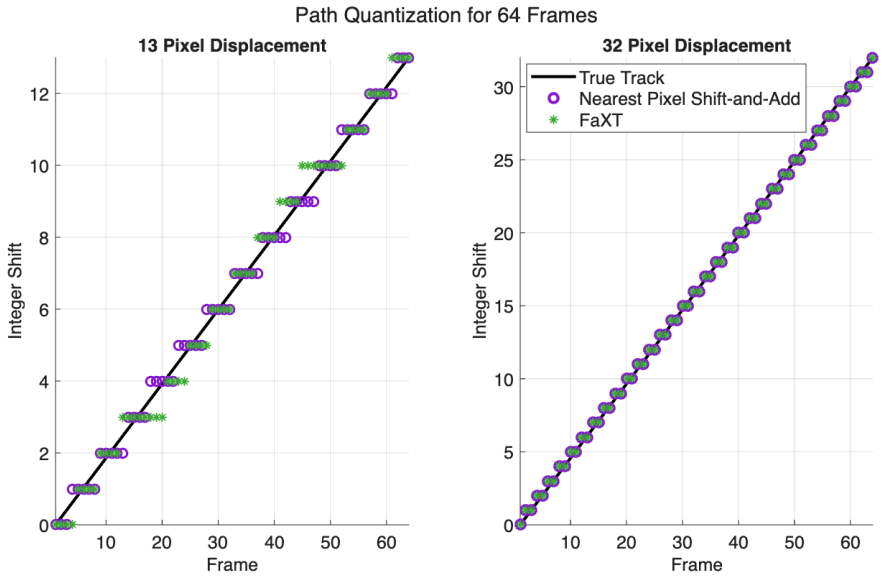


Fig. 1: The FaXT and nearest pixel shift-and-add shifting paths for two displacement hypotheses. FaXT and nearest pixel shift-and-add use different shifting paths for $d = 13$ but use the same path for $d = 32$.

integer displacement hypotheses in x and y, respectively. Therefore, there are a total of 4,096 shifting paths.) The shifting paths were calculated using Eqs. 2 and 3-5. The track error is evaluated in terms of RMS error in pixels, describing the average distance between the hypothesis track and shifting paths in a given frame.

Fig. 2 shows the average track error for every possible integer displacement hypothesis and the associated shifting path for both shift-and-add algorithms over the number of frames in an image stack. Notably, the algorithms use the same shifting path for all possible velocity hypotheses when the number of frames is eight or fewer. Beyond $N = 8$, the average track error diverges, as shown in Fig. 2. Nearest pixel shift-and-add can be thought of as a uniform quantizer with a step size of one pixel. The mean square error of a uniform quantizer with a step size of one is $\frac{1}{12}$ [1]. The average RMS error for the nearest pixel shift-and-add algorithm plateaus at $\sqrt{2/12}$ (≈ 0.41) pixels, which corresponds to the mean square error of a uniform quantizer. The average track error for FaXT increases monotonically with the number of frames. Although the track of FaXT continues to increase beyond the bounds show in Fig. 2, performing a blind search for very large search volumes, is not practical for either TBD method due to computational limits.

Figs. 3a and 3b expand on the data shown in Fig. 2 for 64 frames. The plots show the track error for each integer displacement hypothesis for both nearest pixel shift-and-add and FaXT. There is little variation in the track error associated with nearest pixel shift-and-add, implying consistent photometric loss across displacement hypotheses. Fig. 3b shows FaXT's lower RMS error for power of two displacements. This is a result of the divide and conquer path calculation described in Eqs. 3-5. In fig. 3b, the track error associated with FaXT's shifting paths show greater variation, implying FaXT's photometric losses are highly dependent on the displacement hypothesis. The wide variation over displacement hypotheses, d, can affect astrometric accuracy; if an object's motion is more accurately represented by a displacement that is associated with significant loss, a less accurate displacement may maximize the signal.

The track error is only one indicator of the photometric accuracy of the algorithms. As stated, if the signal aperture encompasses the error of the shifting path, the photometric losses may be mitigated. The following section explores the relationship between photometric accuracy, signal spread, and signal aperture.

3.2 Photometric Accuracy

Photometric accuracy in this work is defined as the ratio of the signal accumulated along the shifting path to the total signal available in the stack of images. Like the error of the shifting path to the displacement hypothesis, photometric accuracy is a function of the number of frames and the velocity hypothesis, although photometric accuracy is also dependent on the point spread function (PSF) of the optical system and the signal aperture used to make detections.

The track error of the shifting path from the displacement hypothesis track does not account for the quantization of

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

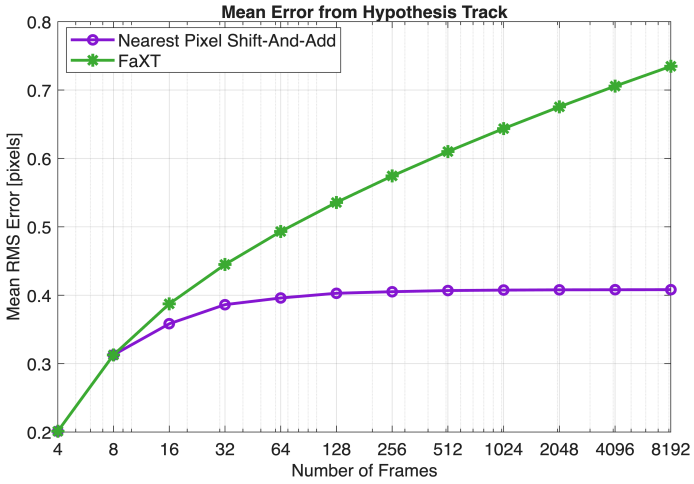
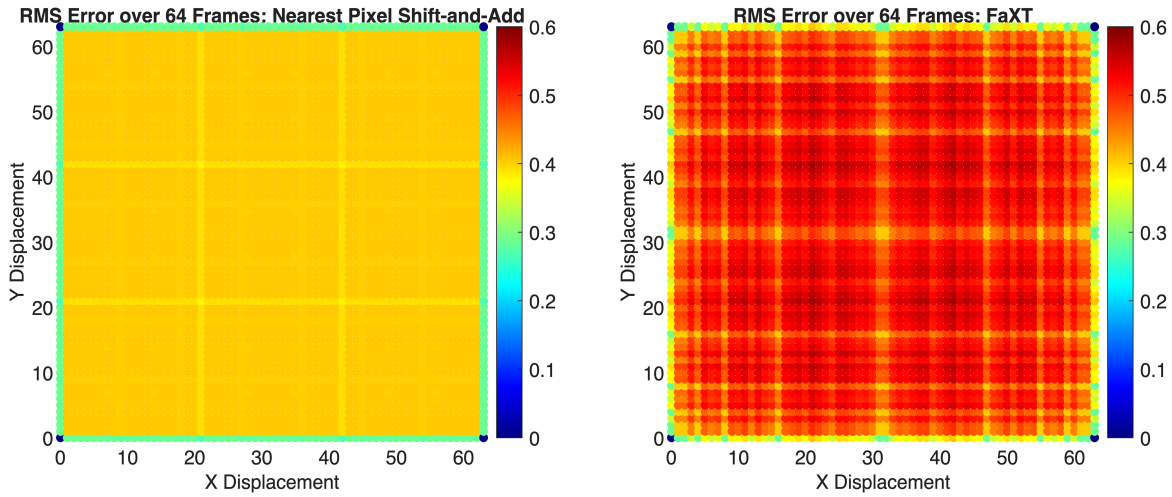


Fig. 2: The average track error of all possible velocity hypotheses given the number of frames in the image stack. This error is a result of the quantization of this shifting path which is inherent in all integer shift-and-add methods. The average track error of FaXT increases monotonically with N, while the average track error of nearest pixel shift-and-add plateaus at the mean RMS error of a uniform quantizer.



(a) Nearest pixel shift-and-add

(b) FaXT

Fig. 3: The track error of TBD algorithm shifting path from displacement hypothesis track for 64 frames. Generally, nearest pixel shift-and-add's shifting paths are closer to the velocity hypothesis than FaXT, implying more accurate astrometry and photometry. For FaXT, power of two displacements over the image stack have lower error values than other displacements.

signals within a frame. The PSF of a signal is quantized over pixels resulting in a pixel response function (PRF) in each frame. To calculate the deterministic photometric accuracy of the algorithms, we modeled a PRF moving through an image stack at every possible displacment hypothesis for 64 frames. We then summed the signal within the signal aperture along both the nearest pixel shift-and-add and FaXT shifting paths. An integrated circular Gaussian parameterized by standard deviation was used to model the PRF in each frame.

The photometric accuracy was assessed over multiple PSFs and signal apertures, as shown in Fig. 4. Nearest pixel shift-and-add outperformed FaXT at smaller PSF sizes and smaller signal apertures. Larger signal apertures encompass the quantization loss associated with integer shifting, resulting in less photometric loss. In practice, a signal aperture is chosen such that the signal-to-noise ratio is maximized [2]. Fig. 4 provides a baseline for the expected photometric

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

accuracy given the PSF and the signal aperture.

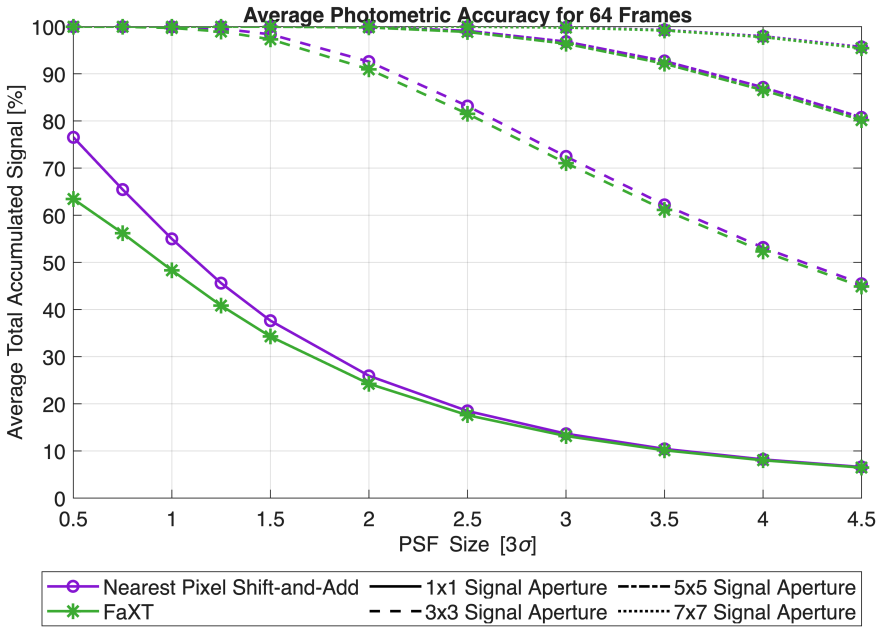


Fig. 4: The average total accumulated signal over every possible integer displacement hypothesis for a 64 frame image stack. The x-axis is the spread of the PSF (modeled as a circular Gaussian) in terms of standard deviation. Four signal apertures were tested. Larger signal apertures result in less loss from path quantization because the aperture encompasses the quantization error. Signal aperture shape and size are determined by the noise conditions and PSF of a given system.

This deterministic analysis of these algorithms focuses on the losses between the shifting path and the displacement hypothesis tracks associated with integer displacements and integer start locations. In practice, signals will move through a series of images with non-integer displacements, and non-integer start locations. This analysis explains the underlying reasons for the difference in performance when nearest pixel shift-and-add and FaXT are applied to more realistic search scenarios.

4. IMAGE GENERATION: SDA-SIM

To compare the performance of nearest pixel shift-and-add and FaXT with, we developed a user-configurable Space Domain Awareness image simulator codebase written in Python, referred to as sda-sim. Sda-sim is capable of generating unresolved images of space objects in various orbits observed by ground-based or space-based sensors. Sda-sim offers a number of options for modeling image noise characteristics and target motion over sequential images. The two methods used for this work are modeling target motion in terms of pixel motion over a series of images and modeling a space object's motion relative to a sensor. All data analyzed in the following sections of this work were modeled with sda-sim.

When modeling objects with pixel motion, we specify the noise characteristics of the images, the PRF, and the linear motion of the signal through a series of images. This modeling option was used to generate data for the analysis of nearest pixel shift-and-add over 64 frames and over N frames detailed in sections 6.1 and 6.2. For this work, the PRF was modeled as an integrated circular Gaussian parameterized by the standard deviation, which was held constant at 0.5 pixels. All signals were modeled with a total flux of 1 and noise was varied in order to vary the per-frame SNR. The pixel motion modeling option is not necessarily representative of real world observation scenarios, but it does allow for thorough testing of target displacements, which was the goal of the analysis in 6.1 and 6.2.

The second modeling option used to generate data for this work is more representative of real world motion. For this modeling option, we define the orbit regime of the targets and sensors. Sda-sim calculates the motion of targets through an image stack using the motion of targets relative to the sensor. For the analysis performed in section 6.3,

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

ground-based sensors and earth orbiting targets, propagated with Keplerian motion, were modeled. The PRF function used was an integrated circular Gaussian parameterized by the standard deviation of the signal. Object motion relative to the focal plane array and noise properties were modeled based on realistic optical system characteristics.

Per-frame SNR is a simulation variable used in all of the following sections. For this work, per-frame SNR is the signal in the signal aperture when the PSF is centered in a pixel. In this work, a 1×1 pixel signal aperture is used for all detections, which is the whole pixel aperture size that maximizes the SNR of the detection across the various PSFs modeled [2]. The amount of signal in one pixel when the PSF is centered in the pixel is determined by the spread of the PSF (i.e. the standard deviation).

5. POST-SEARCH REFINEMENT METHODS

Detection refinement is intended to mitigate photometric and astrometric inaccuracies of a TBD detection. A TBD detection made with an integer shift-and-add method suffers from photometric and astrometric inaccuracies due to path quantization loss and astrometric inaccuracies due to the discrete grid of possible velocity hypotheses and the linear motion assumption. There are a number of means to perform refinement. For this analysis, we minimize a cost function designed to mitigate both astrometric and photometric inaccuracies.

$$C(x_c, y_c, v_x, v_y, \alpha, I_0) = \sum_{n=0}^{N-1} \sum_{x,y} |I_n(x,y) - \alpha \mathrm{PRF}(x - x_c - v_x n, y - y_c - v_y n) - I_0|^2 \tag{6}$$

$$C(x_c, y_c, v_x, v_y, a_x, a_y, \alpha, I_0) = \sum_{n=0}^{N-1} \sum_{x,y} |I_n(x,y) - \alpha \mathrm{PRF}(x - x_c - v_x n - \frac{1}{2} a_x n^2, y - y_c - v_y n - \frac{1}{2} a_y n^2) - I_0|^2 \tag{7}$$

The cost function described in Eq. 6 is modified from the cost function used to refine detections made in Zhai et al. [7]. The function is modified to return the position of the object in the first frame, rather than the mid-epoch, to be consistent with detections made by FaXT. Additionally, this cost function specifies the use of the signal's PRF rather than PSF. In Eq. 7, x_c and y_c are the subpixel coordinates of the signal in the first frame. v_x and v_y represent the velocity of the object in pixels per frame. α represents the total, per-frame flux of the signal. I_0 is a constant background level of the images. Eq. 7 is a further modified version of the cost function with higher order motion terms. The a_x and a_y terms represent the acceleration of the object through the image stack. Eq. 7 can be used when target motion may not necessarily be linear over the image stack.

The optimization method uses the raw output of the TBD method as an initial guess of the detection parameters. The TBD output includes the location in pixels of the detection at the time of the first image, the displacement over the image stack, and the signal flux or SNR. As mentioned, the results of both TBD algorithms are integer locations and displacements. The optimizer solves for non-integer values of the location and displacement.

Over the course of this analysis, we found that the optimization was sensitive to the initial guess for α. For the analysis detailed in section 6, we use the modeled flux value as the initial guess for α. In practice, the SNR of the TBD detection could be compensated with the average assumed loss associated with the integer shift-and-add method used to make detections. Alternatively, subpixel shift-and-add can be used to estimate the initial guess for α by shifting along the motion of the TBD detection.

$$SNR = \frac{\alpha \mathrm{PRF}(x,y)\sqrt{N}}{\sigma_n} \tag{8}$$

The signal-to-noise ratio of the detection refined by the optimizer can be estimated with Eq. 8 [7]. In Eq. 8, α is the total per-frame flux of the signal, $\mathrm{PRF}(x,y)$ is the PSF integrated over the signal aperture, N is the number of frames in the image stack, and σ_n is the per-frame noise scaled by the size of the signal aperture. As stated, we found the optimization to be sensitive to the initial guess for α. If desired, the SNR of the refined detection can be calculated with subpixel shift-and-add using the optimized motion [7]. In this work, we compare the value of α returned by the optimization to the theoretical TBD SNR described in Eq. 1.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

6. RESULTS AND DISCUSSION

Two analyses were conducted to compare the performance of nearest pixel shift-and-add and FaXT. Both utilized data simulated with sda-sim. The first analysis compares performance of the two algorithms over non-integer displacements, non-integer start locations, per-frame SNRs, and search space size. The first part of the analysis focuses on performance when $N = 64$. The second part analyzes performance over N. The second investigation is comprised of two case studies. The case studies are intended to compare the performance of the algorithms for realistic observation scenarios.

6.1 Parametric Sweep Over Fixed Number of Frames

To assess the performance of nearest pixel shift-and-add and FaXT over per-frame SNR, the algorithms are compared over multiple sets of simulated images. Image data is generated using an integrated circular Gaussian with a standard deviation of 0.5 pixels and a total flux of one. Each image stack is made up of 64 images with one signal moving through the series of images. The displacements, start locations, and per-frame noise levels are varied, as described in Tbl. 1.

Number of Frames	N	64
Displacements	d_x or d_y	[0:0.5:63]
Start Locations	(x, y)	[(0,0), (0.2, 0.2), (0.4, 0.4)]
Per-Frame SNR	SNR_n	[0.875, 1.25, 1.75, 2.0]

Table 1: The simulation variables for the parametric sweep. For this analysis, the number of frames was help constant at 64, while the displacement hypotheses, start locations, and per-frame SNR were varied.

Nearest pixel shift-and-add and FaXT are used to make detections across the data set. The same search space is used for both algorithms: all integer displacements from 0 to $N-1$. An SNR detection threshold of seven is used to maintain a false positive rate of 3.5841e-05 for each image stack (with dimensions $84 \times 84 \times 64$), based on Gaussian statistics [7]. A true positive detection is defined as having an RMS error less than or equal to 1.4 pixels. As expected, nearest pixel shift-and-add outperformed FaXT in number of detections, astrometric accuracy, and photometric accuracy.

The theoretical improvement of SNR via TBD algorithms is shown in Eq. 1 [7, 4]. The lowest per-frame SNR tested in this analysis is 0.875, which can theoretically be improved to 7.0 by co-adding 64 frames. To reach this goal there cannot be any path quantization loss. Fig. 5 shows the detection rate for both nearest pixel shift-and-add and FaXT over the per-frame SNR. With a detection threshold of 7.0, the detection rate at the theoretical limit of TBD for the simulated image data is approximately 6% and 3% for nearest pixel shift-and-add and FaXT respectively. The low detection rate is indicative of the losses incurred by both algorithms due to path quantization. FaXT's detection rate is lower than nearest pixel shift-and-add's detection rate, which is consistent with the comparison of the algorithms in [4]. The difference in detection rate is due to the difference in shifting algorithms.

The mean SNR values of the detections are shown in Fig. 6. Fig. 6 shows the theoretical SNR value of a TBD detection given the per-frame SNR calculated with Eq. 1. The detection rates for the 0.875 and 1.25 per-frame SNR cases are low, biasing the mean values of the detection SNR. For cases 0.875 and 1.25, the theoretical detection SNR is compensated using the detection SNR distribution. When the detection rate is near or equal to 100%, the photometric losses are 25% and 30% for nearest pixel shift-and-add and FaXT, respectively.

The photometric losses are consistent with the expected losses shown in Fig. 4. Given the modeling parameters and signal aperture, the maximum amount of signal that can be accumulated in the 1×1 aperture is approximately 47% (area under the modeled PSF within one pixel) of the total signal in the entire image stack. The loss from 47% on Fig. 4 for nearest pixel shift-and-add is approximately 20% and the loss for FaXT is approximately 27%. The losses described in the deterministic analysis only encapsulate the losses from the hypothesis track to the shifting path. For this analysis, there are additional losses from non-integer velocities and non-integer start locations.

There was little variation in track error across per-frame SNR. The mean track error for the nearest pixel shift-and-add detections is ≈ 0.33 pixels, which is approximately the best-case expected RMS between true tracks and discrete representations of tracks, given the simulation setup. The mean track error of the FaXT detections is ≈ 0.37 pixels. As expected, nearest pixel shift-and-add is more astrometrically accurate than FaXT.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

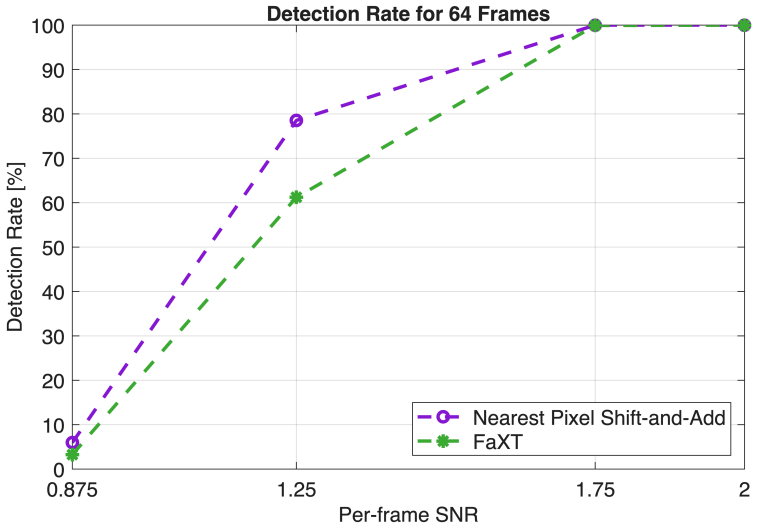


Fig. 5: Detection rate of nearest pixel shift-and-add and FaXT over per-frame SNR. FaXT has a lower detection rate, which is consistent with comparisons of FaXT and nearest pixel shift-and-add in literature [4].

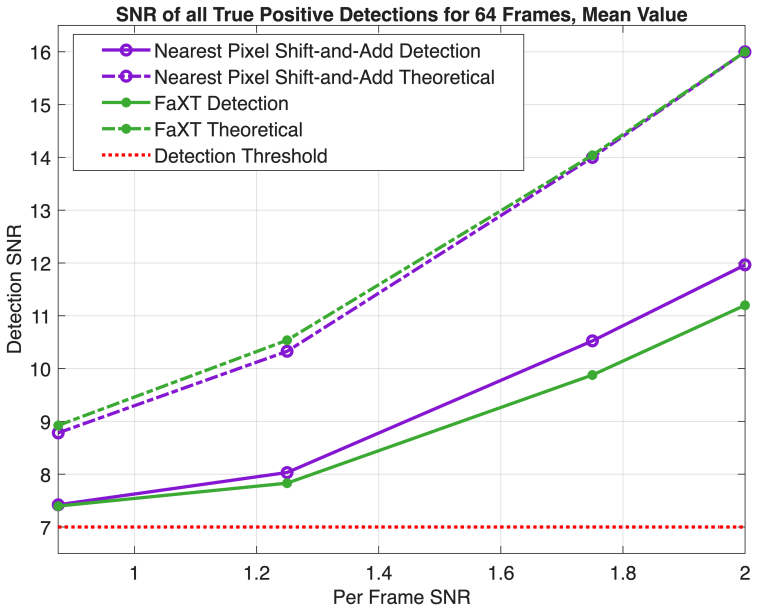
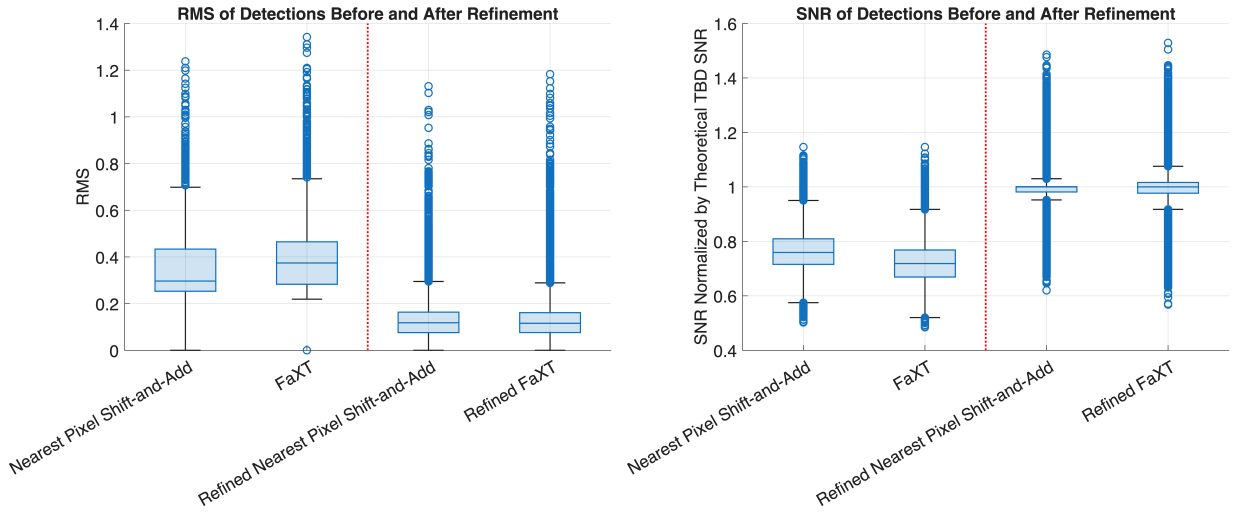


Fig. 6: The mean SNR of the TBD detection over the per-frame SNR. A detection threshold of 7.0 was used, so the detection distributions for the 0.875 and 1.25 per-frame SNR are not complete, biasing the mean of the detections. The theoretical SNR mean was compensated with the SNR distributions of the 0.875 and 1.25 cases.

The detection made by both algorithms were refined by minimizing the cost function described in Eq. 6. Each nearest pixel shift-and-add and FaXT detection has an associated x, y position pair, an x, y displacement pair, and SNR. The position and displacements from the detections were used as part of the initial guess for the optimization. The true value of the signal flux, 1, was used as an initial guess for all optimizations. The results of the optimization are shown in Figs. 7a and 7b. We believe the wide range of alpha values would be improved by tuning the tolerances of the optimizer. More work needs to be done to improve refinement performance. These figures show that, despite the greater inaccuracies of FaXT, the refined detections are comparable in both track error and SNR.

We find that the performance of the algorithms does not change significantly with per-frame SNR. Additionally, the higher track error associated with FaXT does not significantly affect the performance of the refinement algorithm

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com



(a) Track Error

(b) $\mathrm{SNR}_{\mathrm{detection}}/\mathrm{SNR}_{\mathrm{TBD}}$

Fig. 7: Photometric and astrometric accuracy of the detections before and after refinement. Only the 1.75 and 2.0 per-frame SNR cases are shown.

tested. On average, a FaXT detection could be optimized within a few seconds of the runtime of the nearest pixel shift-and-add algorithm. The following analysis details the runtime performance of the algorithms as N, and therefore the velocity search space, increases.

6.2 Runtime Analysis

To assess the runtime of the algorithms, we simulate 1000 image stacks for $N = 64$, 128, and 256. A signal with a per-frame SNR of 4.0 was injected into each stack. The x and y velocities are randomly generated. We chose to analyze 240 image stacks for the $N = 256$ for nearest pixel shift-and-add due to practical constraints of the long runtime.

The astrometric accuracy of both algorithms was consistent over N. Nearest pixel shift-and-add was slightly more accurate than FaXT. The photometric accuracy of nearest pixel shift-and-add was mostly consistent across N; losses were between 28% and 27%, showing a slight decrease over N. The gradual increase in photometric loss follows the trend of track error between the shifting path and hypothesis track for nearest pixel shift-and-add shown in Fig. 2, which gradually increases from $N = 64$ to $N = 256$ and ultimately plateaus at approximately 256 frames. The photometric accuracy of FaXT decreased as the number of frames increased. For 64 frames, the mean FaXT photometric loss was 32%, and for 256 frames the mean photometric loss was 37%. The increase in loss follows the trend of track error shown in Fig. 2.

The runtimes of both TBD algorithms and refinement are detailed in Fig. 8. As expected, as the number of frames in the image stack increases, the runtime of nearest pixel shift-and-add grows significantly faster than FaXT. For a 256 frame image stack, the runtime of nearest pixel shift-and-add is on average approximately 100x slower than FaXT. The runtime of the refinement method does increase with the number of frames. For a 64 frame image stack, the combined runtime of FaXT and refinement is similar to the runtime of nearest pixel shift-and-add. For both a 128 and 256 frame image stack, FaXT and detection refinement can be performed well within the runtime of nearest pixel shift-and-add.

6.3 Case Studies

The following case studies are designed to compare the performances of nearest pixel shift-and-add and FaXT in realistic SDA mission scenarios. The first case study is a ground-based sensor viewing an object in an inclined GEO orbit, enabling the comparison of the algorithms for a single non-linear trajectory. The second case study is a ground-based sensor viewing five objects in LEO. For this case study, the performance of the both algorithms is compared when multiple objects are moving through sequential images.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

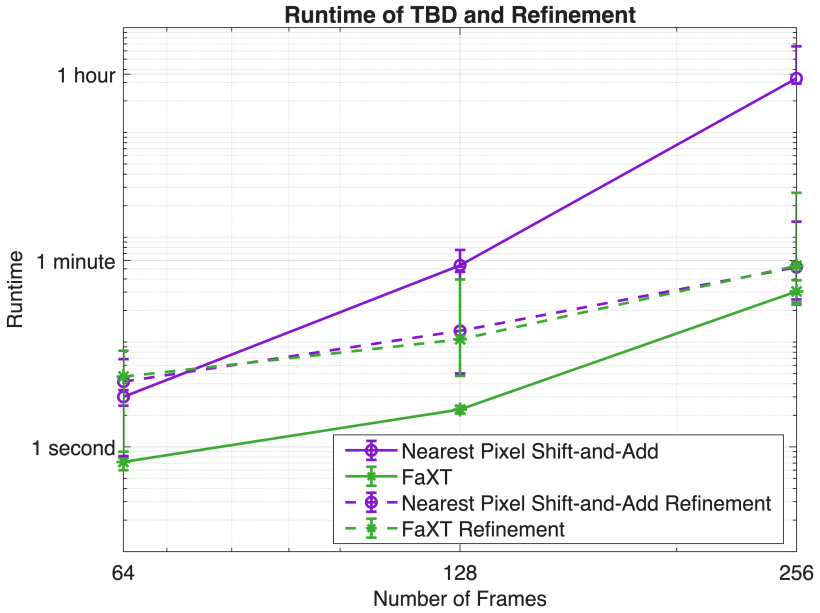


Fig. 8: The runtimes of the TBD algorithms and the refinement of the TBD detections over the number of frames in the image stack. For high values of N the runtime of FaXT remains very low (less than a minute), while the runtime of nearest pixel shift-and-add grows significantly. The refinement runtime for FaXT and nearest pixel shift-and-add detections is nearly identical.

6.3.1 Ground-based Sensor Viewing Object in Inclined GEO

Integer shift-and-add methods assume linear motion along an optical system's focal plane array [2, 4, 6, 7]. The linear motion assumption is true for short arc lengths [2]. The time for which the linear motion assumption can be made depends on the orbit of the observed objects, sensor configuration, and tracking mode [2]. The linear motion assumption can be another source of both astrometric and photometric loss for TBD detections. If an object's motion is not truly linear, the TBD algorithm will return a "linear fit" of the object's motion. In the case of nearest pixel shift-and-add and FaXT, the "linear fit" is the linear displacements associated with the shifting path that accumulates the highest signal. When an object's motion is not linear, both algorithms will accumulate less signal as they add along a nearly linear path and will return detections with less astrometric accuracy because neither account for higher-order motion terms.

A wide field-of-view ground-station observing an object in an inclined geosynchronous orbit was modeled to simulate 256 images. The per-frame SNR of the signal in each image is 1.2, which would result in a theoretical TBD SNR of 19.1, assuming perfect signal accumulation. The motion through the image stack, shown in Fig. 9, is almost linear, but can be described more accurately by a second degree polynomial. The higher order motion of the satellite is more easily seen with the residual of the TBD detections, shown in Fig. 10. Both nearest pixel shift-and-add and FaXT were used to detect the modeled object in simulated images.

The TBD detection results are detailed in Tbl. 2. Both algorithms experience $\approx 50\%$ photometric loss from theoretical. The RMS error of the nearest pixel shift-and-add detection is ≈ 0.19 pixels less than that of the FaXT detection. The lower RMS error from the true track resulted in the slightly higher SNR of the nearest pixel shift-and-add detection.

The detections were refined with the cost function described by Eq. 7 with MATLAB's fminsearch function, which uses the Nedler-Mead algorithm [3]. Both detections were optimized twice. The initial values used for the refinement were the detection location and displacements. The first refinement used the signal of the detection SNR compensated with a 30% loss, which aligns with expected loss of a TBD detection, for the initial value of α. The second optimization used the signal value used to model the data as the initial value of α. We find that the optimization is sensitive to the initial α parameter and, as expected, refinement yielded more accurate results when the modeled signal value was used. The results of the second refinement were nearly identical, as shown in Fig. 10 and Tbl. 2. Both algorithms made the same

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

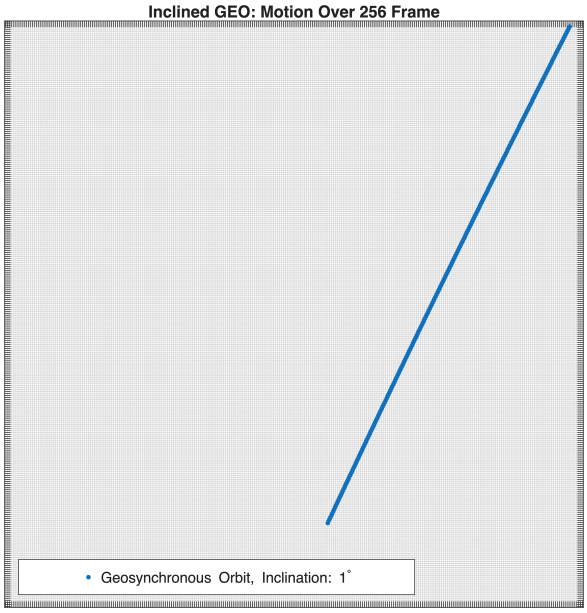


Fig. 9: The motion of the geosynchronous satellite over the 256 frame image stack. While the motion appears mostly linear in this representation, it has non-negligible second-order terms.

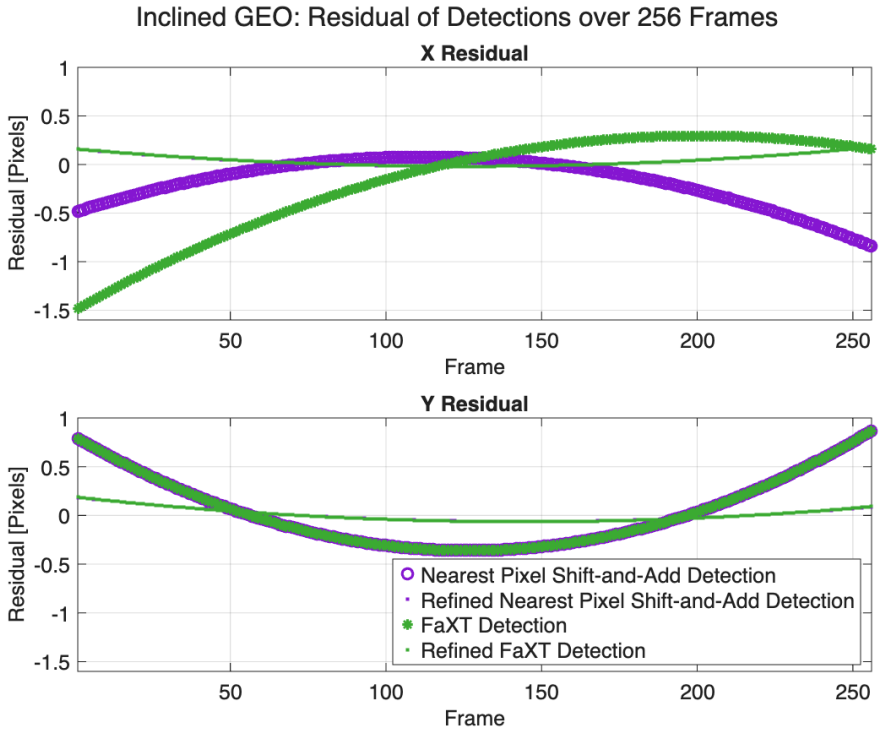


Fig. 10: The residual of the nearest pixel shift-and-add and FaXT detections before and after refinement. The residual error in the initial detections illustrates the extent of non-linearity in the satellite's motion over the image stack. Both algorithms made the same detection in the y direction but different detections in the x detections. The refinement yielded nearly the same results for both detections (the residual lines are on top of each other in both x and y). For both detections, the refined detection residual curve is more flat than the initial detection residual curves, indicating the refinement algorithm was able to recover some of the higher-order motion of the signal.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

TBD-Algorithm	Nearest Pixel Shift-and-Add	FaXT
Detection RMS [pixels]	0.47	0.66
Refined RMS [pixels]	0.10	0.10
$SNR_{detection}/SNR_{TBD}$	0.50	0.48
$SNR_{refined}/SNR_{TBD}$	0.94	0.94
TBD Runtime [Normalized by FaXT Runtime]	18	1
Refinement Runtime [Normalized by FaXT Runtime]	86	25

Table 2: Inclined GEO Raw and Refined Detections

detection in the *y* direction but different detections in the *x* detections. The refinement yielded nearly the same results for both detections (the residual lines are on top of each other in both *x* and *y*). For both detections, the refined detection residual curve is more flat than the initial detection residual curves, indicating the refinement algorithm was able to recover some of the higher-order motion of the signal. The refinement of both detections improve the photometric and astrometric accuracy of the initial detections.

The runtime of the refinement algorithm is a function of many parameters including the dimensions of the image searched, the proximity of the initial detection to truth, and tolerances. A contributing factor to the high runtimes shown in Tbl. 2 are the low tolerances used in the optimization of both detections. For this case, it was possible to significantly improve the astrometry and photometry of the detection within the runtime of the nearest pixel shift-and-add algorithm. Within the runtime of nearest pixel shift-and-add, the FaXT detection was refined resulting in a RMS error of 0.22 and an SNR of 17.17, exceeding the accuracy of the nearest pixel shift-and-add detection.

This case study shows that TBD algorithms can detect objects when motion is not entirely linear over an image stack, though this is highly dependent on the per-frame SNR of the signal and the degree of nonlinearity. The TBD detection can then be refined with higher order motion terms to improve astrometric and photometric accuracy. While the accuracy of the FaXT detection can be improved significantly within the runtime of the nearest pixel shift-and-add search, more work must be done to improve the performance of the refinement algorithm to maintain low runtimes.

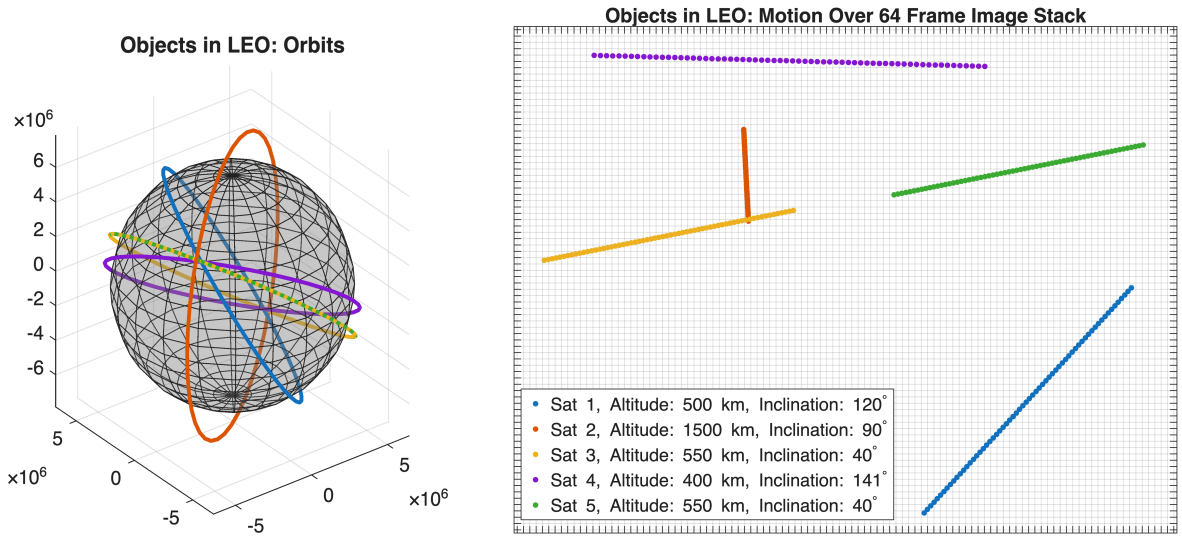
6.3.2 Ground-based Sensor Viewing Objects in LEO

Thus far, the performance of nearest pixel shift-and-add and FaXT have been compared when one signal is injected into the image stack. When performing a blind search in a large volume, there is typically more than one object moving through a scene. For this case study, five objects were modeled to generate 64 images. The per-frame SNR of each object was varied from 1.0 to 1.5. The PSF for signals was modeled as a circular Gaussian with a standard deviation of 0.6. The objects were modeled in various LEO orbital altitudes and therefore have varied displacements along the focal plane array of the modeled sensor, as shown in Fig.11a and Fig. 11b. Satellites 3 and 5 were modeled as trailing satellites in the same orbital plane while the orbital planes of satellites 1 and 5 were randomly selected. Satellite 2 is modeled in a polar orbit.

Satellite	TBD Algorithm	x	y	dx	dy	$SNR_{detection}$	$SNR_{detection}/SNR_{TBD}$	RMS Error [pixels]
Sat 1	Nearest Pixel Shift-and-Add	176	84	33	36	7.37	0.92	0.20
	FaXT	176	84	33	36	6.74	0.84	0.20
Sat 2	Nearest Pixel Shift-and-Add	147	145	1	-15	7.22	0.65	0.30
	FaXT	148	145	0	-15	6.98	0.62	0.37
Sat 3	Nearest Pixel Shift-and-Add	156	132	-40	-8	7.22	0.72	0.51
	FaXT	156	131	-40	-5	8.10	0.82	1.09
Sat 4	Nearest Pixel Shift-and-Add	123	156	63	-1	11.22	0.94	0.43
	FaXT	123	156	63	-1	11.23	0.94	0.43
Sat 5	Nearest Pixel Shift-and-Add	211	142	-40	-7	7.95	0.80	0.33
	FaXT	211	142	-40	-7	8.12	0.81	0.33

Table 3: Detections of Objects in LEO

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com



(a) Orbits

(b) Motion over the image stack

Fig. 11: The orbits and motion over the image stack of the simulated objects.

Tbl. 3 details the detections made by both nearest pixel shift-and-add and FaXT. Over the five objects the algorithms performed similarly, with a mean photometric loss of approximately 20% (consistent with expected loss shown in Fig. 4). FaXT had a higher mean RMS error over the five detections than nearest pixel shift-and-add. As explained previously, the shifting paths for each algorithm can be equivalent or different depending on the displacement hypothesis. For satellite 4, the algorithms make the same detection because the shifting paths for both the x and y displacements are identical for both algorithms. The detections made for satellites 1 and 5 have the same astrometric accuracy but the detections have different SNR values. The difference in SNR is due to the difference in shifting paths for both algorithms for the associated displacements.

For satellites 1, the FaXT detection has a lower SNR than the nearest pixel shift-and-add detection, while for satellites 5, the opposite is true. The deterministic comparison of FaXT and nearest pixel shift-and-add show that, when compared to the hypothesis path, nearest pixel shift-and-add will outperform or match the performance of FaXT in terms of astrometry and photometry. The hypothesis path is a quantization of an objects' true trajectories, so although the RMS error of the nearest pixel shift-and-add shifting paths are less than those of FaXT, there is no guarantee the shifting path of one algorithm will always be closer to an object's true trajectory than the other. For satellite 1, the nearest pixel shift-and-add path is nearer to the satellite 1's true trajectory than the FaXT shifting path. For satellite 5, the FaXT shifting path is nearer to the satellite's true trajectory than nearest pixel shift-and-add's shifting path, explaining the difference in photometric accuracy between the two algorithms for both detections. For satellites 2 and 3 the algorithms associate different motion with the detections. This can be explained by the photometric loss and RMS error associated with specific displacements, as detailed in the deterministic analysis section of this work.

Satellite	TBD Algorithm	$SNR_{detection}/SNR_{TBD}$	$SNR_{refined}/SNR_{TBD}$	Detection RMS Error	Refined RMS Error
Sat 1	Nearest Pixel Shift-and-Add	0.92	1.00	0.20	0.20
	FaXT	0.84	1.00	0.20	0.20
Sat 2	Nearest Pixel Shift-and-Add	0.65	0.78	0.30	0.22
	FaXT	0.62	0.78	0.37	0.22
Sat 3	Nearest Pixel Shift-and-Add	0.72	0.95	0.51	0.14
	FaXT	0.82	0.93	1.09	0.17
Sat 4	Nearest Pixel Shift-and-Add	0.94	1.0	0.43	0.15
	FaXT	0.94	1.0	0.43	0.15
Sat 5	Nearest Pixel Shift-and-Add	0.80	1.0	0.33	0.24
	FaXT	0.80	1.0	0.33	0.24

Table 4: Refined Detections of Objects in LEO

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

The detections were refined using the cost function described in Eq. 6 with SciPy's L-BFGS-B minimization algorithm [5], the same one used to optimize detections in sections 6.1 and 6.2. Tbl. 4 shows the refined results. The optimization method improved the astrometric and photometric accuracy of both the nearest pixel shift-and-add and FaXT detections. Some tuning of the optimization tolerances was required to achieve the results above. Implementing this refinement method in future work will require a more in depth analysis of performance across minimization algorithms and optimization tolerances.

This case study is a demonstration of nearest pixel shift-and-add and FaXT performance when more than one object is moving through a series of images. In this work's previous analysis, the detection threshold of 7.0 was used. Two of the FaXT detections made in this section have an SNR below 7.0, while all nearest pixel shift-and-add detections are above that threshold. While this work has proven that the photometric and astrometric inaccuracies of FaXT can be mitigated, the detection rate of FaXT is lower than that of nearest pixel shift-and-add due to the additional losses FaXT incurs. More work should be done to investigate how to maintain a low false positive rate while lowering a detection threshold.

7. CONCLUSION

We use three metrics to describe and compare the performance of nearest pixel shift-and-add and FaXT: astrometric accuracy, photometric accuracy and runtime. We find that FaXT's unique shifting path results in astrometric and photometric inaccuracies slightly higher than those of nearest pixel shift-and-add. Despite the additional losses, FaXT's runtimes are consistently significantly lower than those of nearest pixel shift-and-add due to the algorithms unique shifting pattern.

In addition to comparing the performance of the TBD algorithms, we investigate the use of a cost function to optimize the results of both algorithms. We find that despite the differences in astrometric accuracy, the refinement method results in similar improvements in astrometry for FaXT and nearest pixel shift-and-add detections. Depending on the tolerances of the optimization and the number of detections, optimizing FaXT detections can be faster than performing a blind search with nearest pixel shift-and-add.

Future work includes investigating methods to improve the detection rate of FaXT, more investigation into detection refinement methods, and testing detection refinement algorithms on real image data. Due to the additional photometric losses incurred by FaXT's shifting path, the algorithm makes fewer detections than nearest pixel shift-and-add with the same detection threshold. Lowering the detection threshold can recover these detections but will increase the false positive rate. Investigating robust methods to reject false positives is a potential way to maintain a false positive rate while lowering a detection threshold. Some manual methods to verify detections are detailed in [2]. Developing robust automated methods of false positive rejection is a current topic of research. Over the course of this work, we found that the refinement method investigated was sensitive to the initial guess. We intend to test more optimization algorithms and tune the tolerances to determine the optimal means of refining detections with the cost functions described in this work. We will also investigate other refinement methods to find a robust means of refining FaXT detections with little additional runtime. We plan to test any refinement methods on real image data to confirm the performance seen with simulated data.

This work expands on comparisons of nearest pixel shift-and-add and FaXT in literature by characterizing the astrometric and photometric loss incurred by the algorithms. Accurate characterization of the algorithms informs usage and design decisions for architectures that rely on track-before-detect methods.

8. ACKNOWLEDGMENTS

This material is based upon work sponsored by the Defense Advanced Research Projects Agency. The content of this material does not necessarily reflect the position of the Government.

9. REFERENCES

[1] R.M. Gray and D.L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6):2325–2383, 1998.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

[2] Aren N. Heinze, Stanimir Metchev, and Joseph Trollo. Digital tracking observations can discover asteroids 10 times fainter than conventional searches. *The Astronomical Journal*, 150(125):25, 2015.
[3] MathWorks. fminsearch. https://www.mathworks.com/help/matlab/ref/fminsearch.html (accessed: Aug. 11, 2026).
[4] Tam Nguyen, Deborah F. Woods, Jessica Ruprecht, and Jonathan Birge. Efficient Search and Detection of Faint Moving Objects in Image Data. *The Astronomical Jounral*, 167(3):113, March 2024.
[5] SciPy. minimize(method='l-bfgs-b'). https://docs.scipy.org/doc/scipy/reference/optimize.minimize-lbfgsb.html (accessed: Aug. 11, 2026).
[6] Michael Shao, Bijan Nemati, Chengxing Zhai, Slava G. Turyshev, Jagmit Sandhu, Gregg Hallinan, and Leon K. Harding. Finding Very Small Near-Earth Asteroids Using Synthetic Tracking. *The Astrophysical Journal*, 782(1):10, February 2014.
[7] Chengxing Zhai, Michael Shao, Bijan Nemati, Thomas Werne, Hanying Zhou, Slava G. Turyshev, Jagmit Sandhu, Gregg Hallinan, and Leon K. Harding. Detection of a Faint Fast-moving Near-Earth Asteroid Using the Synthetic Tracking Technique. *The Astrophysical Journal*, 792(1):60, September 2014.
[8] Chengxing Zhai, Michael Shao, Navtej Saini, Philip Choi, Nez Evans, Russell Trahan, Kutay Nazli, and Max Zhan. Near-Earth Object Observations using Synthetic Tracking. *Publications of the Astronomical Society of the Pacific*, 136(3), March 2024.

Copyright © 2026 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com