An Autonomous Data Reduction Pipeline for Wide Angle EO Systems

Grant Privett, Simon George, William Feline, Andrew Ash, Graham Routledge.

Defence Science & Technology Laboratory, Porton Down, UK.
gjprivett@dstl.gov.uk

Abstract

The UK's National Space and Security Policy states that the identification of potential on-orbit collisions and re-entry warning over the UK is of high importance, and is driving requirements for indigenous Space Situational Awareness (SSA) systems. To meet these requirements options are being examined, including the creation of a distributed network of simple, low cost commercial–off-the-shelf electro-optical sensors to support survey work and catalogue maintenance. This paper outlines work at Dstl examining whether data obtained using readily-deployable equipment could significantly enhance UK SSA capability and support cross-cueing between multiple deployed systems.

To effectively exploit data from this distributed sensor architecture, a data handling system is required to autonomously detect satellite trails in a manner that pragmatically handles highly variable target intensities, periodicity and rates of apparent motion. The processing and collection strategies must be tailored to specific mission sets to ensure effective detections of platforms as diverse as stable geostationary satellites and low altitude CubeSats.

Data captured during the Automated Transfer Vehicle-5 (ATV-5) de-orbit trial and images captured of a rocket body break up and a deployed deorbit sail have been employed to inform the development of a prototype processing pipeline for autonomous on-site processing. The approach taken employs tools such as *Astrometry.Net* and *DAOPHOT* from the astronomical community, together with image processing and orbit determination software developed in-house by Dstl.

Interim results from the automated analysis of data collected from wide angle sensors are described, together with the current perceived limitations of the proposed system and our plans for future development.

1. Introduction

The ever increasing number of satellites and debris circling the Earth is rendering space a congested domain. This is placing increasing demands on Space Situational Awareness (SSA) systems employed to enable the safe and secure operation of national assets and infrastructure. In recognition of this, the United Kingdom (UK) National Space and Security Policy (NSSP) [1] acknowledged the need to undertake work on options that might support the creation of an enhanced UK SSA picture. Several packages of work are underway at the UK Defence Science and Technology Laboratory (Dstl) in support of this aim, including experimenting with the employment of radar systems such as the UK's Science and Technology Facilities Council (STFC) Chilbolton Observatory - a fully steerable 25m radar normally employed for meteorological research - for satellite tracking via upgraded transmitters and receivers.

However, facilities such as Chilbolton require a considerable investment and so are unlikely to form the basis of a widely distributed sensor system. The vast majority of radars also lack sufficient power to detect objects higher than low Earth orbit (LEO). This limits their utility for SSA which, by its nature, is a global problem requiring multiple sensors at widely distributed locations to provide sufficient coverage across a range of orbital regimes. Consequently, an effort has been made to identify electro-optical (EO) systems that might help form a design outline for a small, rapidly deployable and affordable demonstration SSA installation; these include SuperWASP [2], MASCARA [3] and the Dragonfly camera array [4] facilities [5]. These, and others, are being examined to determine how we might build and maintain a system capable of fully automated day-to-day operations and the pipelined analysis. By doing so, data could quickly be made available for use, while gaining confidence and experience of operating remote SSA equipment systems. This experience is essential to support our later interactions with potential contractors.

The activity reported herein pertains to the analysis of data generated by low cost electro-optical sensor options deployed in the UK and New Zealand. The particular focus for this paper is the development at Dstl of an automated image analysis system.

2. Equipment Specification

The imagery studied was obtained by the use of consumer level commercial Canon or Nikkor lenses (typically 20mm or 24mm at f2.8) coupled with Starlight Xpress H18 or Trius SX694 Peltier cooled Charge Coupled Device (CCD) cameras. These cameras use Kodak KAF8300 and Sony ICX694AL EXVIEW chips and provide quantum efficiencies of 0.51 and 0.77 (at 550nm wavelength) respectively. The H18 camera includes a mechanical shutter with an actuation latency of approximately 20ms.

Data were collected at the Dstl range at Porton Down in Wiltshire, England on various dates and at the Lauder Atmospheric Research Station near Otaga, New Zealand (NZ) during the 2015 joint UK/NZ ATV-5 observation trial reported previously [6].

To test a number of shutter techniques and applications, 30GB of imagery was collected against a wide range of target satellites, serendipitous satellites and debris, using a normal electronic shutter and also a mechanical shutter under software control. Consequently, the test dataset contained operational satellites and also systems that were clearly tumbling or with surfaces generating flares in brightness. This variability ensured the software robustness was tested against a varied dataset. Examples are shown in Figure 1.

Figure 1. Example of a single exposure/CCD readout frame with an electronic shutter (left) and (right) a single/exposure CCD readout frame when a programmable mechanical shutter was employed to produce exposures of unequal duration.

3. Methodology

The processing of images to extract and measure trails arising from the passage of moving and/or transient sources has been the subject of study for some years [7, 8, 9]. Over that time a number of approaches to the process have been adopted and tried, but most are based around the same core principles.

1, **Calibrate and "reduce" the data** – apply dark subtraction and flatfielding, mend or mask defective pixels/columns and hot/cold/unreliable pixels.

2, **Determine the location of stars within the image** – apply a technique that highlights the location of sources of a profile similar to the point spread function (PSF) of the optical system used.

3, **Deduce the image background** - map the mixture of sky gradient, Milky-Way, Zodiacal Light and Gegenschein that leads to a sky background that is not uniformly planar and unchanging.

4, **Remove the stars and image background from each image** – thereby leaving only image noise and moving sources.

5, **Locate the presence of trails in the residual image** – look for elongated clumps of associated pixels that are statistically unlikely to be the result of image noise. Depending on the method used to observe the sky (stare mode, sidereal tracking, target-tracking, etc.), discriminating target from star trails using e.g. trail length may be necessary.

6, **Extract orientation, size, the pixel coordinates of the trails start and end points** - calculate the position angle, ellipticity and length of the target track.

7, **Transform the pixel coordinates into Right Ascension (RA) and Declination (Dec) values** – create a polynomial that is exploited to provide precise sky coordinates for each image pixel.

Obviously, many variations and subtle differences in how each stage is executed are possible with the choices made often defined by whether the code is optimised for speed, signal to noise ratio (SNR), robustness of execution or other considerations.

In the work reported herein, the following assumptions were made:

- the optical/CCD systems will be high-end consumer quality, rather than cutting edge,
- multiple, sequential wide-angle image frames are available,
- the targets are larger than CubeSats (SNR >5),
- the observation site is remote and unmanned,
- the bandwidth of the local system may be moderately constrained,
- processing and transmission are to take place in approximately real time,
- processing power is similar to that of a single quad core Intel i7-based laptop,
- the image processing code will be modified in future to lower thresholds and detect fainter streaks.

Consequently, the focus has been on creating a demonstration system that requires minimal computational effort, handles the idiosyncrasies of the sensors used effectively and, where appropriate, employs existing or modified tools and techniques.

4. Results

Data Reduction

The first stage of processing includes the standard data reduction/calibration techniques of dark subtraction and flat fielding [10, 11]. The darks and flats employed are both the result of median stacking more than 10 frames. This approach eliminates the potential impact of cosmic ray strikes against the sensor during the collection of the flat or dark frame.

A refinement which involves the mapping of defective pixels within the CCD array is used. Defective pixels manifest themselves as a few very responsive ("hot") pixels, a similar number of less responsive "cold" pixels and a small percentage of pixels that are deemed unreliable in their response – all arising from manufacturing flaws. These are identified by comparison of the pixel values against the local median value and also the variance in the values returned by each pixel across 10 frames. With the consumer level CCDs used, masking approximately 0.5% of all pixels proved adequate for our purposes.

The second stage involves the detection of stars and objects of generally star-like profile. Several candidates were identified for this purpose including: *Source Extractor* by Bertin and Arnout [12], *DAOPHOT* by Stetson [13] and *PISA* from Starlink [14]. *Source Extractor* and *PISA* were designed for the detection of galaxies while DAOPHOT was designed for crowded field star detection.

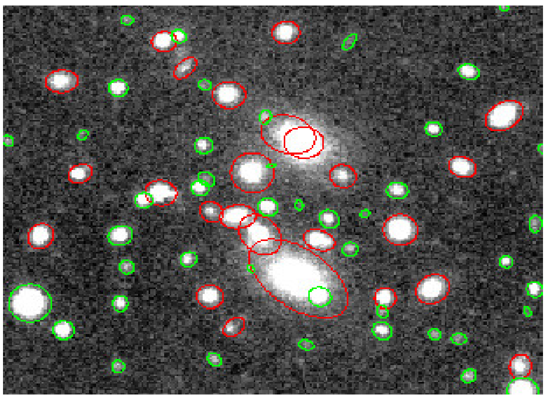

Figure 2. Example of PISA in use against an image frame containing several galaxies. In this instance, members of a galaxy catalogue have been highlighted in red, with other sources in green. Image: Peter Draper.

All three programs execute quite quickly, but in tests *PISA* proved to be by far the most memory efficient. Within the astronomical community *Source Extractor* is often seen as a default good choice and superior to *PISA* and *DAOPHOT,* but its documentation is incomplete, there appear to be unimplemented input parameters and no Python, Matlab or IDL version appears to be available. Similarly, *PISA* is poorly documented FORTRAN code which is not readily amenable to conversion to other languages. *DAOPHOT* is available in FORTRAN and IDL and the code is easy to understand and relatively easy to amend. To test their performance, we undertook a comparison of the packages against isolated dim unresolved targets within normally distributed noise fields. The tests showed they performed very similarly for circular broadly star-like targets and so a modified version of *DAOPHOT*, more tolerant of non-stellar sources, was adopted as our prime detector.

It is worth noting that the detectors worked most consistently when the estimate of image standard deviation (σ) was derived from the width of the histogram rather than from a direct calculation, apparently because non-Gaussian residual instrument noise exaggerates the value of σ. This approach was also found to be more successful than using a less sensitive statistic such as absolute deviation. A clipped mean type metric of the deviation might be an equally viable approach.

The detector was used with a 10σ threshold to determine the location of bright fiducial stars within the image, which were used to identify triangles of stars shared between the frames and thereby infer the transformation and image rotations that were required to align the images to each other. These parameters might be expected to progress linearly from frame to frame, but they were not fit with a linear regression, as winds could affect the pointing.

During the next stage of processing, the image frames were processed in sequence. For each frame in the series, a master image (frame N) was created by median stacking some of the frames immediately before and after it. Each frame then had its own master frame subtracted from it, which lowered the signal from the stars present and greatly reduced (but did not entirely eliminate) much of the unresolved background variability arising from the Milky Way or other light sources – avoiding the need to iteratively assess the image background.

However, it is worth noting that while increasing the number of frames stacked means the star profiles are better estimated, ensuring smaller residuals in the vicinity of stars, it has the disadvantage of smearing any pixel-to-pixel variations unaccounted for by the flat field and dark subtraction, which will appear as noise in the master frame subtracted images. Consequently, there was a balance to be struck between the number of frames stacked, exposure length, the plate scale (and thus optical system employed) and the timeliness requirement for the creation of a processed frame suitable for detection algorithms to work upon. Given the constraints upon processing power likely to be available the pipeline is usually run using median stacking of 5 images i.e. frame N-2 to N+2 when exposures are 1-2 seconds and of similar duration to the download time. Some tuning would be essential for a specific operational system.

Our modified DAOPHOT detector is run against each frame at a 3σ threshold level to determine the location of all the stars on the frame. A second routine then determines the extent of the stars and replaces the pixels covered by a star with a "magic" value which identifies stars potentially contaminating a satellite trail and could be used as part of the error metric.

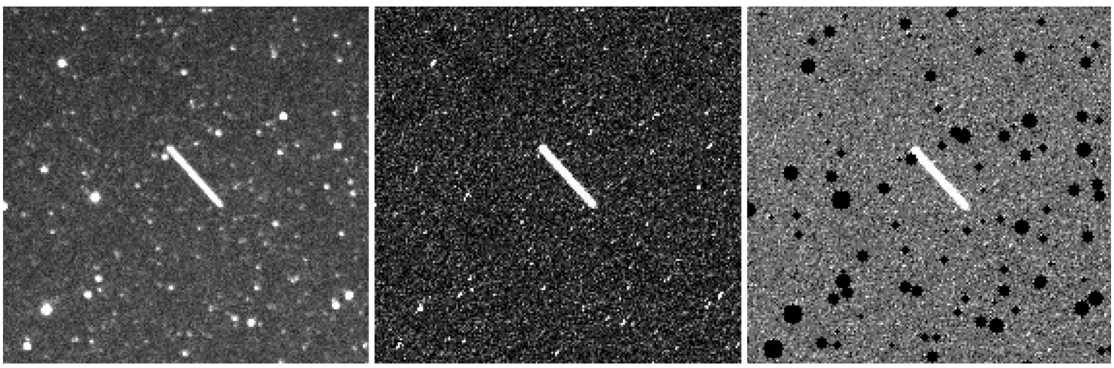

Figure 3. The output at 3 stages of the processing. After dark subtraction/flat-fielding (left), after an aligned image stack is subtracted (middle). After the detected stars have been blanked and the residual background modelled and removed (right.)

The final processing stage is the fitting of a polynomial to each processed image to remove any residual background variation. This has been included to make the system more robust to the presence of cloud, haze and mists.

Track Detection

The frames created by the processing described above are largely free of background variation and the stars present in the scene have been masked out and set to values that allow them to be excluded from subsequent processing. At this point, anything consisting of a few pixels above the image noise is deemed to be either a viable potential detection or simply a statistical outlier.

At this point several approaches for target detection may be considered.

- Use a simple "clump" detector – IDL and MATLAB have one and the Starlink CUPID package [15] has several - that deems any associated group of above threshold pixels to be a potential satellite trail. This has the virtue that potential trail are examined and subjected to tests of ellipticity but is less effective at very low brightness pixels where Hough transforms may do better.

- Employ a Hough (or Radon) [16] transform to examine the image for the presence of distinct line-type features. In such an approach images are converted to a binary form where those pixels above a set threshold are set to 1 and all others to zero. The transform then generates an image from which the location and direction of motion of trails can be identified. However, fully automating this step in a robust manner proved problematic; the nature of the Hough transform technique being highly sensitive to the noise statistics and image artefacts, which makes automated thresholding difficult to implement. Consequently, it is typically cued to process over a constrained sub-image around a trail which has been detected by the simple "clump" detector and simple filtering eccentricity filtering.

- Use a variant of the Hough approach by applying the transformation to the full bit depth image rather than a binary version of the image. This is a more computationally intensive approach, but is more sensitive to the presence of extended low SNR targets. Again it needs to be employed against a sub-image for best results. Unfortunately, it is also has the potential to create more false maxima (alarms) within the transform surface created. In addition, we noticed that when using this approach the MATLAB implementation showed a tendency to provide trajectories slightly misaligned with the trail centres. This is being investigated.

- Apply a simple correlation based detector to generate detection surface images and employ a Track Before Detect (TBD) [17] approach to determine which of the potential detections within the scene is real. This approach is highly computationally intensive and grossly memory inefficient – with most variants requiring one instantiation of the TBD for every suspected target within the scene.

Each of these methods has its virtues and disadvantages and so, combining our constrained hardware specification and the knowledge that 4 satellites in the same frame is quite a common occurrence and our field width, current activity on developing the CPU/memory hungry TBD method were suspended. It was also decided that the benefits and behaviour of the potentially slower 16-bit Hough approach needed a more substantial examination before implementation.

Consequently, the binary Hough transform and the simple clumper methods were adopted as the basis of the current pipeline configuration, yielding a trajectory estimate and the end points of each track within a frame. Both techniques are relatively simple to understand and to code, leaving automation of the configuration the only remaining challenge. The detection process operates on individual image frames independently and on a cropped area of the total image frame, with no *a priori* information about either the number of targets in the scene or the expected position of a known target in the subsequent frame.

Positional Refinement

To support orbit determination, the start and stop locations for each satellite trail must be determined in a manner that takes into account the convolution of the unresolved satellite with the PSF of the optical imaging system. These locations coincide with the only defined times along the satellite trail which can be extracted from the optical image i.e. the beginning and end of CCD charge accumulation or the operation of a mechanical shutter. For our study, CCDs using both electronic and mechanical shutters were employed.

Although the Hough and clumper approaches efficiently identify candidate target trails, their accuracy is typically limited to approximately 1 pixel resolution. For much of the imagery used, this corresponds to an accuracy of 74 arc seconds (equivalent to a 2.4km positional error at a typical LEO altitude satellite), so a refinement step that achieves sub-pixel accuracy is highly desirable. Two methods of end point refinement were explored, using the Hough/clumper outputs as initial estimates of the position and orientation of the trail:

In the first method, a trail profile orientated to the trajectory identified by the Hough/clumper method, and of similar length and intensity is convolved with a 2-D Gaussian function representative of the PSF of the optical system to create a template image. The characteristics of the template image – PSF, trail length, intensity, start point position and orientation – are then iteratively fitted to the observed trail by chi-squared minimisation. The minimisation algorithm employed was a multidimensional downhill simplex minimisation derived from the Nelder-Mead "AMOEBA" [18]. In general, the algorithm converged within 100-150 iterations.

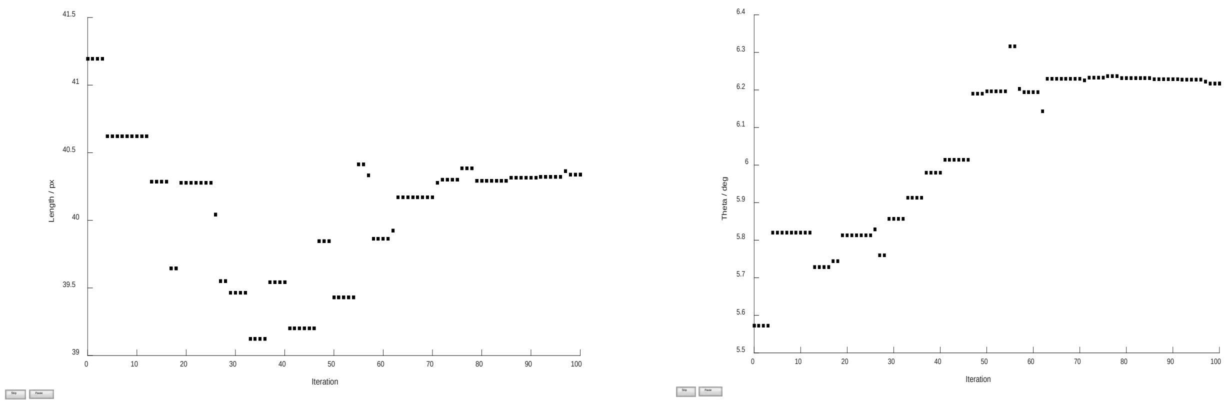

Figure 4. Simplex minimisation examples showing how the value of the angle and trail length parameters vary as the iterative process progressed.

The alternative refinement method examines the trail cross-section at various locations along its length and estimates the optical PSF of the system, which in turn allows the trail orientation to be more precisely determined. This approach makes the assumption that the image PSF is Gaussian, which holds broadly true for all but the most coma-afflicted parts of the original image frame. The variation in pixel intensity along the trail's central axis is then estimated and attempts are made to define the shape of the intensity roll-offs at the ends of each trail, approximately sigmoid in shape with Gaussian-like curvature, in order to estimate the trail start and stop positions.

Both methods have proved effective in refining the initial estimates of the trail delivered by the Hough and clumper approaches, but are sensitive to rapid variations in satellite brightness and may struggle with low SNR trails where photon counts are low. Both methods have proved effective in refining estimates of trail start and end points, but their failings and in particular they will have issues with rapidly tumbling satellites where the intensity profile varies quickly – an example being the CanX-7 solar sail test satellite where, during a recent observation, it was seen to vary in brightness by nearly 3 magnitudes within a second. Further work is needed to make both systems more robust.

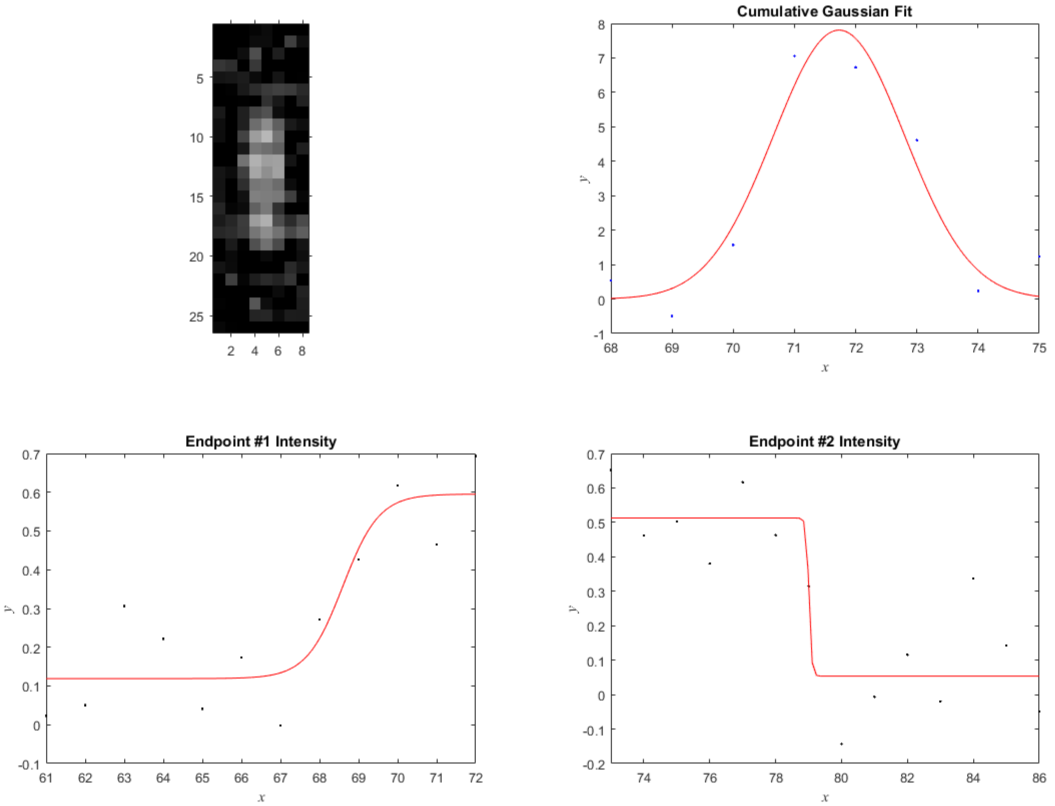

Figure 5. (top, left) cropped region around a single satellite trail; (top, right) Gaussian profile fitted to the across-track profile of the trail; (bottom) Sigmoid functions fitted to the ends of the satellite trail. Note the differing Sigmoid fits assigned to each end; future work will look to model these more accurately based on the known optical PSF. The small number of pixel data points combined with noise remaining in the image background degrades the ability to routinely assign a robust fit.

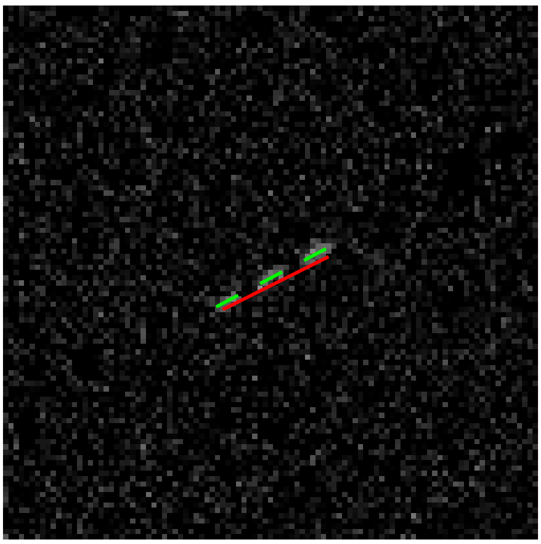

Figure 6. Positional refinement. The red line indicates the streak position estimated by a Hough transform. The green lines indicate the refined positional estimate obtained by chi-squared minimisation. The (cropped) image shown has been pre-processed as described in the text to remove background stars and variations.

Astrometry

Following the application of a successful endpoint-refinement method, the system yields the satellite position in *x*, *y* image space which must be converted to an RA-DEC angle pair. The Linux package *Astrometry.NET* [19], funded by the US National Science Foundation and others, was utilised to perform blind-solving astrometric calibration for our image frames. It employs pattern-matching techniques against accurate star catalogues, comparing triangles formed from stars within the scene with triangles generated using catalogues. Importantly, it does not require any *a priori* information on the image scale and has proved very robust.

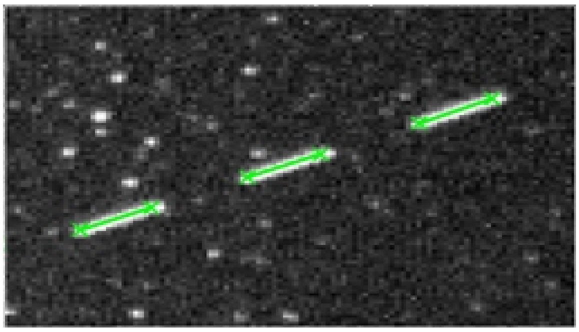

Figure 7. Example 3-streak (6 shutter operations during CCD exposure) trail showing its limits as deduced by the trail cross section refinement method.

To reduce the development time, our current instantiation of the pipeline calls the *Astrometry.NET* web server via its Application Program Interface (API) and supplies small, trail-centred image subsets for solution. The system then involves pattern matching against the many millions of triangles that might be formed by stars in the system database. Our studies found that the accuracy of the final result depends upon a number of factors including the size of the image sample supplied, its location on the image frame, the order and type of the polynomial employed to store the World Co-ordinate System (WCS) transformation.

The astrometric accuracy of the results of the solutions increased as the sub-image size decreased, but appeared to reach an optimum value for sub-images of between 300-500 pixels wide, when the mean astrometric error found for catalogue stars was less: 0.007°, or around 0.3 pixels. The results were also found to be best when using a 4^{th}-order polynomial with minimal improvement observed at higher orders, as might be expected with lenses covering such a wide field of view.

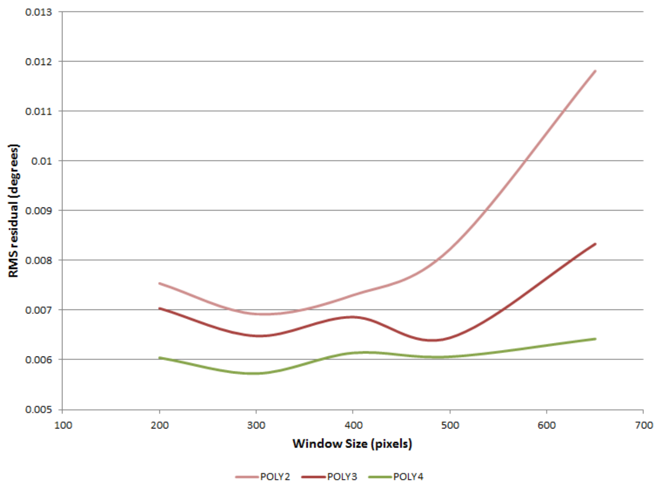

Figure 8. The error residuals (degrees) as a function of size of sub-image employed, for 2^{nd}-, 3^{rd}- and 4^{th}-order WCS polynomial fits. Results were generated for varying window sizes at 4 different locations of the original image; the mean RMS angle error across all 4 locations is displayed here.

Tests of the accuracy against the catalogue position of stars suggests that, as employed, the errors associated with the solution are smaller than the pixel size and comparable with the error associated with the uncertainty in system time. The timing error is associated with a Windows based laptop clock locked to GPS using the NMEATIME2 software rather than via a GPS receiver’s 1pps output.

The components described so far have been combined to form an image analysis pipeline that has been mainly coded in MATLAB with some portions written in IDL. There is clearly considerable scope for refinement and including the use of GPUs and multi-CPU systems, but this should not be undertaken until efficient processes have been identified. The code was written for ease of understanding and coding to minimise effort at the testing and debugging stages. Work is underway to convert the IDL sections into MATLAB to allow a unified system with the likely future development path for the code being as a Python-based system.

5. Conclusion

The components of a simple demonstration pipeline suitable for the automated processing of satellite trails found on imagery sourced from sidereal tracking or fixed wide angle optical sensors have been created. The hardware used for this demonstration lacks the sensitivity to capture most CubeSats [20], but the software would work equally well with a more capable system.

We have demonstrated the use of solutions that are relatively easy to code (and maintain), that are not computationally intensive and which when used with relatively modest hardware could be optimised to provide near real-time data creation.

In addition, the system is expected to be relatively easy to refine and enhance at a later date, including the implementation of a "slow time" processing phase to be employed during the observatory daytime, when more intensive processing may be applied.

Collection of data and the development of the software has provided many useful pointers that will help define a future software system to be employed operationally.

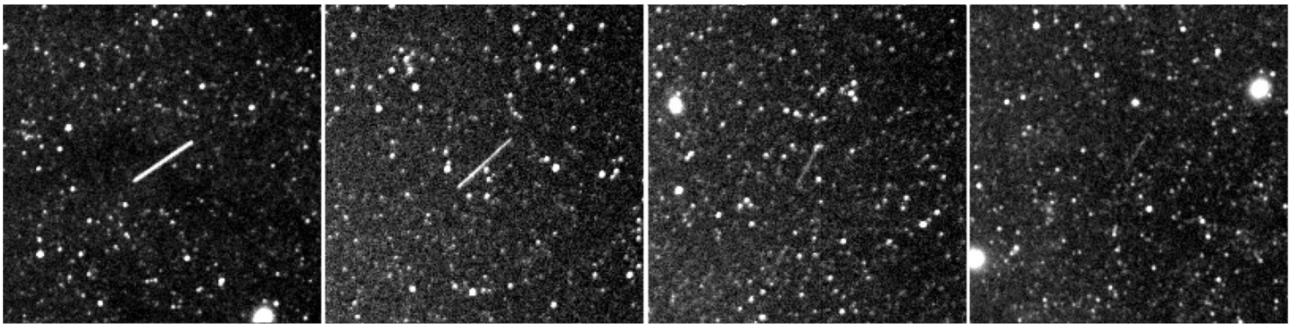

Figure 9. Example detections from 1s exposures in a region of the Milky Way showing trails down to a level where the pixels are only 3σ above sky background brightness.

The current system works only for trails made up of contiguous sets of associated pixels above a 3σ threshold and work is required to examine slow-time analysis techniques to be run when the system is otherwise idle. Future work will also include examining whether the direct integration of camera control into the analysis system provides benefits or whether it can should treated as a separate standalone system. It will also include the examination of track and associate software using Finite Set Statistics and TBD.

6. References

1, Her Majesty's Government, *United Kingdom National Space Security Policy*, Her Majesty's Stationery Office, URN:UKSA/13, 1292, April 2014.

2. Street, R.A., Pollacco, D.L., Fitzsimmons, A., *et al*, *SuperWASP: Wide Angle Search for Planets, Scientific Frontiers in Research on Extrasolar Planets*, ASP Conf. Series, Vol 294, Eds. Drake&Seager, ISBN 1-58381-141-9, 2003.

3, Talens G.J.J., Spronck J.F.P., Lesage A.L., Otten G.P.P.L., *et al*, *The Multi-site All-Sky CAmeRA (MASCARA)*, Astronomy and Astrophysics, Vol 601, May 2017.

4, Abraham R.G., Van Dokkum P., *Ultra Low Surface brightness Imaging with the Dragonfly Telephoto Array*, Publications of the Astronomical Society of the Pacific, 126, 2014.

5, Ackermann M.R., Cox D.D., McGraw J.T., Zimmer P.C., *Lens and Camera Arrays for Sky Surveys and Space Surveillance*, Proceedings AMOSTECH SSA Conference, Maui, 2016. (SAND2016-8077).

6, Ash, A., Donnelly R.P., *The critical Role of Experimentation to Further SSA Understanding*, Proceedings AMOSTECH SSA Conference, Maui, 2015.

7, Stöveken, E., Schildknecht, T., *Algorithms for the Optical Detection of Space Debris Objects*, Proceedings of the European Conference on Space Debris, 2005.

8, Levesque M.P., *Automatic Reacquisition of Satellite Positions by Detecting Their Expected Streaks in Astronomical Images*, Proceedings AMOSTECH, Maui, 2009.

9, Virtanen, J., Poikonen, J., Säntti, T., Komulainen, T., *et al*, T., *Streak Detection and Analysis Pipeline for Space-Debris Optical Images*, Advances in Space Research, 2015.

10, Privett, G.J., *Creating and Enhancing Digital Astro Images*, Springer, ISBN-1846285801, 2010.

11, Howell S.B., *Handbook of CCD Astronomy*, Cambridge University Press, ISBN-9780521617628, 2000.

12, Bertin E., Arnouts S., *SExtractor: Software for source extraction*, Astron. Astrophys. Suppl. Seri 117, 1996.

13, Stetson P. B., *DAOPHOT – A computer program for crowded field stellar photometry*, Astronomical Society of the Pacific, Publications, vol 99, March 1987.

14, Draper P.W., Eaton N., *PISA – Position Intensity and Shape Analysis*, SUN 109, Starlink Project, UK STFC, 2002. AMOSTECH, 2015.

15, Berry D.S., *CUPID – A 3D Clump Identification Package,* SUN 255, Starlink Project, UK STFC, 2016.

16, Duda R.O., Hart P.E., *Use of the Hough Transformation to Detect Lines and Curves in Pictures*, Communications Association for Computing Machinery, Vol 15, January 1972.

17., Salmond D.J., Birch H., *A Particle Filter for Track-Before-Detect*, Proceedings of American Control Conference, Arlington, June, 2001.

18, Nelder, J.A., Mead R., *A Simplex Method for Function Minimization*, Computer Journal. 7: 308–313., 1965.

19, Lang D., Hogg D.W., Mierle K., Blanton M., Roweis S., *Astrometry.NET: Blind Astrometric calibration of arbitrary astronomical images*, The Astronomical Journal, 139, 2010.

20, Ackermann, M.R., McGraw J.T., Martin J.B., Zimmer P.C., *Blind Search for Micro Satellites in LEO: Optical Signatures and Search Strategies*, Proceedings AMOSTECH, 2003.