DEVELOPMENT AND OPERATION OF CICLOPS, THE COMPACT IMPERIAL COLLEGE LONDON OPTICAL SENSOR

E. Kaygorodov[1], M. Hjelmstad Rattigan[2], M. I. Hallgarten La Casta[3], and D. Amato[4]

Imperial College London, London SW7 2AZ, United Kingdom
egor.kaygorodov21@imperial.ac.uk[1],
mathias.hjelmstad-rattigan21@imperial.ac.uk[2],
m.hallgarten-la-casta21@imperial.ac.uk[3],
d.amato@imperial.ac.uk[4]

ABSTRACT

This paper outlines the development of the Raven-class optical telescope CICLOPS (Compact Imperial College London OPtical Sensor), from conceptualization to first light. CICLOPS is a portable telescope providing a testbench for the implementation and validation of novel computational methods for Space Situational Awareness (SSA). The primary objective of CICLOPS is to generate astrometric and photometric measurements of space objects to support both ongoing and future research activities in SSA at Imperial College London. The paper provides an overview of the system hardware and main functional blocks for operations, as well as the associated algorithms for scheduling observations and reducing images. The telescope works in either fixed altazimuth pointing and in ballistic tracking modes. Observation plans are generated for each night through an Observation Scheduling Tool (OST) providing a list of visible candidate objects based on the US Space Catalogue and the ESA DISCOS databse. A full astrometric reduction pipeline has been implemented, including image acquisition, background removal, streak detection, and plate solving. The pipeline operates in near-real-time on a standard laptop, potentially enabling closed-loop tracking operations. An overview of the standard operating procedures during each observation night is provided as well. First light was achieved from central London during February 2025, and an observation campaign of GEO objects is currently underway. CICLOPS will enable the development of algorithms for catalogue verification, advanced state estimation, and behavioural analysis in a theory-to-hardware framework.

Keywords: astrodynamics, Space Situational Awareness, Space Surveillance and Tracking, optical measurements, computer vision.

1. INTRODUCTION

A key challenge facing research groups in the domain of Space Situational Awareness (SSA) specifically and Astrodynamics as a whole is the shortage of accurate and timely data describing the brightness and orbital parameters of live space objects. The development of new uncertainty quantification and orbital propagation methodologies is reliant on the availability of data that can be used for testing and validation. The rise of low-cost portable cameras and astrographs means that individual research groups can now obtain this data themselves, by setting up and developing their own optical systems.

This paper describes the initial design, development and calibration of a fully integrated optical satellite tracking system, the Compact Imperial College London OPtical Station or CICLOPS for short. A combination of an Observational Scheduling Tool with a high-performance image processing pipeline is used with high-precision telescope hardware to develop the capability to perform high-accuracy astrometry and photometry. Observations are conducted from London with significant sources of background light pollution present, and hence a large part of the system design revolved around mitigating this.

The purpose of the system is to be used as a testbench for the Computational Astrodynamics research group at Imperial rather than the creation or maintenance of a catalogue of space debris. Regular measurement and verification of specific objects will enable the testing and validation of newly developed algorithms, including for uncertainty quantification and orbital propagation. The data will also be used to verify and augment space data from commercial and government providers (e.g. through the US Space Catalogue).

The paper occupies serves as a link between several high-level design papers outlining qualitative aspects for a broad range of SSA systems, and low-level literature focussed on the implementation of specific techniques useful for satellite tracking, including proposed background removal algorithms such as those proposed by Popow-

Proc. 9th European Conference on Space Debris, Bonn, Germany, 1–4 April 2025, published by the ESA Space Debris Office
Editors: S. Lemmens, T. Flohrer & F. Schmitz, (http://conference.sdo.esoc.esa.int, April 2025)

icz et Smolka [1] or techniques for improving ballistic tracking results in the TAROT telescope [2]. References such as the work by Coder et al describing the Georgia Tech Space Object Research Telescope [3], as well as the seminal report on the development of Raven-class optical telescope systems [4] brilliantly outline the key qualitative concerns when designing an SSA system, but are not in themselves a direct blueprint. Our work fits into a similar niche to the work by the AGO70 telescope team [5] and the work outlining operations at the Zimmerwald observatory [6]. Both of those systems however are much larger in scale and are not portable configurations.

The intention is to provide a detailed blueprint of how newly acquired optical tracking systems can be set-up, outlining in detail specific concerns that come from a practical implementation and how systems can be adapted to operate in light-polluted metropolitan areas. The capabilities of cooled modern astrophotography cameras and developments in the field of high performance computer vision are leveraged to allow for a real-time image processing pipeline, which is described in detail in this paper alongside a full radiometric model for predicting the system's performance and a detailed description of its operating procedures. We hope to provide insight in setting up similar systems at universities and research institutions, which will allow gathering high-accuracy astrometric and photometric measurements with a low capital investment.

2. SYSTEM DESIGN

2.1. Hardware Specifications

Figure 1 shows the physical assembled system, composed of a tripod, altazimuth mount, and an astrograph including an optical tube as well as a camera.

An 8" Celestron Rowe-Ackermann Schmidt Astrograph (RASA) is used, with an f/2.0 focal ratio with a 203mm aperture and 400mm focal length, providing a 3.2° field of view. It operates in the $390\,\mathrm{nm}$-$800\,\mathrm{nm}$ band and supports up to a 9.1kg camera.

The imaging system employs the Altair Hypercam 26C APS-C astronomy camera, equipped with a Sony IMX571 colour sensor offering a resolution of 6224×4168 pixels and a $3.76\,\mu\mathrm{m}$ pixel size. It features a peak quantum efficiency exceeding 90% and a full well capacity of 51ke^{-}, supporting a dynamic range of 86.8dB. Thermoelectric cooling enables operation 35°C below ambient.

The mount is the 10Micron HPS1000AZ Alt-Az mount, featuring a backlash-free transmission system and dual-axis brushless servo motors. It supports a payload capacity of up to 25 kg and achieves a pointing accuracy better than 20 arcseconds, with an average tracking accuracy of under 1 arcsecond over 15 minutes. The mount operates

Figure 1. CICLOPS assembly without dew shield attached. The assembled stack is about 1.5 m tall.

at slewing speeds of 2°/s to 15°/s. Physical support is provided by a wooden Berlebach Planet tripod.

Additional accessories include a Celestron focuser motor for automated focus control and the mount control box. The camera and mount are connected to a central laptop processing hub through USB-C.

2.2. Scheduling and Mount Control Workflow

The full workflow of the CICLOPS system is presented in figure 2. Once an observation window has been defined based on weather conditions, the Observation Scheduling Tool described in section 3 is used to generate a list of candidate satellites that satisfy geometric and optical visibility constraints. The user selects the desired targets and retrieves their orbital data and TLEs from the tool, which is then imported into the mount control software.

On the night of observation, a satellite database containing the TLEs of the selected candidates is first loaded into the mount. The mount software uses this information to compute the predicted passes, the start times in UTC are then displayed on the user interface. Upon selecting a pass, the mount slews to the satellite's position and initiates tracking along the computed trajectory. The mount control box continuously calculates and issues the required slew commands using real-time positional data from the GPS module, combined with feedback from the mount's internal encoders, to maintain accurate tracking throughout the observation.

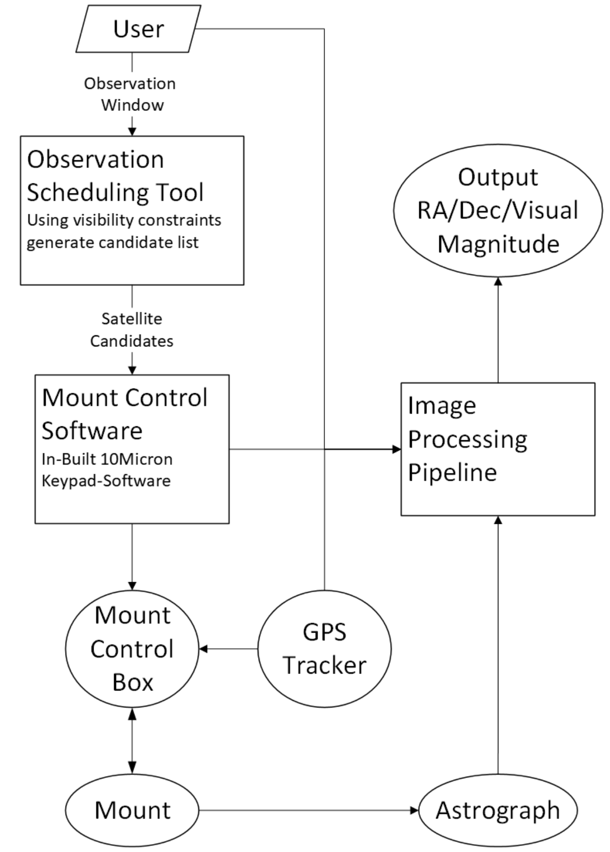

Figure 2. Workflow of the CICLOPS System

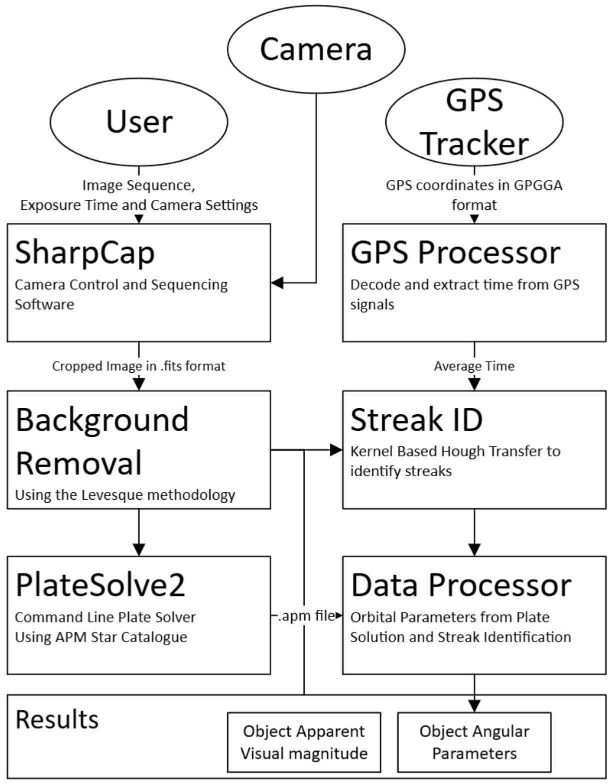

Figure 3. Flowchart describing the key elements of the Image Processing Pipeline block in figure 2.

2.3. Image Processing Pipeline Overview

Conducted astrometry and photometry on raw image data acquired by the telescope requires the implementation of an image processing pipeline. The system is designed to work according to either fixed altazimuthal pointing or ballistic tracking mode.

The fundamental purpose of the image processing pipeline outlined in figure 3 is to command the camera to take a raw astronomical image, and to astrometrically reduce the image by extracting the Right Ascension and Declination of a space object in the Geocentric Celestial Reference Frame (GCRF).

A detailed outline of the specific data flow is provided in section 4, which goes into detail on the techniques used to reduce the image and to deal with the noise sources present.

3. OBSERVATION SCHEDULING TOOL AND MOUNT CONTROL

The Observation Scheduling Tool (OST) serves as a critical component within the CICLOPS workflow, responsible for determining the observation schedule by applying geometric and optical constraints to space object orbit predictions. The development of an in-house scheduling tool ensures direct access to the codebase, enabling iterative improvements and rapid integration of novel computational algorithms for SSA.

Given the portable nature of CICLOPS, the OST requires user-defined input parameters, unlike a fixed ground station where these variables would typically remain constant. The user inputs include observer geodetic latitude and longitude, a horizon constraint angle, the minimum elevation above which satellites must appear to be considered observable, and a site-specific telescope limiting magnitude, whose quantification is performed through the radiometric model developed in section 5. The user also defines an observation time window based on weather forecasts, primarily focusing on acceptable cloud coverage.

After the user-defined parameters are specified, the OST retrieves the most recent Two-Line Elements (TLEs) from the SpaceTrack database [7] and space object dimensions from ESA's DISCOS database [8], resulting in a preliminary candidate list of approximately 15,000 space objects with data including both ephemeris and physical parameters. The orbits are then propagated for the user-defined observation window using the Simplified General Perturbations-4 (SGP4) orbit propagator. SGP4 initially returns satellite positions in a True Equator Mean Equinox (TEME) frame, which is subsequently converted

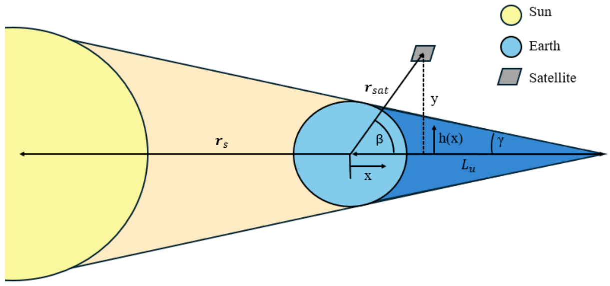

Figure 4. Geometry of Earth's shadow.

to the topocentric East-North-Up frame (ENU) based on the user's geodetic coordinates [9]. With satellite positions on the sky defined through azimuth and elevation in the ENU frame, the OST applies a sequence of visibility constraints to generate the final list of space objects suitable for observation.

3.1. Visibility Constraints

3.1.1. Elevation mask

After orbit propagation and coordinate transformation to the ENU frame, passes are defined as contiguous time intervals during which the space object elevation satisfies:

$$\mathrm{El} \geq \mathrm{El}_{\mathrm{min}} \tag{1}$$

where $\mathrm{El}_{\mathrm{min}}$ is the observer-defined minimum elevation angle. Since preliminary observations are being conducted in London, where urban obstructions significantly limit visibility, a site-specific elevation limit must be defined; otherwise, the default threshold is 0°.

3.1.2. Earth shadow constraint

Satellites become invisible when passing through the Earth's umbra, as they no longer reflect sunlight. The OST models Earth's shadow as a geometric cone extending away from the Sun. The geometry of the umbra is fully determined by the aperture angle of the umbra cone γ and the length of the umbra axis L_u[10]:

$$\gamma = \tan^{-1}\left(\frac{R_S - R_E}{|r_S|}\right) \tag{2}$$

$$L_u = \frac{R_E}{\tan\gamma}, \tag{3}$$

where R_S and R_E are the radii of the Sun and Earth respectively, and r_S is the Sun-Earth distance. The radius of the umbral shadow at a distance x along the Sun-Earth vector is found through

$$h(x) = \tan\gamma \cdot (L_u - x). \tag{4}$$

To evaluate if a satellite is within Earth's umbra, its position vector $\mathbf{r}_{\mathrm{sat}}$ is projected onto the Earth-Sun line $-\hat{\mathbf{r}}_S$. The satellite's position relative to Earth's shadow axis is determined by:

$$\begin{aligned} x = r_{\mathrm{sat}} \cdot \cos\beta, \quad y = r_{\mathrm{sat}} \cdot \sin\beta \\ \text{where} \quad \cos\beta = -\hat{\mathbf{r}}_S \cdot \hat{\mathbf{r}}_{\mathrm{sat}}. \end{aligned} \tag{5}$$

A satellite is considered within the Earth's umbra—and thus invisible—if the perpendicular distance y from the satellite to the Sun-Earth vector is smaller than the umbra radius $h(x)$ at the satellite's projected position x along this vector.

3.1.3. Optical Visibility Constraint

The visibility of a space object pass is further constrained by the telescope's limiting magnitude. Passes whose apparent magnitude exceeds the telescope's detection threshold are deemed unobservable and subsequently discarded. Precise prediction of apparent magnitude would require detailed knowledge of satellite material composition, orientation, shape, and size; however, due to the extensive number of candidates and practical limitations, satellites are modeled as uniformly reflective spheres with the first-order radiometric model developed in section 5.1.

The telescope's limiting magnitude criterion is therefore expressed as:

$$m_{\mathrm{sat}} < m_{\mathrm{limiting}}. \tag{6}$$

Note that satellites passing within the penumbra, that is the region of space in which the Sun disk is only partially obstructed by the Earth, will be partially sunlit and thus may be visible although with increased magnitude. Future work will include improved modelling of the visible magnitude to take this account and thus expand the length of the satellite pass.

3.1.4. Moon Visibility Constraint

An additional visibility constraint arises from the Moon, whose brightness can significantly impact visibility through Rayleigh scattering of moonlight through the atmosphere. This constraint is implemented by defining a minimum angular separation of 5° between the satellite and the Moon.

Consequently, satellite passes are either shortened, retaining only segments that satisfy the visibility criteria or fully discarded, if no part of the pass meets the criteria. Short duration passes below a 60 second threshold are also discarded to exclude too short arcs (TSAs). This results in observation windows containing only intervals that fulfil all visibility conditions giving actionable observation data that can be fed to the mount control software.

3.2. Propagation and implementation

The Observation Scheduling Tool (OST) was implemented in Python, leveraging the Skyfield library [11] for astrometric computations. Skyfield provides built-in methods for orbit propagation with automatic coordinate transformations, directly converting satellite TLEs into topocentric elevation, azimuth and range (the latter used for expected optical magnitude calculation).

The orbits are propagated for the user-defined observation window using the Simplified General Perturbations-4 (SGP4) orbit propagator. SGP4 initially returns satellite positions in a True Equator Mean Equinox (TEME) frame. Skyfield transforms satellite positions from the TEME frame to the International Terrestrial Reference Frame (ITRF) by first applying a z-axis rotation by the Greenwich Mean Sidereal Time angle giving a pseudo-Earth-Fixed frame. A subsequent correction is applied to account for polar motion giving the position in the ITRF. Skyfield transforms the position in the ITRF frame to the observer's ENU frame by first translating the position vector from the Earth centred origin to the observer's location and then applying a time-dependent rotation based on the observer's geodetic coordinates and observation time. The resulting vector, now expressed in the topocentric East-North-Up frame (ENU) frame, is converted into spherical coordinates, yielding the satellite's elevation, azimuth and range [11], [12].

The TLE for each satellite is stored in a dedicated satellite data class and updates if the element set epoch exceeds a 12-hour threshold. Each satellite object in these classes stores other essential information such as the NORAD catalogue ID, physical dimensions and a list of computed passes, which include time-stamped elevation, azimuth, range and predicted apparent magnitude.

Initially, orbit data covering the entire observation window is propagated. Passes are then defined as described in Section 3.1.1, followed by filtering based on the visibility constraints outlined in Section 3.1.

The OST features a Graphical User Interface (GUI) through which users interact with the program by specifying observation parameters and initiating computations. The GUI provides real-time feedback, detailing each processing step and allowing the user to pause operations if necessary. Upon completion, the interface presents a prioritized candidate list ordered primarily by the rise time of the pass, with secondary prioritization favouring higher elevation passes. Users can select preferred candidates and export pass data in a `csv` file, containing time-series, in UTC, of elevation, azimuth, range and predicted magnitude. Additionally, the GUI exports an image of the pass with rise, set and closest approach to zenith times. An interactive animation further aids user selection by visually displaying predicted satellite trajectories on a sky plot, illustrating their positions throughout the planned observation window.

The OST is designed according to a modular paradigm in order to generate pointing commands to track trajectories generated arbitrarily, e.g. by using debiased TLE data with higher accuracy rather than naive TLEs [13].

3.3. Mount Control

Preliminary observations are conducted using the built-in 10Micron control software. This is done by loading in a satellite database to the mount, the TLEs of the selected candidates. From the TLEs the mount defines their passes which are displayed on the UI. When the pass has been selected, the mount aims at the satellite. If the satellite is not visible yet, the mount aims at where the satellite is expected to appear.

Once the target coordinates are provided the control software transmits the commanded pointing direction to the mount control box, which processes the received signal to convert it to control commands sent directly to the mount.

4. IMAGE PROCESSING PIPELINE

The basic flow of the pipeline shown in figure 3 starts with the SharpCap camera control software being used to operate the camera. The obtained images are then saved as FITS files [14] and are operated on using an algorithm proposed by Levesque [15] which is outlined in detail in section 4.1 to remove their background and noise elements. This is a crucial step as it improves the probability of detecting an observed object and enables shorter exposure times.

The processed images with the background and noise removed from them are then saved and read into a plate solver programme, which uses an internal catalogue of stars in order to match the stars on the image to a location in the sky. This process yields the Right Ascension (RA) and Declination (DEC) of the centre of the image, and provides the parameters required to find the RA and DEC of any point on the image plane.

The processed image held in the background removal program is simultaneously operated on to identify the space objects streaks on the image, which are formed due to the movement of the space object with respect to the direction that the sensor is pointing. This streaking is a feature both in fixed pointing, where the sensor is rotating with the Earth and to a lesser extent in ballistic tracking where some smearing of the space object still occurs. A line-detection technique is used to detect these streaks; their geometric centroid is then found and assigned the RA/DEC value read from the plate solver.

This section outlines this process in detail and describes the specific techniques used to achieve near real-time performance.

4.1. Background removal

The background created in the image by light pollution and other sources is identified and isolated using an iterative algorithm first proposed in [16] and then subsequently refined to use local statistics as opposed to a polynomial fit in [15]. The following is a brief summary of the methodology and implementation of the refined method based on local statistics derived by Levesque et al.; the above papers should be consulted for the detailed implementation.

An initial estimate of the background is first derived by smoothing the image with an iterative, local averaging box filter. The filter computes the first estimate of the background $B_0(u, v)$ at pixel coordinates u, v through

$$B_0(u, v) = \mu_w(I_0(u, v)) \tag{7}$$

where $I_0(u, v)$ denotes the pixel intensity at coordinates u, v. The image coordinate system has its origin in the top left corner, v positive down and u positive to the right, and μ_w denotes a local average obtained using a window of size $w \times w$.

At each iteration, the local mean is calculated using box filters of two window sizes: w_1, 5 times the size of the approximate expected Point Spread Function (PSF) of the streak, and w_2, 10 times the size. The background estimate is then taken to be the element-wise minimum of the outputs of the two filters as per

$$B_i(u, v) = \min\left[\mu_{w_1} I_i(u, v), \mu_{w_2} I_i(u, v)\right]. \tag{8}$$

This allows for a sharp filter that is insensitive to very bright stars. The larger window, upon encountering a star, produces a large local intensity bump with a lower amplitude as the brightness of the star is "smeared" across a larger area, whereas the smaller window produces a smaller local intensity bump but with a greater amplitude[15].

At each iteration, the local standard deviation is found across the image using the smaller window size, which enables for the effects of bright stars to be eliminated via sigma clipping as follows:

$$I_i(u, v) = \min\left[I_{i-1}(u, v), B_i(u, v) + 2\sigma_n(u, v)\right]. \tag{9}$$

Any sources of brightness that are two standard deviations greater than the background estimate are clipped as a result. The algorithm is repeated until the relative difference between subsequent summations of the image pixel values is less than an assigned tolerance ϵ. A value of $\epsilon = 10^{-6}$ is used in the current implementation.

$$\epsilon > \frac{\sum_{i,j} \left(I_i(u_i, v_j) - I_{i-1}(u_i, v_j)\right)}{\sum_{i,j} I_{i-1}(u_i, v_j)}. \tag{10}$$

4.2. High-performance implementation

An aspirational aim driving the design of the image processing pipeline is the ability to track LEO objects, and hence to operate in real-time with closed-loop tracking. Several improvements to the background removal algorithm were developed with the aim of reducing the computational cost.

A key driver of the cost is the large number of convolutions required in the background removal method described in section 4.1. Achieving real-time performance requires leveraging the fact that the local averaging convolutions are not done with a kernel made up of distinct terms, but rather with a box filter consisting of a $w \times w$ matrix of ones.

We reduce computational cost of the background removal by exploiting Integral Images. Integral Images were first proposed for the purpose of applying local averaging filters and finding local standard deviations in digital images [17]; in this work, we follow the methodology for creating an integral image in reference [18]. The key property of such an image is that each pixel on it corresponds to the summation of all pixels above and to the left of the corresponding pixel in the original image.

In a naive convolution the local average $\bar{S}$ around the pixel at coordinates (u, v) is computed as

$$\bar{S}(u, v) = \frac{1}{w^2} \sum_{i=-r}^{r} \sum_{j=-r}^{r} S(u + i, v + j), \tag{11}$$

which requires w^2 multiplications. The key idea is to rewrite the local average as

$$\begin{aligned} \bar{S}(u-1, v-1) = \frac{1}{w^2}(&I_{II}(u + r, v + r) \\ &- I_{II}(u + r - w, v + r) \\ &- I_{II}(u + r, v + r - w) \\ &+ I_{II}(u + r - w, v + r - w)) \end{aligned}, \tag{12}$$

which only requires four operations including the division by w^2. In equation (13), $I_{II}(u, v)$ is the integral image of $I(u, v)$, and $r = (w - 1)/2$ is the radius of the averaging window. This means that in order to calculate the local average around a chosen pixel, instead of performing w^2 multiplications as would be required with a naive convolutional approach shown in equation (11), only 4 operations are required as can be seen from equation (13).

An additional cost is also incurred from having to calculate the integral image itself. This cost only needs to be

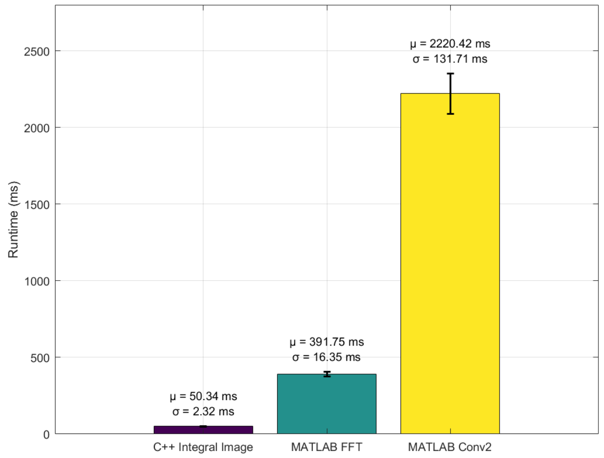

Figure 5. Runtime comparison of several different convolution algorithms for images of size 512 × 1024.

incurred once per iteration of the background removal algorithm, as the same integral image can be used for both the small and large box filters. The following equation from reference [18] is used to calculate it:

$$
\begin{aligned}
I_{II}(u,v) = (&I_{II}(u,v-1) \\
&+ I_{II}(u-1,v) \\
&+ I_i(u-1,v-1) \\
&- I_{II}(u-1,v-1))
\end{aligned}, \tag{13}
$$

Using integral images enables the computational cost of local averaging to be reduced from $\mathcal{O}(w^2)$ per image pixel to $\mathcal{O}(1)$ per image pixel. The corresponding reduction of computational time for a test image is shown in figure 5, representing the gain both from using this method and from switching to a more performant programming language.

As in [15] the edges and corners are handled by dynamically altering the size of the convolution kernel around them. Smaller window sizes are used in order to avoid edge artefacts.

Table 1 demonstrates that the largest computational cost comes from the background removal algorithm. Further performance improvements are possible by using parallelisation to accelerate the calculation of the summed area tables and their subsequent manipulation. Levesque's algorithm lends itself very well to parallelisation due to having a comparatively small program initialisation overhead, with most of the cost coming from matrix convolution.

4.3. Streak Identification Pre-Processing

Streak identification is performed once the background removal is completed. In fixed pointing the satellite streak is likely the only streak in the image, whereas in ballistic tracking care needs to be taken to avoid false positives created by the stars being streaked. The image with background removed must first be converted into a binary mask via the following clamping process:

Figure 6. Star field test image overlaid with synthetic streak.

$$
I_b(u,v) = \begin{cases} 0, & \text{if } I'(u,v) < \mu'_I + 2\sigma'_I \\ 255, & \text{if } I'(u,v) > 255 \\ I'(u,v), & \text{otherwise} \end{cases}, \tag{14}
$$

where I_b and I' are the binary mask and the background-less image respectively and μ'_I and σ'_I are the global background-less image pixel mean and standard deviation. I_b is then discretised by rounding its pixel values to the closest integer.

To improve the effectiveness of the line identification algorithm used for identifying the streaks, the approach suggested in [19] is used. A single pass of the local averaging box filter is applied, followed by the Canny edge detection algorithm [20, 21]. The implementation of the Canny edge detection algorithm in the OpenCV computer vision library is used in this work [22].

The Canny edge detector is used due to the fact that atmospheric diffraction causes the satellite streaks to be several pixels wide in an astronomic image. Naive line identification on such an image will return several artefact lines. The test image demonstrated in 6 is based on the overlaying of a synthetic satellite streak over a real astronomic image taken by CICLOPS during an observation. Figure 7 demonstrates the effect of applying the background removal and the pre-processing algorithms.

4.4. Kernel-Based Hough Transform

In computer vision applications the Hough transform can be used for detection of lines and curves in images [23]. The fundamental principle of the transform

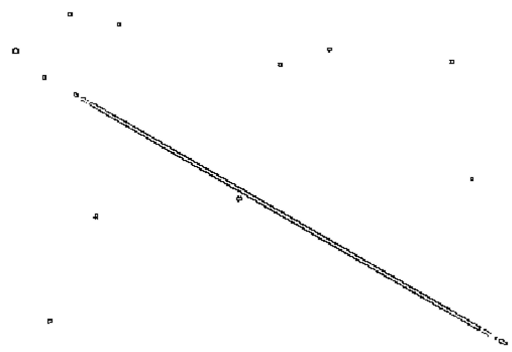

Figure 7. Background removed and pre-processed version of the test image.

Method	Mean (ms)	Std Dev (ms)
Background Removal	154.973	6.235
Canny Edge Detection	16.641	2.886
KHT	10.075	0.801

Table 1. Mean and standard deviation of processing times based on 10 runs of the image processing pipeline.

is to parametrise the image using a voting procedure wherein each significant pixel (obtained by edge detection) "votes" by listing a set of all possible lines going through it, with a certain pre-determined discretisation. The result of the voting is stored in the Hough plane, where each line is parametrised through its distance from the origin ρ and angle from the x-axis θ. If several pixels lie on a straight line, this means that the parametric equation of that line will be "voted" for repeatedly, and hence appear as a peak in the Hough plane. The naive voting procedure is computationally expensive, and dedicated hardware is required to make it run in real time [24], which is unfeasible within the scope of this project.

An alternative called the Kernel-based Hough Transform (KHT) has been developed in reference [24], which uses an elliptical Gaussian kernel to improve the robustness, accuracy and speed of the method. The code from the reference implementation of the algorithm, available from the authors' GitHub[1] was used in order to achieve near real-time detection of satellite streaks, with run-times for 1024 x 1536 images being in the order of 10ms as can be seen in table 1. Detailed discussion of the algorithm is outside of the scope of this paper and can be found in the above paper.

An example of the output of the KHT algorithm can be seen in figure 8, which also demonstrates that the algorithm still has the potential to output spurious identified streaks. We are currently working on eliminating spurious streak detections by grouping approximately similar lines into clusters, and using those clusters to identify significant pixels, as opposed to testing each detected line,

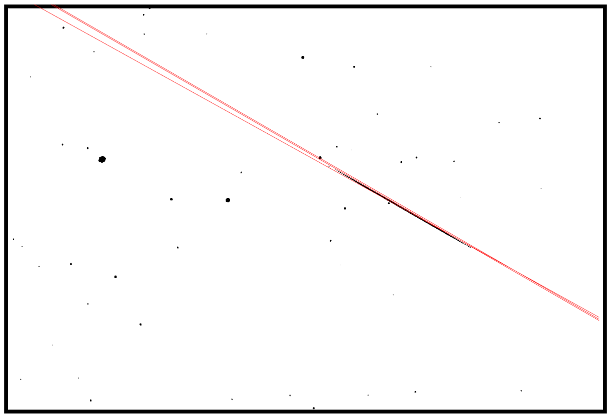

Figure 8. Output of the KHT algorithm (red lines) overlaid on the binary mask of a test image.

which is expected to reduce computational cost and improves robustness. It is expected that the algorithm would only need to output one line, as there is only one streak on the image.

4.5. Plate Solver

The next step is to identify the Right Ascension and Declination of the satellite. Due to the fact that it appears as a streak as opposed to a point, the Right Ascension and Declination of the streak's centroid must be used. The images obtained from the Levesque background removal routine are saved in the FITS format and read into the `Platesolve2` plate solver. To enable automated closed-loop tracking and for efficient calibration of the mount, the software is run via the command line via a script. The format of the input is provided in Appendix A of this paper for the sake of reproducibility, as there is limited documentation available for `Platesolve2`.

Real-time operation of the `Platesolve2` solver requires the OST to produce sufficiently close initial guesses for RA and DEC, as a blind solution or even a solution that has to search through several regions takes significantly longer than is permissible for live tracking. `Platesolve2` uses a spiral search centred on the initial guess, with the Cambridge APM star catalogue [25] used to identify the Right Ascension and Declination of the centre of the image.

The output of the solver is saved to a file with the custom `.apm` file extension, which contains the Right Ascension and Declination of the centre of the image, as well as the angle of the solved plate. This data is then used to correctly label identified satellite streaks. Figure 9 demonstrates a visual representation of the solution, in which red plus symbols represent objects identified as stars, blue crosses represent the expected catalogue star locations, and green circles represent identified stars. Note that green circles are not present on some of the identified stars because the maximum number of

[1]URL: https://github.com/laffernandes/kht, last accessed: 13.03.25

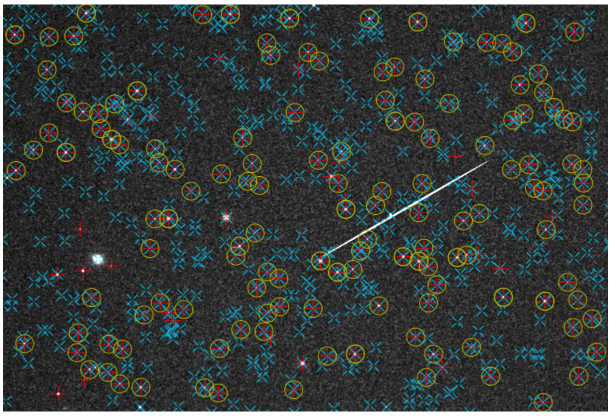

Figure 9. Plate solution for example image.

stars used for the plate solution is a parameter that was set to 45. Increasing this parameter any further does not improve the precision of the solution but just incurs additional computational cost.

4.6. Streak Labelling

The `.apm` file contains several parameters that can be used to find the RA and DEC of any pixel in the image. Ordinarily, the header of a FITS file would contain a CD matrix which could be used to transform the pixel coordinates into RA/DEC coordinates. `Platesolve2` however returns the raw plate geometry instead. The information in the `.apm` file output by `Platesolve2` is used to find the RA and DEC of any arbitrary pixel in the image through a linear relationship,

$$\begin{bmatrix} \text{RA} \\ \text{DEC} \end{bmatrix} = \begin{bmatrix} \text{RA}_\text{c} \\ \text{DEC}_\text{c} \end{bmatrix} + \begin{bmatrix} \frac{\text{FOV}_\text{RA}}{p_w}\cos(\Theta) & \frac{\text{FOV}_\text{DEC}}{p_h}\sin(\Theta) \\ \frac{\text{FOV}_\text{RA}}{p_w}\sin(\Theta) & \frac{\text{FOV}_\text{DEC}}{p_h}\cos(\Theta) \end{bmatrix} \cdot \begin{bmatrix} p_x - p_{x_\text{c}} \\ p_y - p_{y_\text{c}} \end{bmatrix} \tag{15}$$

Where $\frac{\text{FOV}_\text{RA}}{p_w}$ is the appropriate pixel scaling factor for the x-coordinate, obtained by dividing the angular field of view of the plate by the width in pixels of the image, and $\frac{\text{FOV}_\text{DEC}}{p_h}$ is the corresponding scale factor for the y-coordinate, p_x and p_y are the coordinates of the pixel and p_{x_c} and p_{y_c} are the coordinates of the centre of the image.

The geometric centroid of the streak is used as an approximation for the location of the satellite during the exposure. The GPS module coupled with time-stamps on the image enable for the average time during the exposure to be found in UTC and plotted on the visual output, as can be seen in figure 10 which demonstrates the read-out from the image processing pipeline.

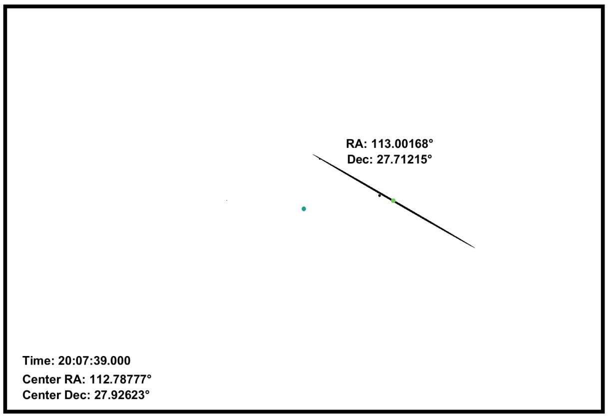

Figure 10. Data pipeline output with identified and labelled synthetic streak. Note times are provided in UTC and are read from the image time-stamp. The Right Ascension and Declination are in the J2000 [25] TEME frame . The centre of the image and centroid of the streak are labelled in green.

5. RADIOMETRIC MODEL

5.1. General Approach

Developing a robust radiometric model for space objects is critical both for developing the Observation Scheduling Tool and selecting appropriate system parameters such as camera exposure time and the required streak detection algorithm sensitivity. Estimating the visual magnitude of orbital objects is crucial towards conducting observations, especially in the high background radiance conditions inherent to London.

In order to determine the relevant system parameters, and use them to estimate satellite visual magnitudes, the methodology in the appendix of reference [26] is used and outlined here.

This approach is a first-order model and should be contrasted with more in-depth approaches undertaken in more detailed models such as [27], which will however require more information on the modelled space objects. Objects are modelled as spherical, with specular reflectivity

$$\rho_\text{spec} = \frac{1}{4\pi} \tag{16}$$

and diffuse reflectivity

$$\rho_\text{diff}(\psi) = \frac{2}{3\pi^2}\left[\sin(\psi) + (\pi - \psi)\cos(\psi)\right], \tag{17}$$

where ψ is the solar phase angle. The apparent visual magnitude of the object is computed according to

$$m_{v,SO} = m_{v,\odot} - 2.5\log_{10} * \left[\frac{A_\text{so} a_\text{so}\left(\rho_\text{spec}(\psi) + \rho_\text{diff}(\psi)\right)}{R^2}\right], \tag{18}$$

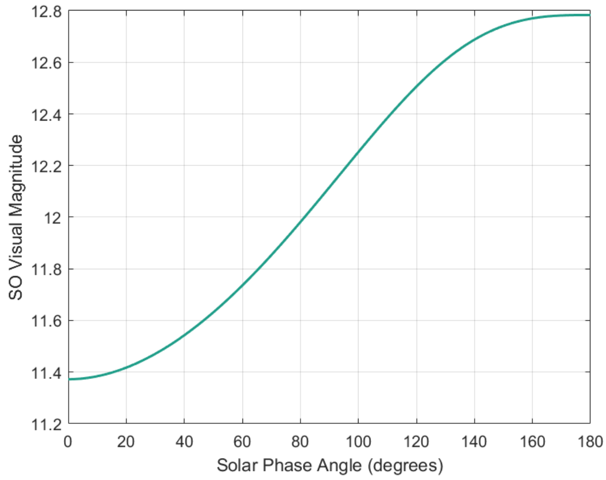

Figure 11. Variation of Sentinel 4 visual magnitude with the solar phase angle.

where R is the range from the observer to the object and $m_{v,\odot} = -26.74$ is the apparent visual magnitude of the sun. The object area A_{so} is obtained from the ESA DISCOS database [8] and the albedo of the satellite, a_{so}, is estimated as 0.175 [28].

To estimate the impact of the variation of solar phase angle, and hence to what extent the exposure time would need to be modified to account for it, a range of solar phase angles were plotted in figure 11 to analyse the impact of varying this parameter. The geostationary satellite Sentinel 4 was selected as a test case for calculating the required parameters, such as the approximate required exposure time.

The driving idea behind the calculations in this section is to calculate the required exposure time to detect an object with a given probability, which requires deriving a relationship between the two.The first step of the process involves calculating the photon flux density from the visual magnitude through

$$\Phi_{\text{SO}} = \Phi_0 \times 10^{-0.4 m_{\text{v,SO}}}, \tag{19}$$

where Φ_0 is the constant value of photon flux density of an object with zero visual magnitude. This value is calculated by assuming that the Sun is a black body using the methodology in [26]. The photon flux captured by the optical system, measured in $\frac{e^-}{s}$ is then given by [26]:

$$q_{\text{SO}} = \Phi_{SO} \tau_{\text{atm}} \tau_{\text{opt}} \left(\frac{\pi D^2}{4} \right) \text{QE}, \tag{20}$$

where D is the telescope aperture, τ_{atm} and τ_{opt} are coefficients related to the transmittance of the atmosphere and the optical assembly and QE is the quantum efficiency of the CCD. Specific values for these parameters can be found in tables 2 and 3.

The τ_{opt} parameter is taken to be equal to the values found in [26, 29] and depends on several complex atmospheric phenomena. A detailed procedure involving ray-tracing of light from the satellite and a full account of atmospheric conditions as described in [30] would enable a more precise estimate but is outside of the scope of this work.

5.2. Variation in Detection Probability

The metric driving the probability of a satellite being detected is the ratio of its signal, as represented by the current induced in the sensor due to photons received from the satellite, to the noise, which is the current induced in the CCD from other sources such as the background light pollution or thermal electron generation.

Successfully detecting a space object requires that its received signal be several times greater than the number of photons detected from natural sources, which corresponds to an object having an SNR past a certain threshold [31]. The approach taken in [26] enables the number of photons q_{sky} due to the background radiance to be obtained from the local background radiant intensity I_{sky} in units of m_v/arcsec^2. The background radiance is first computed through

$$L_{sky} = \Phi_0 10^{-0.4 I_{\text{sky}}} \left(\frac{180}{\pi} \right)^2 3600^2, \tag{21}$$

with the corresponding current induced in the sensor being

$$q_{\text{p,sky}} = \frac{L_{\text{sky}} \tau_{\text{opt}} \pi p^2 \text{QE}}{1 + 4N^2}, \tag{22}$$

where p is the pixel size and N is the ratio between the focal length and the aperture diameter.

Using the CCD equation for the SNR [31], the values calculated above can be used in order to find the signal to noise ratio for the exposure time t,

$$\text{SNR} = \frac{q_{\text{SO}} t}{\sqrt{q_{\text{SO}} t + m_p \left(1 + \frac{1}{p^2}\right) \left[\left(q_{p,\text{sky}} + q_{p,\text{dark}}\right) t + \frac{\sigma_r^2}{n^2} \right]}}, \tag{23}$$

where m_p is the number of pixels occupied by the object, $q_{p,dark}$ and $q_{p,sky}$ are noise due to the background and thermal dark current respectively, n is the binning factor which represents pixels being combined to improve light sensitivity but reduce resolution (set to be equal to one for CICLOPS, representing no binning) and σ_r^2 is the CCD read-out noise. For the chosen Altair Hypercam 26C APS-C camera the manufacturer quotes CCD read noise as 0.79 $e-$, which is significantly less than the noise from the background radiant intensity. Modern cooled astrophotography CCDs furthermore do not experience significant amounts of thermal dark current for the orders of magnitudes of our exposures [32], meaning therefore $q_{p,sky} >> q_{p,dark}$ and $q_{p,sky} >> \sigma_r^2$.

Table 2. Optical System Parameters

Symbol	Units	Value
λ_{upper}	m	6.90×10^{-7}
λ_{lower}	m	4.00×10^{-7}
D	m	2.03×10^{-1}
QE	N/A	9.00×10^{-1}
f	m	4.00×10^{-1}
p_{length}	m	3.76×10^{-6}
h_{pix}	pixels	6224
w_{pix}	pixels	4168
σ_{read}	e^-	0.79
n	-	1

Table 3. Environmental and Optical Parameters

Symbol	Value	Source
τ_{opt}	9.00×10^{-1}	Coder et al. 2016 [26]
A_{SO}	14.2692	ESA DISCOS
α_{SO}	1.75×10^{-1}	Mulrooney et al. [28]
r	3.58×10^{7} m	ESA DISCOS
τ_{atm}	5.00×10^{-1}	Coder et al.2016 [26]
I_{sky}	17.75	[33]

A detailed summary of the parameters used in equations 20, 22 and 23 for the purpose of calculating the photometric parameters for observations undertaken on Imperial College's Silwood Park Campus, using the Sentinel 4 GEO satellite as a target, can be seen in table 2 and are obtained from the datasheets of the manufacturers. Table 3 shows the parameters used in the equation that were not obtained from either the telescope or camera manufacturers' datasheet, and gives the source for each.

The parameter for background radiant light intensity was obtained using the open source lightpollutionmap tool [2] and was found to be approximately 17.75 $m_v/arcsec^2$ [33].

Figures 12 and 13 were then plotted using the equations described above and informed the selection of a preliminary range of exposure times. Figure 13 especially demonstrates that increasing the exposure time yields diminishing returns and that a key parameter when selecting a streak detection algorithm is its threshold SNR. The high intensity of the background light means that observations are made more difficult and longer exposure times are required in order to achieve the same signal to noise ratio.

[2]URL: https://www.lightpollutionmap.info, last accessed: 13 March 2025.

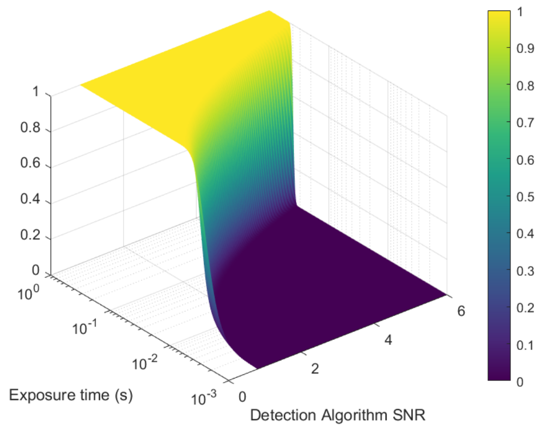

Figure 12. Surface plot of detection probabilities for an object with $m_{v,SO} = 16$.

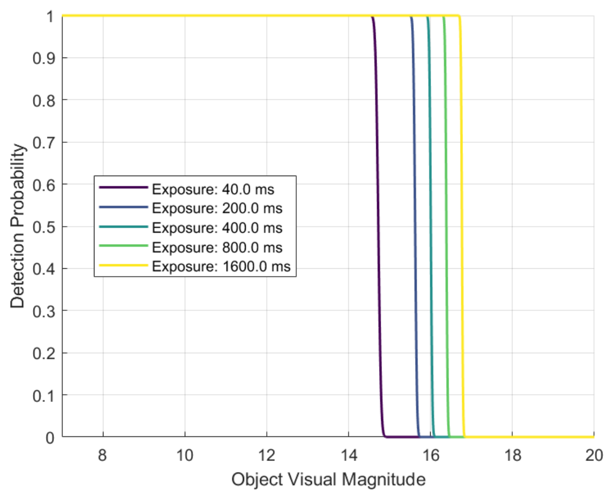

Figure 13. Variation of the detection probability of Sentinel 4 with exposure time for the CICLOPS setup.

6. SYSTEM CALIBRATION AND OPERATIONS

6.1. Optical Calibration

In order to improve the quality of the images taken, and hence the probability of a satellite being detected for a given exposure time, the acquisition of three types of calibration frames is planned, specifically dark, flat, and dark flat frames. All of these frames are re-usable for several months provided that the astrograph assembly is not significantly changed.

For dark frames, the temperature, exposure time and gain must be the same as the light frames, i.e the frames that are produced during the observations. Their purpose is to account for the effects of noise that arises as a result of dark current produced by the thermal generation of electrons during normal operation of the CMOS sensors [34]. Using the radiometric model in section 5 and data from the manufacturers website, it is estimated that the noise from the pixels is negligible compared to the background effects, and hence it is likely that the dark frames will be unnecessary for this experimental set-up.

The purpose of flat frames is to take into account the effects of imperfections in light distribution across the assembly, that may arise as a result of vignetting or of dust on the optics. The technique for taking flat frames involves adjusting the exposure time until the peak of the image pixel intensity histogram is in the middle of the range. Flat darks are used with the same exposure as the flat frames, and are there to correct for thermal effects that may affect the flat frames. Both frames will be taken before any future observation campaigns are conducted with the astrograph

6.2. Mount Calibration

Due to the portable nature of CICLOPS, the system must be manually aligned and levelled for each observation, introducing alignment errors if the base is not perfectly levelled and the east reference direction is not accurately set. Additionally, mechanical deviations such as optical axis misalignment, structural flexure from temperature fluctuations, and inherent orthogonality errors from manufacturing tolerances contribute to pointing inaccuracies. Although the 10Micron AZ1000HPS mount is designed to achieve a pointing accuracy of 1 arcsecond, improper setup can result in inaccuracies as large as 3 degrees, exceeding the system's field of view.

To mitigate these errors, the mount employs a multi-star alignment calibration procedure to construct a correction function to offset the pointing inaccuracies. The calibration procedure involves slewing the telescope to a known reference star, capturing an image and using the plate solver to determine the Right Ascension and Declination at the centre of the field. The user then manually adjusts the pointing direction to centre the image before running the plate solver again, iterating this process until the Right Ascension and Declination match the commanded pointing direction to within 1 arcsecond. This measured pointing error is incorporated into the mount model. This procedure is repeated for at least 10 well-spaced calibration stars, enabling the system to generate a mount model and correction function that compensates for mechanical and alignment errors. The process can be repeated for as few as 3 reference stars however, they must be sufficiently spaced and doing so will reduce pointing accuracy.

The mount provides an estimated root-mean-square pointing accuracy estimation based on the calibration results. However, orthogonality errors- resulting from non-perpendicularity between axes- cannot be corrected through software and must be manually addressed by inserting shims into the mounting system. While this has not been necessary in the current implementation, long term mechanical wear may require future manual correction.

Figure 14. Example full size image taken by the telescope

Additionally, the mount accounts for atmospheric refraction effects by incorporating site-specific temperature and pressure measurements, recorded by the user, into its correction function.

7. DISCUSSION

The CICLOPS system is operational and has seen first light during preliminary testing in February 2025. A range of images with a range of exposures has however been taken to test the image processing pipeline, an example of which can be seen in figure 14.

The system is operational and is capable of acquiring images which can be reduced and plate solved to perform astrometry. The system is currently capable of performing open-loop ballistic tracking and fixed pointing. As the image processing pipeline is operating close to real-time. With additional optimisation of the image processing codebase and GPU parallelisation it is expected to move the image processing towards real-time operation, which will in turn enable closed-loop ballistic tracking.

The next immediate step will be the implementation of a photometric calibration pipeline. An observation campaign to perform astrometry on GEO and LEO objects is planned to take place during spring 2025, taking advantage of better weather conditions in the UK. One of the primary outcomes of the observation campaign will be the robust quantification of uncertainty (bias and noise) affecting the astrometric measurements of the telescope, which will be achieved through the quantification of the PSF associated to individual streaks. We expect the results of the first observation campaign to prove that it is possible to perform astrometry with accuracy typically superior to that of propagated TLEs with a low-cost, commercial astrograph from a light-polluted environment.

8. CONCLUSIONS AND FURTHER WORK

Overall, the CICLOPS system is capable of being operated in sidereal tracking mode in order to obtain angular measurements of objects in Geostationary Orbit, with testing so far being done in fixed pointing mode. The system architecture was designed to operate in ballistic tracking mode, and further work is being done to make this possible.

The image processing pipeline operates in near real-time due to an efficient implementation of Levesque's Background Removal algorithm and the use of the Kernel-Based Hough-Transform for streak detection. Further improvements in order to achieve full real-time performance are being implemented and will leverage GPU-based acceleration using CUDA to achieve results.

The implemented radiometric model enabled for a rough estimate of the required exposure time to be derived, and was used in order to build the constraints for the Observation Scheduling Tool.

The OST currently satisfies the baseline functional requirements to generate candidate satellite lists for preliminary observations. However, due to the limited validated data, the accuracy of its predictions remains unverified. Subsequent development should prioritise optimising computational efficiency, improving the graphical user interface to facilitate candidate selection, adding a mode to identify the next observation window for user-specified space objects, establishing a direct interface with the `pyPOGS` control software and if required revisiting the orbit propagation method used.

Future system operations will incorporate ESA's `pyPOGS` software, which offers integrated mount control, image capture and image reduction. This integration will enabled semi-automated calibration and streamline observation workflows. Planned extensions to `pyPOGS` will enable the import of custom pointing sequences from `csv` files, allowing support for alternative orbit propagation models.

The system produces images that are reduced and plate solved, with Right Ascension and Declinations in the J2000 GCRF frame outputted at a sufficiently high rate to enable real-time tracking of satellites.

ACKNOWLEDGMENTS

This project was funded through the Imperial Institute of Security Science and Technology seed fund. We gratefully acknowledge the assistance of Prof Marcus Holzinger (University of Colorado Boulder) who provided advice on the initial setup as well as test images used for preliminary testing.

REFERENCES

1. Popowicz A, Smolka B. A method of complex background estimation in astronomical images. Monthly notices of the Royal Astronomical Society. 2015 Sep 1;452(1):809-23. Available from: https://www.proquest.com/docview/1701285207.
2. Laas-Bourez M, Blanchet G, Boër M, Ducrotté E, Klotz A. A new algorithm for optical observations of space debris with the TAROT telescopes. Advances in Space Research. 2009 Dec 1;44(11):1270-8. Available from: https://dx.doi.org/10.1016/j.asr.2009.06.013.
3. Coder RD, Jaunzemis AD, Mathew MV, Worthy JL, Holzinger MJ. Georgia Tech Space Object Research Telescope. Journal of spacecraft and rockets. 2017 Nov 1;54(6):1399-403. Available from: http://arc.aiaa.org/doi/full/10.2514/1.A33852.
4. Sabol C, Luu KK, Kervin P, Nishimoto D, Hamada K, Sydney P. Recent Developments of the Raven Small Telescope Program. Advances in the Astronautical Sciences. 2002 Jan 1;112:397-416. Available from: https://search.proquest.com/docview/27678343.
5. Šilha J, Krajčovič S, Zigo M, Tóth J, Žilková D, Zigo P, et al. Space debris observations with the Slovak AGO70 telescope: Astrometry and light curves. Advances in space research. 2020 Apr 15;65(8):2018-35. Available from: https://dx.doi.org/10.1016/j.asr.2020.01.038.
6. Ploner M, Schildknecht T, Früh C, Vananti A, Herzog J. SPACE SURVEILLANCE OBSERVATIONS AT THE ZIMMERWALD OBSERVATORY. 6th European Conference on Space Debris. 2013 January.
7. United States Space Command. Space-Track Satellite Catalog; 2024. Accessed: March 14, 2024. Available from: https://www.space-track.org.
8. Agency ES. Database and Information System Characterising Objects in Space;. Available from: https://discosweb.esoc.esa.int/.
9. Vallado DA. Fundamentals of astrodynamics and applications. California: Microcosm Press; 2022.
10. R Ortix Longo C, L Rickman S. Method for the Calculation of Spacecraft Umbra and Penumbra Shadow Terminator Points; 1995. Available from: https://ntrs.nasa.gov/api/citations/19950023025/downloads/19950023025.pdf.
11. Rhodes B. Skyfield: High precision research-grade positions for planets and Earth satellites generator; 2019. Astrophysics Source Code Library, record ascl:1907.024.
12. Vallado D, Crawford P, Hujsak R, Kelso S. Revisiting Spacetrack Report No. 3. Reston: American Institute of Aeronautics and Astronautics; Jan 1, 2006. Available from: https://www.proquest.com/docview/1430916175.

13. Hallgarten MI, Amato D. Generalisable bias correction of Two-Line Element sets in the Medium Earth Orbit regime. European Space Agency; 2025. .
14. Pence WD, Chiappetti L, Page CG, Shaw RA, Stobie E. Definition of the Flexible Image Transport System (FITS), version 3.0. Astronomy & Astrophysics. 2010 Dec 1;524:A42. Available from: `https://api.istex.fr/ark:/67375/80W-9XCCF70W-9/fulltext.pdf`.
15. Levesque MP, Lelievre M. Evaluation of the Iterative Method for Image Background Removal in Astronomical Images; 2008. Available from: `https://apps.dtic.mil/sti/citations/ADA479337`.
16. Levesque MP, Buteau S. Image Processing Technique for Automatic Detection of Satellite Streaks; 2007. Available from: `https://apps.dtic.mil/sti/citations/ADA596420`.
17. Lewis JP. Fast Template Matching; 1995.
18. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. vol. 1. IEEE; 2001. p. I. Available from: `https://ieeexplore.ieee.org/document/990517`.
19. Rawls M, Bektesevic D, Iverson V, Biswas A, Santos A, Chambers K. Quantifying the Impact of Satellite Streaks in Astronomical Images;. Available from: `https://uwescience.github.io/DSSG2022-Satellite-Streaks/`.
20. Canny J. A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1986;PAMI-8(6):679-98. ID: 1.
21. Abraham J, Wloka C. Edge Detection for Satellite Images without Deep Networks; 2021. Available from: `https://arxiv.org/abs/2105.12633`.
22. Bradski G. The OpenCV library. DrDobb's Journal. 2000;25(11):120-5. Available from: `https://www.proquest.com/trade-journals/opencv-library/docview/202684726/se-2?accountid=16260`.
23. Duda RO, Hart PE. Use of the Hough transformation to detect lines and curves in pictures. Communications of the ACM. 1972 jan;15(1):11–15. Journal: Commun. ACM. Available from: `https://doi.org/10.1145/361237.361242`.
24. Fernandes LAF, Oliveira MM. Real-time line detection through an improved Hough transform voting scheme. Pattern Recognition. 2008;41(1):299-314. Available from: `https://dx.doi.org/10.1016/j.patcog.2007.04.003`.
25. McMahon RG, Irwin MJ, Maddox SJ. README for the catalogue The APM-POSS1 Sky Catalogue 1.0; 2000.
26. Ryan C, Marcus H. Multi-objective design of optical systems for space situational awareness. Acta astronautica. 2016 Nov;128:669-84. Available from: `https://dx.doi.org/10.1016/j.actaastro.2016.07.008`.
27. Fankhauser F, Tyson JA, Askari J. Satellite Optical Brightness. The Astronomical journal. 2023 Aug 1;166(2):59. Available from: `https://iopscience.iop.org/article/10.3847/1538-3881/ace047`.
28. Mulrooney MK, Matney MJ, Hejduk MD, Barker ES. An Investigation of Global Albedo Values. 2008 AMOS Conference Proceedings (Advanced Maui Optical and Optical and Space Surveillance Technologies Conference). 2008 Jan 1:1. Available from: `https://search.proquest.com/docview/35521779`.
29. Ryan C, Marcus H. Sizing of a Raven-Class Telescope Using Performance Sensitivities; 2013. p. E28. Available from: `https://ui.adsabs.harvard.edu/abs/2013amos.confE..28C`.
30. Liaudat TI, Starck JL, Kilbinger M. Point spread function modelling for astronomical telescopes: a review focused on weak gravitational lensing studies. FrontAstronSpace Sci. 2023 Oct 9;10. Available from: `https://hal.science/hal-04160213`.
31. Schildknecht T. Optical astrometry of fast moving objects using CCD detectors [PhD thesis]; 1994. Available from: `http://www.ub.unibe.ch/content/bibliotheken_sammlungen/sondersammlungen/dissen_bestellformular/index_ger.html`.
32. Craig M, Chambers L. CCD Data Reduction Guide;.
33. Falchi F, Cinzano P, Duriscoe D, Kyba CCM, Elvidge CD, Baugh K, et al.. Supplement to: The New World Atlas of Artificial Night Sky Brightness; 2016. Available from: `https://search.datacite.org/works/10.5880/gfz.1.4.2016.001`.
34. Hecht E. Optics. Fifth edition, global edition ed. Boston: Pearson; 2017.

APPENDIX

A. PLATE SOLVER COMMANDS

The PlateSolve2 plate solver is run from the command line with the following format:

$$
\begin{aligned}
&\textit{PlateSolve2.exe}(\textit{Right ascension in radians},\\
&\quad \textit{Declination in radians},\\
&\quad \textit{x dimension in radians},\\
&\quad \textit{y dimension in radians},\\
&\quad \textit{Number of regions to search},\\
&\quad \textit{FITS filename},\\
&\quad \textit{Wait time at the end})
\end{aligned}
$$

Example:

$$
\begin{aligned}
&\textit{PlateSolve2.exe}(\textit{1.9679}, \textit{0.4808}, \textit{0.0147},\\
&\quad \textit{0.0091}, \textit{20}, \textit{Example.fits}, \textit{0})
\end{aligned}
$$