RIJKSUNIVERSITEIT GRONINGEN

BACHELOR THESIS

The effects and implications of satellites for ground based astronomy with small telescopes

---

Author:
Wyncko Tonckens
S4394623

Supervisors:
Dr. J. Noel-Storr
Dr. M.H.F. Wilkinson

July 11, 2023

Abstract

In this thesis the magnitude of 15 satellites has been analyzed. The average absolute (m_{1000}) magnitude is found to be 5.72 ± 0.86 in the V band and 6.07 ± 0.79 in the R band. For Starlink satellites the weighted average of absolute magnitude in the V band is found to be 6.66 ± 0.39. Overall all the analyzed satellites are found to be too bright in the V band at the perigee of their orbit according to SatCon-1 recommendations, and only half of the satellites comply to this recommendation within the uncertainty of their measurement. A tool to check if observations will be affected by satellites is proposed and made. The program can be found here on github.

Contents

1 Introduction

Satellites are used for a wide variety of purposes ranging from espionage to telecommunication. When looking at the night sky satellites can even be visible by the naked eye, they look like a plane without flashing lights, or like a faint star that is moving across the sky. Starlink satellites can even be seen flocking in straight lines just after they have been launched. The proliferation of satellites in orbit, espicially due to mega constellations such as Starlink and Oneweb, threatens ground based astronmy due to the light pollution it brings with it (Falchi et al., 2023). This thesis focuses on the impact of satellites on observations with the Gratama and LDS telescope, but the conclusions drawn from the results will also be applicable to small telescopes all over the world using similar setups. Small telescopes play a large role in collecting astronomical data, such as variable star measurements and transient events. (Kondo, 2003)

A dark sky is seen as a common heritage to which everyone is entitled and ensuring the existence of the starry night sky is a responsibility carried by all of us, similar to keeping the environment clean. Just as air pollutants like NO_x and CO_2 ,light can be seen as a pollutant as well and similarly, measures have to be taken to limit their impact. (Cipriano, 2007) We as astronomers and scientists carry the responsibility to create awareness of the adverse affects of light pollution to further try to keep the sky dark, so that generations after us can look at the night sky and feel the same admiration as we once felt. In this thesis we will investigate the implications of satellites for ground based astronomy.

2 Background

2.1 Satellite constellations

A single satellite can only cover up to half of the world and a minimum of three would be needed for worldwide coverage (Clarke, 1945). To create a network for fast worldwide communication, multiple satellites are needed which form a constellation. A satellite constellation can be described as a set of multiple similar satellites, serving the same function, moving in complementary orbits and which are under shared control (Wood, 2003).

We can classify satellite constellation networks by different properties such as for example:

- Orbital altitude, such as Low Earth Orbits (LEO) of 180km up to 2000km, high earth and geosynchronous orbits of 35,780km upwards and Medium Earth Orbits of altitudes in between LEO and high earth orbits(Riebeek, 2009).
- Intended purpose such as telecommunication or navigation.
- Constellation geometry or coverage.
- Frequency range over which it operates

2.2 Observational contamination from satellite constellations

The contamination of astronomical observations due to satellites is dependent on a number of parameters which can be categorized into two categories: constellation and observation parameters. The first of which contains the number of satellites in the constellation, the range of altitudes and inclination of the satellites and the magnitude and variability in brightness of a single satellite. Observational parameters include the time and location of an observation, the atmospheric extinction, the spectral band and the inclination of the targeted object for observation, the field of view and exposure time (Hall, 2023). One of the key parameters is the solar depression angle (SDA) which defines how far the sun is below the local horizon for an ground based observer.

Sunset and sunrise occur when the SDA is equal to zero and night can be divided into four stages dependent on SDA:

- Civil twilight: $0° \leq \text{SDA} < 6°$
- Nautical twilight: $6° \leq \text{SDA} < 12°$
- Astronomical twilight: $12° \leq \text{SDA} < 18°$
- Astronomical night: $18° \leq \text{SDA}$

A SatCon-1 workshop from august 2020 set up a list of recommendations for observatories and constellation operators to mitigate the effects of the increase in satellite constellations on ground based astronomy. They found that the disturbance due to satellites on astronomical ground based observations is the greatest during twilight and near the horizon, due to the fact that LEO satellites, dependent on the design of the constellation, are covered by earth's shadow during astronomical night. This means that observations that require to be done in astronomical twilight, such as the search for Near Earth Objects (NEO's), are affected the most by LEO satellites (Walker et al., 2020). Satellites orbiting below 600km are only visible for a few hours during astronomical twilight, while satellites orbiting above 600km can be visible during the whole night.

(Hainaut and Williams, 2020) found similar results by estimating the SDA when a satellite of certain height above zenith is just illuminated, i.e. when half of the constellation is illuminated or the height of earths shadow in zenith. Figure 1 shows the SDA and height of earth shadow above zenith for the night during the summer and winter solstice for the LDST.

In the following sections we will discuss different sources of contamination to ground based astronomical observations due to satellites.

2.2.1 Brightness of a satellite

One of these recommendations made by SatCon-1 is that a satellite may not exceed a V-band magnitude dependent on their height given by:

$$M(h) = 7.0 + 2.5\log_{10}(\frac{h}{550km}) \tag{1}$$

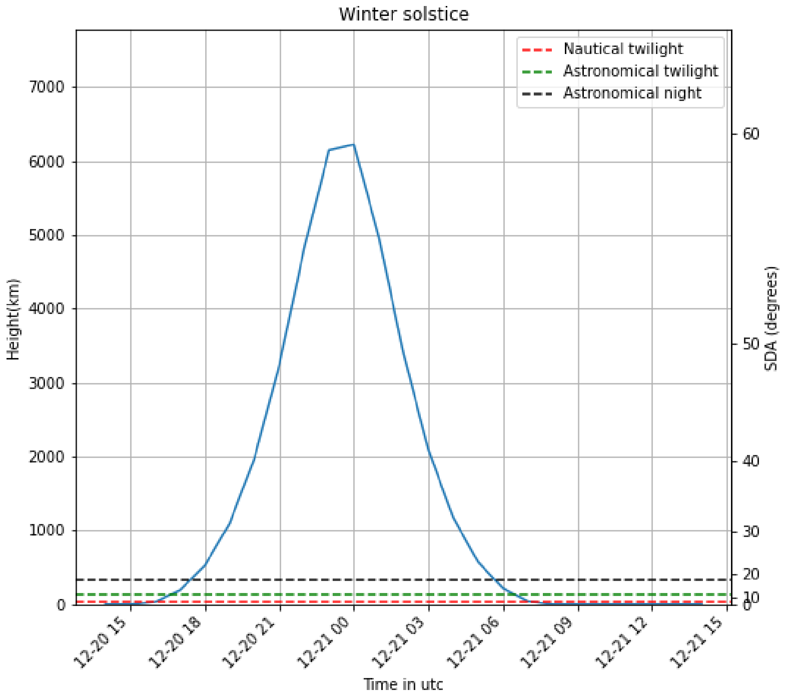
where h is the height of the orbit of the satellite. This equation can be interpreted as that the magnitude of a satellite orbiting at 550 km may not exceed 7 magnitudes.

Simulations performed by (Hainaut and Williams, 2020) estimate starlink satellites orbiting at 550km to have magnitudes in the range of 4.2 to 5.9 and measurements performed by (Tregloan-Reed et al., 2020) on starlink satellites have validated this estimation.

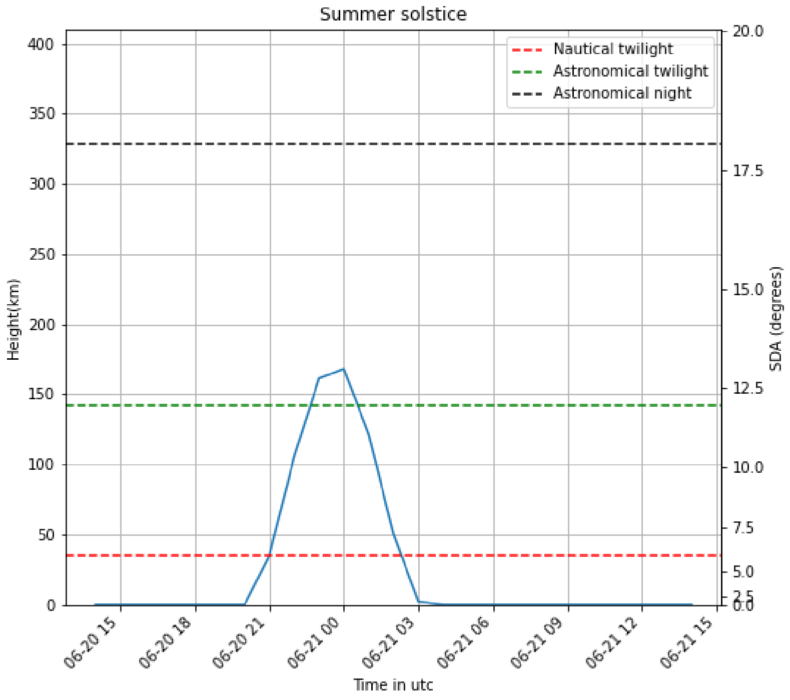
Mini-MegaTORTORA (MMT-9) is a automated observatory located in Russia which has analyzed the magnitude of 12176 satellites at the time of this thesis (Karpov et al., 2016). (Mallama, 2022) analyzed over 80.000 magnitudes of observed oneweb satellites from this database and found them to have an mean magnitude, adjusted to 1000km, of 7.05 ± 0.66. (Mallama, 2021) performed the same analysis for over 100,000 starlink magnitude measurments and found the average starlink satellite to have a magnitude, adjusted to 1000km, of 5.89 ± 0.46 or 7.21 ± 0.89 in the case of satellites equipped with VisorSat (a visor that shades the satellite from sunlight).

2.2.2 Light pollution

Not only satellites but also space debris can add an artificial skyglow component to the dark night sky, which can cause the sky to become brighter. A paper from 2021 by M Kocifaj (Kocifaj et al., 2021) analyzed the increase in artificial night sky brightness by space objects (satellites and space debris) and estimated that this skyglow component, at the time of publishing, could



(a)



(b)

Figure 1: SDA and height of earths shadow versus time for different dates: a) for the winter solstice and b) for the summer solstice

reach an intensity of the order of 21.1 μcdm^{-2}, at the beginning and end of astronomical night ,and might approach 25 μcdm^{-2} in 2030. This value is more than 10% of the natural dark sky brightness of 200 μcdm^{-2} which is given by (Cayrel et al., 1980) as a critical level for light pollution. This skyglow component is dominated mostly by small objects (mm and below space debris) and (Bassa et al., 2021) found the contribution from satellites to the surface brightness to be in the 0.3-0.7 μcdm^{-2} range with peaks around 2.7 μcdm^{-2} making the satellites only contribute to 1% of sky brightness in the worst case.

2.2.3 Satellite flares

Satellite flares occur when a satellite has one or more surfaces that are more susceptible to reflection, than other parts of the satellite, making the satellite light up for a short period of time above the observer. The first Iridium constellation launched in 1998 was notorious for its satellite flares which could reach magnitudes of up to -8 -or about 30 times as bright as Venus- due to reflection from its silver 1.6 square meter silver coated main mission antenna (satobs, 2007). (Hainaut and Williams, 2020) estimated from simulations that contamination of observations due to satellite flares would only effect very long exposure (1800s) wide field (1 sq.deg) observations up to a 10^{-4} level and that other exposures, with shorter exposure times, are affected much below that level. For small telescopes the observational contamination due to satellite flares is negligible.

2.2.4 Satellite trails

When a satellite passes over an observation, it leaves a trail in the image, an example of an observation being contaminated by a satellite trail can be seen below in Figure 2.

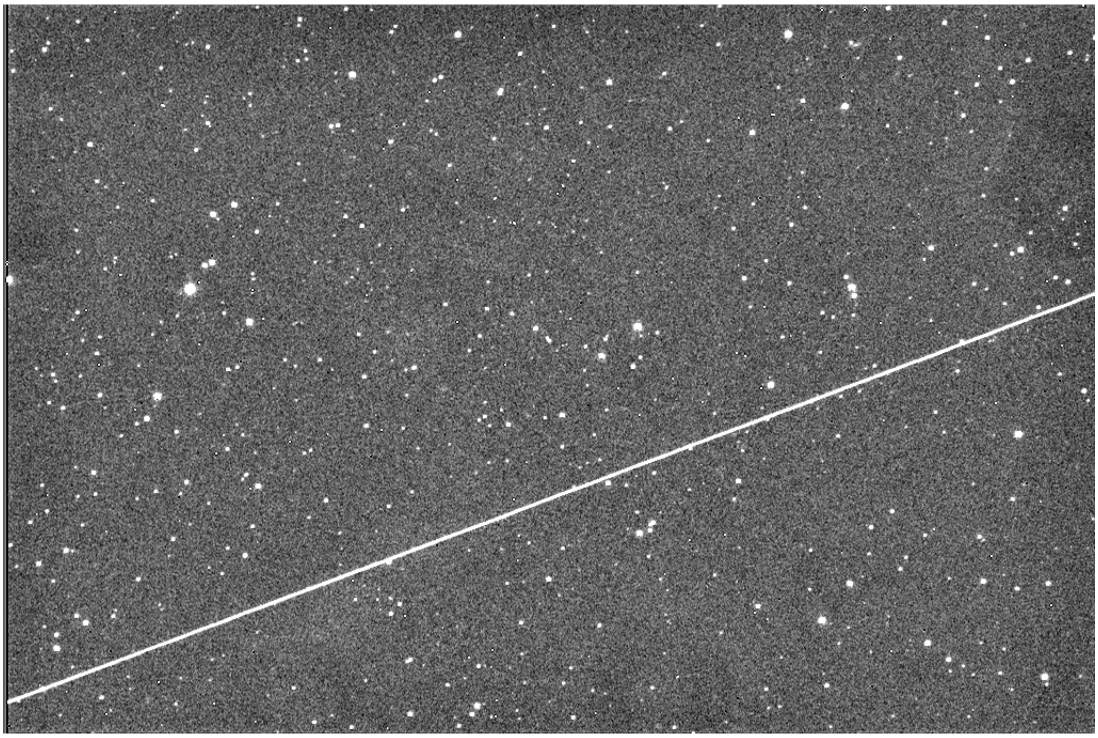


Figure 2: An observation of the Gratama telescope contaminated by a satellite taken on 02-05-2022

The number of satellite trails in an exposure scales with the size of the field of view and exposure time, so the contamination differs for each telescope(Bassa et al., 2021). Using simulations[1] from the previous cited paper, we can get an estimate of how different satellite constellations, also determined in the previous citation, will affect the Gratama and Lauwersmeer Dark Sky (LDS) telescope. These graphs are similar for other small telescopes with similar properties. The relevant properties for the telescopes are listed in the table below.

	LDST	**Gratama**
Latitude (degrees)	53.38	53.24
Max Field of View (degrees)	1.03	0.5
Resolution or pixel size(arcsec)	0.802	0.566
Limiting magnitude	19	17

Table 1: Relevant properties for the Gratama and LDS Telescope

Where we obtained the field of view and pixel size by uploading an arbitrary observation for each telescope to astrometry.net (Lang et al., 2010). The graphs produced from the previous mentioned simulations can be found below in Figure 3 on page 9 and show the number of satellites per square degree for a solar declination of -18 degrees, for the satellite constellations active in 2021 and for an estimate of the number of satellites active in 2050.

2.2.5 Occultation

Occultations occur when a non-illuminated satellite passes in front of a source blocking the light from that source. Simulations performed by (Hainaut and Williams, 2020) found that in the worst case occultations would effect 10^{-4} of 10s exposures. The overall effect of these occultations range from 2×10^{-5} magnitude for 10s exposures with low orbiting satellites to 1×10^{-2} magnitude for 0.1s exposures with high orbiting satellites. Since the LDST, Gratama and other small telescopes have even longer exposure times the effects of occultation will be negligible and thus will not be taken into account in this thesis.

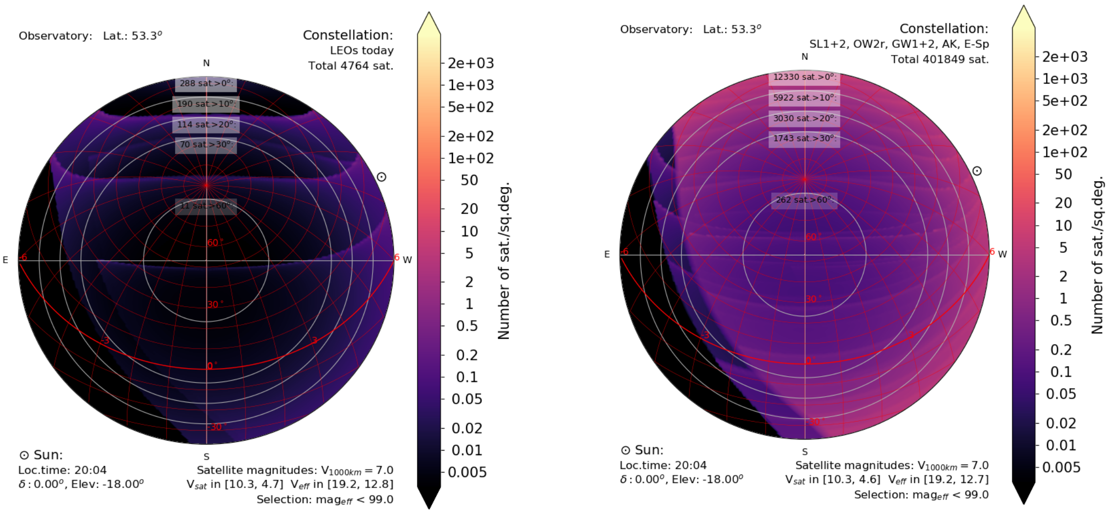
2.3 Mitigation measurres

2.3.1 Scheduling of observations

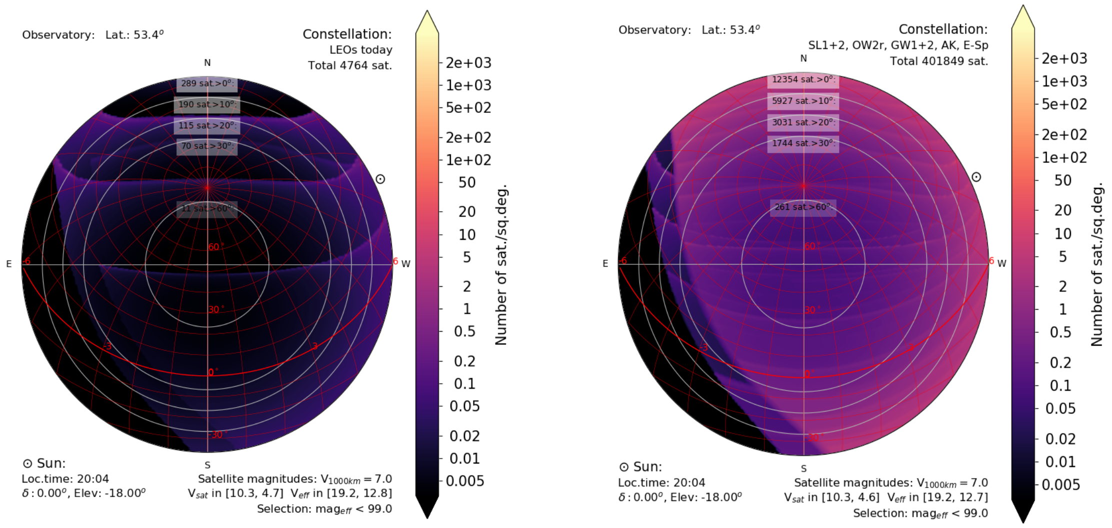
A way to avoid observations being contaminated due to satellite tracks would be to schedule observations so that no satellite will pass during your exposure. To do this, an extensive database of satellites and their orbital parameters would be needed. The celestrak (Celestrak, 2023) database offers these parameters for a wide variety of satellites and using a python module called Skyfield, which makes uses of SGP4 protocols, (Rhodes, 2019) one can model where a satellite is in the sky at any arbitrary time. The Celestrak database provides orbital parameters of satellites in Two Line Element sets or TLE's. These TLE's give the satellites orbital parameters on a certain epoch and the skyfield module calculates the position the satellite is likely to have had on a certain time from that epoch. The epoch is a time at which the orbital parameters of the satellite, such as position and velocity are accurately determined. Since satellites orbits are difficult to predict, the results degrade the further away you get from the epoch. To quote the "Revisiting Spacetrack Report 3": "The maximum accuracy for a TLE is limited by the number of decimal places in each field (Vallado, 2007:116). In general, TLE data is accurate to about a kilometer or so at epoch and it quickly degrades (Hartman, 1993)." (Vallado et al., 2006)

---

[1] https://eso.org/ ohainaut/satellites/simulators.html



(a) Gratama 2021 (b) Gratama 2050



(c) LDST 2021 (d) LDST 2050

Figure 3: Satellites per square degree for the Gratama and LDS telescope in 2021 and 2050 based on simulations. The red lines indicate the number of satellites above a certain elevation

2.3.2 Starlink VisorSat and DarkSat

Starlink is actively trying to reduce the brightness of their satellites. They have come up whith different means to achieve this such as VisorSat, a visor that blocks the satellite from sunlight and DarkSat a coating that decreases the reflectivity of the satellites. The effectiveness of these measures will be discussed in later sections of this thesis in more detail.

2.4 goals

The main goals of the thesis are the following:

1. Determine the magnitude of the satellites and how they compare to the satcon recommendation for limiting magnitude found in Equation1
2. Develop a tool to check whether, and to what extend, an observation will be affected by satellites to schedule observations
3. Determine the number of frames contaminated by satellites from the Gratama and LDS telescope over time

To achieve these goals we collaborate with Oliver Parades, a computing science student who will write a algorithm to find satellite tracks in an image and will supply us with all the frames contaminated by satellite trails from the Gratama and LDS telescope. The contaminated frames will provide us with an extensive data set of different satellites over a time range from 2009 to now. In the following section will elaborate on the methods used to achieve these goals.

3 Methods

3.1 Magnitude determination

Since the satellite is moving it leaves a trail in our observation that has a length of $\omega_{sat}t_{exp}$, where ω_{sat} is the apparent angular speed of the satellite and t_{exp} is the exposure time of the observation. The light from the satellite in this frame is spread along this trail for which we have to account accordingly. Typically this length is larger than the field of view of our instrument (Bassa et al., 2021). The effective time it takes a satellite to cross an individual resolution element would be $t_{eff} = \frac{r}{\omega_{sat}}$ where r is the size of a single resolution element of the ccd of the telescope. This gives rise to the term *effective magnitude*, as the magnitude of a static point source that, in a time t_{exp}, would produce the same accumulated light in resolution elements as the satellite would in those elements in a time t_{eff}.
The frames which came up as contaminated by the algorithm made by Oliver Parades first have to be reduced. For each contaminated observation, we take all the images of the same observation and align and normalize them so that they are all equally bright. Then the median of all the images can be removed from the image containing the trail to increase the signal to noise ratio of the trail and so that we are left with an image that contains only the trail. From here we can isolate the trail in a box and sum the count value in each column of the box. Since we assume the satellite to be a point source at each point, all the light in a column of the box is assumed to be from the satellite. An example of this can be seen below in Figure 4. To convert these counts to magnitude we have to know the magnitude of a star in the frame to calibrate our magnitude scale. From the median frame we isolate one star and determine the counts that star has and by consulting astrometry.net we can find the magnitude the star is supposed to have to find the zeropoint for our magnitude scale. By using Equation 2 below we can now determine the effective magnitude of the satellite:

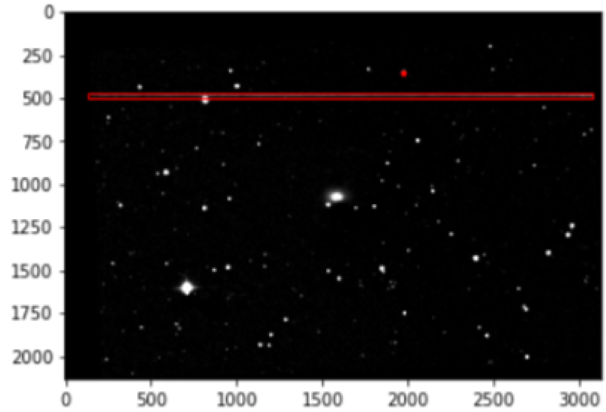


Figure 1: 110128_Li_.00000052.FIT with the aperture used to find zeropoint

Figure 2: The box, from previous image enclosing the satellite trail, plotted horizontally

Figure 4: An image showing how the counts of the satellite trail are obtained

$$m_{eff} = -2.5 * \log_{10}(counts_{satellite}) + zeropoint \tag{2}$$

where $counts_{satellite}$ is the light accumulated in each column of the box containing the satellite and the zeropoint is the calibration constant discussed above, which can be expressed as $2.5 * \log(counts_{star}) + m_{star}$. We take the mean of the effective magnitude along the trail as the effective magnitude of the satellite and we take the difference between the maximum and mean effective magnitude along the trail to represent the error. From comparing the satellite counts in its resolution elements to the counts of a star with a known magnitude it is not possible to determine the apparent magnitude, since the light from the satellite in the frame is spread along the trail and the stars contributes counts to its resolution elements during the whole exposure time, as discussed above. Therefore we have to transform the effective magnitude to apparent magnitude, taking into account the different times in which counts have been accumulated. We can do this by using Equation 3 below

$$m_{apparent} = m_{eff} + 2.5 * \log_{10}(\frac{t_{eff}}{t_{exp}}) = m_{eff} + 2.5 * \log_{10}(\frac{r}{\omega_{sat} * t_{exp}}) \tag{3}$$

where t_{eff} has been substituted for the value derived above. For determining the apparent angular velocity of the satellite (ω_{sat}) we use the following approximation

$$\omega_{sat} = \frac{360 * 3600}{period_{satellite}} \tag{4}$$

where the numerator is the number of arc seconds for a full rotation of 360 degrees and the denominator is the period of the satellite, which will be looked up once we determined which satellite was in the frame, which will be discussed in more detail below. A more complete derivation of Equation 3 can be found in Appendix section A of this thesis. Because the distance for each satellite varies, we have to normalize them to a single distance to compare how bright different satellites are in relation to one another. For this *absolute distance* we take 1000km, which is common convention for satellites and similar as found in (Bassa et al., 2021). Equation 5 below is used to convert from apparent to *absolute magnitude* (m_{1000km})

$$m_{1000km} = m_{apparent} - 5 * \log_{10}(\frac{d(km)}{1000km}) \tag{5}$$

Where d is the distance to the satellite which will be determined by methods described in the section below. To determine whether these found magnitudes are below or above the SatCon-1 recommendation for the V band magnitude a satellite may not exceed, given in Equation 1, we need to know at which height the satellite in our frame is orbiting. To do this we can obtain TLE's for an extensive number of satellites from Celestrak (Celestrak, 2023) and by using a python module called Skyfield (Rhodes, 2019) we can extrapolate which satellites were in our frame at the time of observing and find the distance to those satellites from our point of observing. Since this method uses TLE's the accuracy of the results degrade the further away the observation is from the epoch of the satellite used; A point which we will discuss in more detail in the Discussion section of this thesis.

Note that since orbits of satellites are difficult to predict, as mentioned before, the satellite which is the closest to our observation frame at the time of the observation is taken as the most probable to have caused the contamination. Since the distance from an observer to a satellite does not equal the height at which the satellite orbits, we have to look up the height of the orbit after we determined which satellite was in our frame. To do this we look up the perigee of the orbit (closest point of the orbit to earth) of the satellite on a site like N2YO.com. Here we can also find the period of the satellite. Then by using Equation 1 we can find the limiting magnitude the satellite may not exceed, at that orbital height, according to SatCon-1 recommendations. Since we determined the magnitude at a distance from the satellite and not at its orbiting height we need to convert the magnitude to represent the magnitude the satellite would have had at its orbital height, we can do this by using Equation 6 below, which is similar to Equation 5, but instead of calculating how bright the satellite would be at 1000 km, we calculate how bright it would be in the perigee of its orbit. The limiting magnitudes and magnitudes at the perigee of the orbits of the satellites can be found in the Results section in Table 5.

$$m_{perigee} = m_{apparent} - 5 * \log_{10}(\frac{d(km)}{perigee(km)}) \tag{6}$$

3.2 Scheduling software

Using a similar method to how we determined which satellite is in a frame at the time of observing, a program which checks whether a satellites are likely to interrupt an observation was made. This program makes use of TLE's from the celestrak database, the python skyfield module and some user input parameters listed in the Table 2 below, and checks which, if any, satellites are likely to contaminate your observation. This software can be used to schedule your observations accordingly to minimize contamination from satellite trails. For each satellite it returns the time, position and distance to the satellite. The program also returns the accuracy of the result depended on how far the observation is from the epoch of the resulting satellite. The accuracy can give 3 results: green for when the observation is less than 10 days from the last epoch, orange when the observation is between 10 and 20 days from the epoch and red when the observation is further than 30 days from the epoch. These ranges were chosen according to the "Revisiting Spacetrack report 3" (Vallado et al., 2006). The difference between epoch and observation was chosen to indicate the accuracy since this is the greatest source of inaccuracy in predicting the satellite's position. The program updates it list of TLE's every time it is ran, making it the more accurate to find satellite's closer to your current date and time.

INPUT	**OUTPUT**
Longitude(degrees)	Satellite name
Latitude(degrees)	Time(UTC)
Time start (UTC)	Right ascension
Time end (UTC)	Declination
Right ascension(hours)	Distance(km)
Declination(degrees)	Accuracy
FOV(degrees)	

Table 2: Table showing the input and output parameters for the scheduling software

3.3 Number of satellite trails

To determine the number of satellite trails in our observations we collaborate with a computing science student who writes a script to automatically recognize satellite tracks in our extensive database of observations from the Gratama and LDS telescope that dates from 2009 to now. Using these results we will be able to analyze the number of contaminated frames from the Gratama and LDS telescope over time.

4 results

4.1 Magnitude determination

The script made by Oliver Parades was unfortunately not fully functioning in time and due to not having unlimited time for this thesis, only the frames that did come out contaminated have been able to be analyzed. Due to the data set being smaller than anticipated, making meaningful plots about the number of frames contaminated over time, the number of satellites observed above certain elevations and the magnitude of Starlink satellites over time is unfortunately not possible. This limitation will be discussed in more detail in the discussion section of this thesis. For eight satellites the V band magnitude has been determined and for 7 satellites the R band magnitude. The name of the satellite, telescope used, distance to the satellite and effective, apparent, and absolute (m_{1000}) magnitude can be found below in Table 4. The perigee of the orbit, the magnitude of the satellite at that perigee and the SatCon-1 recommendation for the magnitude a satellite may not exceed at that orbital height can be found below in Table 5. Data such as the date, time, right ascension and declination of the frame, period of the satellite and how many days the observation differs from the epoch of the TLE used to determine the satellite can be found in Appendix section B of this thesis.
The weighted average for Starlink and Oneweb satellites magnitudes at a distance of 1000km and at the perigee of their orbit can be found below in Table 3. An example notebook for the data reduction of a single night can be found in the Appendix section C of this thesis and here on github.

	weighted average m_{1000}	**weighted average at perigee of orbit**
STARLINK V band	6.66 ± 0.39	5.37 ± 0.39
ONEWEB R band	7.06 ± 1.11	7.47 ± 1.11

Table 3: Table showing the weighted average of the magnitude for starlink and oneweb satellites found from results at 1000km and at the perigee of the orbit

Figure 5 and 6 show the absolute (m_{1000}) V and R band magnitudes of the satellites respectively and indicates the average magnitude. Figure 5 also shows the SatCon-1 recommendation for the V band magnitude a satellite may not exceed at a height of 1000km. The average absolute (m_{1000}) magnitude is found to be 5.72 ± 0.86 in the V band and 6.07 ± 0.79 in the R band. The brightest satellite we measured is COSMOS 2501 [GLONASS-K], a Russian satellite with an absolute V band magnitude of -0.30 ± 0.15.

Telescope	**Satellite**	$m_{effective}$			$m_{apparent}$			**Distance (km)**	m_{1000}		
		B	V	R	B	V	R		B	V	R
Gratama	SKYSAT-C14			$(1.69 \pm 0.11) \times 10^1$			5.93 ± 1.10	693.92			6.72 ± 1.10
	PHASE 3B (AO-10)			(1.80 ± 0.09) x 10^1			8.78 ± 0.94	30981.81			1.34 ± 0.94
	GENESIS 1		$(1.72 \pm 0.19) \times 10^1$	$(1.70 \pm 0.03) \times 10^1$		6.22 ± 1.86	6.05 ± 0.27	753.99		6.83 ± 1.86	6.67 ± 0.27
	COSMOS 2501 [GLONASS-K]		$(1.51 \pm 0.01) \times 10^1$			6.32 ± 0.15		21074.82		-0.30 ± 0.15	
	STARLINK-1969		$(1.72 \pm 0.05) \times 10^1$			6.23 ± 0.45		813.63		6.68 ± 0.45	
	YAOGAN 9C		$(1.80 \pm 0.06) \times 10^1$			5.41 ± 0.57		1496.76		4.53 ± 0.57	
LDST	MOHAMMED VI-A		$(1.64 \pm 0.10) \times 10^1$			6.61 ± 0.96		708.48		7.36 ± 0.96	
	2022-019J		$(1.60 \pm 0.14) \times 10^1$			6.22 ± 1.40		576.43		7.42 ± 1.40	
	STARLINK-5051		$(1.56 \pm 0.10) \times 10^1$			5.76 ± 1.10		569.55		6.98 ± 1.10	
	STARLINK-1150		$(1.49 \pm 0.10) \times 10^1$			5.04 ± 1.03		574.33		6.25 ± 1.03	
	ONEWEB-0442			$(1.89 \pm 0.13) \times 10^1$			8.02 ± 1.26	1711.06			6.85 ± 1.26
	ONEWEB-0090			$(1.94 \pm 0.23) \times 10^1$			8.56 ± 2.31	1435.84			7.77 ± 2.31
	STARLINK-2326			$(1.79 \pm 0.15) \times 10^1$			6.90 ± 1.47	663.77			7.79 ± 1.47
	COSMOS 2477 [GLONASS-M]			$(2.08 \pm 0.20) \times 10^1$			11.88 ± 1.98	20077.87			5.37 ± 1.98

Table 4: Table showing the effective, apparent and absolute (m_{1000}) magnitudes of a number of satellites observed by the Gratama or LDS telescope.

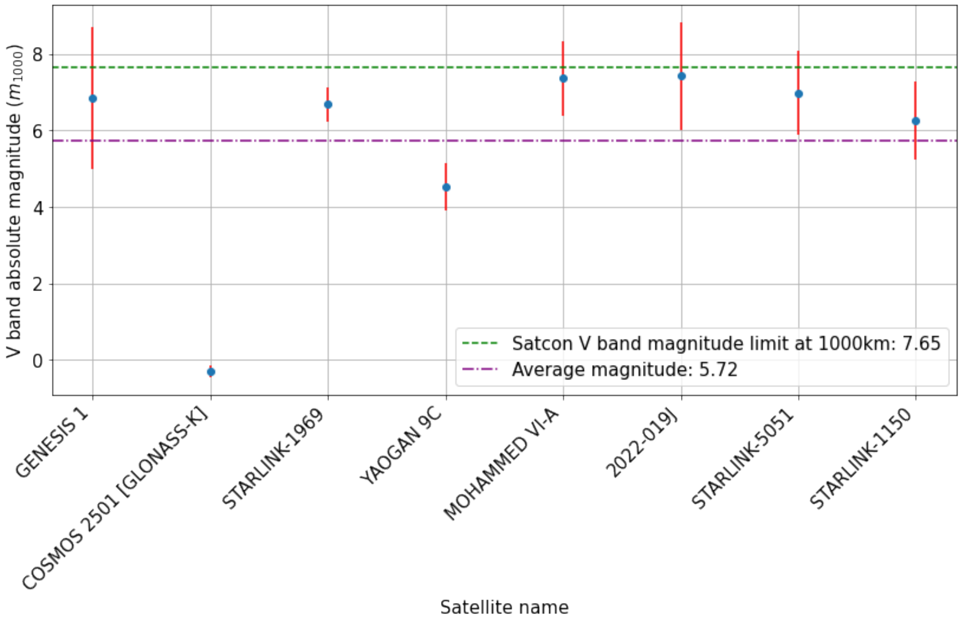


Figure 5: Showing the V band magnitude at 1000km for different satellites. The dashed line indicates the Satcon V band magnitude limit that a satellite may not exceed at 1000km. The dashed dotted line indicates the average magnitude

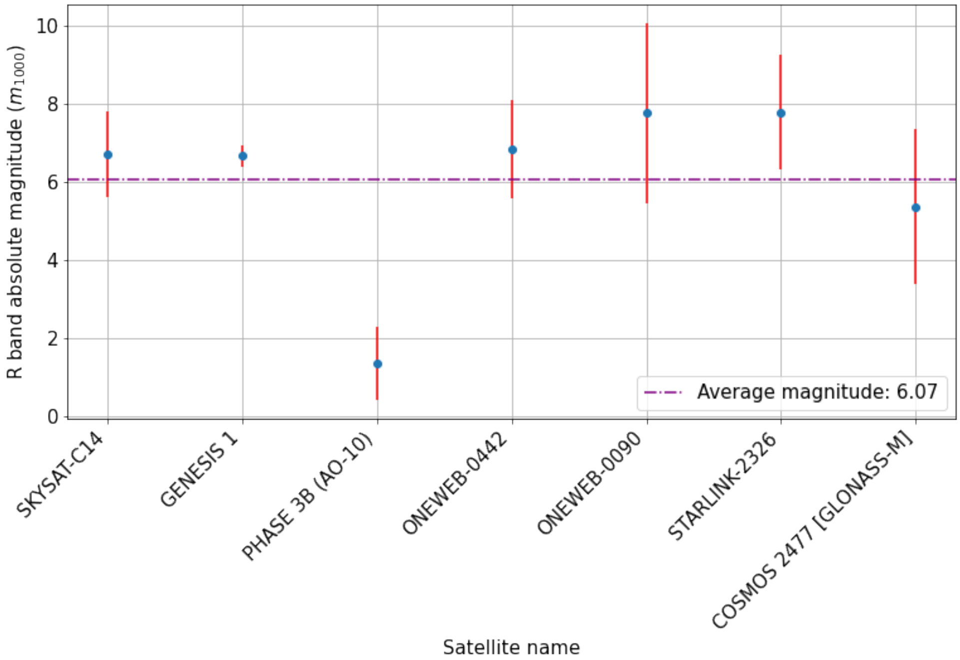


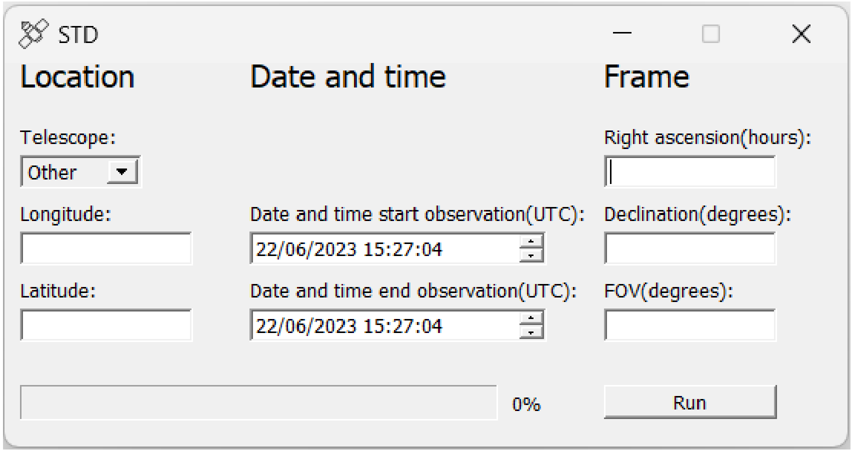
Figure 6: Showing the R band magnitude at 1000km for different satellites. The dashed dotted line indicates the average magnitude

Telescope	Satellite	Orbit perigee (km)	m_orbit B	V	R	Satcon Recommendation
Gratama	SKYSAT-C14	433.7			4.91 ± 1.10	6.74
	PHASE 3B (AO-10)	3982.5			4.34 ± 0.93	9.15
	GENESIS 1	475.2		5.22 ± 1.86	5.05 ± 0.27	6.84
	COSMOS 2501 [GLONASS-K]	19086.3		6.10 ± 0.15		10.85
	STARLINK-1969	553.6		5.40 ± 0.45		7.01
	YAOGAN 9C	764.6		3.95 ± 0.57		7.36
LDST	MOHAMMED VI-A	644.8		6.40 ± 0.96		7.17
	2022-019J	528.6		6.03 ± 1.40		6.96
	STARLINK-5051	546.4		5.67 ± 1.10		6.99
	STARLINK-1150	553.6		4.96 ± 1.03		7.01
	ONEWEB-0442	1198.8			7.25 ± 1.26	7.85
	ONEWEB-0090	1223.3			8.21 ± 2.31	7.87
	STARLINK-2326	553.3			6.50 ± 1.47	7.01
	COSMOS 2477 [GLONASS-M]	19086.1			11.77 ± 1.98	10.85

Table 5: Table showing the satellite's name, telescope with which the observation was made, perigee of the orbit, magnitude at the perigee of the orbit and the satcon recommendation for the V band magnitude a satellite may not exceed at the perigee of their orbit

4.2 Scheduling software

The final program to check whether a satellite is likely contaminate an observation, to schedule observations, can be found here on github. A user interface to easily retrieve results from the program was made using Pyqt5. This also makes the program usable for astronomers with small telescopes all over the world. The interface takes the input parameters listed in Table 2 and outputs the parameters found in the same table. For choosing with which telescope you are observing, there are three options: LDST, Gratama and other. LDST and Gratama automatically set the longitude and latitude to the values of those telescopes and for other, two new input boxes appear to set the longitude and latitude of your telescope manually. All the input boxes only accept float values between the appropriate ranges; for example the longitude box only accepts values between -180 and 180. The software finds the satellites in your specified frame and time range and outputs the results as a pandas data frame. The user interface also has a functioning progress bar to show the progress of the code. The program deletes the loaded TLE file, extracted from Celestrak, every time it is ran to ensure the program uses the most up to date orbital information and epoch of the satellites.



Telescope: Other

Longitude:

Latitude:

Date and time start observation(UTC): 22/06/2023 15:27:04

Date and time end observation(UTC): 22/06/2023 15:27:04

Right ascension(hours):

Declination(degrees):

FOV(degrees):

0%

Figure 7: The user interface of the scheduling software

5 Discussion & Conclusion

5.1 Magnitude determination

The magnitudes determined for the observed satellites range from 1.34 to 7.79 in the R band and -0.30 to 7.42 in the V band. We have not observed any satellites in the blue band.
The weighted average of Starlink satellite magnitudes was found to be 6.66 ± 0.39 in the V band normalized to a distance of a 1000km. This value compares to a magnitude of 6.43 at 1000km found by (Mallama, 2021) from the MMT-9 database for Starlink satellites with VisorSat. This leads us to believe that all the Starlink satellites we measured are equipped with VisorSat to reduce their brightness, since the previous mentioned source found a magnitude difference of 1.38 for Starlink satellites with and without VisorSat, which is not visible in our data. Althought these satellites appear to be equipped with VisorSat, they are still to bright to comply with SatCon-1 recommendations for V band magnitude as can be seen in Figure 5.
For Oneweb satellites the weighted average of the magnitude in the R band was found to be 7.06 ± 1.11 normalized to a distance of 1000km. This result is in close agreement with values produced by (Mallama, 2022) from the MMT-9 database, who found a magnitude of 7.05 ± 0.66, also normalized to a distance of 1000km.
At the perigee of the orbit, we found the weighted average of the Starlink satellites in the V band to be 5.37 ± 0.39 this is in the range of values predicted from simulations by (Hainaut and Williams, 2020). All these averages can be found in Table 3.
Overall, none of the satellites analyzed have been found to comply with the SatCon-1 recommendations for V band magnitude, and only four out of eight within the uncertainty of their measurement. All the satellites we measured in the V band were too bright at the perigee of their orbit. This is similar to results produced from MMT-9 data by (Hall, 2023) who found that 99.86% of 17245 starlink satellites equipped with VisorSat are too bright at their orbit. The brightest satellite we measured, COSMOS 2501 [GLONASS-K], has an apparent magnitude of 6.32 ± 0.15 in V band at a distance of 21075km, which translates to an absolute (m_{1000}) of -0.30 ± 0.15. This result seems to be too bright, since for another COSMOS satellite we measured an apparent magnitude of 11.88 ± 1.98 in the R band at a distance of 20078km which results in an absolute (m_{1000}) magnitude of 5.37 ± 1.98. A probable explanation for this discrepancy would be that the bright satellite we measured is not a COSMOS satellite, but another satellite orbiting at a lower altitude, since the greater distance of the COSMOS satellite would make the measured satellite appear brighter at 1000km. The main problem with determining which satellite is contaminating the frame is that the accuracy of the prediction quickly degrades the further you are from the epoch of the satellite (Vallado et al., 2006). A way to increase the accuracy of the predictions would be to implement a database of TLE's in the satellite schedule software such that it can load the orbital information of satellites with epoch's the closest to your date of observing. So far there has been no luck finding such a database of TLE's, but I hope to be able to implement this feature in the future. Table 6 in Appendix section B, shows how far each observed satellite differs from its latest known epoch.
The biggest limitation to the results is the number of frames contaminated by satellite trails. Since the script made by Oliver Parades was not fully functioning in time, only 15 frames have been able to be analyzed. Once the script is fully functional we will be able to identify all the contaminated frames from the Gratama and LDS telescope and analyze them. More satellites, over a wider range of time and declination, need to be measured before meaningful plots can be made. Especially comparing the number of satellites above a certain elevation, to simulations made by (Bassa et al., 2021) seen in Figure 3 or comparing the number of satellites spotted to the solar declination angle would be interesting topics for follow-up research. It would also

be interesting to see how the brightness of Starlink satellites differ over time, since Starlink is actively trying to decrease the brightness of their satellites before they are launched into space. The current data reduction used to determine the magnitude of the satellites still requires the user to perform some manual operations. The box in which the satellite is isolated is indexed manually, but this could be automated by finding the angle and position of the satellite trail using the script that Oliver Parades made and by indexing the trail accordingly. The magnitude of the reference star also needs to be input manually, but this could be automated by making use of the SIMBAD astroquery. Another problem the script still has is that it can not distinguish when different objects are observed on the same night and tries to align them, a way to fix this would be to look at the header object information of the fits files to group files according object. Implementing these features would make the script autonomous and makes it able to be ran over the whole range of contaminated frames at once or to even be included in the, currently still in progress, LDST pipeline. Aperature photometry could also be applied to single observation frames at the cost of a lower sigal to noise ratio for the satellite trail. To determine the angular velocity of the satellites we did not take into account the rotation of the earth, but since the earth has a period of 24 hours, which is way longer than the periods of the satellites measured, the approximation used is assumed to hold.

5.2 Scheduling software

Since the accuracy of predicting the positions of satellites using the Celestrak database and Skyfield model is largely depended on how far your observation is from the known epoch of the satellite, running it far (more than a month or so) back or forward in time increases the uncertainty in predictions. A way to improve the accuracy would be to more precisely analyze how much predicted positions differ from observed positions depended on how far the observation is from the epoch to get a better feel for these uncertainties. Rigorous testing would be needed to find in what time range from an epoch an prediction is still accurate. The program is thus, currently, better suited to predict satellite positions on the night of observing itself when the observation date is close to the latest known epoch of the satellite. This is not a drawback since its main purpose of the software is to plan observations to avoid satellites when observing. An extended purpose for the scheduling software would be another menu that allows you to look up where a specific satellite will be to observe it, or to find the best visible satellite's for you to observe in a time frame. This could make the so called "hunting" for satellite's easier. Another interesting option to implement would be to show which satellites are visible above a certain declination. Another function to be implemented, is connecting the software to the LDST so that it can automatically find where the telescope is pointing and check for satellites at that position. This would also be useful when the LDST receives requests for observations from the public, making it possible to schedule observations depending on satellites in the frame. A disturbance indicator scale would be useful for this to distinguish between close and bright and far and fainter satellites. Below is a list of features I wish to implement during the next academic year as part of the Practical Astronomy Crew

- Make predicting satellite trails from the past more accurate by implementing old TLE's
- Make a section especially to track satellites for satellite "hunting"
- Implement the position where the LDST is pointing
- Make a disturbance indicator scale for satellite trail contamination

5.3 Conclusion

To conclude, of the satellites analysed only a small percentage comply with the SatCon-1 brightness recommendations. Measures should be taken in order to mitigate the light reflected from satellites before satellites are launched into space, since it is difficult to do anything about the problem once the satellites are in orbit. Measures especially have to be taken before mega constellations such as Starlink will have reached their planned population, since these are by far the most massive sources of satellites. These measures have to be taken to try to keep our skies clear and unpolluted by satellite trails and to fight the further endangerment of observations from small telescopes.
Starlink's VisorSat-1 solution shows good promise into reducing the magnitude of starlink satellites up to almost 1.5 magnitudes, as shown by (Mallama, 2021) but even then Starlink satellites are still on the edge of what SatCon-1 recommends. (Tregloan-Reed et al., 2020) showed that Starlink DarkSat satellites, which have an extra dark coat applied to them, can reduce the light reflected by a satellite up to a factor of 2. In the future I plan to continue this research and the development of the satellite scheduler software. Especially determining the magnitudes of all the contaminated frames found in data taken by the Gratama and LDS telescope is something I want to automate and plan to continue to work on in the future.

6 Appendix

6.1 Apendix A

$$m_{star} - m_{sat} = -2.5 * \log_{10}(\frac{\frac{f_{star}}{t_{exp}}}{\frac{f_{sat}}{t_{eff}}}) = -2.5 * (\log_{10}(f_{star}) - \log_{10}(f_{sat}) + \log_{10}(\frac{t_{eff}}{t_{exp}})) \tag{7}$$

where m_sat is the apparent magnitude of the satellite and m_star is the magnitude of our reference star. Both fluxes are normalized to 1 second by dividing by the corresponding exposure times.

$$m_{sat} = -2.5 * \log_{10}(f_{sat}) + 2.5 * \log_{10}(f_{star}) + m_{star} + 2.5 * \log_{10}(\frac{t_{eff}}{t_{exp}}) \tag{8}$$

where $-2.5 * \log_{10}(f_{sat}) + 2.5 * \log_{10}(f_{star}) + m_{star}$ is equal to the effective magnitude of the satellite (m_{eff}) so this becomes,

$$m_{sat} = m_{eff} + 2.5 * \log_{10}(\frac{t_{eff}}{t_{exp}}) \tag{9}$$

6.2 Apendix B

Telescope	Satelllite	Date of observation(dd/mm/yyyy)	Time of observation	Right ascension	Declination	Days from epoch	Period(s)	File name
Gratama	SKYSAT-C14	02-05-2022	01:42:21	16h 43m 57s	+02°59’ 50”	412.71	5598	220501_Li_.00000206.Entered_Coordinates.BLUE.FIT
	GENESIS 1 (R band)	02-05-2022	01:56:37	16h 43m 55s	+02°59’ 50”	412.56	5658	220501_Li_.00000218.Entered_Coordinates.BLUE.FIT
	GENESIS 1 (V band)	02-05-2022	01:55:27	16h 43m 57s	+02°59’ 50”	412.56	5658	220501_Li_.00000217.Entered_Coordinates.GREEN.FIT
	COSMOS 2501 [GLONASS-K]	02-05-2022	01:23:44	16h 43m 56s	+02°59’ 50”	410.68	40542	220501_Li_.00000193.Entered_Coordinates.GREEN.FIT
	STARLINK-1969	02-05-2022	01:59:01	16h 43m 57s	+02°59’ 50”	412.59	5736	220501_Li_.00000220.Entered_Coordinates.GREEN.FIT
	PHASE 3B (AO-10)	08-05-2017	21:08:07	12h 43m 01s	-00° 32’ 16"	2231.91	41970	170508_Li_00000041.fits
	YAOGAN 9C	29-01-2011	05:37:34	08h 55m 42s	+78° 13’ 32"	4524.60	6420	110128_Li_.00000052.FIT
LDST	MOHAMMED VI-A	03-06-2023	21:06:56	15h 22m 20s	+36° 46’ 20”	15.30	5850	20230603_LDST-0015_G.fit
	2022-019J	03-06-2023	21:08:35	15h 22m 20s	+36° 46’ 20”	15.15	5712	20230603_LDST-0016_G.fit
	STARLINK-5051	03-06-2023	21:25:02	15h 22m 20s	+36° 46’ 21”	15.15	5724	20230603_LDST-0026_G.fit
	STARLINK-1150	03-06-2023	21:48:02	15h 22m 20s	+36° 46’ 21”	14.91	5736	20230603_LDST-0040_G.fit
	ONEWEB-0442	05-06-2023	21:54:17	15h 05m 17s	+1° 28’ 13”	14.19	6558	20230605_LDST-0014_R.fit
	ONEWEB-0090	03-06-2023	22:17:18	18h 09m 24s	+31° 06’ 59”	13.95	6588	20230605_LDST-0018_R.fit
	STARLINK-2326	03-06-2023	22:22:01	18h 09m 24s	+31° 07’ 00”	14.18	5736	20230605_LDST-0019_R.fit
	COSMOS 2477 [GLONASS-M]	03-06-2023	22:26:45	18h 09m 24s	+31° 07’ 00”	13.61	40542	20230605_LDST-0020_R.fit

Table 6: Table showing the observed satellite’s name, telescope used, name of the frame with the trail and the date, time, right ascension and declinaton of the observation. It also shows how far the epoch used to determine which satellite was in the frame is from the date of the observation.

6.3 Appendix C

```python
#!/usr/bin/env python
# coding: utf-8

# In[1]:


import numpy as np
from scipy import ndimage
from astropy.io import fits
import os
import astroalign as aa
from astropy.stats import SigmaClip, sigma_clipped_stats
from photutils.background import Background2D, MedianBackground
from photutils.detection import DAOStarFinder
from photutils.aperture import CircularAperture
from photutils.aperture import aperture_photometry
import matplotlib.pyplot as plt
from skyfield.api import load, wgs84


# # Master Bias

# In[2]:


#Master Bias


date = '110128'
bias = []
for subdir, dir, files in os.walk(f'/net/dataserver3/data/users/sterrenwacht/images/{date}/STL-6303E/i'):#for loop to iterate over all fits files
    for filename in files: #loop over only the files
        # open the bias files and append them to a master bias
        hdulist = fits.open(f"/net/dataserver3/data/users/sterrenwacht/images/{date}/STL-6303E/i/{filename}")# open the bias files and appen
        hdr = hdulist[0].header #header of bias files
        if hdr['IMAGETYP'] == 'Bias Frame':
            data = hdulist[0].data  #raw bias files
            bias.append(data)

#stacks the bias arrays along the z axis and take the median
master_bias = np.median(np.stack(bias, axis=0), axis=0)

#write the masterbias to disk
hdu = fits.PrimaryHDU(master_bias)
hdul = fits.HDUList([hdu])
hdul.writeto(f'/Users/users/tonckens/BULK/LDSThesis/master_bias.fits', output_verify='ignore', overwrite=True)


# # Master Dark

# In[3]:


#Master Dark
```

```
dark = []
#open the masterbias
master_bias = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_bias.fits"
    )[0].data

for subdir, dir, files in os.walk(f'/net/dataserver3/data/users/sterrenwacht/
    images/{date}/STL-6303E/i'):#for loop to iterate over all fits files
    for filename in files: #loop over only the files
        hdulist = fits.open(f"/net/dataserver3/data/users/sterrenwacht/images/{
            date}/STL-6303E/i/{filename}")# open the dARK files and appen
        hdr = hdulist[0].header
        if hdr['IMAGETYP'] == 'Dark Frame':#condition to only allow dark files
            in the next step
            data = hdulist[0].data  #raw dark files
            #subtract the master bias and divide by exposure time
            data = (data-master_bias)/300
            dark.append(data)

#stack the dark frames and take the median
master_dark = np.median(np.stack(dark, axis=0), axis=0)

#write masterdark to disk
hdu = fits.PrimaryHDU(master_dark)
hdul = fits.HDUList([hdu])
hdul.writeto(f'/Users/users/tonckens/BULK/LDSThesis/master_dark.fits',
    output_verify='ignore', overwrite=True)



# # Master Flat Field

# In[4]:


#imports the master and dark bias frames made before in order to create the
    Flatfield
hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_bias.fits")
master_bias = hdulist[0].data

hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_dark.fits")
master_dark = hdulist[0].data

flatR = []
flatB = []
flatV = []
for subdir, dir, files in os.walk(f'/net/dataserver3/data/users/sterrenwacht/
    images/{date}/STL-6303E/i'): #for loop to iterate over all fits files
    for filename in files:
        hdulist = fits.open(f"/net/dataserver3/data/users/sterrenwacht/images/{
            date}/STL-6303E/i/{filename}")
        hdr = hdulist[0].header
        if hdr['IMAGETYP'] == 'Flat Field':
            data = hdulist[0].data
            data = data - master_bias
            data = data - master_dark*hdr['EXPTIME'] #scales the master dark to
                the exposure time of the flatfield frame
            data = data/np.median(data)
            if hdr['FILTER'] == 'R':
                flatR.append(data)
            if hdr['FILTER'] == 'B':
```

```
                flatB.append(data)
            if hdr[’FILTER’] == ’V’:
                flatV.append(data)


#check if there are flatfields made
if len(flatR) !=0:
    master_flatR = np.median(np.stack(flatR,axis=0), axis=0) #take the median along the images
    #saves the flatfield in directory
    hdu = fits.PrimaryHDU(master_flatR)
    hdul = fits.HDUList([hdu])
    hdul.writeto(f’/Users/users/tonckens/BULK/LDSThesis/master_flatR.fits’, output_verify=’ignore’, overwrite=True)
    print("The Flatfield has been saved as a fits file in the directory.")
if len(flatB) !=0:
    master_flatB = np.median(np.stack(flatB,axis=0), axis=0) #take the median along the images
    #saves the flatfield in directory
    hdu = fits.PrimaryHDU(master_flatB)
    hdul = fits.HDUList([hdu])
    hdul.writeto(f’/Users/users/tonckens/BULK/LDSThesis/master_flatB.fits’, output_verify=’ignore’, overwrite=True)
    print("The Flatfield has been saved as a fits file in the directory.")
if len(flatV) !=0:
    master_flatV = np.median(np.stack(flatV,axis=0), axis=0) #take the median along the images
    #saves the flatfield in directory
    hdu = fits.PrimaryHDU(master_flatV)
    hdul = fits.HDUList([hdu])
    hdul.writeto(f’/Users/users/tonckens/BULK/LDSThesis/master_flatV.fits’, output_verify=’ignore’, overwrite=True)
    print("The Flatfield has been saved as a fits file in the directory.")


# # Noise reduction on the RGB frames

# In[5]:


#open the dark and bias files
hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_bias.fits")
master_bias = hdulist[0].data

hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_dark.fits")
master_dark = hdulist[0].data

if len(flatR) != 0:
    hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_flatR.fits")
    master_flatR = hdulist[0].data
if len(flatB) != 0:
    hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_flatB.fits")
    master_flatB = hdulist[0].data
if len(flatV) != 0:
    hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/master_flatV.fits")
    master_flatV = hdulist[0].data

```

```python
#define needed parameters found on the photutils site
sigma_clip = SigmaClip(sigma=3.)
bkg_estimator = MedianBackground()

for subdir, dir, files in os.walk(f’/net/dataserver3/data/users/sterrenwacht/images/{date}/STL-6303E/i’): #for loop to iterate over all fits files
    for filename in sorted(files):
        #opening data

        hdulist = fits.open(f"/net/dataserver3/data/users/sterrenwacht/images/{date}/STL-6303E/i/{filename}")
        hdr = hdulist[0].header
        if hdr[’FILTER ’] == ’R’ and hdr[’IMAGETYP’] == ’Light Frame’:

            data = hdulist[0].data
            #bias, dark, flat and background reduction
            data = data - master_bias
            data = data - master_dark*hdr[’EXPTIME’]
            if len(flatR) != 0:
                data = data/master_flatR
            bkg = Background2D(data, (50, 50), filter_size=(3, 3),sigma_clip=sigma_clip, bkg_estimator=bkg_estimator) # create the background estimation
            data = data-bkg.background
            #saving the data
            hdu = fits.PrimaryHDU(data,header=hdr)
            hdul = fits.HDUList([hdu])
            hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/red/{filename}", output_verify=’ignore’, overwrite=True) #saves the reduced images in the same directory as notebook
        if hdr[’FILTER ’] == ’B’ and hdr[’IMAGETYP’] == ’Light Frame’:
            #opening data

            hdr = hdulist[0].header
            data = hdulist[0].data
            #bias, dark, flat and background reduction
            data = data - master_bias
            data = data - master_dark*hdr[’EXPTIME’]
            if len(flatB) != 0:
                data = data/master_flatB
            bkg = Background2D(data, (50, 50), filter_size=(3, 3),sigma_clip=sigma_clip, bkg_estimator=bkg_estimator) # create the background estimation
            data = data-bkg.background
            #saving the data
            hdu = fits.PrimaryHDU(data,header=hdr)
            hdul = fits.HDUList([hdu])
            hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/blue/{filename}", output_verify=’ignore’, overwrite=True) #saves the reduced images in the same directory as notebook
        if hdr[’FILTER ’] == ’V’ and hdr[’IMAGETYP’] == ’Light Frame’:
            #opening data
            hdr = hdulist[0].header
            data = hdulist[0].data
            #bias, dark, flat and background reduction
            data = data - master_bias
            data = data - master_dark*hdr[’EXPTIME’]
            if len(flatV) != 0:
                data = data/master_flatV
```

```
            bkg = Background2D(data, (50, 50), filter_size=(3, 3),sigma_clip=sigma_clip, bkg_estimator=bkg_estimator) # create the background estimation
            data = data-bkg.background
            #saving the data
            hdu = fits.PrimaryHDU(data,header=hdr)
            hdul = fits.HDUList([hdu])
            hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/green/{filename}", output_verify='ignore', overwrite=True) #saves the reduced images in the same directory as notebook

print("All the images have been noise reduced and are now saved in the directory as fits files.")


# # align the images and normalize the flux to a single frame

# In[6]:


#lists for easy managment
red = []
alignedlist = []
red_name = np.array([])
red_hdr = []
alignedlistred = []

green = []
green_name = np.array([])
green_hdr = []
alignedlistgreen = []

blue = []
blue_name = np.array([])
blue_hdr = []
alignedlistblue = []

#allign the red images
for subdir, dir, files in os.walk(f'/Users/users/tonckens/BULK/LDSThesis/{date}/red'): #for loop to iterate over all fits files
    for filename in sorted(files):
        hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/{date}/red/{filename}")
        hdr = hdulist[0].header
        data = hdulist[0].data
        red.append(data)
        red_name = np.append(red_name,filename)
        red_hdr.append(hdr)


red = np.stack(red)

#image aligner and flux normalizer to a single frame
for t in range(len(red)):
    #aligns every image to reference image number 0
    aligned, footprint = aa.register(red[t], red[0])

    #source extractor for target flux
    mean, median, std = sigma_clipped_stats(red[0],sigma=3.0)
    daofind = DAOStarFinder(fwhm=3.0,threshold=5.*std)
```

```
    sources_source = daofind(red[0])

    #source extractor for source flux
    mean, median, std = sigma_clipped_stats(aligned,sigma=3.0)
    daofind = DAOStarFinder(fwhm=3.0,threshold=5.*std)
    sources_target = daofind(aligned)

    #normalize source to target by a factor of peak intensity of the same star
    #print(sources_target['peak'][np.argmax(sources_target['xcentroid'])], sources_source['peak'][np.argmax(sources_source['xcentroid'])])
    factor = sources_source['peak'][np.argmax(sources_source['xcentroid'])]/sources_target['peak'][np.argmax(sources_target['xcentroid'])]
    aligned = aligned*factor
    sources_target2 = daofind(aligned)
    #print(sources_target2['peak'][np.argmax(sources_target2['xcentroid'])])

    #save images
    alignedlistred.append(aligned)
    hdu = fits.PrimaryHDU(aligned,red_hdr[t])
    hdul = fits.HDUList([hdu])
    hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/red/{red_name[t]}", output_verify='ignore', overwrite=True)

hdu = fits.PrimaryHDU(np.stack(alignedlistred))
hdul = fits.HDUList([hdu])
hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/redaligned.fits", output_verify='ignore', overwrite=True)



#allign the blue images
for subdir, dir, files in os.walk(f'/Users/users/tonckens/BULK/LDSThesis/{date}/blue'): #for loop to iterate over all fits files
    for filename in sorted(files):
        hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/{date}/blue/{filename}")
        hdr = hdulist[0].header
        data = hdulist[0].data
        blue.append(data)
        blue_name = np.append(blue_name,filename)
        blue_hdr.append(hdr)

if len(blue) != 0:
    blue = np.stack(blue)

#image aligner and flux normalizer to a single frame
    for t in range(len(blue)):
        aligned, footprint = aa.register(blue[t], blue[0])#aligns every image to reference image number 0

    #source extractor for target flux
        mean, median, std = sigma_clipped_stats(blue[0],sigma=3.0)
        daofind = DAOStarFinder(fwhm=3.0,threshold=5.*std)
        sources_source = daofind(blue[0])

    #apply to all images by a factor
        mean, median, std = sigma_clipped_stats(aligned,sigma=3.0)
        daofind = DAOStarFinder(fwhm=3.0,threshold=5.*std)
        sources_target = daofind(aligned)

```

```
    #normalize source to target by a factor of peak intensity of the same star
        factor = sources_source['peak'][np.argmax(sources_source['xcentroid'])]/sources_target['peak'][np.argmax(sources_target['xcentroid'])]
        aligned = aligned*factor


    #save images
        alignedlistblue.append(aligned)
        hdu = fits.PrimaryHDU(aligned,blue_hdr[t])
        hdul = fits.HDUList([hdu])
        hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/blue/{blue_name[t]}", output_verify='ignore', overwrite=True)


    hdu = fits.PrimaryHDU(np.stack(alignedlistblue))
    hdul = fits.HDUList([hdu])
    hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/bluealigned.fits", output_verify='ignore', overwrite=True)

#allign the green images
for subdir, dir, files in os.walk(f'/Users/users/tonckens/BULK/LDSThesis/{date}/green'): #for loop to iterate over all fits files
    for filename in sorted(files):
        hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/{date}/green/{filename}")
        hdr = hdulist[0].header
        data = hdulist[0].data
        green.append(data)
        green_name = np.append(green_name,filename)
        green_hdr.append(hdr)


green = np.stack(green)


for t in range(len(green)):
    aligned, footprint = aa.register(green[t], green[0])#aligns every image to reference image number 0

    #source extractor for target flux
    mean, median, std = sigma_clipped_stats(green[0],sigma=3.0)
    daofind = DAOStarFinder(fwhm=3.0,threshold=5.*std)
    sources_source = daofind(green[0])

    #apply to all images by a factor
    mean, median, std = sigma_clipped_stats(aligned,sigma=3.0)
    daofind = DAOStarFinder(fwhm=3.0,threshold=5.*std)

    #normalize source to target by a factor of peak intensity of the same star
    sources_target = daofind(aligned)
    factor = sources_source['peak'][np.argmax(sources_source['xcentroid'])]/sources_target['peak'][np.argmax(sources_target['xcentroid'])]
    aligned = aligned*factor

    #save images
    hdu = fits.PrimaryHDU(aligned,green_hdr[t])
    hdul = fits.HDUList([hdu])
    alignedlistgreen.append(aligned)
    hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/green/{green_name[t]}", output_verify='ignore', overwrite=True)
```

```


hdu = fits.PrimaryHDU(np.stack(alignedlistgreen))
hdul = fits.HDUList([hdu])
hdul.writeto(f"/Users/users/tonckens/BULK/LDSThesis/{date}/greenaligned.fits",
    output_verify='ignore', overwrite=True)


# # Isolating the satellite trail

# In[8]:


hdulist = fits.open(f"/Users/users/tonckens/BULK/LDSThesis/{date}/greenaligned.
    fits") #open the file with a satellite track
dat = hdulist[0].data
track = dat[16] # image with the track
frame = np.median(dat,axis=0)#median of all images
track_iso = track-frame
#make the apeture for star
x_star = 1977
y_star = 354
r_star = 11
aperture = CircularAperture((x_star,y_star), r=r_star) #define the circle
circle1 = plt.Circle((x_star,y_star), r_star, color='r')

#make the box for the trail
xy = (110,470)
width = 25
length = 2995
rotation = 1.68
track_iso = ndimage.rotate(track_iso,rotation) #correct rotation
box = plt.Rectangle(xy, length, width, linewidth=1, edgecolor='r', facecolor='
    none')
IMAGE = (track_iso[xy[1]:xy[1]+width,xy[0]:xy[0]+length])




#plotting for checking
ax = plt.gca()
ax.cla()
ax.add_patch(box)
ax.add_patch(circle1)
plt.imshow(ndimage.rotate(track,rotation),cmap='gray', vmin=0, vmax=255)
txt = 'Figure 1: 110128_Li_.00000052.FIT with the aperture used to find
    zeropoint' #caption for the image
plt.figtext(0.5, 0.01, txt, wrap=True, horizontalalignment='center', fontsize
    =15)
plt.show()


#plotting the isolated trail
plt.imshow(IMAGE,cmap='gray', vmin=0, vmax=255)
plt.show()
print("Figure 2: The box, from previous image enclosing the satellite trail,
    plotted horizontally \n \n")

#find the counts and zeropoint for the star for magnitude determination
magnitude_star = 11.98 # star TYC 4544-1790-1
```

```
counts_star = ((aperture_photometry(ndimage.rotate(frame,rotation), aperture))['
    aperture_sum'])/red_hdr[0]['exptime']
zeropoint = magnitude_star + 2.5*np.log10(int(counts_star)) #determine the
    zeropoint


#find the maximum counts for the satellite trail
counts = np.array([np.sum(IMAGE[:,i]) for i in range(length)])
counts = counts/red_hdr[0]['exptime']
dif = np.max(counts)-np.mean(counts)

#find the magnitudes of the counts
magnitude_sat_eff = -2.5*np.log10(np.mean(counts))+zeropoint#calculate the
    magnitudes of the asteroid
magnitude_sat_eff_max = -2.5*np.log10(np.max(counts))+zeropoint#calculate the
    magnitudes of the asteroid
err = np.abs(magnitude_sat_eff_max-magnitude_sat_eff)

#magnitude calculations
period = 6420
angular_velocity = (360*3600)/period
D = 1496.76
t_exp = 300
r = 0.566
perigee = 764.6
magnitude_sat_apparent = magnitude_sat_eff + 2.5*np.log10(r/(angular_velocity*
    t_exp))
magnitude_sat_abs = magnitude_sat_apparent - 5*np.log10(D/1000)
magnitude_sat_perigee = magnitude_sat_apparent - 5*np.log10(D/perigee)

print(f"Effective V band magnitude satellite: {magnitude_sat_eff}+/- {err}")
print(f"Apparent V band magnitude satellite: {magnitude_sat_apparent}+/- {err}")
print(f"Absolute (1000m) V band magnitude satellite: {magnitude_sat_abs}+/- {err
    }")
print(f"V band magnitude satellite at perigee: {magnitude_sat_perigee}+/- {err}"
    )
```

7 Bibliography

References

Bassa, C. G., Hainaut, O. R., and Galadi-Enriquez, D. (2021). Analytical simulations of the effect of satellite constellations on optical and near-infrared observations.

Cayrel, R., Fisher, A., de Boer, J., Union, I. A., and on Illumination, I. C. (1980). *Guide Lines for Minimizing Urban Sky Glow Near Astronomical Observatories*. International Astronomical Union: Publication IAU/CIE. International Astronomical Union.

Celestrak (2023). CelesTrak: Current GP Element Sets — celestrak.org. `http://celestrak.org/NORAD/elements/`. [Accessed 23-May-2023].

Cipriano, Marín Jafar, J. (2007). *StarLight: A Common Heritage*. STARLIGHT INITIATIVE INSTITUTO DE ASTROFÍSICA DE CANARIAS (IAC).

Clarke, A. (1945). Extra-terrestrial relays. *Wireless World*, pages 305–308.

Falchi, F., Bará, S., Cinzano, P., Lima, R. C., and Pawley, M. (2023). A call for scientists to halt the spoiling of the night sky with artificial light and satellites. *Nature Astronomy*, 7(3):237–239.

Hainaut, O. R. and Williams, A. P. (2020). Impact of satellite constellations on astronomical observations with ESO telescopes in the visible and infrared domains. *Astronomy &amp Astrophysics*, 636:A121.

Hall, D. T. (2023). Semi-empirical astronomical light pollution evaluation of satellite constellations. *The Journal of the Astronautical Sciences*, 69(6):1893–1928.

Karpov, S., Katkova, E., Beskin, G., Biryukov, A., Bondar, S., Davydov, E., Ivanov, E., Perkov, A., and Sasyuk, V. (2016). Massive photometry of low-altitude artificial satellites on Mini-Mega-TORTORA. In *Revista Mexicana de Astronomia y Astrofisica Conference Series*, volume 48 of *Revista Mexicana de Astronomia y Astrofisica Conference Series*, pages 112–113.

Kocifaj, M., Kundracik, F., Barentine, J. C., and Bará, S. (2021). The proliferation of space objects is a rapidly increasing source of artificial night sky brightness. *Monthly Notices of the Royal Astronomical Society: Letters*, 504(1):L40–L44.

Kondo, Y. (2003). *Important Roles of Small Telescopes in Space*, pages 227–237. Springer Netherlands, Dordrecht.

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2010). Astrometry.net: Blind Astrometric Calibration of Arbitrary Astronomical Images. , 139(5):1782–1800.

Mallama, A. (2021). Starlink satellite brightness – characterized from 100, 000 visible light magnitudes.

Mallama, A. (2022). Oneweb satellite brightness – characterized from 80, 000 visible light magnitudes.

Rhodes, B. (2019). Skyfield: High precision research-grade positions for planets and Earth satellites generator. Astrophysics Source Code Library, record ascl:1907.024.

Riebeek, H. (2009). Catalog of earth satellite orbits. `https://earthobservatory.nasa.gov/features/OrbitsCatalog/page1.php`. (Accessed on 05/01/2023).

satobs (2007). Iridium flares. `http://www.satobs.org/iridium.html#observation`. (Accessed on 05/10/2023).

Tregloan-Reed, J., Otarola, A., Ortiz, E., Molina, V., Anais, J., González, R., Colque, J. P., and Unda-Sanzana, E. (2020). First observations and magnitude measurement of starlink's darksat. *Astronomy &amp Astrophysics*, 637:L1.

Vallado, D., Crawford, P., Hujsak, R., and Kelso, T. (2006). Revisiting spacetrack report #3. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*. American Institute of Aeronautics and Astronautics.

Walker, C., Hall, J., Allen, L., Green, R., Seitzer, P., Tyson, T., Bauer, A., Krafton, K., Lowenthal, J., Parriott, J., and et al. (2020). Impact of satellite constellations on optical astronomy and recommendations toward mitigations.

Wood, L. (2003). *Satellite Constellation Networks*, pages 13–34. Springer US, Boston, MA.