Don't Interfero with me: Exploring Radio Interferometry at X-Band

Khushi Nesarikar
April 10, 2024

ABSTRACT

In this paper, I investigate interferometry and learn how to use scripts to control telescopes to track the Sun. Furthermore, I used the data from tracking the sun from horizon to horizon to calculate the baseline, which is the distance between the telescopes in the east-west direction and the north-south direction. I found this to be 15.01 ± 0.35 m in the east-west direction and 1.06 ± 0.16 m in the north-south direction. I also used the fact that the signal from tracking the Sun was a Bessel function to calculate the angular radius of the Sun, which I found to be 0.00438 ± 0.00002 radians.

1. Introduction

Interferometry is a very essential part of radio astronomy and is a great tool to observe sources and derive information from the observed sources. On New Campbell Hall, we have two interferometers on the roof that can be controlled remotely using a Rasberry Pi. There is also a SNAP spectrometer which digitises, Fourier transforms, cross-correlates and integrates the two signals we got from the dishes for time. This gives us all the tools we need to be able to observe sources, such as the Sun, the Moon and point sources, to calculate the distance between the two telescopes and the radius of the source. In this paper, I tracked the Sun and used that to calculate the baseline and the angular radius of the Sun. We did attempt to track the Moon and 3C144 (the Crab Nebulae, a point source), however, the signals were not bright enough for us to get any intelligible data out of it which was unfortunate.

In §2 we introduce the techniques used in this paper. In §3 we detail our process of writing code to track the Sun and take data using the SNAP spectrometer, conduct our analysis and show the results of the baseline between the two interferometers we calculated and the radius of the Sun. We conclude in §4.

2. Background

The main theoretical concept to understand here is the concept of Interferometry. Interferometry is the process of using the interference between two (or more) telescopic arrays to be able to observe astronomical sources. Interference is when two or more waves, either in or out of phase, interact so as to add their displacements together. The way our interferometer

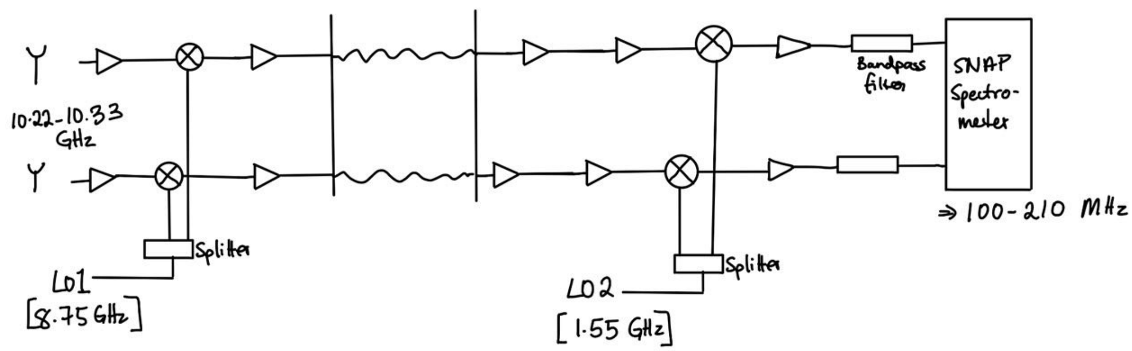

Fig. 1.— Diagram of the interferometry setup that includes the two interferometers and a complicated system of amplifiers, bandpass filters and two different local oscillators (LOs) being mixed in. The SNAP Spectrometer then samples the received signal at 500 Msps and the signal has a center of approximately 155 MHz.

setup works in New Campbell Hall is that we have 2 telescopes on the roof. Astronomical sources emit electromagnetic radiation, and since they are so incredibly far away from us, by the time they arrive at our telescopes they are essentially plane waves. The telescopes are placed at a certain distance away from each other though, and the distance between the two of them is known as the baseline vector, **b**. Due to this separation, the radiation arrives at the two telescopes at different times (2), which is known as the geometric delay. This delay changes over time as the source moves across the sky. The time delay is given as:

$$\tau = \frac{\mathbf{b} \cdot \hat{\mathbf{s}}}{c} \tag{1}$$

The signal is sampled with a SNAP spectrometer, which samples at 500 Msps. The interferometer setup basically projects a giant sine wave in the sky, and through our sampling with the SNAP spectrometer we get the visibility or the brightness of the source. The visibility is basically the Fourier transform of the perceived sky. The geometric delay is due to the difference between the distance from the interferometers to the source which changes throughout the day. When the source is at the horizon, the difference between the distance is equal to the baseline and when the source is directly above the interferometers the difference is zero. This allows us to be able to calculate the baseline with the visibility spectrum that we get since the can get the fringe frequency, which represents how the phase changes. This fringe frequency, however, depends on the time that the signal is received and the specific hour angle. Therefore, there is the concept of a local fringe frequency, which is the fringe frequency at a given hour angle and declination as shown is 2. The ω in this equation is the angular rotation of the Earth.

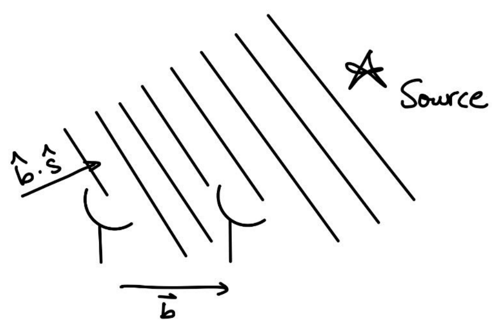

Fig. 2.— Diagram of how the signal is received by the interferometers. Since the source is so far away, by the time the radiation arrives at the interferometers they are plane wave, therefore there is a time delay between the signal arriving at one interferometer vs the other. Furthermore, you can also see the baseline vector being the distance between the two interferometers.

$$\frac{f_f}{\omega} = \left[\frac{b_{ew}}{\lambda}\cos\delta\right]\sin h_s + \left[\frac{b_{ns}}{\lambda}\sin L\cos\delta\right]\cos h_s \tag{2}$$

We specifically observe the Sun in this lab. When we sample the Sun in one specific frequency channel through time, we can see that the shape of the visibility spectrum is a Bessel function (or an airy disk). This is because when the take the Fourier transform of a circle in the sky, this gives us an airy disk and a cut through an airy disk is a Bessel function. As we can see in the Bessel function for the Sun (3), it contains information about the baseline vector as well as the angular radius of the source. Therefore, this can be used to estimate the radius of the source given an accurate baseline.

$$\begin{gathered}\alpha\left(\frac{|J_1\left(2\pi u_{\max}\cos(h_s)r_{\odot}\right)|}{2\pi u_{\max}\cos(h_s)r_{\odot}}\right)+\beta \\ u_{\max}=\frac{|b|}{\lambda}=\frac{\sqrt{b_{ew}^2+b_{ns}^2}}{\lambda}\end{gathered} \tag{3}$$

3. Interferometry

In performing this experiment, we had to develop scripts that would be able to track the movements of a given source and record data at the same time. Then we observed the Sun from horizon to horizon and performed analysis of the data. From the Sun data, we derive the baseline of our interferometers and the radius of the Sun.

3.1. Tracking the Sun

To track the Sun, we made extensive use of `ugradio coord`, specifically `sunpos`. We also relied heavily on threading, which enabled us to run multiple functions simultaneously by running them on different threads. We had three main functions: a function to track the position of the sun using `sunpos`, a thread to take data from the SNAP spectrometer and a function to keep track of time. The time function enabled us to specify how long we wanted to record data for and set a global variable (initially set to True) to be set to False once that amount of time had passed which would effectively stop the other functions from running as well.

We updated the position of the interferometers every 30 seconds and took data continuously. We found the average time between each data collection to be 1.25 seconds. From the SNAP spectrometer, we got the spectrum and the time that the data was collected in Unix time. We ensured to save the data as soon as we received it from the SNAP spec to ensure that we didn't miss any data points. I also wrote similar scripts to track the Moon (using `ugradio coord moonpos`) and to track the Crab Nebulae. However, after obtaining a couple of hours worth of data from the Moon and Crab we found that the signal was too faint to be able to observe the interference fringes and thus focused solely on the Sun. We first conducted this over shorter time chunks like 30 minutes and 2 hours to ensure our tracking code was fully functional and then took data from horizon to horizon. Due to some technical issues with the interferometers, we ended up taking data from 11 am to 7 pm.

3.2. Analysis

Now, with our data in hand, we could finally investigate and try to answer the questions we had. To be able to visualise our data in a cohesive way and to observe the interference fringes from the Sun, I made waterfall plots of our data (3). The waterfall plots basically stack the Fourier-transformed, cross-correlated data vertically next to each other using `imshow`. The sinusoidal pattern from the interference thus gets translated into darker parts (peaks) and lighter parts (troughs) which is the visibility at that point. Therefore, when we stack them we can observe the fringes from the Sun.

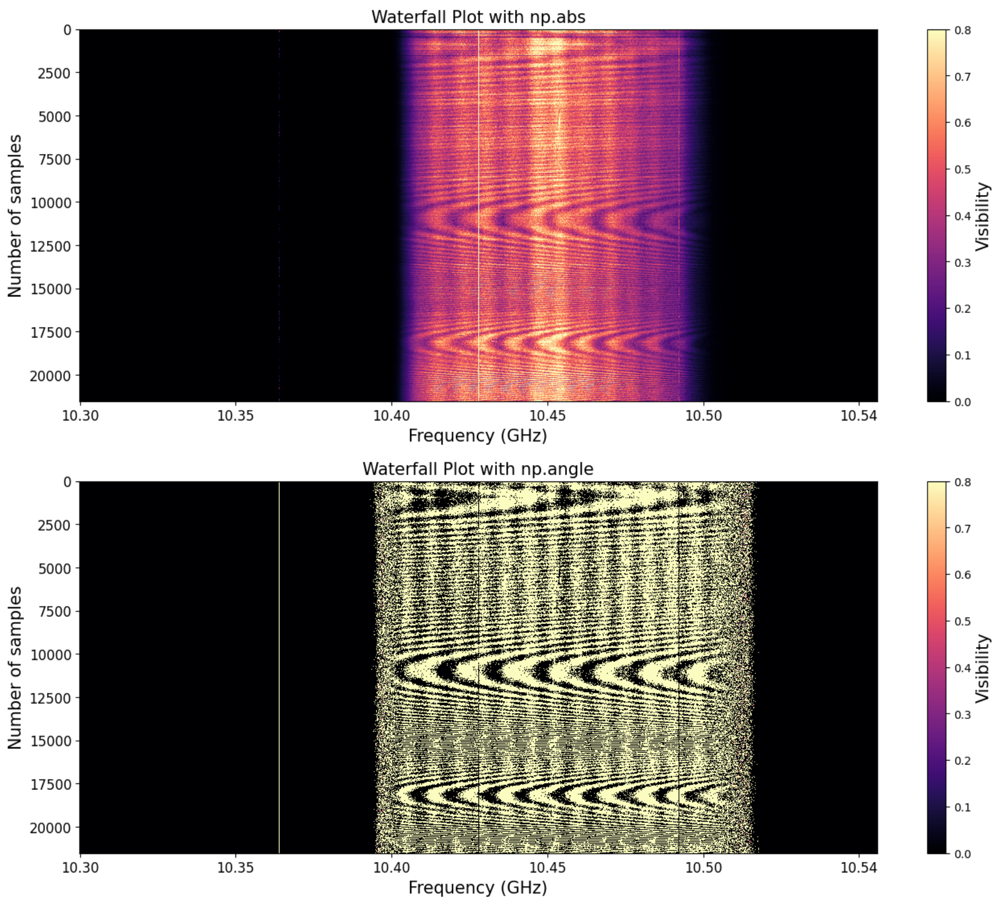

Fig. 3.— **Waterfall Plots** *Top Panel:* This waterfall plot shows the absolute value of the data from observing the Sun for 7 hours. We can observe the fringes in the lines that run across the plot. Due to our computer's compression of the images, however, aliasing occurs which causes some parts to look curvy like a tree trunk. This plot does not really show the intricacies of the fringes. *Bottom Panel:* This waterfall plot shows the counterclockwise angle from the positive real axis on the complex plane. Here, we can see the fringes more clearly. We can also still see the aliasing. Furthermore, there is also a line at around 10.36 GHz for which I'm not too sure the origin is.

Now that we have been able to visualize the data and do some preliminary analysis, we can move on to using the data to find the baseline. For this, first I had to get the local fringe frequencies. To do this, I performed a short-time Fourier transform on my data. This involves taking small chunks of data in time and performing a Fourier transform on each chunk. For the sampling frequency, I used one over the average time in between each data point. From this, I was able to visualise how the local fringe frequency changed over time (4). After that, I obtained the peak fringe frequency for each chunk of time sampled and fit a model of the local fringe frequencies using 2 and `scipy curvefit` to get optimised values for the baseline in the east-west direction and in the north-south direction.

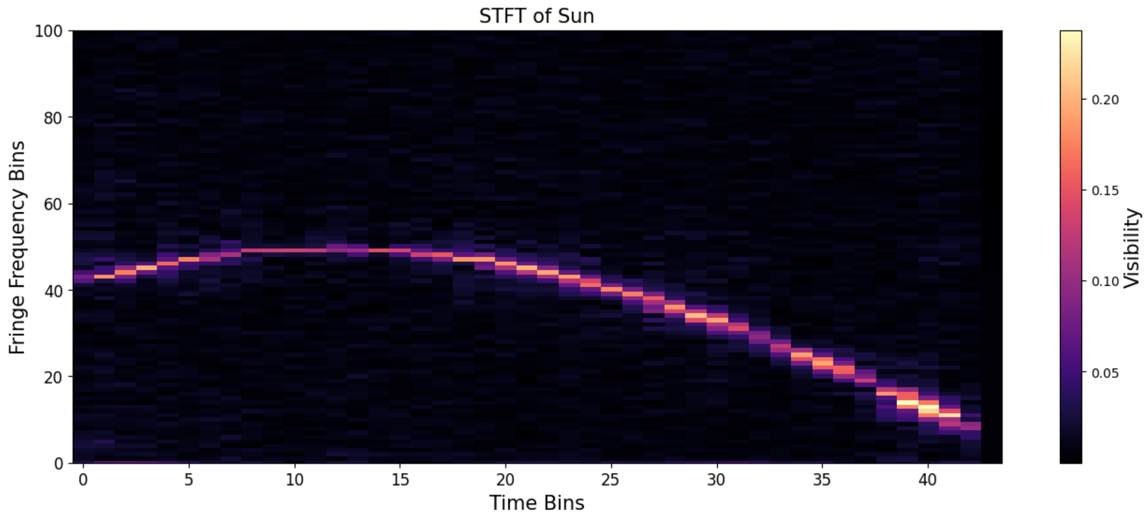

Fig. 4.— This figure shows the short-time Fourier transform (STFT) of the data to obtain the local fringe frequencies. The sampling frequency is $\frac{1}{1.25}$ where 1.25 seconds is the average time in between data points. I chose the length of each segment to be 1024 which was the number of samples in each spectrum.

As the Sun moves across the sky, the visibility resembles a Bessel function (6) since you would expect the Fourier transform of a sphere to be a Bessel function (or airy disk). Using the baseline that we just derived and its projection towards the Sun, we can use 3 to fit for the angular radius of the Sun. Alternatively, we can also fit for the "zeros" or the times when the Bessel function goes to zero to get the radius. I decided to use the first method.

3.3. Results

For the East-West baseline, we got the value to be 15.01 metres and the North-South baseline to be 1.06 metres. To get the errors on the values, we used the brute-force method to essentially re-fit the data and minimise the χ^2 error for a range of values for East-West and North-South baselines. Based on this, we found the range of the spread within 2 standard deviations of that minimised χ^2 value for both variables to determine the error in them.

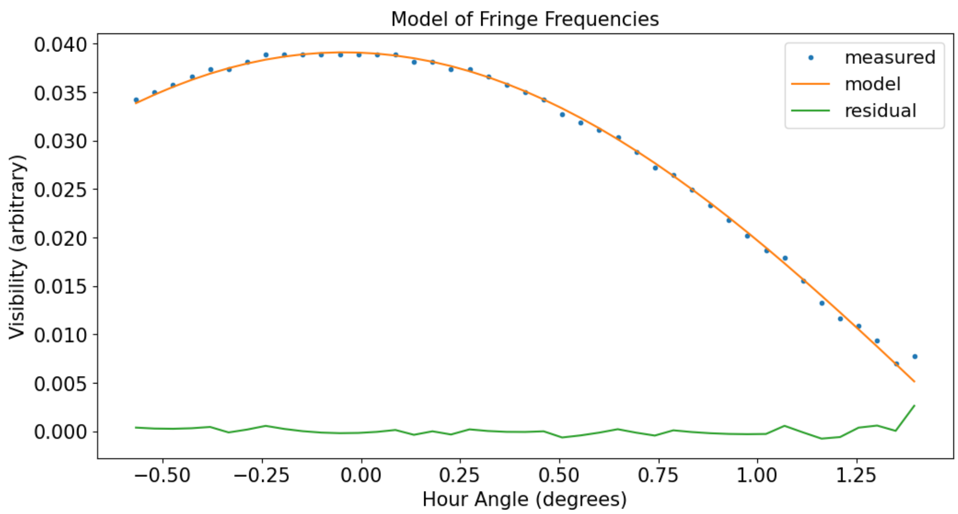

Fig. 5.— This figure shows the fitted model for the local fringe frequency. The model visually appears to fit well and the residuals have no apparent pattern and are close to zero. With this model I got the east-west baseline to be 15.01 metres and the north-south baseline to be 1.06 metres with a reduced χ^2 of 1.05 which seemed acceptable.

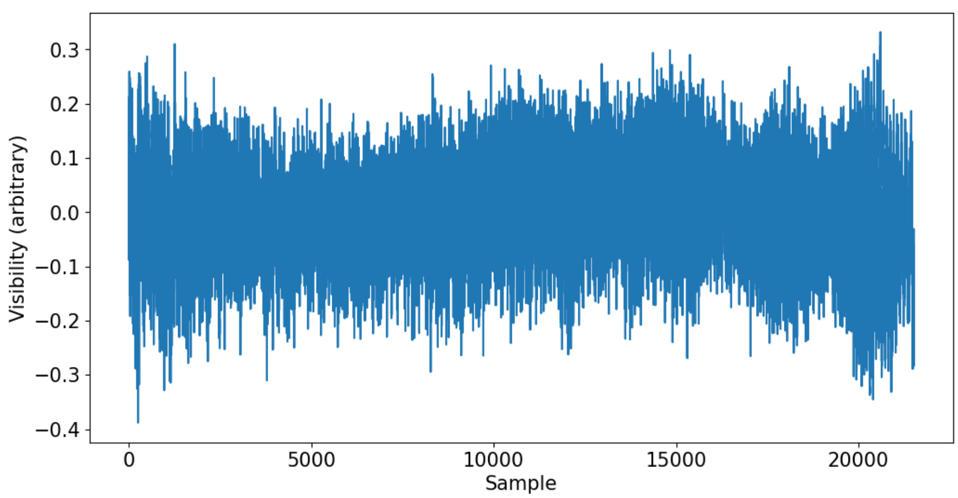

Fig. 6.— This figure shows the visibility spectrum for a single frequency channel through time. Through investigating and looking through many channels, I found channel 660 to have a clear Bessel shape. Towards the end, you can see the around four lobes that are characteristic of a Bessel function. Since we did not start taking data exactly at sunrise, there are not really visible lobes in the beginning.

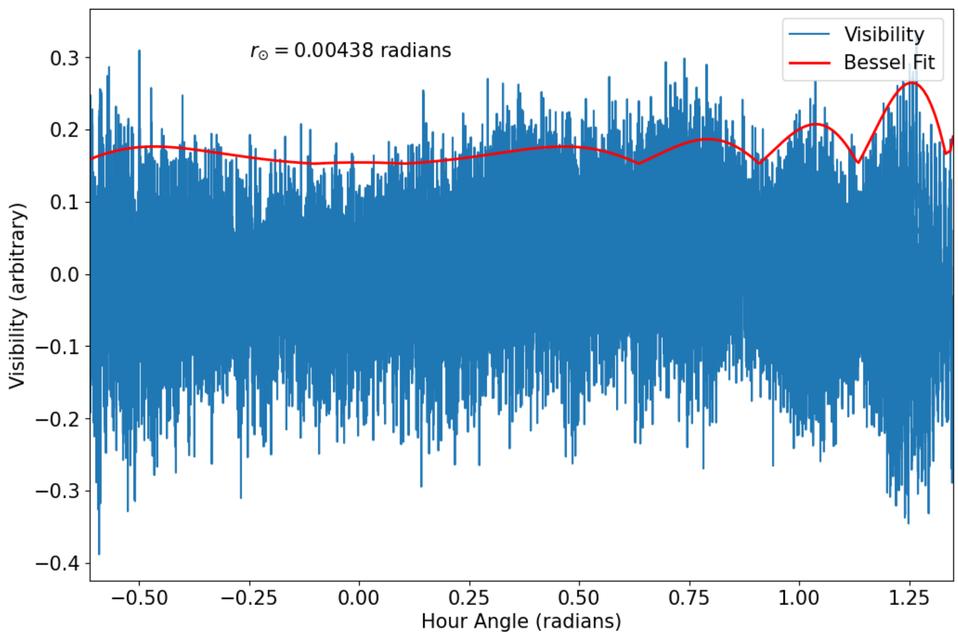

Fig. 7.— This figure shows the Bessel fit for the frequency channel 660. The fit seems to fit pretty well and shows 4 lobes on the right side and 1 lobe on the left side. This fit gives the radius to be 0.00438 radians.

Doing this, we found the values to be 15.01±0.35 m in the East-West direction and 1.06±0.16 m in the North-South direction. Based on comparison with other groups, this seems to be a reasonable result and based on our reduced χ^2, which was 1.05, I believe that these are within reason.

For the angular radius, I obtained the error with the `curvefit` function. My final results were 0.00438 ± 0.00003 radians. In degrees this is 0.251 ± 0.002 degrees. Based on estimations and some Googling, I find the actual radius of the sun to be 0.25 degrees, or 0.00436 radians, which is within my margin of error making me confident in my results and fit.

4. Conclusion

In conclusion, we were successfully able to track and observe the Sun using self-written scripts and obtain visibility spectra that could be used to determine the baseline distance between our two interferometers and the angular radius of the Sun. We were also able to determine the errors in our values by minimizing the χ^2 and obtaining the error bars around

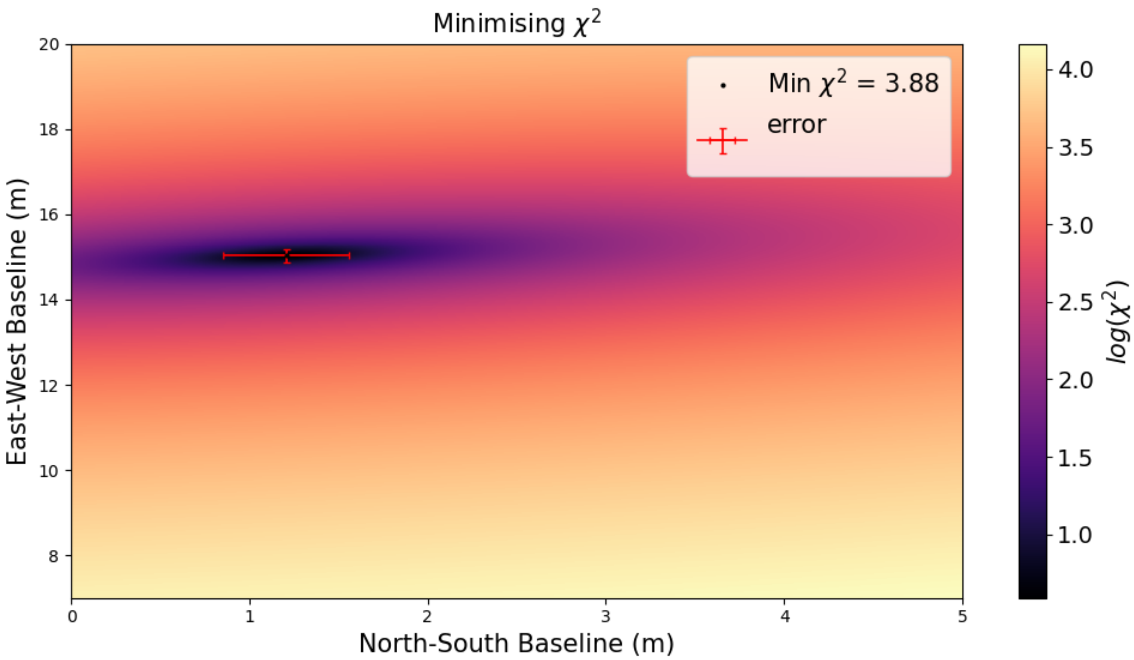

Fig. 8.— This is an `imshow` plot of the χ^2 values for different combinations of east-west baseline (from 7 to 20 metres) and north-south baseline (from 0 to 5 metres). We can see that the lowest χ^2 values are between 14 for east-west and 0 to 3 for north-south. The spread is more in the north-south direction since we did not have the complete horizon to horizon data making it harder to estimate the north-south baseline since the Sun does not move as significantly in the north-south direction over a shorter period of time.

that value. Our results are summarised in 1.

Variable	**Value**	**Error**
East-West Baseline	15.01	0.35
North-South baseline	1.06	0.16
Angular Radius of Sun	0.00438	0.00002

Table 1: Summary of all the important results obtained including the error in the values I got.

Acknowledgements

I firstly would like to acknowledge my group members, Vassili McAnney and Luca Castle with whom I spent time collecting data and analysing it. I would also like to acknowledge

Professor Aaron Parsons for his invaluable feedback and help and the UGSI Mallory for her support and help. I also would not have been able to complete this lab without the help of SUSSY, especially Sumbal and Charlie and their help and support. Finally, I want to acknowledge the software packages I used that were an essential part of the analysis: (`numpy`, `ugradio`, `scipy`, `matplotlib`,`astropy`, `glob` and `seaborn`.