Tracking Coronal Mass Ejections using Fourier Transform

Keshav Janyani

National Initiative on Undergraduate Studies

Winter Project
Report

Supervisor:
Prof. Dipankar Banerjee
Indian Institute of Astrophysics

December 2017

Contents

Chapter 1

Introduction

A Coronal Mass Ejection(CME) is a large eruption of plasma and magnetic field from the Sun. A typical CME has a mass of around 10^{11}–10^{12} kg and has a speed between 400 and 1,000 km/s.[1] There has been scientific interest to more about CMEs as they can have adverse effects on space weather. There have been various instruments which continuously monitor the sun for observing CMEs. Some of these are LASCO-SOHO and STEREO-SECCHI. There are Automated algorithms that detect CMEs from coronographs taken from these instruments like CACTus and SEEDS. These algorithms however fail to apply accurately to a particular instrument COR-1 which is a part of STEREO-SECCHI. My aim during this winter project was to study CMEs and be able to implement a pre-processing algorithm based on Fourier transforms that would reduce noise from COR-1 data so that CACTus can be implemented on this data.

1.1 Coronal Mass Ejections

Coronal mass ejections consist of large structures containing plasma and magnetic fields that are expelled from the Sun into the heliosphere. They are of interest for both scientific and technological reasons. Scientifically they are of interest because they remove built-up magnetic energy and plasma from the solar corona, and technologically they are of interest because they are responsible for the most extreme space weather effects at Earth, as well as at other planets and spacecraft throughout the heliosphere.[2] When CMEs impact the Earth's magnetosphere, they are responsible for geomagnetic storms and enhanced aurora. CMEs originate from highly twisted magnetic field structures, or "flux ropes", on the Sun. [3]

1.1.1 Observation

Until the early years of this century, images of CMEs had been made near the Sun primarily by coronagraphs on board spacecraft. Coronagraphs view the outward flow of density structures emanating from the Sun by observing Thomson-scattered sunlight from the free electrons in coronal and heliospheric plasma.[2] They were observed used spacecraft based coronagraphs from early 1970s.
In 1996, Solar and Heliospheric Observatory (SOHO) was functional in space to study the Sun. Originally planned as a two-year mission, SOHO continues to operate after over 20 years in space. In November 2016, a mission extension lasting until December 2018 was approved. Large Angle and Spectrometric Coronagraph (LASCO) is one of the instruments aboard SOHO which consists of coronagraphs to monitor

the corona of the Sun. LASCO comprises of three telescopes (C1, C2 and C3), each of which looks at an increasingly large area surrounding the Sun. However, C1 stopped functioning around 1998. Telescopes C1 and C2 still send us data on daily basis. In 2006, Solar Terrestrial Relations Observatory(STEREO),

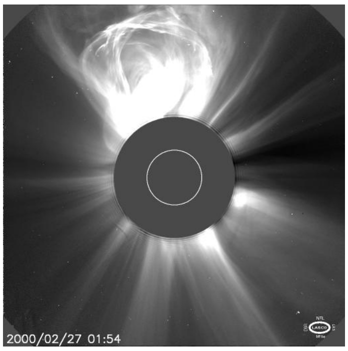

Figure 1.1: Image Taken by C2 onboard SOHO

which consisted of two nearly identical spacecrafts was launched. One of the spacecrafts is ahead of Earth in its orbit while the other one trails earth. Sun Earth Connection Coronal and Heliospheric Investigation (SECCHI) is one of the suites of instruments on both of these spacecrafts. The suite consists of 5 scientific telescopes. The primary goal of the STEREO mission is to advance the understanding of the three-dimensional structure of the Sun's corona, especially regarding the origin of coronal mass ejections (CMEs), their evolution in the interplanetary medium, and the dynamic coupling between CMEs and the Earth environment. COR1 and COR2 observe the inner (1.4-4 R_{sun}) and outer (2-15 R_{sun}) corona with greater frequency and polarization precision than ever before.

1.2 Corona

Corona consists of the plasma that surrounds the Sun and extends for up to millions of kilometers in space. It is visible to the naked eye during the solar eclipse. Continuous observation of corona is carried out by corona graphs which block out the sun. The temperature of Corona is extremely high and the explanation behind it is still an open question in Solar Physics popularly known as the Coronal Heating Problem. Corona has several components emitting in entire visible spectrum.
K(continuous)-Corona is visible due to Thomson scattering of sun light by free electrons in the plasma. The spectrum of this corona is continuous. K corona is found to be strongly polarized. [12] The F(Fraunhofer)-Corona is due to scattering of sunlight by dust particles. It has Fraunhofer spectra with absorption lines.The radial gradients of the K-and the F-components are such that the K-corona dominates inside about 2.5 R_{sun}, while the F-corona dominates beyond. E-Corona is due to a line emission spectrum of highly ionized atoms of Fe, Ni and Ca. This covers the light that is actually emitted by the particles in Corona.
The light received on COR 1 primarily consists of K-Corona, F-Corona and Instrumental scattered light. F-Corona mostly consists of static features of corona and hence to observe CMEs, we are mostly focused on K-Corona. While separating different spectrum we make use of the fact that K-Corona is polarized and that F-corona is fairly constant. We take the minimum of each pixel ranging over a particular time-period from Total Brightness images to get the F-corona and instrumental scattered light. This minimum is then

subtracted to get higher proportion of K-corona in the resulting image. Similarly, we can use polarised brightness images at 3 different angles(0°,120°,240°).

1.3 Fourier Transforms

Fourier Transforms are used to transform a function from time or spatial domains to frequency or spatial frequency domains. A Fourier transform (when defined) of a function $f(t)$ is given by

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt$$

Note that in general $F(\omega)$ is a complex valued function. Fourier transforms on functions with multiple variables are defined similarly.For example for a function $f(x,t)$, the Fourier Transform is shown below

$$F(k,\omega) = \iint_{-\infty}^{\infty} f(x,t)e^{-i(\omega t+kx)}dxdt$$

The Fourier Transform is used in a wide range of applications related to image processing, such as image analysis, image filtering, image reconstruction and image compression.The output of the transformation represents the image in the Fourier or frequency domain.[4]

1.3.1 Discrete Fourier Transforms

The Discrete Fourier Transform (DFT) is the equivalent of the continuous Fourier Transform for signals known only at instants separated by sample times T (i.e. a finite sequence of data).[5]

$$\begin{aligned} X_k &= \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N} \\ &= \sum_{n=0}^{N-1} x_n \cdot [\cos(2\pi kn/N) - i \cdot \sin(2\pi kn/N)], \end{aligned}$$

DFT has some interesting properties or characteristics such as the Nyquist limit.

A sampled waveform needs at least two sample points per cycle. Put another way, the wave's frequency must not be above half the sampling frequency. This limit is called the Nyquist limit of a given sampling frequency.A sine wave higher than the Nyquist frequency is sampled, a sine wave of lower frequency results. This effect is called aliasing.[6]

It is this form of Fourier Transforms that is used in image processing as images can be thought of two dimensional discrete functions of two variables. For a square image of size NM, the two-dimensional DFT is given by:

$$F(k,l) = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f(n,m)e^{-i2\pi(kn/N+lm/M)}$$

where $f(a,b)$ is the image in the spatial domain and the exponential term is the basis function corresponding to each point $F(k,l)$ in the Fourier space. The equation can be interpreted as: the value of each point $F(k,l)$ is obtained by multiplying the spatial image with the corresponding base function and summing the result.$F(0,0)$ represents the DC-component of the image which corresponds to the average brightness and $F(N-1,N-1)$ represents the highest frequency.[4]

Computer programs make use of Fast Fourier Transform(FFT) as an algorithm to calculate DFTs as it is much faster and gives the exact same result as DFT. FFTs make use of the symmetries that exist in DFT.

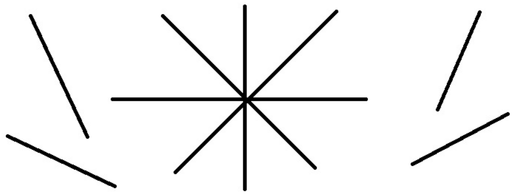

Figure 1.2: A sample image containing lines of various slopes

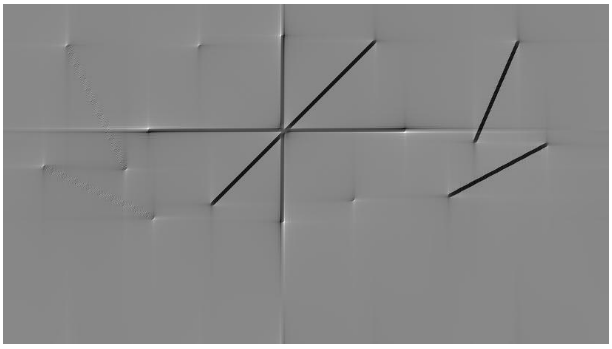

Figure 1.3: A filtered image without lines of negative slope

1.3.2 Use of Fourier Transform in the Algorithm

Fourier Transform was used to localize moving features. The basic idea behind the usage being that we can isolate features having a particular range of slopes using Fourier transforms. A very example of that demonstrates this is shown in the figure 1.1 and 1.2.
Since CMEs are radially outward moving features. It helps to remove noise or other features that have radially inward velocity. The idea is inspired from the paper referred [7], which uses this method on a very different topic. We first create time-distance maps on which this method is applied.
Consider a time–distance image $I(r, t)$ that maps the value of some quantity as a function of position (r) and time (t). Then I(r,t) can be decomposed into velocities as

$$I(r, t) = \int f_v(r - vt)dv$$

where f_v is a function of $s_v = r - vt$ Applying 2D Fourier transform to a single velocity component $f_v(s_v)dv$ gives

$$F_v(k_r, \omega) = (2\pi)^{-\frac{1}{2}}\delta(vk_r + \omega)\int e^{-ik_r s} f_v(s)ds$$

which is zero everywhere except the line $\omega = -k_r v$. Hence a feature having radial speed v appears on the line $\omega = -k_r v$ in the Fourier transform.[7]
Hence we can mask out every velocity except in our desired range of velocities to reduce noise.

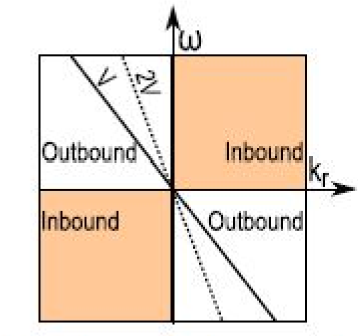

Figure 1.4: A feature moving with velocity v outward gets localized along the line indicated[7]

Chapter 2

Algorithm

The Algorithm described here has been applied on COR1A Data for some specific dates. The images used to demonstrate the results are from the images of the date $2010/08/01$ on which as many as 6 CMEs have been noted down in the manual catalog.[8]

2.1 Preliminaries

Before applying the algorithm, we need to first create the data-set in the required form. This involves getting the data in level 1 form. The transformation of level 0.5 data to level 1 can be done with the secchi_prep command of Interactive Data Language(IDL) available in the solarsoft(ssw) package. [9]
On board corrections like missing block correction, solar north corrections are applied before creating Level 0.5 data. Conversion to level 1 data involves flat field correction and dark/bias subtraction. Dark subtraction is applied because even when no signal is given, the detectors have some of their pixel readings non zero.

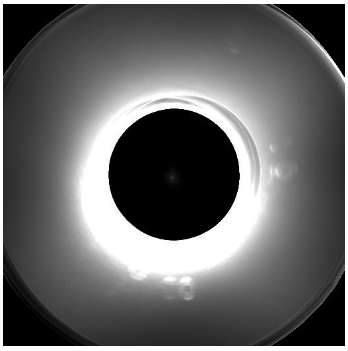

Figure 2.1: An example of a level 1 image

2.1.1 Getting Background Subtracted Images

The available Background Images online use the algorithm described in the paper referred. [9] However, these background images were not of use in this algorithm due to some scaling issues and not much clarity obtained for the use of detecting CMEs as the backgrounds available are more suitable for study of streamers and other features. So Backgrounds were generated for this purpose separately.

We first remove faulty images which occur due to some technical errors in COR1 A. To do this, we first find the mean intensity value of all the images and then the mean of these means say ’m’ and the standard deviation of these means say ’std’. All those images whose mean doesn’t lie in $(m - k * std, m + k * std)$ (where $k = 2$ is found to give good results) are replaced with images that high a constant high pixel value. This is done so that these images do not interfere in the background image calculation process. These images are not deleted as it would then cause confusion on the time-axis of time-distance maps.

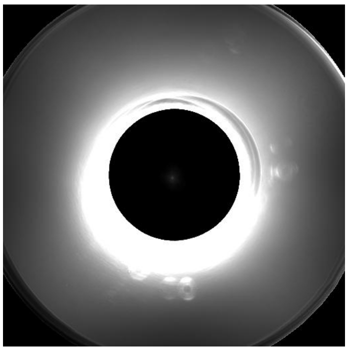

Figure 2.2: An example of a background image

Now, from the data-set minimum value of intensity for each pixel is calculated and stored to get a background image. Note that IDL at times cannot handle such a large data-set, hence it is advised to divide the data-set into three smaller data-set and get three background images, one for each set and then taking the minimum over them. Then we just subtract each original level 1 image from the background obtained to get Background Subtracted images. We need to make sure the header is stored in the Background Subtracted Images as well.

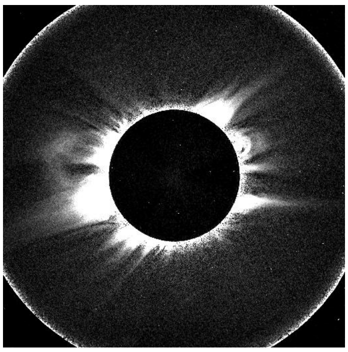

Figure 2.3: An example of a background subtracted image

2.2 Polar Transform and Time-Distance Maps

Now, the Background subtracted images are transformed into polar images with the coordinate system having the origin at the center of the Sun. For this, we first calculate the pixel number of the center of the Sun using the method given in the paper referred[10]. Then the coordinate transform is carried out using standard procedure. We use 720 bins instead of 360 bins so that information accurate to half a degree is stored and under-sampling does not occur.

While doing the polar transform, it is essential that the time at which the image was taken in stored in the image FITS header. Hence it would be appropriate to copy the header of Cartesian image into the polar one. Also since the starting pixels are blocked by the occulter and the later pixels are not uniform, we just keep the intermediate pixels. For a $1024X1024$ keeping information about radial pixels 215 to 500 is sufficient while that for a $512X512$ (After 2010) information for 100 to 255 radial pixels is good enough. Note that these numbers change for other instruments accordingly.

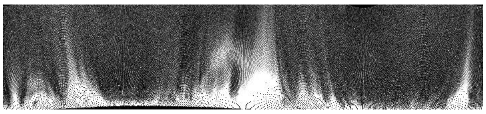

Figure 2.4: An example of a Polar image

Now, these Polar images are transformed to time-distance maps by first storing these images in a 3D array containing radial, theta and t dimensions and then storing radial and time dimensions together for each theta.Now our images are ready for the filter to be applied.

2.3 Fourier Filtering and Regenerating Images

The suggested ranges for the velocity are 100-2500 km/s.[11] The corresponding pixel ranges can be obtained by looking at the cadence in the image headers.A filter mask is created having pixel value 1 within the required ranges and value 0 out of these ranges. Also the pixel values corresponding to higher frequencies(near the center) are removed so as to reduce noise. Also streamers are found to lie in the lower frequency range, so we filter them out too.

We first get the Fourier transform of the image using 'fft' function and then multiply the filter to it. We then take the inverse Fourier transform to get the filtered image back. Note that in order to store or display the image, it might be necessary to take real part of the image as in general the inverse Fourier transform might be complex.

We then transform these time-distance maps to polar form. While storing the Cartesian form, we must make sure that the center if the image is transformed to the correct pixel of the Sun so that further processing can take place. For this, it is essential that the information of the header is maintained throughout. So, while transforming images to polar form it is necessary that the time information is stored in the header, so that the polar image can be linked to the original image. This will allow the header of the original image to be used and copied to the new filtered image. This completes our algorithm to filter out unwanted noise from our images that lie outside our interested velocity ranges.

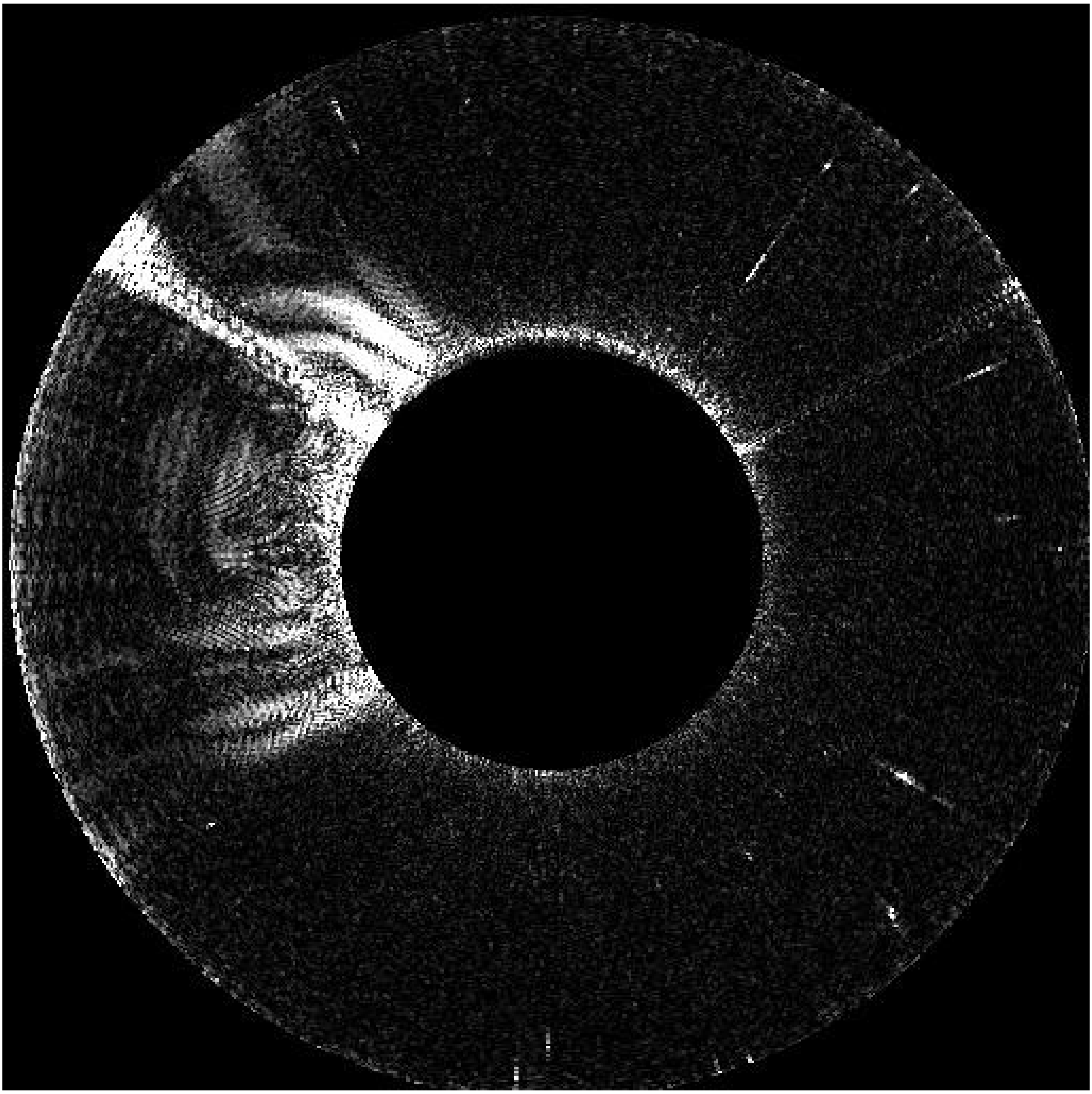

Figure 2.5: An image containing CME after applying the algorithm

Bibliography

1. Timothy Howard: Coronal Mass Ejections

2. Coronal Mass Ejections: Observations

3. `http://www.swpc.noaa.gov/phenomena/coronal-mass-ejections`

4. `https://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm`

5. `http://www.robots.ox.ac.uk/~sjrob/Teaching/SP/l7.pdf`

6. `http://www.slack.net/~ant/bl-synth/3.nyquist.html`

7. Inbound waves in the solar corona: a direct indicator of Alfven surface location

8. `https://cor1.gsfc.nasa.gov/catalog/cme/2010/HongXie_COR1_preliminary_event_list_2010-08.html`

9. Background Subtraction for the SECCHI/COR1 Telescope Aboard STEREO

10. The SolarSoft WCS Routines: A Tutorial

11. Automated Detection of Coronal Mass Ejections in STEREO Heliospheric Imager data

12. Structure of the Solar Dust Corona and its Interaction with the other Coronal Components

13. `https://github.com/kjanyani/NIUS-14.2`
    Repository containing the code used

14. `http://www.thefouriertransform.com/`
    A useful site to understand Fourier transforms