# AUTOMATED DETECTION OF CMES

**D. Berghmans**[1]

[1]Royal Observatory of Belgium, Ringlaan - 3 - Avenue Circulaire, B-1180 Brussels, Belgium

## ABSTRACT

We have developed a software package for 'Computer Aided CME Tracking' (CACTus), that autonomously detects CMEs in image sequences from LASCO. The crux of the CACTus software is the detection of CMEs as bright ridges in [height, time] maps using the Hough transform. The output is a list of events, similar to the classic catalogs, with principle angle, angular width and velocity estimation for each CME. In contrast to catalogs assembled by human operators, these CME detections by software can be faster and possibly also more objective, as the detection criterion is written explicitly in a program. In this paper we discuss two applications of the software. In a first application, we validate the performance by comparing the CACTus output with the classical, visually assembled CME catalogs. We discuss its present success rate (about 75 %) and prospects for improvement. By giving an example we show that the software can also reveal CMEs that have been not been listed in the catalogs. Such unreported cases might be of influence on CME statistics and prove that also the present catalogs do not have a 100 % success rate. The second application, is an operational version of CACTus that scans in real time, the last available LASCO images for recent CMEs.

Key words: Sun: corona, Sun: particle emission.

## 1. INTRODUCTION

For more than 6 years now, coronal mass ejections (CMEs) have been detected routinely by visually checking each image from the Large Angle Spectrometric Coronagraph (LASCO, Brueckner et al. (1995)). Event catalogs have been assembled continuously and are made publicly available (http://lasco-www.nrl.navy.mil/cmelist.html, http://cdaw.gsfc.nasa.gov/CME_list). Each observed CME is listed with its time of appearance in the inner C2 coronagraph, the angle of its central axis, its angular span, velocity, acceleration estimations and a short morphological description. These catalogs are used as a reference and form a valuable resource for further statistical analysis on the nature of CMEs (St. Cyr et al. (2000); Vourlidas et al. (2002)).

The visual detection of CMEs in the flood of incoming LASCO data is a labor intensive task. It is up till now essentially the human eye that detects a CME occurrence and a scientist that collects all the CME parameters in the catalogs. From 2005 onwards, new missions will bring several more coronagraphs in space. The SECCHI instrument suite on each of the 2 STEREO spacecrafts, as well as the SHARPP instrument suite on the SDO mission, contain coronagraphs that will boost the data rate by orders of magnitude. Continuing visual detection of CMEs will become a big investment of man power.

Meanwhile near real time alerts for halo CMEs are needed by the space weather community (Clette et al., 2002). Although in typical cases such halo CMEs take a few days to arrive to the Earth, their detection is timely as CPU-time intensive 3D MHD simulation are required to estimate their geoeffectiveness. This implies that CME halo alerts should be issued 24 hours per day.

Finally, the subjective interpretation by a human operator or scientist makes it doubtful whether this visual CME detection is stable over a solar cycle, as the operator gains experience or personnel is replaced. There is probably not much confusion for big, well structured events but small and/or weak events might be arbitrarily detected or not. This definetly influences conclusions based on statistics of large number of CMEs.

For all these reasons, we have developed the software package CACTus ('Computer Aided CME Tracking') that detects CMEs in coronagraphic images. In this paper we present the inner machinery of this package. A first preprocessing module (section 2) merges the C2 and C3 images, cleans, rebins and reformats them with every step optimized for improving the CME contrast. The second image-recognition module (section 3) then extracts motion patterns and groups the patterns in CMEs. In section 4, we estimate the performance of the package. Section 5 briefly discusses real time operations and we conclude in section 6 with some ideas for further improvement.

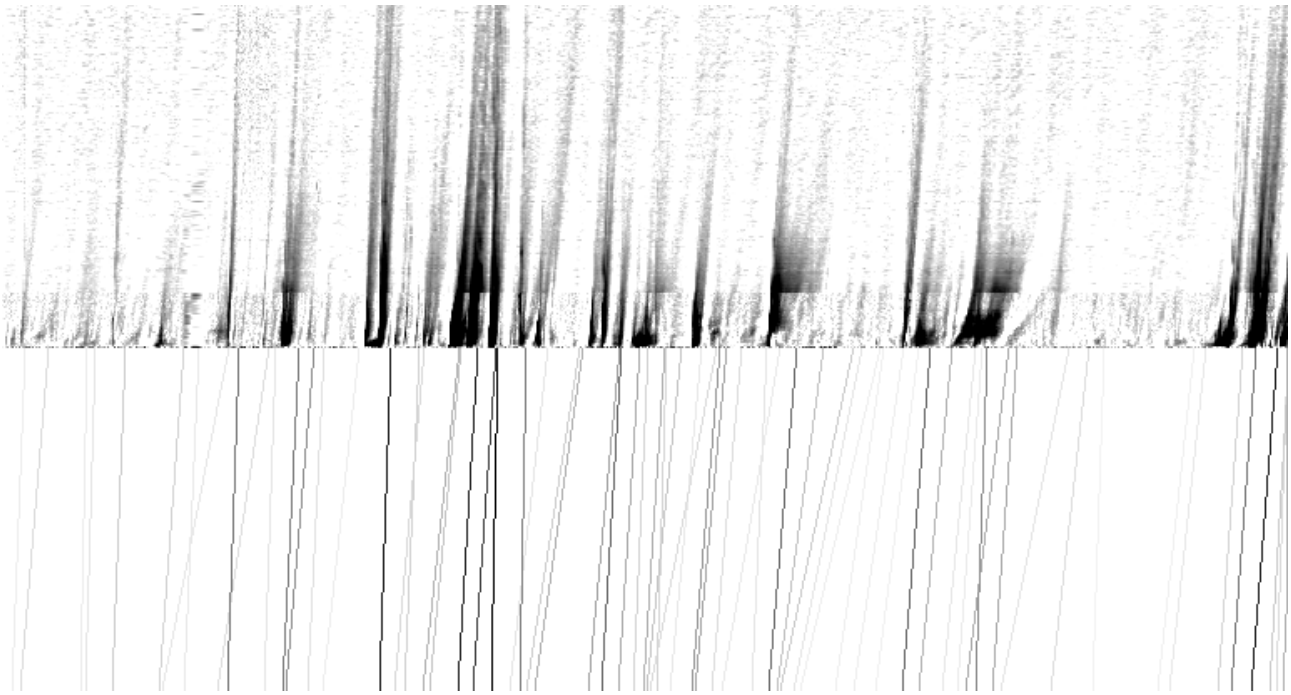

*Figure 1. Example of a [time,height] slice through the datacube (top) and the ridges detected in it with the Hough transform (bottom). The inclination angle of the ridges corresponds to the propagation velocity. The horizontal range corresponds to the month May 1998. The vertical range corresponds to the combined C2/C3 field of view.*

## 2. THE PREPROCESSING MODULE

The default processing of the LASCO images (see eg latest images on http://sohowww.nascom.nasa.gov) is not optimized for CME detection. The CME signature is convolved with quasi-static K-corona streamer structures and with slowly moving stars, planets and comets as well as the instrumental stray light and F-corona backgrounds. Also, towards the edge of the field-of-view (FOV), the noise levels increase sharply because the CME contrast diminishes, and this makes it difficult to follow outmoving features. The images are relatively large with a spatial resolution far beyond what is needed for CME detection. A typical CME is only a relatively weak variation in intensity and only visible in a few subsequent images.

All this means that the signal which we want to trigger on, is only very scarcely present in the huge amount of incoming data. Straight application of image recognition techniques on the usual 1024x1024 images would therefore result in a giant computational overhead. Finally the different spatial and temporal resolution of C2 and C3 data make a combined analysis difficult. To avoid all these complications, a preprocessing module is applied that reformats the input images:

- Each "level 0.5" image from LASCO/C2 and C3 is read in. Exposure time normalization is applied and bright point like sources (cosmic ray hits, but also planets and stars) are removed.
- A polar transformation is made of each image: the $[x, y]$ FOV becomes a $[r, \theta]$ FOV, with $\theta$ the poloidal angle around the Sun and $r$ the distance from the limb. By choosing the $r$-range appropriately, the dark occulter and corner regions are easily avoided. While transforming we also rebin, from 1024x1024 pixels for the $[x, y]$ FOV to a 200x360 pixels $[r, \theta]$ FOV. This increases the signal to noise ratio significantly, especially far away from the disc, as the size $r\Delta\theta\Delta r$ of the 'footprint' of a $[r, \theta]$ pixel in $[x, y]$ images grows as $r$.
- The $[r, \theta]$ images originating from C2 and C3 are combined in a single composite image by rescaling/matching the different spatial and temporal resolution of the two coronagraphs. Since the LASCO C2 FOV is much smaller than that of C3, this step essentially comes down to adding a small C2 strip at the bottom of the $[r, \theta]$ C3 images. To take into account the different observation times of the C2 and C3 images, a cubic spline interpolation is applied so that C2 images are matched to the default 1 hour cadence of C3.
- Stacking the resulting composite $[r, \theta]$ images in a $[r, \theta, t]$ datacube, we calculate a 'background' as a running average over 1 day. For each $[r, \theta]$ pixel a CME passage results in a short lived positive deviation from the running average. In an iterative, pixelwise procedure, such bright short lived deviations are identified and removed for the next iteration of the background calculation. After a few iterations, the resulting background effectively contains only the variability on a timescales larger than 1 day. In what follows, we will only consider relative deviations

from the original $[r, \theta, t]$ datacube with respect to this background. This effectively removes the dust corona but also streamers that rotate into and out of the FOV.

Note that the default background used for LASCO images has the same function, but it only removes variability on timescales larger than 1 month. Our technique has the advantage that slow rotation and evolution of streamers is additionally removed. We have also avoided differencing images, because the 'black and white' confusion in difference images impedes to detect the full spatial extent of a CME. The output of all this is a $[r, \theta, t]$ datacube which is much smaller than the total of the original input data, and in which most of the non-CME signal is removed or strongly attenuated.

## 3. THE IMAGE RECOGNITION MODULE

CMEs are seen as bright features moving outward from the Sun. It turned out not to be feasible to identify in each separate image the location and extension of individual CME by segmentation techniques. The CMEs are too variable in appearance, they are often too weak to identify their extension (especially their trailing edge), and they might erroneously be merged with one another.

Instead of trying to detect CMEs in each $[r, \theta]$ image, we looked at $[t, r]$ slices (Figure 1, top) for each $\theta$ in the $[r, \theta, t]$ datacube. If a $[t, r]$ slice at an angle $\theta$ cuts through a CME, an inclined ridge is seen in the $[t, r]$ slice. Detecting CMEs in $[t, r]$ slice was first introduced by Sheeley et al. (1999). Thanks to the preprocessing module however, our $[t, r]$ slices, and the CME ridges in them, have a much better contrast and contain less noise. Working with $[t, r]$ slice has the advantage that all CMEs look the same (inclined ridges) and that even weak CMEs show up with a clear signature. Finally, as a bonus, the propagation speed of the CME can be determined from the inclination angle of the CME.

A well known technique for detecting ridges or straight lines is the so-called Hough transform (Jähne, 1997). Given an image I(x,y), the Hough transform $I(a, b)$ is constructed by evaluating the integral $I(a, b) = \int I(x, ax + b)dx$. If a bright line $y = a_i x + b_i$ is present in the image $I(x, y)$, then the Hough transformed image $I(a, b)$ will have a local maximum at $[a = a_i, b = b_i]$. Detecting ridges in the image $I(x, y)$ thus comes down to thresholding peaks in the transformed image $I(a, b)$.

We make an Hough transformation of every $[t, r]$ slice. In transformed space, we filter the most significant signals, which after inversion correspond to the required ridges (Figure 1, bottom). Each ridge $R$ in an $[t, r]$ slice at an angle $\theta_R$ is characterized by its onset time $t_R$, its velocity $v_R$ and its brightness $b_R$. We can now build up a datacube $[v, \theta, t]$ by setting for each ridge $[v = t_R, \theta = \theta_R, t = t_R] = b_R$.

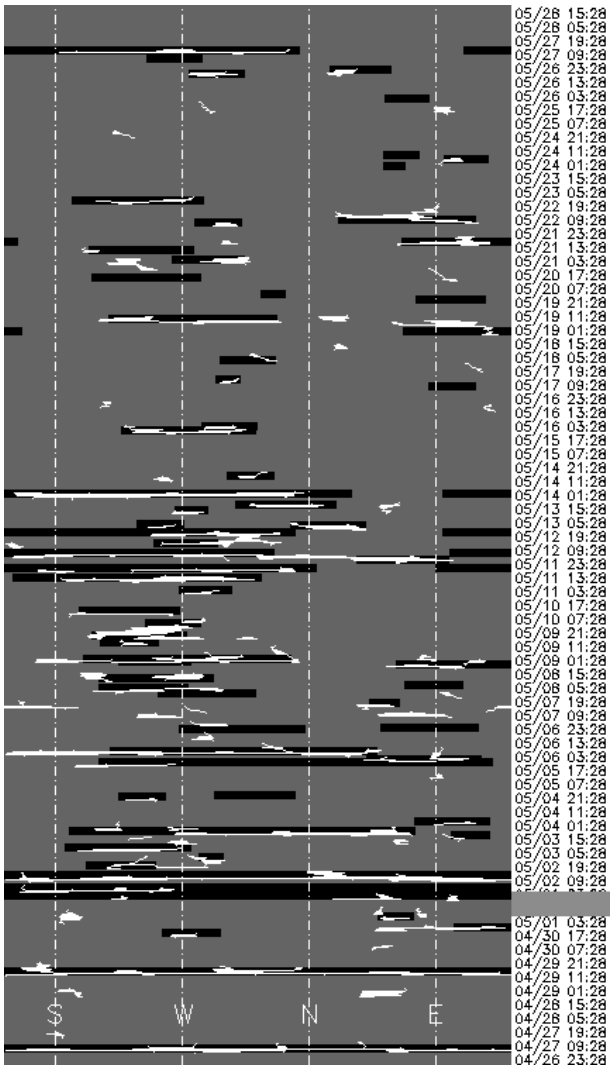


*Figure 2. Comparison of the catalog CMEs (dark) and those found by CACTUS (white). Time runs vertically over most of May, 1998. The poloidal angle runs counterclockwise from left (near C3 pylon) to right. The catalog time of appearance corresponds to the bottom of the dark boxes. The 'thickness' of the boxes is arbitrarily set to 8 hours.*

Since a CME is a large scale structure, the onset time and velocity will only slightly differ from angle to angle. This means that a CME is represented in the $[v, \theta, t]$ datacube as a dense cluster of datapoints. The problem of detecting CMEs has thus been reduced to identifying clusters in a 3d scatter plot.

For the time being, we simply integrate the $[v, \theta, t]$ cube along the $v$-direction and identify the location of clusters in the resulting $[\theta, t]$ map as the time of occurrence and angular span of CMEs. The result of this is shown in Figure 2. More performant identification of the clusters in the $[v, \theta, t]$ cube will be the subject of future upgrades.

## 4. VALIDATION OF THE CME DETECTION PACKAGE

We applied to above scheme to the LASCO data from April, 27, 1998 to May, 27, 1998. This month is among the latest months for which a 'final CME catalog' exists (http://lasco-www.nrl.navy.mil/cmelist.html). Each CME is listed with the time of appearance in the inner C2 coronagraph, the angle of the central axis of the CME, the angular span, a velocity and acceleration estimation and a short morphological description. For the period mentioned, the catalog lists 71 CMEs of which 4 were halo CMEs. In Figure 2 we show the angular span and time of occurrence of these catalog CMEs as dark boxes.

The CACTus software found 95 events. They are shown as white elongated regions in Figure 2. The overal distribution in (angle, time) space is very similar and the number of CMEs found is of the same order of magnitude. Comparing the two sets in more detail is a delicate exercise. The 'success rate' of our software obviously depends on the tolerance allowed on the deviations.

Of the 71 catalog CMEs, there are 19 CMEs (27 %) that are reproduced with nearly identical time of appearance and angular location. Allowing for a reasonable tolerance on the time of appearance (at least within 3 hours) and on the angular span (at least 50 % of the catalog span), the number of reproduced CMEs increases to 53 (75 %). In this 'success rate' we have also included those cases were the CACTus software 'merged' events that were listed as separate CMEs. At the other hand, about 10 catalog CMEs (14 %) are completely missed by CACTus. The remaining 11 % are marginal/disputable detections eg when a catalog CME seems to be detected but the detected time of appearance deviates more than three hours.

For space weather applications it is important to note that out of the 4 catalog halo CMEs, 2 are indeed reproduced as halo CMEs. The remaining 2 are missed because of a data gap (see grey zone in time bar of Fig. 2).

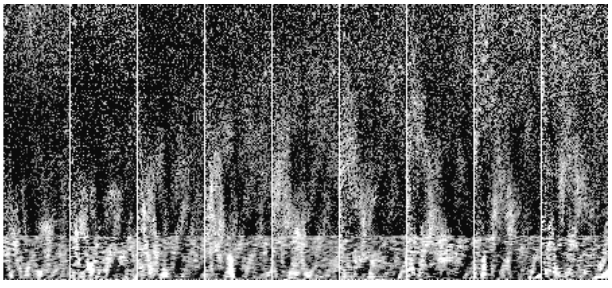

*Figure 3. Example of an unreported CME, lifting off at 19h28 on April 27, 1998 with a speed of about 200 km/s. The subfield shown is 50 degrees wide, centered around the South direction. There is a 3 hours lapse between the different subfields.*

The CACTUS software found 95 events whereas there are only 71 CME entries in the catalog. Part of the difference between the two numbers can be explained by the cases in which different parts of a CME are erroneously detected as separate events. Yet, we found about 15 (21 %) detected events which are 'far from' any catalog CME. Some of these are due to false alerts generated by fast streamer evolution. In at least some cases, our software has found 'unreported CMEs'. An example of such a case is shown in Figure 3. This means that also the catalogs do not have a 100 % success rate.

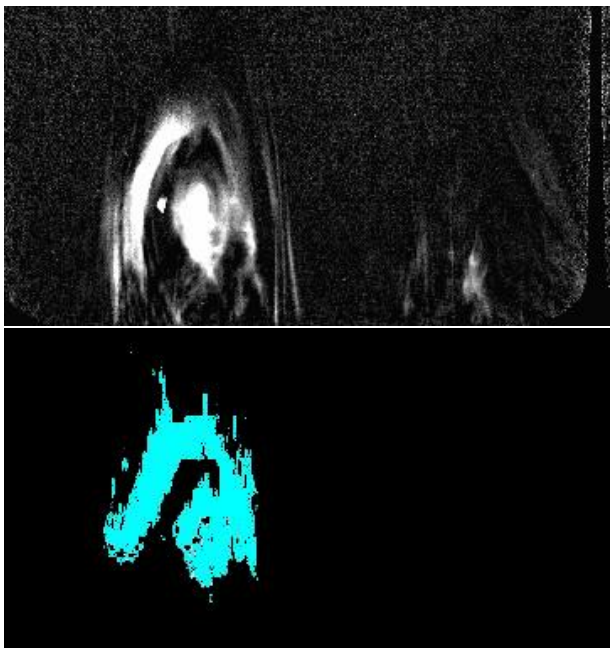

*Figure 4. $[r, \theta]$ image showing a CME on 2002/09/01, 14h30 UT (top panel). The bottom panel shows the same CME as detected by CACTus in near real time.*

## 5. REAL-TIME CACTUS

Besides the above described application of CACTus to a test-case dataset (May 98), we are also developing a real time version of CACTus. Triggered by a UNIX/crontab script, this version automatically fetches the latest quicklook fits images from the LASCO data processing pipeline at Goddard. The preprocessing module and image-recognition module are then applied as soon as a new LASCO image is received. The output is a list of CMEs that occurred in the last week. The performance and development of this realtime CACTus can be followed online at http://homepage.oma.be/david/cactus. At the time of this paper, the real-time version of CACTus only uses C3 images, but this will be extended to C2 in the near future.

## 6. CONCLUSIONS

This paper shows that it is possible to develop software that fully automatically detects CMEs in coronagraphic image sequences, estimates its main characteristics and produces CME catalogs just as currently human operators do it. The current version of our program recovers about 75 % of the catalog CMEs. Several improvements in the preprocessing module can still be envisioned and there is hope to reach a 85 % success rate. The remaining 15 % are due to weak, marginal cases but also to cases of intelligent action/guessing by the human operator eg with data gaps or with partially corrupted images. It should be noted also that for some cases, the software did detect weak CMEs which were missing in the catalog, so also the human operators do not have a 100 % success rate.

## ACKNOWLEDGEMENTS

This work started while D.B. was a research fellow at ESA/ESTEC. It is a pleasure to acknowledge discussions with B. Fleck and C. St. Cyr. This work is part of an ESA/PRODEX and a Belgian OSTC project on space weather and image recognition software. SOHO is a project of international collaboration between ESA and NASA.

## REFERENCES

Brueckner G.E., Howard R.A., Koomen M.J., et al., 1995, Sol. Phys., 162, 357

Clette F., Van der Linden R., Cugnon P., et al., 2002, ESA SP, this conference

Jähne B., 1997, Digital Image Processing, 463, Springer-Verlag

Sheeley N.R., Walters J.H., Wang Y.M., Howard R.A., Nov. 1999, J. Geophys. Res., 104, 24739

St. Cyr O.C., Howard R.A., Sheeley N.R., et al., Aug. 2000, J. Geophys. Res., 105, 18169

Vourlidas A., Buzasi D., Howard R.A., 2002, ESA SP, this conference