Preprint typeset using LaTeX style openjournal v. 09/06/15

CORONAL MASS EJECTION IMAGE EDGE DETECTION IN HELIOSPHERIC IMAGER STEREO SECCHI DATA

Marc D. Nichitiu
The Stony Brook School, NY, USA
submitted February 2022

ABSTRACT

We present an algorithm to detect the outer edges of Coronal Mass Ejection (CME) events as seen in differences of Heliospheric Imager STEREO SECCHI HI-1 images from either A or B spacecraft, as well as its implementation in Python.

1. INTRODUCTION

Studying and monitoring Sun's activity, and in particular, its stronger radiation and particle emission abrupt events, is important for both science and practical purpose, for our life on and around Earth. In particular, Coronal Mass Ejections (CME), which energetically carry away accelerated charged particles, can create electromagnetic storms on our planet, and also endanger astronauts or space equipment. Increased interest in studying their origins, structure, and trajectories resulted in more and more international space missions, among which several ones are still operational. The STEREO pair A and B of NASA spacecraft is one of such latest missions, continuously imaging the Sun and the space between Sun and Earth, with several instruments, as described in Eyles et al. (2009); Eyles et al. (2007) by orbiting around the Sun at roughly the same distance as the Earth[1]. Two visible-light cameras HI-1 and HI-2 on each spacecraft are thus precisely aimed in order to cover the line Sun-Earth, while being shielded from the Sun itself. Their images show various parts of Sun's corona, receiving free-electron scattered light from the K-corona, dust scattered light from the F-corona, as well as light from background stars. Many studies of these images start from running differences, as described by Sheeley et al. (1997) for coronagraph images, to enhance the faint K-corona light and remove the slowly-varying but much stronger F-corona light, as also explained in Davies et al. (2009). CME perturb the K-corona, and thus appear as bright regions in these images. Successive images, obtained from space missions such as the SOHO LASCO Yashiro et al. (2004), and STEREO coronagraph and HI cameras, show their propagation and evolution. Considerable research studying the CME origin mechanisms, shape, and trajectories is in progress, but open problems remain, as described in the very comprehensive review from Webb & Howard (2012), in particular regarding propelling forces and interaction with solar wind: such models can be found in Xie et al. (2006); Michalek et al. (2006); Gopalswamy & Yashiro (2007); Howard et al. (2008). Given the large amount of data as well as the need for tracking, computerized pre-analysis tools can provide important help, in particular to identify the CME boundaries and kinematics.

Generally speaking, many computer vision techniques for edge detection have been devised, such as the ones based on gradient, e.g the Canny algorithm Canny (1986), as well as the family of active contours Caselles et al. (1993); Chan & Vese (2001); Márquez-Neila et al. (2014). However, as Young & Gallagher (2008) explain, it is not straightforward to use existing techniques because of the diffuse nature of CMEs. Moreover, they also have complex inner features, and their images appear next to light from many background stars, which adds to the noise.

Thus, several complex algorithmic methods specialized for this task have been devised: CACTus Robbrecht & Berghmans (2004), SEEDS Olmedo et al. (2008), and multiscale wavelet analysis Young & Gallagher (2008); Byrne et al. (2009), mainly focused on the coronagraph images. These methods isolate the regions of interest in specific ways, and then estimate geometric and kinematic parameters, using underlying assumptions such as constant acceleration and elliptical shapes.

The algorithm presented here only focuses on the image segmentation stage, identifying the outer edge of CME regions in STEREO HI images using a staged approach composed of simple steps. The algorithm starts by filtering and smoothing the image difference, subsequently exploring it from the center of the bright CME region, marking its outer edges.

Through subsequent work, it can be integrated into various kinematic estimation schemes. Given its modular nature, the algorithm itself could also be further adapted to examine specific internal features apparent in the images.

The rest of the paper is organized as follows. In Section 2 we describe the algorithm outline and its subsequent processing of an image. In Section 3 we show its results on several other STEREO HI-1 images of known CME, and in Section 4 we discuss future lines of work.

2. ALGORITHM

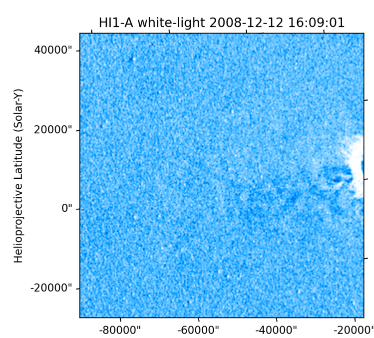

Figure 1. Background-removed 0-to-255 normalized diff image from STEREO HI-1 Spacecraft A, used as input

[1] Currently only STEREO-A is operational, but several years of raw and processed image data are available from both.

In Figure 1 we see an example of what the algorithm starts with, which is an early phase of a CME visible in the differenced image from December 12, 2008. The Sun is behind the right edge of the image, which has $n_c \times n_r = 1024 \times 1024$ pixels.

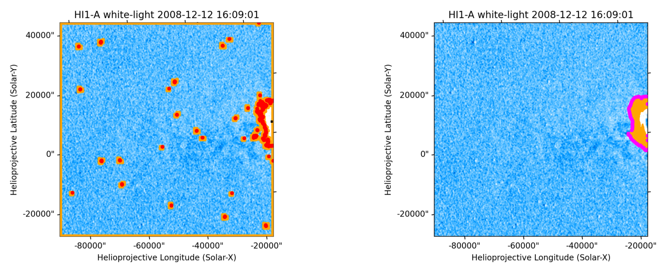

Figure 2. Intermediate processing stages

The main steps, graphically illustrated in Figure 2, are as follows, where $I[c, r]$ represents the pixel intensity for coordinates column c and row r of the input image. The origin is the bottom left of the image.

1. Find the center of mass (x_S, y_S) of the bright area and set $C = (c_S, y_S)$ where c_S is close to the image edge c_0 towards the Sun (right for spacecraft A, so typically $c_0 = 1023$, and left for B, i.e. $c_0 = 0$).

2. Going from c_S to c_E (1023 for B, 0 for A), on each image row r, mark the farthest bright pixels seen along. Then do the same on each image column c, first from y_S up, and then from y_S down, with $t_1 = 240$ for 0-255 gray level images.

```
e1 ← []
for each image row r do
    q ←False
    v ← c0
    for each image column c from cS to cE do
        if I[c, r] > t1 then
            v ← c
            q ←True
        end if
    end for
    if q then
        e1+ ← [v, r]
    end if
end for
for each image column c do
    for each image row r from yS to rE do
        if I[c, r] > t1 then
            v ← r
            q ←True
        end if
    end for
    if q then
        e1+ ← [c, v]
    end if
    for each image row r from yS to 0 do
        if I[c, r] > t1 then
            v ← r
            q ←True
        end if
    end for
    if q then
        e1+ ← [c, v]
    end if
end for
```

3. Smooth the image I into I_M, using a sum-scale-and-modulo-256 5x5 filter to enhance contrast at the edges

4. Mark the contrasting pixels created in step 3, with $t_2 = 120$ for 0-255 gray level images

```
e2 ← []
for each image row r do
    for each image column c except the last one do
        if |IM[c, r] − IM[c + 1, r]| > t2 then
            e2+ ← [c, r]
        end if
    end for
end for
for each image column c do
    for each image row r except the last one do
        if |IM[c, r] − IM[c, r + 1]| > t2 then
            e2+ ← [c, r]
        end if
    end for
end for
```

5. Mark the centers where both kinds of previous marks are present in a $w_i \times w_i$ window, as created in step 2 and step 4, namely from the e_1 and e_2 lists:

```
I1 = ones[nc, nr]
I2 = ones[nc, nr]
for each element [x, y] of e1 do
    I1[x, y] ← 2
end for
for each element [x, y] of e2 do
    I2[x, y] ← 3
end for
I3 ← I1 + I2
I4 ← filter(I3, prod, wi)
e3 ← list of [x, y] such that I4[x, y] is a multiple of
    2 and 3, or of 5
```

6. Estimate the density of the markers from step 5 in a larger window $w_d \times w_d$ and mark those above half into a boolean mask I_D.

7. Circularly sweeping from the center, for each such radial half-line gather the last mark as created in step 6, i.e. from I_D, and build the perimeter:

```
e4 ← []
for each angle α from 0 to 2π in steps of ε do
    v ←Nothing
    for each pixel [x, y] along the α ray from
    [cS, yS] to any edge of the image do
        if ID[x, y] then
            v ← [x, y]
        end if
    end for
    if v then
        e4+ ← v
    end if
end for
```

8. Remove spikes (outliers) created in step 7 in the list e_4 of pairs of pixel coordinates, reconnect and validate the final perimeter, based on CME geometry from Fisher (1984); Fisher & Munro (1984); Crifo (1983).

The idea behind step 3 is to augment existing gradual contrasting zones by creating an abrupt transition through the modulo operation.

Given the larger density of markers around the true edge, even if the rays in step 7 are one or two pixels wide, they should not miss them when they cross it outwards.

Outliers appear occasionally because of brighter background light points far away from the CM, and can be removed in step 8. The validation of the perimeter, to eliminate unphysical results, is based on CME general geometry, using bubble and loop models as described in Crifo (1983). Specifically, as Fisher (1984) and Fisher (1984) point out, the width to depth ratio is typically 3:2, so we compute and compare these diameters.

3. RESULTS

We have tracked CME events using a set of 100 STEREO 11-day background-removed L2 HI-1 a and b images from the UK Solar System Data Center, `https://www.ukssdc.ac.uk`. The algorithm performed with overall high predictive abilities and without any fine-tuning or training set. We also conducted a manual identification of the CME regions, and then compared it with the automatic one. We counted the relative differences in areas between the two perimeters as errors, and we measured by hand that the percentage error of region identifications was below 15% for the most irregular events. We did not find false positives. The images in Figure 3 and Figure 5 illustrate these cases.

A simplified initial calculation appears in agreement with speeds of the order of 300km/s from Davis et al. (2009) for the 2008 December 12 CME event, as effectively measured by the SWEPAM instrument aboard the Advanced Composition Explorer spacecraft Stone et al. (1998). We used the transformation (4) described in Thompson (2006):

$$x \simeq D_{\odot}\left(\frac{\pi}{180^{\circ}}\right)\theta_x$$

to compute the displacement of the front, from the Helioprojective longitude displacement of $\theta_x = 0.27^{\circ} = 1000$ seconds of arc from 18:09 to 18:49 of two algorithm-generated contours for HI-1 B, shown in Figure 4. Thus, $\Delta x = 730000$km, and $\Delta t = 2400s$, so $v = 300$km/s, using only two significant digits.

Succesive images, such as in Figure 5 can allow more accurate estimations of kinematics.

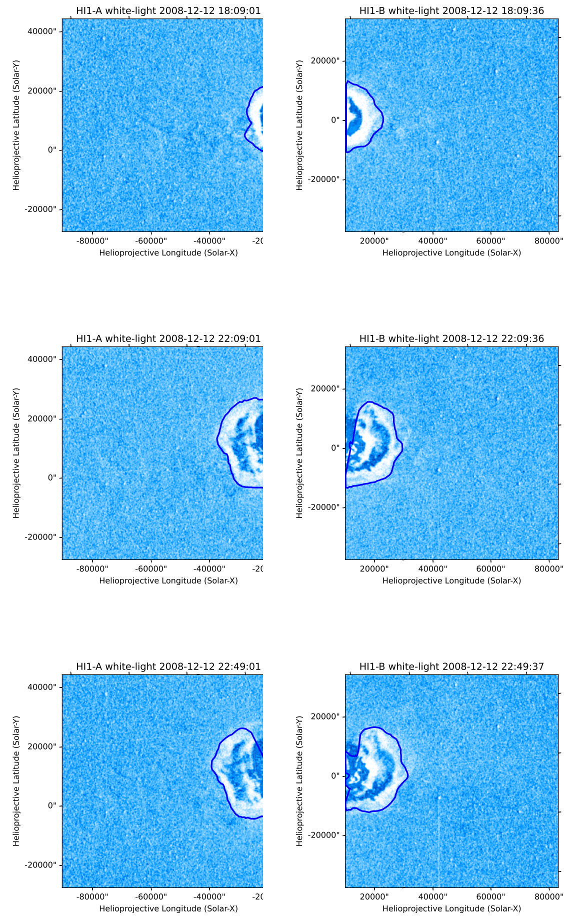

Figure 3. A few examples of processing

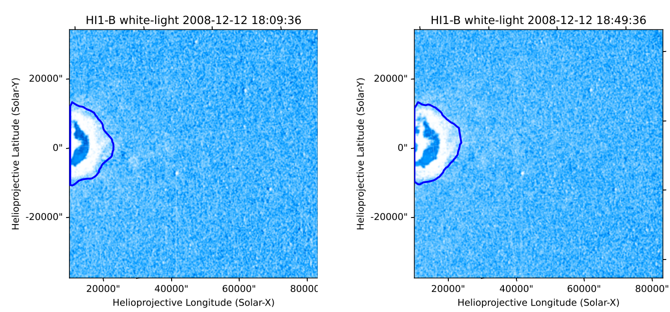

Figure 4. Two-image sequence to estimate front speed

4. DISCUSSION

To improve the smoothness and completeness of the perimeter detection, we have also tried a subsequent stage of the morphological geodesic active contour method from Márquez-Neila et al. (2014), based on level sets seeded with the interior of an initially-found contour. So far, it did not seem to succeed, mainly because of the of the various bright-dark alternating inner fronts, and also because of the background noise from the light of the stars. The latter could be further reduced by carefully realigning the images before subtraction, as suggested by Davies et al. (2009).

A possible line of improvement would be to better adapt the active contour methods to the already-found contour to fine-align it along less-contrasting borders. A more precise way to distinguish the far-away background could also help.

Finally, we are planning on working on integrating it with existing models to estimate CME velocities and 3-D shape changes, and then systematically comparing it with other methods.

5. ACKNOWLEDGEMENTS

We have implemented the algorithm in Python, and made it available at `https://github.com/MDNich/CME-Image-Edge-Detection-in-Python-from-STEREO-data`, and we are thus grateful to the community mainaining

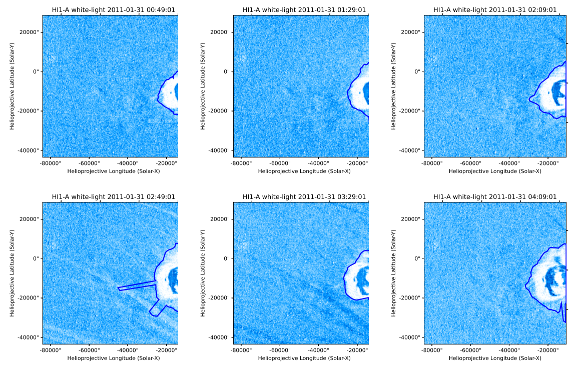

Figure 5. Longer sequence

Numpy, SciPy and Matplotlib, as well as The SunPy Community et al. (2020).

REFERENCES

Byrne, J. P., Gallagher, P. T., McAteer, R. T. J., & Young, C. A. 2009, A&A, 495, 325
Canny, J. 1986, IEEE Transactions on Pattern Analysis and Machine Intelligence, 8, 679
Caselles, V., Catté, F., Coll, T., & Dibos, F. 1993, Numer. Math., 66, 1
Chan, T. F. & Vese, L. A. 2001, IEEE Transactions on Image Processing, 10, 266
Crifo, F. 1983, Solar Physics, 83, 143
Davies, J. A., Harrison, R. A., Rouillard, A. P., et al. 2009, Geophys. Res. Lett., 36, L02102
Davis, C. J., Davies, J. A., Lockwood, M., et al. 2009, Geophys. Res. Lett., 36, L08102
Eyles, C., Davis, C., Harrison, R., et al. 2007, in Solar Physics and Space Weather Instrumentation II, ed. S. Fineschi & R. A. Viereck, Vol. 6689, International Society for Optics and Photonics (SPIE), 40 – 52
Eyles, C. J., Harrison, R. A., Davis, C. J., et al. 2009, Sol. Phys., 254, 387
Fisher, R. 1984, Adv. Space Res., 4, 163
Fisher, R. & Munro, H. R. 1984, Astrophys. J., 280, 163
Gopalswamy, N. & Yashiro, S. 2007, Journal of Geophysical Research, 112
Howard, T., Nandy, D., & Koepke, A. 2008, Journal of Geophysical Research, 113
Michalek, G., Gopalswamy, N., Lara, A., & Yashiro, S. 2006, Space Weather, 4, S10003
Márquez-Neila, P., Baumela, L., & Alvarez, L. 2014, IEEE Transactions on Pattern Analysis and Machine Intelligence, 36, 2
Olmedo, O., Zhang, J., Wechsler, H., Poland, A., & Borne, K. 2008, in AGU Spring Meeting Abstracts, Vol. 2008, SP43A–02
Robbrecht, E. & Berghmans, D. 2004, A&A, 425, 1097
Sheeley, N. R., Wang, Y. M., Hawley, S. H., et al. 1997, ApJ, 484, 472
Stone, E. C., Frandsen, A. M., Mewaldt, R. A., et al. 1998, Space Sci. Rev., 86, 1
The SunPy Community, Barnes, W. T., Bobra, M. G., et al. 2020, The Astrophysical Journal, 890, 68
Thompson, W. T. 2006, Astronomy and Astrophysics, 449, 791
Webb, D. F. & Howard, T. A. 2012, Living Reviews in Solar Physics, 9
Xie, H., Gopalswamy, N., Ofman, L., et al. 2006, Space Weather, 4, S10002
Yashiro, S., Gopalswamy, N., Michalek, G., et al. 2004, Journal of Geophysical Research (Space Physics), 109, A07105
Young, C. A. & Gallagher, P. T. 2008, Sol. Phys., 248, 457

This paper was built using the Open Journal of Astrophysics LaTeX template. The OJA is a journal which provides fast and easy peer review for new papers in the `astro-ph` section of the arXiv, making the reviewing process simpler for authors and referees alike. Learn more at `http://astro.theoj.org`.