

A Machine-Learning-Ready Dataset Prepared from the Solar and Heliospheric Observatory Mission

Carl Shneider^{1,*}, Andong Hu¹, Ajay K. Tiwari¹, Monica G. Bobra², Karl Battams⁵, Jannis Teunissen¹, and Enrico Camporeale^{3,4}

¹Center for Mathematics and Computer Science, Multiscale Dynamics, Amsterdam, 1098 XG, the Netherlands

²W.W. Hansen Experimental Physics Laboratory, Stanford University, Stanford, CA, 94305, USA

³CIRES, University of Colorado, Boulder, CO, 80309, USA

⁴NOAA, Space Weather Prediction Center, Boulder, CO, 80305, USA

⁵US Naval Research Laboratory, Washington DC, USA

*corresponding author(s): Carl Shneider (shneider.carl@gmail.com)

ABSTRACT

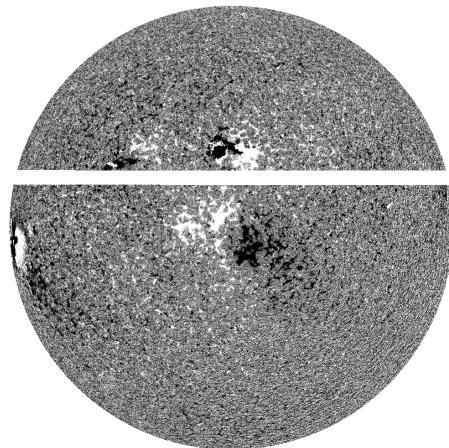
We present a Python tool to generate a standard dataset from solar images that allows for user-defined selection criteria and a range of pre-processing steps. Our Python tool works with all image products from both the Solar and Heliospheric Observatory (SoHO) and Solar Dynamics Observatory (SDO) missions. We discuss a dataset produced from the SoHO mission's multi-spectral images which is free of missing or corrupt data as well as planetary transits in coronagraph images, and is temporally synced making it ready for input to a machine learning system. Machine-learning-ready images are a valuable resource for the community because they can be used, for example, for forecasting space weather parameters. We illustrate the use of this data with a 3-5 day-ahead forecast of the north-south component of the interplanetary magnetic field (IMF) observed at Lagrange point one (L1). For this use case, we apply a deep convolutional neural network (CNN) to a subset of the full SoHO dataset and compare with baseline results from a Gaussian Naive Bayes classifier.

1 Background & Summary

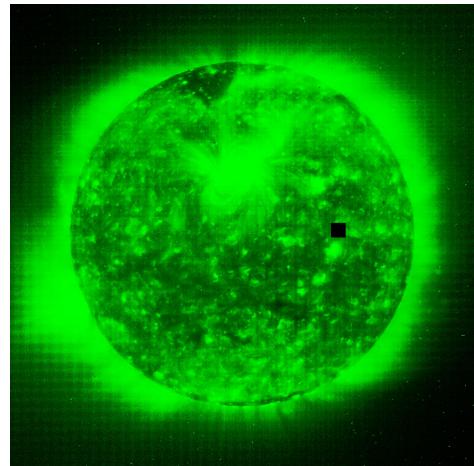
Studies based on physics models have shown that solar magnetic field captured with magnetograms contain crucial information for estimating the speed of the solar wind, while the dynamical features of CMEs (angular width, initial speed, etc.) are routinely inferred from coronagraph images^{1,2}. Hence, it is expected that applying machine learning (ML) techniques to the high temporal coverage data of both the Solar and Heliospheric Observatory³ (SoHO) mission and the Solar Dynamics Observatory⁴ (SDO) mission is a feasible venture, that can potentially improve the space weather forecasting capability of current models^{5–7}. With the quality of input data remaining paramount to the success of these ML-methods^{8–10} and to ensure reproducible scientific research^{11,12}, the preparation of a community-wide standard dataset with a standard software is crucial.

At present, SoHO has provided more temporal coverage of the Sun than its successor, NASA's SDO, and has also fully covered Solar Cycle 23 and 24 with a suite of on-board instruments³ including those specific to solar imaging: the Michelson Doppler Imager¹³ (MDI) for the solar photosphere, the Extreme ultraviolet Imaging Telescope¹⁴ (EIT) for the stellar atmosphere to low corona, and the Large Angle and Spectrometric Coronagraph¹⁵ (LASCO) covering the corona from $1.5 – 30 R_s$, detailed in Table 1. Recently, a white-light coronal brightness index (CBI), constructed from the full LASCO C2 mission archive, was used to explore correlations between the solar corona and several geophysical indices¹⁶. Although SDO has even higher resolution and cadence, SoHO continues to uniquely provide coronagraph products and serves as a mission critical backup to the SDO for solar flare and CME forecasting in the event of SDO failure.

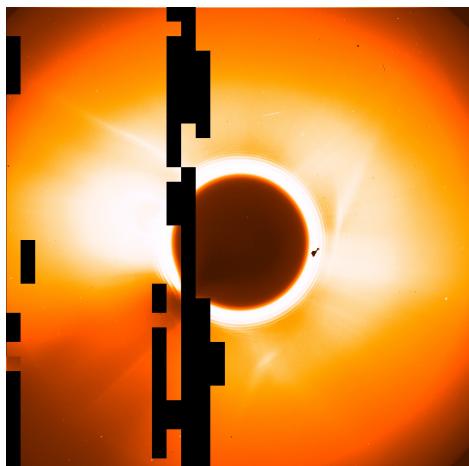
Stanford University's Joint Science Operation Center (JSOC) stores data from SoHO MDI, SDO HMI and AIA, and various other solar instruments. The SunPy-affiliated package DRMS enables querying of these images^{17,18}. All of these individual image products from JSOC are at the same processing level and are supplied in a Flexible Image Transport System (FITS) format that contains only scalar values. The NASA Solar Data Analysis Center's (SDAC) Virtual Solar Observatory¹⁹ (VSO) tool enables data queries from a number of individual data providers. SDAC's terabytes of available EIT and LASCO images are also in FITS format. However, the SDAC data is highly heterogeneous. Not only are there intrinsic differences among these SoHO products (e.g., individual cadence as shown in Table 1), but there is also an irregular assortment of image file sizes and processing levels. These varying file sizes can correspond to different image resolutions, calibrations and orbital maneuvers, and multi-frame recordings. In addition, all four of the publicly available EIT products require calibration to



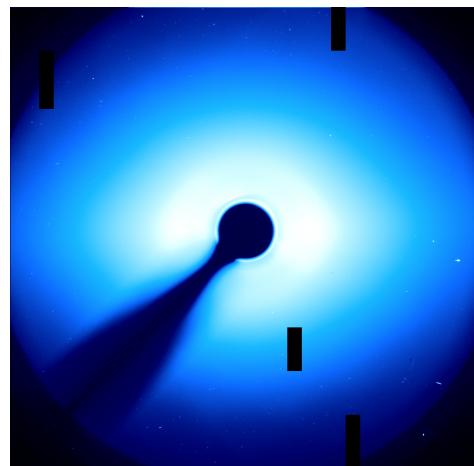
(a) MDI



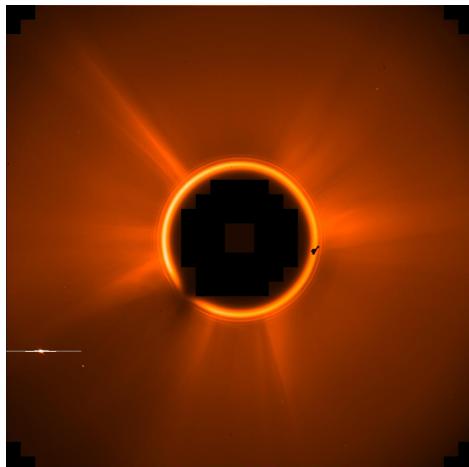
(b) EIT 195



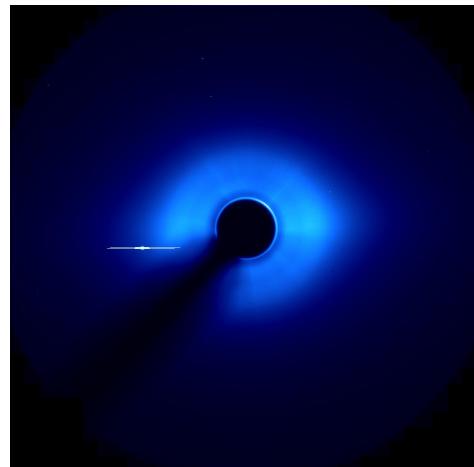
(c) LASCO C2



(d) LASCO C3



(e) Mercury Transit in LASCO C2



(f) Mercury Transit in LASCO C3

Figure 1. Examples of missing pixel data in the various SoHO products (a-d) as well as planetary transits in the LASCO coronagraphs (e-f). Telemetry errors take the form of large missing portions of the field of view, single or multiple image ‘holes’, and ‘stripes’. Colors similar to those used on the NASA SoHO site have been used.

Instrument	Detector	Observation	Observed Region	$\lambda(\text{\AA})$	Cadence (min)	Date Range
MDI	MDI	LOS Mag. Fld.	Full Disk	6768 (Ni I)	~ 96	1996.05.01 - 2011.04.12
EIT	EIT	Intensity	Full Disk	171 (Fe IX/X)	~ 360	1996.01.01 →
EIT	EIT	Intensity	Full Disk	195 (Fe XII)	~ 12	1996.01.01 →
EIT	EIT	Intensity	Full Disk	284 (Fe XV)	~ 360	1996.01.01 →
EIT	EIT	Intensity	Full Disk	304 (He II)	~ 360	1996.01.01 →
LASCO	C2	Intensity	Corona ($1.5 - 6 R_s$)	Visible	~ 20	1995.12.08 →
LASCO	C3	Intensity	Corona ($3.5 - 30 R_s$)	Visible	~ 20	1995.12.08 →

Table 1. Suite of SoHO Instruments utilized. LOS Mag. Fld. denotes the line-of-sight magnetic field, $\lambda(\text{\AA})$ is wavelength measured in angstroms, and R_s is the Sun's radius. LASCO C1 ($1.1 - 3 R_s$) is not included in this work since it was only operational till Aug. 9, 2000.

account for instrument degradation and the presence of the burn-in caused by continuous exposure to the sun. SunPy's^{18,20} Federated Internet Data Obtainer (Fido) interface with the VSO helps to reduce the search space by enabling a query based on a priori knowledge of the appropriate file sizes but is currently limited in its time sampling capabilities. In addition, a few percent of the total data for each of these mission products contains artefacts, illustrated in Fig. 1 (a-d). Besides holes, bright lines caused by comet transits, speckle patterns arising from cosmic rays impinging on the cameras, and spurious image artefacts, all contaminate the two coronagraphs' field of view. However, the dominant contribution, by far, to image contamination comes from the regularity of planetary transits that all leave a bright horizontal signature on the LASCO image products, an example of which is shown in Fig. 1 (e-f). Hence, preparing an ML-ready dataset from solar images, that is a dataset that can be ingested by a machine learning algorithm, remains challenging, time-consuming, and error-prone.

In the event that a sufficiently complex machine learning algorithm would be able to extract sufficient information from the images, either to improve the state-of-the-art forecasting, or to deepen our understanding of the underlying physical processes, this would constitute an invaluable resource for the community. On the other hand, a null result would also be an important step that would highlight the present limitations of solar images for space weather prediction.

2 Methods

In this section we provide an overview of the two-step process to the Python tool (<https://github.com/cshneider/soho-ml-data-ready>) for producing a machine-learning-ready community standard dataset from solar images.

2.1 Data preparation

The data preparation pipeline starts with a data generation step. This is illustrated in the leftmost flowchart of Fig. 2.

Step 1 (data generation):

1. User inputs:

- The user selects the SoHO or SDO mission followed by a set of mission products,
- Appropriate mission start and end dates are required,
- A user-defined time window specifies the time sampling rate of the data in hours,
- In order to be able to fit all of the data on a several GB GPU to speed up machine learning experiments, it is necessary to down-sample the native resolution of the images (i.e., (1024 x 1024) pixels for SoHO and (4096 x 4096) pixels for SDO) to a user-defined output image size. Since the information content of the magnetogram, EUV/UV/Visible, and coronagraph data products is fundamentally different, four different resize strategies are provided per code execution. For instance, one can perform three separate code runs with min-pooling on MDI, max-pooling on EIT, and sub-sampling for LASCO. In the event that the user wants to retain SoHO products that contain missing data that meets an acceptable threshold, an interpolation option is also provided for down-sampling. The definitions of these operations are as follows: i. sub-sampling selects pixels in every image row and column that are separated by a step size given by a scale factor. This scale factor is equal to the ratio of the original image's axis length to the desired image output size (e.g., 1024/128). ii. the interpolation option is a linear interpolation that shrinks the image by the scale factor, has spatial anti-aliasing which minimizes distortion artefacts known as aliasing that occur when a high-resolution image is represented at a lower resolution, and preserves the original

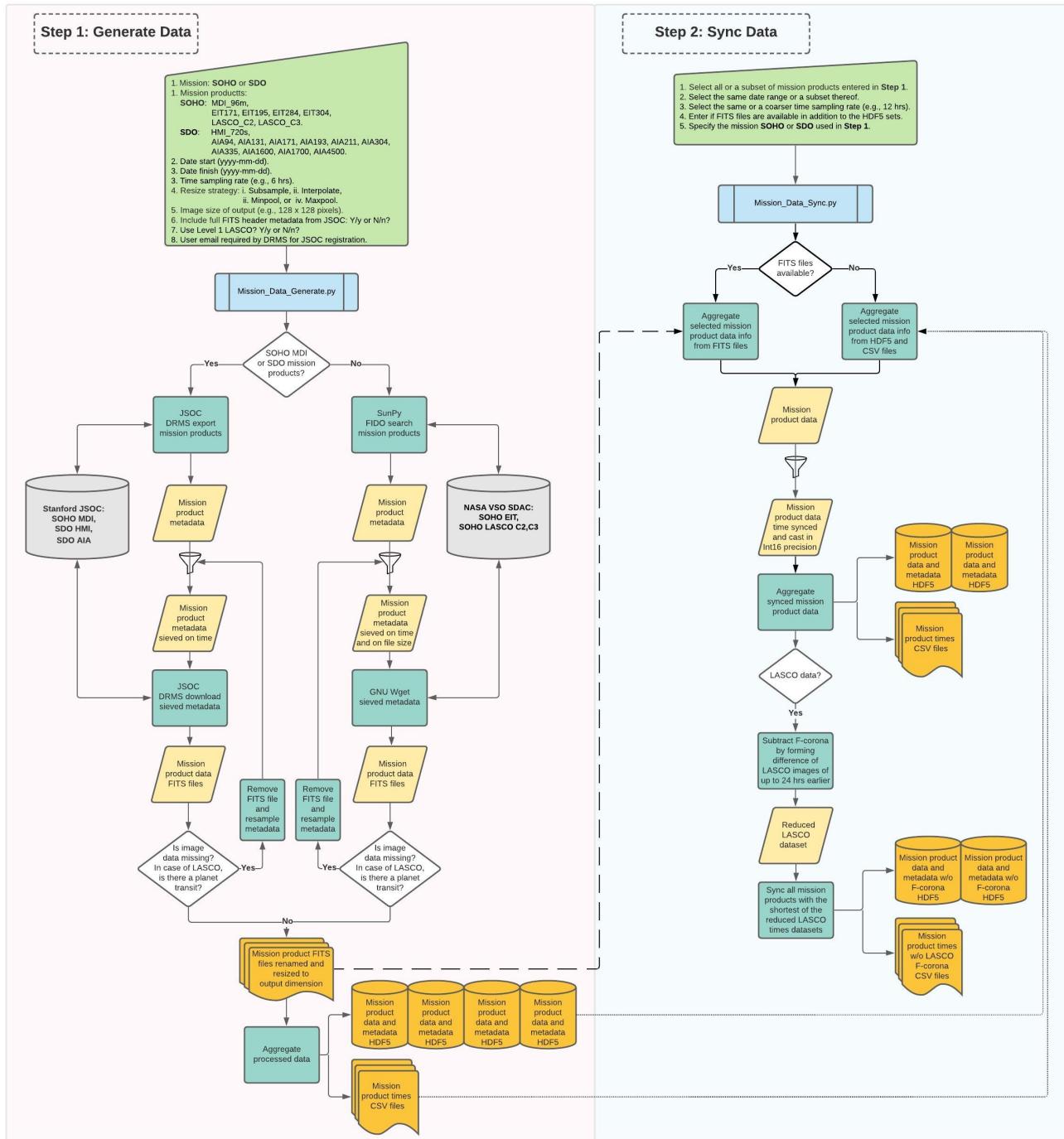


Figure 2. The two flowcharts describe the two main steps of the standard dataset preparation pipeline: data generation and data synchronization. The following color scheme is used: light green denotes software options provided for the user, light blue is for the executing program, turquoise is for key main actions, gray is for external data, yellow is for flowing data, white is for a decision, and orange is for output data products.

range of values in the output image. iii-iv. min-pooling and max-pooling both use the block-reduce function from Scikit-image²¹ to tile the image in blocks of size (scale factor) x (scale factor) and compute the minimum or maximum on each such block,

- FITS headers are down-sampled using the FITS convention with compliant SunPy keywords corresponding to the

77 down-sampling of the respective image products,

- 78 • An option of including the updated FITS metadata with compliant SunPy keywords for JSOC products is also
79 provided but it adds to the processing time as the server needs to first prepare the requested files,
- 80 • As an alternative to using raw, uncalibrated LASCO Level-0.5 data, the option of using Level-1.0 LASCO images
81 is provided. LASCO Level-1.0 data products have the following corrections applied: i. corrections for bias,
82 and calibration to units of mean solar brightness, ii. correction to account for vignetting, particularly around
83 the occulting disk and pylon, iii. warping image to yield flat geometry for entire field of view, iv. up to several
84 minutes clock correction, v. updates to metadata parameters such as the Sun-center location and exposure time.
85 Furthermore, since 2003, the SOHO spacecraft performs 180-degree flips approximately every three months to
86 account for an antenna issue; the Level-1.0 data product provides precise corrections for these roll periods in the
87 metadata, though does not rectify images. In our opinion, these corrections are not needed for input to an ML
88 system for three reasons: a. image flipping is generally an invariant operation for an ML system and is used for
89 image augmentation to generate more examples for ML systems; b. image warping may interfere with the integrity
90 of the raw data; and c. using the LASCO Level-1.0 data in calibrated physical units provides no advantage over the
91 raw Level-0.5 data which has units of photon counts (DN). In regard to calibrated EIT images, it is anticipated that
92 Level-1.0 EIT images will become publicly available for querying via an online interface hosted by ESA later this
93 year,
- 94 • For use of the DRMS system, JSOC requires that the user preregister their email address at http://jsoc.stanford.edu/ajax/register_email.html.

96 2. Databases and queries:

- 97 • With the user selection of SoHO MDI or SDO mission products, Stanford's JSOC database is queried using DRMS
98 export rather than DRMS query which also lists 'ghost' files that are of size zero. The following JSOC series are
99 used: i. mdi.fd_M_96m_lev182 for SoHO MDI at a natural cadence of 96 minutes, ii. hmi.M_720s for
100 SDO HMI at a natural cadence of every 720 seconds, iii. aia.level_euv_12s for SDO EUV AIA (94, 131,
101 171, 193, 211, 304, 335 Å) at a natural cadence of every 12 seconds, iv. aia.level_uv_24s for SDO UV AIA
102 (1600 and 1700 Å) at a natural cadence of 24 seconds, v. aia.level_vis_1h for SDO visible AIA (4500 Å)
103 at a natural cadence of 1 hour. For SDO AIA, both 'images' and 'spikes' (for calibration) are available and so
104 'images' are explicitly specified. DRMS download is used to retrieve the DRMS exported data,
- 105 • With the user selection of SoHO EIT, SoHO LASCO C2, and/or LASCO C3, SunPy's Fido is used to query the
106 NASA VSO SDAC database. GNU's Wget²² is used to fetch the images since it avoids timing out on large datasets,
- 107 • An internal step size of 60 days is used to produce the initial queries for all SoHO mission and SDO mission
108 products and it respects SDAC's 10k results quota per search as well as DRMS' 100 GB export limit.

109 3. Reduction of data search space:

- 110 • The key difference between the two query tracks in terms of metadata sieving is that JSOC products need only to
111 be sieved on time (i.e., user-defined time sampling rate or cadence) whereas SDAC products need to be sieved
112 on time and on file size. Sieving is performed by computing a dynamic set of indices which is updated each
113 time that a candidate file, that has been locally downloaded, is found to have missing data or to have the wrong
114 shape or to have planetary transits in the case of LASCO products. The actual download proceeds from singleton
115 download requests which are sequentially issued based on these indices. Selecting the proper sized file sizes from
116 the database, a priori, offers a major performance advantage over having to locally download and individually
117 examine FITS headers. Local downloading of files is consequently held off until the cleaning step of the pipeline
118 which utilizes Astropy^{23,24} to access the FITS data and FITS keywords,
- 119 • The planet transit filter works by applying the Probabilistic Hough Transform on an edge-filtered image, obtained
120 from using the Canny algorithm, to specifically detect only horizontal lines. Scikit-image is used for both
121 algorithms,
- 122 • Native SoHO image data corresponds to a resolution of (1024 x 1024) pixels with the exception of EIT 195 which
123 is also frequently available in (512 x 512) pixel resolution. These product resolutions correspond to specific file

124 sizes as shown in Table 2. In order to efficiently reduce the VSO query size for these proper data from the very
125 beginning, SunPy's Fido 'search' function is used to return a 'QueryResponseBlock' dictionary object which
126 contains the keyword 'size' in addition to other keywords corresponding to columns 1-5 and 7 of Table 1 as well
127 as the 'fileid' url pointer for file download. In just a handful of cases, the file size on its own is not sufficient to
128 guarantee the proper data type which necessitates 'naxis=2' in the FITS header to be true in order to have a 2D
129 image as opposed to a 3D data cube.

130 4. Mission product FITS with data and metadata cubes and corresponding times:

- 131 • In the event of an unlikely interruption in downloading which crashes the program, the user can restart on the last
132 FITS file date downloaded,
- 133 • FITS files are first generated from all data satisfying user specifications,
- 134 • The data from these FITS files is then aggregated and delivered in compressed Hierarchical Data Format version
135 5^{25,26} (HDF5) data cubes per data product specified. Accompanying each of these data cubes are comma-separated
136 values (CSV) files containing the time axis points of the respective data cube. These data cubes are now ready for
137 input to the second step of the software pipeline which is data synchronization.

138 A major problem in using solar images for machine learning is that, as shown in Table 1, each image comes with its own
139 time cadence. In our Python tool, the user-defined time sampling rate fixes the cadence of all products to the same rate and
140 syncing makes the user-selected set of SoHO products all fall within the same time window. This is illustrated in the rightmost
141 flowchart in Fig. 2. This operation further reduces the volume of data to a more manageable temporal resolution.

142 **Step 2 (time synchronization):**

143 1. User inputs

- 144 • The user selects the mission entered in Step 1 together with all or a subset of the mission products and all or a
145 subset of the date range, as well as the same or coarser time sampling rate sub-sampled by an integer multiple.
146 The user also has the choice of combining images of the same output dimension but obtained with different resize
147 strategies across the different products,
- 148 • The user is presented with two options: i. if the user has simply downloaded the HDF5 cubes and CSV files without
149 locally generating them, then the data from the HDF5 cubes will be taken up along with the corresponding times
150 from the accompanying CSV file for each product specified, ii. if the user has run the Python tool and obtained
151 both the HDF5 data cubes along with the corresponding FITS files then the FITS files will be used to obtain both
152 the data and the times since the time stamps are written into the names of the FITS files themselves, In the case of
153 FITS files being present, the times are directly read from the names of the FITS files because this is more robust
154 against any interruption in the data generation step,
- 155 • The sync algorithm then finds the product with the shortest time list and moves along its time axis checking for
156 overlap with the times of the other products. The overlap time is defined as +/- half the time window. The closest
157 times from all products to that time are taken and if equal times are found for a product then the first such time
158 is always selected. If a matching time for a product for a particular point on the time axis is not found, all other
159 product times obtained for that time axis point are discarded. Therefore, it follows that the more products specified,
160 the harder it is for a time axis point to be shared among all the products,
- 161 • The outputs are synced HDF5 cubes in a numeric format suitable for efficient GPU utilization and corresponding
162 synced times per product as CSV files,
- 163 • If a LASCO product is present, the F-corona is subtracted by taking the time difference between the given LASCO
164 image with itself up to 24 hours earlier. This results in a reduced LASCO image set which is then synced with the
165 rest of the selected products,
- 166 • This new reduced set is then output as the second set of HDF5 and CSV files.

167 Cleaning and processing from the first step and syncing from the second step together yield a machine-learning-ready
168 standardized dataset.

SoHO product	File size (KB)	Resolution (pixels)	Calibration level	Provider
MDI	~ 1400-1600	(1024 x 1024)	L1.8.2	JSOC
MDI	4115	(1024 x 1024)	L1.5, L1.8.2	Removed from SDAC
EIT 171	2059	(1024 x 1024)	L0.5	SDAC
EIT 195	2059 or 523	(1024 x 1024) or (512 x 512)	L0.5	SDAC
EIT 284	2059	(1024 x 1024)	L0.5	SDAC
EIT 304	2059	(1024 x 1024)	L0.5	SDAC
LASCO C2	~ 2100, 4106	(1024 x 1024)	L0.5, L1.0	SDAC
LASCO C3	~ 2100, 4106	(1024 x 1024)	L0.5, L1.0	SDAC

Table 2. Proper file sizes identified for all available calibration levels of SoHO products from the SDAC and JSOC data providers with their respective resolutions. LASCO file sizes are given as approximate due to some minor size variations noted over the years. File size is not relevant for calibrated MDI images obtained from JSOC. It is anticipated that Level-1.0 EIT images will become publicly available for querying via an online interface hosted by ESA later this year.

SoHO product	No. FITS files	HDF5 cube size (MB)	No. excluded files with holes (& planetary transits)
MDI	15456	561	66
EIT 171	10234	187	1045
EIT 195	14439	254	1391
EIT 284	9601	151	895
EIT 304	9439	158	936
LASCO C2	15383	329	489 (& 7511)
LASCO C3	13354	277	699 (& 23901)

Table 3. Individual SoHO products queried at a cadence of 6 hours and time span of Jan. 1, 1999 - Dec. 31, 2010. Image output dimensions are (128 x 128) pixels. The number of images that have been excluded per image product due to holes and, in the case of LASCO, planetary transits is also provided.

3 Data Records

The input data and provenance corresponding to the suite of SoHO instruments described in Table 1 is provided by Table 2. The datasets analysed during the current study, consisting of data plus metadata serialized with JSON, are available as compressed HDF5 data cubes through SURF at (<https://surfdrive.surf.nl/files/index.php/s/NYHm1b9hOKMMcW0>) and are all contained in the “Mission_ML_Pipeline” directory.

The HDF5 data cubes come with corresponding time stamps per image slice provided by the accompanying CSV files. Time stamps are provided as a separate CSV file primarily because a CSV file provides ease-of-access for a beginning user and enables to quickly determine the corresponding number of images in the respective data cube. Each slice of the data cube has one fixed, user-specified output image size which, in this case, is (128 x 128) pixels obtained from the sub-sampling resize strategy. An image output size of (128 x 128) pixels is used for all image products in this paper because all of the data for a given experiment can be efficiently loaded into GPU memory. The original size of the images is (1024 x 1024) pixels with the exception of EIT 195 which also comes in (512 x 512) sizes.

The first set of data cubes encompass all seven SoHO products as shown in Table 3 and has a temporal cadence of six hours and time span of Jan. 1, 1999 - Dec. 31, 2010. The second set of data cubes are the five synced experiments with the same time span and temporal cadence of six hours and separately comprised of one, three, and seven image products as shown in Table 4. Within the three and seven product experiments, the effect of both including and subtracting the F-corona from the LASCO products is investigated. Explicitly, the experiments are: i. MDI-only, ii-iii. MDI, EIT 195, LASCO C2 (minus F-corona), iv-v. all seven SoHO products (minus F-corona). With more products there is a higher probability that some of them are unavailable at the specified times, so that fewer synced times are available.

The sub-directories consist of HDF5 data cubes along with their respective CSV files containing all of the time stamps and are as follows:

- “19990101_20101231_6_128_7_products” are the generated data cubes for all seven products individually,
- “19990101_20101231_6_128_MDI_self_synced” is the self synced MDI data cube,
- “19990101_20101231_6_128_3_products_synced” are the synced data cubes corresponding to MDI, EIT 195, and

SoHO experiment	Total sync output	Train	Validation	Test	Total sync used
MDI only	15455	9090	1269	1227	11586
MDI, EIT 195, LASCO C2	12299	7384	953	943	9280
MDI, EIT 195, LASCO C2 Δ	12132	7292	940	926	9158
MDI, EIT 171, 195, 284, 304, LASCO C2, C3	4348	2369	275	347	2991
MDI, EIT 171, 195, 284, 304, LASCO C2 Δ , C3 Δ	3360	1833	195	268	2296

Table 4. Five SoHO experiments selected to be synced with a cadence of 6 hours and time span of Jan. 1, 1999 to Dec. 31, 2010. The total synced output comes from the second step of the pipeline. For the three product and seven product experiments, the subtraction of the F-corona by time differences of up to 24 hours prior to a LASCO image and itself are used and represented by ‘ Δ ’ after C2,C3. The train, validation, and test data split is formed by applying a progressive rolling indices scheme by successive year. Image dimensions are the same as reported in Table 3. Note that due to the synchronization step the MDI-only experiment contains one less file than in Table 3.

193 LASCO C2. A further sub-directory “Fcorona_subtracted_3products” contains the corresponding F-corona subtracted
194 products,

- 195 • “19990101_20101231_6_128_7_products_synced” are the synced data cubes of all seven SoHO products. A further
196 sub-directory “Fcorona_subtracted_7products” contains the corresponding F-corona subtracted products.

197 These datasets were derived from the following public domain resources: VSO (<https://sdac.virtualsolar.org/cgi/search>) and JSOC (http://jsoc.stanford.edu/MDI/MDI_Magnetograms.html). The example
198 datasets are 6.6 GB in total.
199

200 4 Technical Validation

201 Having a standardized dataset facilitates data exploration including, for example, studies of correlation between different
202 channels. Figure 3 shows a comparison of mean intensity variation across all synced SoHO products from 1999 - 2010.
203 The long-term variation in the mean value of the intensity of the solar disk for EIT extreme ultraviolet (EUV) images at
204 wavelengths of 171, 195, 284, 304 Å show an interesting downward trend corresponding to the CCD degradation over time.
205 Also in this trend are regular clean-up operations which involve reheating the detector to account for the burn-in from the sun
206 which leave ‘break-like’ signatures in the spectrum. The wiggles on the spectrum have a periodicity of roughly 27 days and
207 correspond to active regions. Less readily visible is the growth of mean intensity with increasing solar cycle 23. One of the
208 most interesting peaks seen in all the EUV channels occurs on October 27, 2003 corresponding to the Halloween storm of
209 2003^{27,28} a combination of the bright active regions and the excessive particle events that ‘blinded’ the camera repeatedly over
210 those few days. The long term variation of the LASCO C2 and LASCO C3 both show a periodic variation in the mean intensity.
211 The cyclical broadening of the peak is the puffing up of the F-corona every half year due to the apparent size of the sun getting
212 larger than the fixed occulter and then getting smaller again. This cyclical variation with the Sun–SoHO distance was also
213 found in previous studies²⁹. The overall amplitude in the LASCO images follows the solar cycle and the tiny wiggles, as for
214 the EIT images, correspond to Carrington rotations and harmonics thereof. The peaks in the mean intensity of the images for
215 LASCO C2 and LASCO C3 are partly attributed to observational artefacts and to CMEs in LASCO field of view. The signature
216 of the aforementioned Halloween storm is also visible in the mean value of the LASCO images. The maximum of the absolute
217 values of the magnetic field measurements by MDI best represents the solar cycle; getting weaker approaching the minimum of
218 solar cycle 23 in the later part of 2008 and rising up again, marking the beginning of solar cycle 24. A visual comparison can
219 be seen between the observations of the MDI measurements with the monthly averaged sunspot number (SSN), marking the
220 maxima and the declining phase of the solar cycle 23 and beginning of the solar cycle 24. The gaps in the data products are due
221 to the prepossessing step where the planetary transits are filtered out from the entire dataset.

222 Accompanying the SoHO product temporal profiles, dips in the north-south component of the interplanetary magnetic field
223 (IMF) $B_z(t)$, as measured at Lagrange point one (L1), signify the geomagnetic storms. The stark difference in the geomagnetic
224 activity at different phases of the solar cycle can be clearly seen. The dips are numerous during the maximum of the solar cycle
225 23. In contrast, the rising phase of the solar cycle does not show significant dips in the B_z measurements at L1. As seen from
226 Figure 3, pre-processing of the data is clearly non-trivial and additional steps downstream of our pipeline would be needed to
227 account for instrument degradation and trends in the data that are entangled with solar cycle variability.

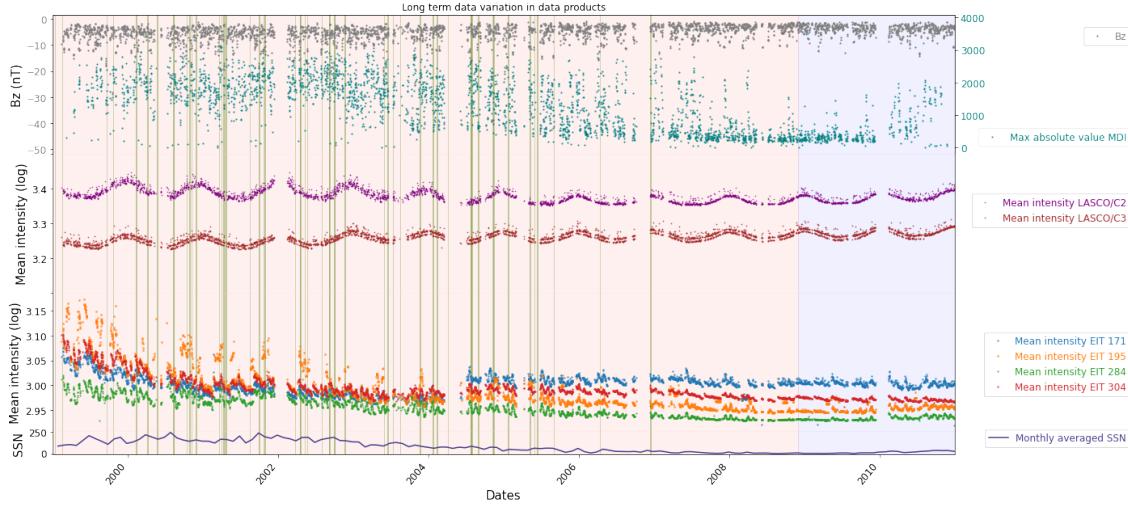


Figure 3. Long term variation (1999 - 2010) of the synced products. Shown are the LASCO images with F-corona subtracted for mean intensity of the solar disc for EIT EUV (blue, red, orange and green curves) in log scale, LASCO white-light (brown and purple curves) in log scale, maximum of absolute value of MDI magnetogram (light blue curve) and B_z values at L1 (gray curves). The light pink and blue backgrounds indicate the data time overlap with solar cycles 23 and 24. The solar cycle is also evident in the bottom plot with the monthly averaged sunspot number (SSN) for the duration of the dataset. The vertical olive-green lines corresponds to large geomagnetic storms with Dst indices smaller than -100.

228 5 Usage Notes

229 Along with the dataset, we provide and discuss results for a simple machine learning model that predicts the onset of a
 230 geomagnetic storm. This is intended as a baseline model that can be used in future studies to assess the skill of more complex
 231 models.

232 5.1 Example application of SoHO dataset to space weather use case

233 SoHO is a cooperative mission between NASA and ESA and was the first space-based telescope to serve as an early warning
 234 system for space weather. SoHO was launched on December 2, 1995 and remains operative after 25 years of service. It has
 235 revolutionized space weather forecasting capabilities by providing several days notice of Earth-directed disturbances such as
 236 Coronal Mass Ejections (CMEs). Space weather refers to conditions on the sun, in the solar wind, magnetosphere, ionosphere,
 237 and thermosphere, that can influence the performance and reliability of space-borne and ground-based technological systems
 238 and can endanger human life or health³⁰.

239 In this section we discuss an example application of the dataset to a challenging space weather problem, namely the forecast
 240 of B_z at L1, in a two-day time window which spans 3 – 5 days ahead of the image data. This is physically justified since the
 241 mean bulk speed³¹ of CMEs is 400 km/s which yields an average propagation time from Sun to Earth of around three days.
 242 Even the slow solar wind from the equatorial regions of the Sun has a mean speed³² of 400 km/s. It is known that that coronal
 243 brightness alone has no relationship with B_z , as recently re-affirmed using two full solar cycles of LASCO C2 data¹⁶.

244 We note that the images contain the full solar disk, so information about events affecting B_z might be present before and after
 245 an image that is assigned a positive event truth label. This can lead to a solar image with an active region toward the left solar
 246 limb being labeled a negative event since its effect has not yet arrived at L1. In order to simplify the problem, the prediction of
 247 B_z is treated as a binary classification problem, with the target being defined as the difference between the minimum of B_z
 248 and the mean of B_z in each two-day window. Values of B_z are obtained from the NASA OMNI low resolution (LRO) dataset
 249 (https://spdf.gsfc.nasa.gov/pub/data/omni/low_res_omni) which provides hourly averaged values. The
 250 mean is subtracted in order to have some variation around the minimum value of B_z encountered during geomagnetic
 251 disturbances since these values tend to dominate the two-day window. The data is selected to span the years 1999 to 2010
 252 inclusive. This is because the intrinsic image quality from 1996 - 1998 is not too reliable.

253 Five experiments are performed using the following image products: i. MDI-only, ii-iii. MDI, EIT 195, LASCO C2 (minus
 254 F-corona), iv-v. all seven SoHO products (minus F-corona). For each experiment the datasets are synced. In order to prevent
 255 temporal correlation between the training, validation and test sets, a rolling split was applied to the data. For 1999, Feb.- Aug.
 256 is used for the training set, Oct. for the validation set, and Nov. for the test set. For each subsequent year, all three of these set
 257 partitions are advanced by one month forward so that for 2010, as an example, it is Jan.-July for training, Sept. for validation,

Product used for Gaussian Naive Bayes (GNB)	%	TP	FP	FN	TN	TSS	MCC
MDI (max, min, std)	25	212	336	85	594	0.35	0.30
MDI (min, std, frac)	20	171	392	67	597	0.32	0.26
MDI (min, std, frac)	15	141	392	50	644	0.36	0.26
MDI (min, std, frac)	10	110	400	36	681	0.38	0.25
MDI (min, std, frac)	5	35	459	5	728	0.49	0.18
MDI (std), EIT 195 (frac), LASCO C2 (0)	25	153	252	67	471	0.35	0.30
MDI (min), EIT 195 (0), LASCO C2 (0)	20	131	338	45	429	0.30	0.24
MDI (std), EIT 195 (frac), LASCO C2 (0)	15	96	273	47	527	0.33	0.24
MDI (std), EIT 195 (frac), LASCO C2 (0)	10	77	298	36	532	0.32	0.21
MDI (std), EIT 195 (min), LASCO C2 (frac)	5	32	441	2	468	0.46	0.17
MDI (frac), EIT 195 (mean), LASCO C2 Δ (frac)	25	159	378	53	336	0.22	0.19
MDI (frac), EIT 195 (std), LASCO C2 Δ (frac)	20	119	333	51	423	0.26	0.20
MDI (frac), EIT 195 (mean), LASCO C2 Δ (frac)	15	117	448	22	339	0.27	0.20
MDI (frac), EIT 195 (mean), LASCO C2 Δ (frac)	10	92	418	17	399	0.33	0.22
MDI (frac), EIT 195 (std), LASCO C2 Δ (frac)	5	30	401	3	492	0.46	0.17
MDI (std), 195 (0), 171 (max), 284 (std), 304 (std), C2 (max), C3 (min)	25	57	72	29	189	0.39	0.35
MDI (frac), 195 (mean), 171 (std), 284 (std), 304 (max), C2 (max), C3 (std)	20	39	66	23	219	0.40	0.33
MDI (min), 195 (mean), 171 (max), 284 (max), 304 (std), C2 (max), C3 (frac)	15	38	101	10	198	0.45	0.32
MDI (frac), 195 (mean), 171 (std), 284 (std), 304 (std), C2 (max), C3 (0)	10	27	63	12	245	0.49	0.35
MDI (frac), 195 (0), 171 (max), 284 (min), 304 (std), C2 (max), C3 (mean)	5	9	45	2	291	0.68	0.33
MDI (frac), 195 (std), 171 (max), 284 (min), 304 (std), C2 Δ (mean), C3 Δ (mean)	25	26	14	39	189	0.33	0.40
MDI (max), 195 (mean), 171 (min), 284 (0), 304 (std), C2 Δ (max), C3 Δ (frac)	20	33	91	13	131	0.31	0.23
MDI (max), 195 (mean), 171 (max), 284 (mean), 304 (std), C2 Δ (std), C3 Δ (0)	15	28	74	8	158	0.46	0.32
MDI (max), 195 (std), 171 (0), 284 (mean), 304 (std), C2 Δ (min), C3 Δ (min)	10	25	84	4	155	0.51	0.32
MDI (max), 195 (frac), 171 (min), 284 (mean), 304 (max), C2 Δ (min), C3 Δ (mean)	5	8	28	1	231	0.78	0.41
Product used for Convolutional Neural Network (CNN)	%	TP	FP	FN	TN	TSS	MCC
MDI only	25	15	13	282	917	0.04	0.10
MDI, EIT 195, LASCO C2	25	29	25	191	698	0.10	0.18
MDI, EIT 195, LASCO C2 Δ	25	53	145	159	569	0.05	0.05
MDI, EIT 195, EIT 171, EIT 284, EIT 304, LASCO C2, LASCO C3	25	27	20	59	241	0.30	0.24
MDI, EIT 195, EIT 171, EIT 284, EIT 304, LASCO C2 Δ , LASCO C3 Δ	25	20	26	45	177	0.18	0.20

Table 5. Results from the five SoHO experiments obtained for two categories of models for binary classification: Gaussian Naive Bayes (GNB) (first five cells) and from a three-layer convolutional neural network (CNN) (final cell). Each model is evaluated at a percentile (25, 20, 15, 10, 5)% of B_z values from the MDI-only training set. These percentiles correspond to B_z values of -6.28, -6.95, -7.76, -8.92, and -11.61 nT, respectively. The best performing CNN models for each of the five experiments are shown. The ‘ Δ ’ denotes F-corona subtraction obtained by taking the time difference between the given LASCO image with itself up to 24 hours earlier. For the GNB category, the best performing models are shown with their corresponding features per SoHO product: ‘max’ is the maximum, ‘min’ is the minimum, ‘mean’ is the mean, ‘std’ is the standard deviation, ‘frac.’ is the Minkowski–Bouligand fractal dimension, and ‘0’ denotes feature absence. Per model, the True Skill Score (TSS), at 50%, and Matthews Correlation Coefficient (MCC) are given with corresponding true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN).

and Oct. for the test set. The other months are not used. The final datasets for each experiment are given in Table 4 and approximately yield an 80-10-10 split for the training, validation, and test sets. Each of the experiments’ test sets is normalized by the maximum of the absolute value of the respective training sets. The corresponding truth labels for each experiment are further assigned a binary value determined by setting the threshold p at five percentiles (25, 20, 15, 10, 5)% corresponding to B_z values of -6.28, -6.95, -7.76, -8.92, and -11.61 nT, respectively. This causes a class imbalance at each threshold which requires the use of class weights for training both the CNN and GNB models to perform binary classification.

5.1.1 Machine learning approach

We apply a three-layer CNN, implemented with TensorFlow³³, to each of the five experiments. The SoHO products used in each of these experiments constitute the input channels. These channels are input in parallel with their individual dense or fully connected layers eventually concatenated as depicted in Fig. 4. Due to the application of a down-sampling strategy on

268 the images, the full data cubes can fit into a several GB GPU memory with the final down-sampled sizes constrained by how
 269 many image products will be run in parallel. In our case, sub-sampling to (128 x 128) pixels fits into the GPU for all five
 270 experiments. This parallel approach is chosen over the traditional, stacking approach, in that the integrity of the individual
 271 product's features may be preserved instead of merging features across different products which will confound the individual
 272 product's contribution. Three successive block iterations consisting of max-pooling, convolution, batch normalization and a
 273 rectified linear unit (ReLU) activation are applied on each parallel network. After the above mentioned concatenation, two
 274 successive blocks of linear activation and dropout, which acts as a regularizing term to prevent over-fitting, are applied. The
 275 final single activation unit is given by the sigmoid function which assigns binary values to the preceding probabilities. A binary
 276 cross entropy loss function is used together with a stochastic gradient descent (SGD) optimizer. A convergence is achieved
 277 within 100 epochs with early stopping enabled triggered by the validation loss. The total number of model parameters is
 278 ~62K, ~178K, and ~409K for the one, three, and seven product experiments, respectively. The network architecture and
 279 hyper-parameters have not been optimized for predicting B_z in this paper so the results should be interpreted as simple baselines.

280 **5.1.2 Baseline model, features, and metrics**

281 The Gaussian Naive Bayes (GNB) classifier is used as a baseline and implemented with Scikit-learn²¹ in order to compare
 282 with the deep CNN approach. The Naive Bayes classifier assumes that the value of a particular feature is independent of the
 283 value of any other feature, ignoring any possible correlations. The advantage of this method is that it only requires a small
 284 number of training data to estimate the parameters necessary for classification. In our approach, we examine combinations
 285 of five input features for each of the five experiments. The first four features that are selected to characterize the image pixel
 286 distribution are the max, min, mean, and standard deviation. The last feature leverages the mathematics of shape, as applied
 287 to space weather prediction in a related but different topological approach³⁴, to compute the Minkowski–Bouligand (MB)
 288 fractal dimension or box-counting dimension. The MB fractal dimension takes on values between 1 and 2 for a 2D image. The
 289 box-counting dimension is computed on sub-sampled SoHO images which are each first transformed into a binary representation
 290 using the respective means of these images as the discriminator. At each of the experiments' percentile thresholds, a best-fit
 291 model is determined. For the single SoHO product experiment, there are 31 distinct feature combinations. For the other two
 292 multi-product experiments, one feature per product is considered which yields 216 combinations for three products and 279,936
 293 combinations for all seven products, of which one in ten is sampled. The GNB approach does not utilize a validation set. As
 294 such, in order to maintain a direct comparison with the CNN results, the validation set is excluded rather than used to make a
 295 larger training set.

296 The predictions from the CNN and GNB are compared using the TSS and MCC scores. TSS is the difference between the
 297 true and false positive rates and is an unbiased estimator with respect to class imbalance⁸. MCC is a correlation coefficient
 298 between the observed and predicted binary classifications and can be used if the classes are of very different sizes. Both
 299 measures return a value between -1 and +1. Each of the GNB model TSS and MCC scores can be directly compared with the
 300 same class of CNN models since the number of positives and negatives is fixed for the same products used at a given threshold.
 301 Expressions for TSS and MCC are given in eqs. 1 and 2 as

$$TSS = TPR - FPR = \frac{TP}{TP + FN} - \frac{FP}{FP + TN}, \quad (1)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}}, \quad (2)$$

302 where TP, TN, FP, FN denote true-positive/negative and false-positive/negative.

303 **5.1.3 Results and discussion**

304 In Table 5, we present baseline results arising from the application of a Gaussian Naive Bayes (GNB) classifier and a deep
 305 convolutional neural network (CNN) to a binary classification of B_z , which serves as a proxy for the onset of geomagnetic
 306 disturbances at Earth. The features for the GNB are a set of standard statistical quantities characterizing the image pixel value
 307 distribution (min, max, mean, standard deviation) with the addition of the Minkowski–Bouligand (MB) fractal dimension³⁵.
 308 Additionally, there is a ‘dummy’ feature for each product that is set to zero. The selection of this dummy feature in Table 5,
 309 indicated by a (0), shows that the respective product was not informative.

310 The best performing CNN model has a True Skill Statistic (TSS) of 0.30 and and Matthews correlation coefficient (MCC)
 311 of 0.24 and is obtained for the experiment with all seven SoHO products, with the F-corona left in for the LASCO products,
 312 evaluated at the 25th percentile of B_z corresponding to -6.28 nT as shown in Table 5. It is interesting that with the F-corona
 313 subtracted, the model performance drops to a TSS of 0.18 and MCC of 0.20. The difference here is that there are ~ 700
 314 fewer synced times available when the F-corona is subtracted as seen from Table 4, which results in 21 fewer true events as
 315 only 65 true events (TP + FN) remain following F-corona subtraction as opposed to the 86 true events starting out when the

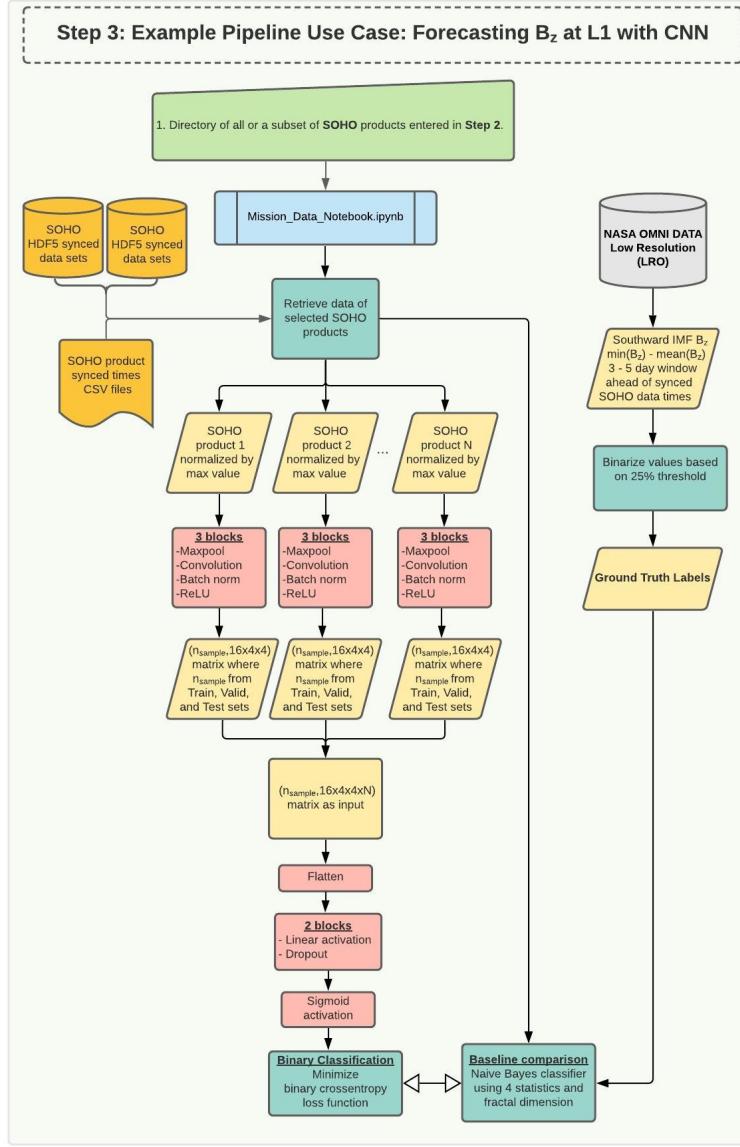


Figure 4. The flowchart describes the steps of applying a deep CNN network for classifying the occurrence of a geomagnetic storm using B_z as an example proxy. This prediction is then compared with a classical baseline. In the box titled ‘Baseline comparison’, the four statistics refer to the max, min, mean, and standard deviation and the fractal dimension is the Minkowski–Bouligand fractal dimension. Three block iterations of successive max-pooling (2×2), convolution (3×3), batch normalization and ReLU activation is applied on the (128×128) pixel synced images. The convolutional layers are each initialized using the Glorot uniform initializer, also called Xavier uniform initializer. After the flattened layer containing the aggregate normalized features from all products come three fully connected layers. Batch normalization is first applied, followed by two iterative blocks of successive linear activation and dropout. Linear activation consists of 128 activation units in the first block and 32 units in the second. A dropout rate of 60% is used as a regularizer to reduce over-fitting. The final single activation unit is given by the sigmoid function which assigns binary values to the preceding probabilities. It is at this stage only that a single bias term is introduced. A binary cross entropy loss function is used together with a SGD optimizer with a fixed learning rate of 0.01. A convergence is achieved within 100 epochs with early stopping enabled triggered by the validation loss. Color scheme is same as in Fig. 2 with the addition of pink which denotes key stages in the CNN architecture.

³¹⁶ F-corona is left in. This is a $\sim 25\%$ reduction in the number of true events which may contribute to the drop in performance.
³¹⁷ For this same experiment, the GNB model has a TSS of 0.39 and MCC of 0.35 with the features: standard deviation of MDI, no

318 EIT 195, maximum of EIT 171, standard deviation of EIT 284, standard deviation of EIT 304, maximum of LASCO C2, and
319 minimum of LASCO C3. Although the GNB outperforms the CNN in this example, we note that the CNN's architecture and
320 hyper-parameters have not been optimally selected. The GNB best-fit model with maximum for MDI, MB fractal dimension for
321 EIT 195, minimum for EIT 171, mean for EIT 284, maximum for EIT 304, minimum for LASCO C2, and mean for LASCO C3
322 with the F-corona in the LASCO images subtracted at the 5% threshold in Table 5 has a sensitivity or true positive rate (TPR)
323 of 89% and the highest GNB specificity or true negative rate (TNR) of 89% making this a satisfactory model for predicting "all
324 clear" scenarios for space weather applications. That this model has a TSS score of 0.78 but only an MCC score of 0.41 also
325 re-affirms that a single metric is insufficient to fully describe a model's performance. Comparing this GNB best-fit model with
326 the F-corona subtracted with the GNB best-fit model at the 5% threshold with the F-corona left in, shows an added \sim 15% in
327 TSS and \sim 24% in MCC performance compared to a TSS of 0.68 and MCC of 0.33. Subtracting the F-corona increases metric
328 performance in the seven product experiments at the 5% to 15% thresholds.

329 It is likely that a future extensive architectural study combined with a hyper-parameter sweep will produce a neural network
330 that can outperform the GNB baseline. Capturing temporal evolution of magnetograms may hold the key for further predictive
331 power. That there is information content in the SoHO images which does allow for forecasting is encouraging to prompt further
332 investigations by the community. Since the code is intrinsically agnostic to the ground truth labels, a variety of space weather
333 parameters can be examined.

334 The machine learning-ready dataset and the software pipeline with an executable Jupyter notebook for ML experiments, such
335 as illustrated by Fig. 4, presented in this paper pave the way toward establishing a community-wide practice that emphasizes
336 reproducible research and benchmark datasets in space weather. An advantage of this pipeline is that it is highly modular thanks
337 to its wrapping around both SunPy Fido and DRMS functionality. It enables a sophisticated user to add on other data providers
338 and data products that are supported by the SunPy community and JSOC. The user can opt to generate FITS files locally or to
339 work with the pre-generated HDF5 data cubes. Another advantage of the pipeline is that different down-sampling strategies
340 may be better suited to different SoHO image products as observed from the features of the best fit GNB models in Table 5.
341 Filters for missing data and planetary transits filter the data from these signal incursions which interfere with the system's
342 learning of the real data. Furthermore, there is a significant compute performance advantage over having to download each
343 and every FITS file locally in order to determine its suitability versus querying the JSOC and SDAC databases on-the-fly to
344 fetch suitable files and performing the image integrity assessment locally. The code also outputs clickable URLs to all products
345 that have been discarded due to having missing data and separate identifiers of planetary transits. Furthermore, the scale of
346 the observed region is kept as in the original images (no re-scaling). This should be appropriate for most machine learning
347 applications. However, there might be some specialized applications, such as super-resolution, where the performance could be
348 improved if the apparent size of the Sun is kept constant. Such re-scaling should then be applied downstream of our pipeline. If
349 it is eventually found that there is not enough information contained in the images for forecasting space weather, this would still
350 be an important step.

351 **6 Code and Data availability**

352 All of our Python code and Jupyter notebooks are open-source. Further details on virtual environment setup, code usage with
353 sample benchmarks, and accompanying Jupyter notebooks for i. performing the CNN and GNB experiments on one, three, and
354 all seven SoHO image products, with and without the F-corona, as described in this paper and ii. illustrating filters for planet
355 and comet transits as well as cosmic ray and blob detection, may be obtained via (<https://github.com/cshneider/soho-ml-data-ready>). Software versions of required libraries are provided in the 'Mission_Requirements' text document
356 and can be automatically installed with the specified conda environment using 'conda install' as in the instructions of the
357 aforementioned GitHub repository. Interested parties are encouraged to get involved in the ongoing developments of the dataset.
358

359 **References**

- 360 1. Owens, M. J., Lang, M., Riley, P. & Stansby, D. Towards Construction of a Solar Wind "Reanalysis" Dataset: Application
361 to the First Perihelion Pass of Parker Solar Probe. *Sol. Phys.* **294**, 83, [10.1007/s11207-019-1479-6](https://doi.org/10.1007/s11207-019-1479-6) (2019).
- 362 2. Howard, T. A., Nandy, D. & Koepke, A. C. Kinematic properties of solar coronal mass ejections: Correction for projection
363 effects in spacecraft coronagraph measurements. *J. Geophys. Res. Space Phys.* **113**, <https://doi.org/10.1029/2007JA012500>
364 (2008). <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2007JA012500>.
- 365 3. Domingo, V., Fleck, B. & Poland, A. I. The SOHO Mission: an Overview. *Sol. Phys.* **162**, 1–37, [10.1007/BF00733425](https://doi.org/10.1007/BF00733425)
366 (1995).
- 367 4. Pesnell, W. D., Thompson, B. J. & Chamberlin, P. C. The Solar Dynamics Observatory (SDO). *Sol. Phys.* **275**, 3–15,
368 [10.1007/s11207-011-9841-3](https://doi.org/10.1007/s11207-011-9841-3) (2012).

- 369 5. Singer, H., Heckman, G. & Hirman, J. *Space Weather Forecasting: A Grand Challenge*, 23–29 (American Geophysical
370 Union (AGU), 2013). <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/GM125p0023>.
- 371 6. Schrijver, C. J. *et al.* Understanding space weather to shield society: A global road map for 2015–2025 commissioned by
372 cospar and ilws. *Adv. Space Res.* **55**, 2745 – 2807, <https://doi.org/10.1016/j.asr.2015.03.023> (2015).
- 373 7. Schwenn, R. Space Weather: The Solar Perspective. *Living Rev. Sol. Phys.* **3**, 2, [10.12942/lrsp-2006-2](https://doi.org/10.12942/lrsp-2006-2) (2006).
- 374 8. Camporeale, E. The challenge of machine learning in space weather: Nowcasting and forecasting. *Space Weather.* **17**,
375 1166–1207, [10.1029/2018SW002061](https://doi.org/10.1029/2018SW002061) (2019). <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2018SW002061>.
- 376 9. Gombosi, T. I. *et al.* What Sustained Multi-Disciplinary Research Can Achieve: The Space Weather Modeling Framework.
377 *arXiv e-prints* arXiv:2105.13227 (2021). [2105.13227](https://arxiv.org/abs/2105.13227).
- 378 10. Bortnik, J. & Camporeale, E. Ten ways to apply machine learning in Earth and space sciences. *Eos* **102**, [10.1029/2021EO160257](https://doi.org/10.1029/2021EO160257) (2021).
- 379 11. Burrell, A. G. *et al.* Snakes on a spaceship—an overview of python in heliophysics. *J. Geophys. Res. Space Phys.* **123**,
380 10,384–10,402, [10.1029/2018JA025877](https://doi.org/10.1029/2018JA025877) (2018). <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2018JA025877>.
- 381 12. Bobra, M. G. & Mason, J. P. *Machine Learning, Statistics, and Data Mining for Heliophysics* (2020).
- 382 13. Scherrer, P. H. *et al.* The Solar Oscillations Investigation - Michelson Doppler Imager. *Sol. Phys.* **162**, 129–188,
383 [10.1007/BF00733429](https://doi.org/10.1007/BF00733429) (1995).
- 384 14. Delaboudinière, J. P. *et al.* EIT: Extreme-Ultraviolet Imaging Telescope for the SOHO Mission. *Sol. Phys.* **162**, 291–312,
385 [10.1007/BF00733432](https://doi.org/10.1007/BF00733432) (1995).
- 386 15. Brueckner, G. E. *et al.* The Large Angle Spectroscopic Coronagraph (LASCO). *Sol. Phys.* **162**, 357–402, [10.1007/BF00733434](https://doi.org/10.1007/BF00733434) (1995).
- 387 16. Battams, K., Howard, R. A., Dennison, H. A., Weigel, R. S. & Lean, J. L. The LASCO Coronal Brightness Index. *Sol.*
388 *Phys.* **295**, 20, [10.1007/s11207-020-1589-1](https://doi.org/10.1007/s11207-020-1589-1) (2020). [2001.07596](https://arxiv.org/abs/2001.07596).
- 389 17. Glogowski, K., Bobra, M. G., Choudhary, N., Amezcuia, A. B. & Mumford, S. J. drms: A python package for accessing
390 hmi and aia data. *J. Open Source Softw.* **4**, 1614, [10.21105/joss.01614](https://doi.org/10.21105/joss.01614) (2019).
- 391 18. Barnes, W. *et al.* The sunpy project: Open source development and status of the version 1.0 core package. *The Astrophys.*
392 *J.* **890**, 68 (2020).
- 393 19. Hill, F. *et al.* The virtual solar observatory—a resource for international heliophysics research. *Earth, Moon, Planets* **104**,
394 315–330, [10.1007/s11038-008-9274-7](https://doi.org/10.1007/s11038-008-9274-7) (2009).
- 395 20. SunPy Community, T. *et al.* SunPy—Python for solar physics. *Comput. Sci. Discov.* **8**, 014009, [10.1088/1749-4699/8/1/014009](https://doi.org/10.1088/1749-4699/8/1/014009) (2015). [1505.02563](https://arxiv.org/abs/1505.02563).
- 396 21. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
- 397 22. Free Software Foundation. GNU wget. <http://www.gnu.org/software/wget/> (2010).
- 398 23. Astropy Collaboration *et al.* Astropy: A community Python package for astronomy. *A & A* **558**, A33, [10.1051/0004-6361/201322068](https://doi.org/10.1051/0004-6361/201322068) (2013). [1307.6212](https://arxiv.org/abs/1307.6212).
- 399 24. Astropy Collaboration *et al.* The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package.
400 *aj* **156**, 123, [10.3847/1538-3881/aabc4f](https://doi.org/10.3847/1538-3881/aabc4f) (2018). [1801.02634](https://arxiv.org/abs/1801.02634).
- 401 25. Collette, A. *Python and HDF5* (O'Reilly, 2013).
- 402 26. The HDF Group. Hierarchical data format version 5 (2000-2010).
- 403 27. Lopez, R. E., Baker, D. N. & Allen, J. Sun Unleashes Halloween Storm. *EOS Transactions* **85**, 105–108, [10.1029/2004EO110002](https://doi.org/10.1029/2004EO110002) (2004).
- 404 28. Gopalswamy, N. *et al.* Introduction to violent Sun-Earth connection events of October-November 2003. *J. Geophys. Res.*
405 (*Space Physics*) **110**, A09S00, [10.1029/2005JA011268](https://doi.org/10.1029/2005JA011268) (2005).
- 406 29. Llebaria, A., Lamy, P., Gilardy, H., Boclet, B. & Loirat, J. Restitution of the K and F Components of the Solar Corona
407 from LASCO-C2 Images over 24 Years [1996–2019]. *arXiv e-prints* arXiv:2011.12920 (2020). [2011.12920](https://arxiv.org/abs/2011.12920).
- 408 30. The National Space Weather Program Strategic Plan, FCM-P30-1995, Washington D.C. (1995).
- 409 31. Yurchyshyn, V., Yashiro, S., Abramenko, V., Wang, H. & Gopalswamy, N. Statistical distributions of speeds of coronal
410 mass ejections. *The Astrophys. J.* **619**, 599, [10.1086/426129](https://doi.org/10.1086/426129) (2008).

- 416 32. McComas, D. J. *et al.* The three-dimensional solar wind around solar maximum. *Geophys. Res. Lett.* **30**, 1517, [10.1029/2003GL017136](https://doi.org/10.1029/2003GL017136) (2003).
- 417 33. Abadi, M. *et al.* TensorFlow: Large-scale machine learning on heterogeneous systems (2015). Software available from tensorflow.org.
- 418 34. Deshmukh, Varad, Berger, Thomas E., Bradley, Elizabeth & Meiss, James D. Leveraging the mathematics of shape for
419 solar magnetic eruption prediction. *J. Space Weather. Space Clim.* **10**, 13, [10.1051/swsc/2020014](https://doi.org/10.1051/swsc/2020014) (2020).
- 420 35. Falconer, K. *Fractal Geometry - Mathematical Foundations and Applications* (Wiley, 2014).
- 421 36. van der Walt, S., Colbert, S. C. & Varoquaux, G. The NumPy Array: A Structure for Efficient Numerical Computation.
422 *Comput. Sci. Eng.* **13**, 22–30, [10.1109/MCSE.2011.37](https://doi.org/10.1109/MCSE.2011.37) (2011). [1102.1523](https://doi.org/10.1109/MCSE.2011.37).
- 423 37. Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Comput. Sci. Eng.* **9**, 90–95, [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55) (2007).
- 424 38. Pérez, F. & Granger, B. E. IPython: a system for interactive scientific computing. *Comput. Sci. Eng.* **9**, 21–29,
425 [10.1109/MCSE.2007.53](https://doi.org/10.1109/MCSE.2007.53) (2007).

426 **Acknowledgements**

427 We would like to thank Frédéric Auchère of Université Paris-Sud 11, Institut d’Astrophysique Spatiale, Orsay, France, for his
428 advice with EIT.

429 We are grateful to all the developers of the following software and machine learning software libraries: TensorFlow³³,
430 Scikit-learn²¹, NumPy³⁶, Matplotlib³⁷, IPython: Jupyter³⁸.

431 CS was partially supported by NWO-Vidi Grant No. 639.072.716. CS would like to thank Drs. Mandar Hemant Chandorkar
432 and Rakesh Sarma for helpful discussions in regard to TensorFlow code implementation.

433 KB was supported by NASA funds to the NRL in support of the SOHO/LASCO project.

434 EC is partially funded by NASA under Grant No. 80NSSC20K1580 (Ensemble Learning for Accurate and Reliable
435 Uncertainty Quantification). This project has received funding from the European Union’s Horizon 2020 research and
436 innovation program under Grant Agreement No. 776262 (AIDA).

437 **Author contributions statement**

438 Carl Shneider wrote the software pipeline, visualized the software pipeline, developed and ran the CNN networks using
439 TensorFlow, visualized the CNN network, bench-marked all of the runs, ran the Gaussian Naive Bayes experiments with
440 features including the fractal dimension, analyzed all the results, visualized all the results (except for Fig. 3), and wrote the
441 paper.

442 Andong Hu helped to develop the CNN networks using PyTorch, visualized the CNN network, run a subset of the Gaussian
443 Naive Bayes experiments and provided analysis on these experiments, and provided insightful comments on the paper.

444 Ajay K. Tiwari produced Fig. 3, helped write Section 4, helped incorporate FITS metadata into the HDF5 data cubes, and
445 provided insightful comments on the paper.

446 Monica G. Bobra provided insightful information on performing queries with Fido and DRMS.

447 Karl Battams provided insightful information on the LASCO coronagraph images and edits in Section 2.

448 Jannis Teunissen advised on the visualization and data analysis, and provided insightful comments on the paper and code.

449 Enrico Camporeale advised on all aspects of the project, helped write the paper, and provided insightful comments on the code.

450 **Competing interests**

451 The authors declare no competing interests.