



Fast Reconstruction of 3D Density Distribution around the Sun Based on the MAS by Deep Learning

Sumiaya Rahman¹, Seunghoon Shin^{1,2} , Hyun-Jin Jeong³ , Ashraf Siddique⁴ , Yong-Jae Moon^{1,3} , Eunsu Park⁵ ,
Jihye Kang³ , and Sung-Ho Bae⁶

¹ School of Space Research, Kyung Hee University, Yongin, 17104, Republic of Korea; moonjy@khu.ac.kr

² Earth Intelligence Division, SI Analytics, Daejeon, 34047, Republic of Korea

³ Department of Astronomy and Space Science, College of Applied Science, Kyung Hee University, Yongin, 17104, Republic of Korea

⁴ Department of Computer Science Engineering, Kyung Hee University, Yongin, 17104, Republic of Korea

⁵ Space Science Division, Korea Astronomy and Space Science Institute, Daejeon, 34055, Republic of Korea

⁶ Department of Computer Science Engineering, College of Software, Kyung Hee University, Yongin, 17104, Republic of Korea; sumiaya@khu.ac.kr

Received 2022 August 29; revised 2023 January 30; accepted 2023 February 16; published 2023 April 28

Abstract

This study is the first attempt to generate a three-dimensional (3D) coronal electron density distribution based on the pix2pixHD model, whose computing time is much shorter than that of the magnetohydrodynamic (MHD) simulation. For this, we consider photospheric solar magnetic fields as input, and electron density distribution simulated with the MHD Algorithm outside a Sphere (MAS) at a given solar radius is taken as output. We consider 155 pairs of Carrington rotations as inputs and outputs from 2010 June to 2022 April for training and testing. We train 152 deep-learning models for 152 solar radii, which are taken up to 30 solar radii. The artificial intelligence (AI) generated 3D electron densities from this study are quite consistent with the simulated ones from lower radii to higher radii, with an average correlation coefficient 0.97. The computing time of testing data sets up to 30 solar radii of 152 deep-learning models is about 45.2 s using the NVIDIA TITAN XP graphics-processing unit, which is much less than the typical simulation time of MAS. We find that the synthetic coronagraphic images estimated from the deep-learning models are similar to the Solar Heliospheric Observatory (SOHO)/Large Angle and Spectroscopic Coronagraph C3 coronagraph data, especially during the solar minimum period. The AI-generated coronal density distribution from this study can be used for space weather models on a near-real-time basis.

Unified Astronomy Thesaurus concepts: The Sun (1693); Solar photosphere (1518); Astronomy data analysis (1858); Astronomy image processing (2306)

1. Introduction

Knowledge of the coronal electron density distribution can provide a better understanding of the physical dynamics of the solar corona (Gómez 2018) and inner heliosphere. To determine the compression ratio of shocks driven by coronal mass ejection (CME), determine the Alfvén Mach number, and predict the CME total mass and propagation, the ambient coronal electron density should be accurately estimated (Vourlidis et al. 2000; de Patoul et al. 2015; Feng et al. 2015). MHD modeling is the initial approach that can bridge self-consistently thoroughly from the heliocentric distances near the Sun to well beyond Earth's orbit (Usmanov 1998; Usmanov & Goldstein 2003; Hayashi 2005; Detman et al. 2006; Riley et al. 2012; Feng et al. 2021). Furthermore, MHD models obtain a 3D distribution that can provide the distribution of the plasma parameters (i.e., density, temperature, and velocity) and the configuration of the magnetic field of the solar corona, inner heliosphere, and outer heliosphere (de Patoul et al. 2015). The MHD Algorithm outside a Sphere (MAS; Linker 1999; Mikić & Linker 1999) model is such a 3D MHD model based on the photospheric magnetic field observations that have been developed to study the large-scale structure and dynamics of the solar corona (from 1 to 30 solar radii) and inner heliosphere (from 30 solar radii to 5 au) (Riley et al. 2009). The MAS model considers the coronal heating, anisotropic

thermal conduction, and radiative losses to develop a more realistic model of the global coronal plasma density distribution (Lionello & Linker 2009). However, this complex simulation technique demands significant computing time, resources, and effort. The entire simulation process, including the determination of the Sun's position, the gathering of the necessary data for one Carrington rotation (CR) process, the collection of data on the underlying magnetic field, and the computation of the simulation for space weather forecasting, takes at most a few days (Caplan & Linker 2019).

Image-to-image translation is a well-known application in deep-learning-based image processing. It is the process of transforming an input image into a corresponding output image, where the main objective is to know the relationship between input and output data. Recently, several forms of network structure, loss functions, learning principles, and techniques have been employed in image translation. The generative adversarial network (GAN; Goodfellow et al. 2014) models like pix2pix (Isola et al. 2017) and pix2pixHD (Wang et al. 2018) prove to be a well-known deep-learning approach that provides state-of-the-art performance in several applications in the field of astronomy and astrophysics, such as reconstruction and synthesis of data (Kim et al. 2019; Park et al. 2019; Jeong et al. 2020; Shin et al. 2020) and denoising (Park & Moon 2019). For high-resolution image translation applications, the pix2pix model (Isola et al. 2017) fails to capture fine details and produces artifacts (Wang et al. 2018). On the other hand, the pix2pixHD model (Wang et al. 2018) can solve the limitations and generate fine features almost without any such artifacts for high-resolution data sets. It



Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

can also increase computational efficiency significantly and minimize computational costs.

Jang et al. (2021) generated the 3D coronal electron density from coronagraphic images for one solar rotation period (27 days). They synthesized the input images from the target 3D electron density in their work by applying the Thomson scattering theory and developed their models only for six selected solar radii. According to our knowledge, this study is the first attempt to generate 3D solar coronal density distribution from the photospheric magnetic field using the deep-learning models. The most impressive thing is that the computation time is much faster than those of an MAS simulation. This study is organized as follows. The data are described in Section 2. The methods used for 3D coronal density distribution generation are discussed in Section 3. The results are given in Section 4. A brief summary and discussion are given in Section 5.

2. Data

The MAS model solves the time-dependent resistive thermodynamic MHD equations in 3D spherical coordinates to study the large-scale structure and dynamics of the solar corona (1 to 30 solar radii) and inner heliosphere (30 Rs to 5 au) (Riley et al. 2009). The MAS model considers a synoptic map of the photospheric magnetic field constructed from a series of observations over 27 days and centered at the central meridian region. The synoptic map is formed by the fundamental boundary condition at the surface of the Sun, which has been described previously in Mikić & Linker (1994), Lionello et al. (1999), Mikić & Linker (1999), Linker et al. (2001), and Lionello & Linker (2001). For this work, both the input and target data sets are collected from the MAS thermodynamic model. We consider the synoptic maps of the photospheric magnetic field data as input and the 3D distributions of coronal electron density up to 30 solar radii as the target. We collect the MAS simulation data from the MAS database (<http://www.predsci.com/data>).

The size of the input (photospheric magnetic field) data is 361×182 (longitude and latitude) from 2010 June to 2022 April (CR 2097 to CR 2256). The coronal electron density data of the MAS model have two different sizes for two different time periods: from 2010 June to 2020 May (CR 2097 to CR 2240), the size is $181 \times 101 \times 151$ (longitude, latitude, and altitude from the solar surface, respectively), and from 2020 June to 2022 April (CR 2241 to CR 2256), it is $300 \times 143 \times 255$. Since the size of the coronal electron density data of the MAS model for the two distinct periods is not the same, we align the coronal electron density data of these two different time periods and fix the size of our target data set as $182 \times 96 \times 151$ from 2010 June to 2022 April (CR 2097 to CR 2256). According to the size of the target, the input data (photospheric magnetic field data from MAS) are also aligned at the nearest latitude and longitude position of the target. After the alignment, the photospheric magnetic field data size is 182×96 . We consider the same alignment process for the photospheric magnetic field data for testing our deep-learning models.

In this study, we consider 155 CRs for training and testing. The deep-learning models are tested using 33 CRs after being trained on 122 CRs. To include the solar maximum and the solar minimum period in training and testing data sets, we consider three consecutive CRs for training and the fourth CR for testing from 2010 June to 2022 April (CR 2097 to CR 2256). Here 26% of the data are used to evaluate the deep-

learning models. As the number of our simulation data is not many, we extend the training data set approximately 30 times by flipping (longitude and latitude) and rotating by 10° in the longitudinal direction. Since the photospheric magnetic field data and coronal density map are asymmetric, we can obtain new magnetic field and density structures by flipping and rotating in order to provide a more diverse data set during training (Shorten & Khoshgoftaar 2019).

We normalize input data sets with mean and standard deviation values according to Equation (1). The main goal of the normalization is recentering and rescaling the data. In addition, it increases the efficiency and stability of deep-learning models by allowing them to train the deep-learning models for various data values at the same scale. Since the density data are not limited in their ranges and they decrease rapidly as the altitude increases, we normalize the density data at each solar radius with their mean values and standard deviation according to Equation (2). The formulae of the normalization are as follows:

$$\hat{X} = \frac{X - \text{mean}(X)}{\text{Std}(X)}, \quad (1)$$

$$\hat{Y}_i = \frac{Y_i - \text{mean}(Y_i)}{\text{Std}(Y_i)}, \quad (2)$$

where X, \hat{X} are the original and normalized input data, respectively. Y_i, \hat{Y}_i are the normalized and original electron density data of the i th solar radii, respectively.

Here we apply random noise and Gaussian blur only to the input data of the training data set. When we consider any ground-based photospheric magnetic field observation with artifacts like noise and blurriness as input, our deep-learning models may overcome the artifacts.

3. Method

The pix2pixHD (Wang et al. 2018) is an improved version of the pix2pix (Isola et al. 2017) algorithm for high-resolution data. It employs conditional GANs (cGANs) with multiscale discriminator and generator architectures, as well as novel adversarial loss function. As the resolution of our data is low, we consider a generator (G) that has U-Net (Ronneberger et al. 2015) architectures and a single discriminator (D), shown in Figure 1. It simultaneously trains D and G during the learning process (Mao et al. 2017). The primary objective of the generator G is to convert the inputs into outputs, and D triggers the generator to create realistic outputs (Wang et al. 2018).

During the training process, the generator extracts features from the input data and reconstructs the outputs from the extracted features. The discriminator works like a classifier to identify whether the output is fake or real. In our training, we apply the Least Squares Generative Adversarial Networks (LSGAN; Mao et al. 2017) instead of GAN to stabilize the training, as followed in Wang et al. (2018). LSGAN penalizes the fake samples and forces the generator to generate outputs closer to the decision boundary. Thus, LSGAN overcomes the vanishing gradients' problem and performs more stably (Mao et al. 2017).

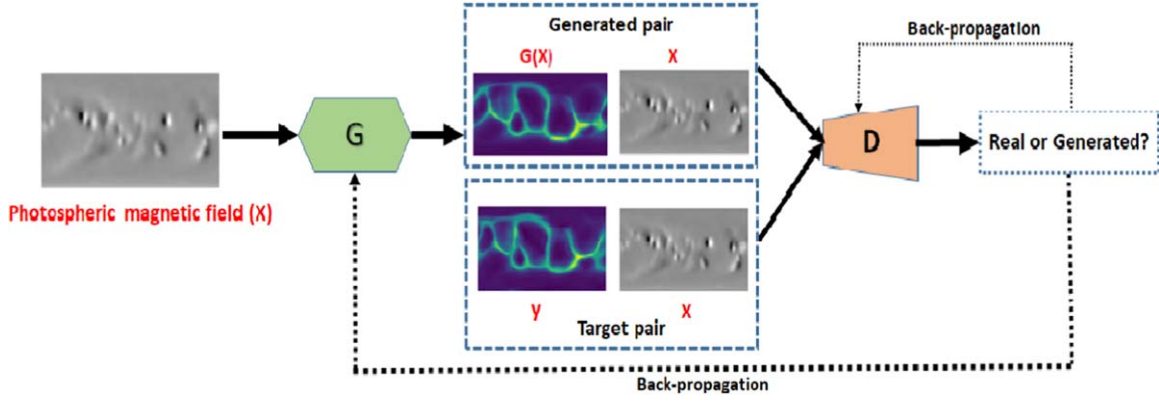


Figure 1. Architecture of our deep-learning model based on pix2pixHD.

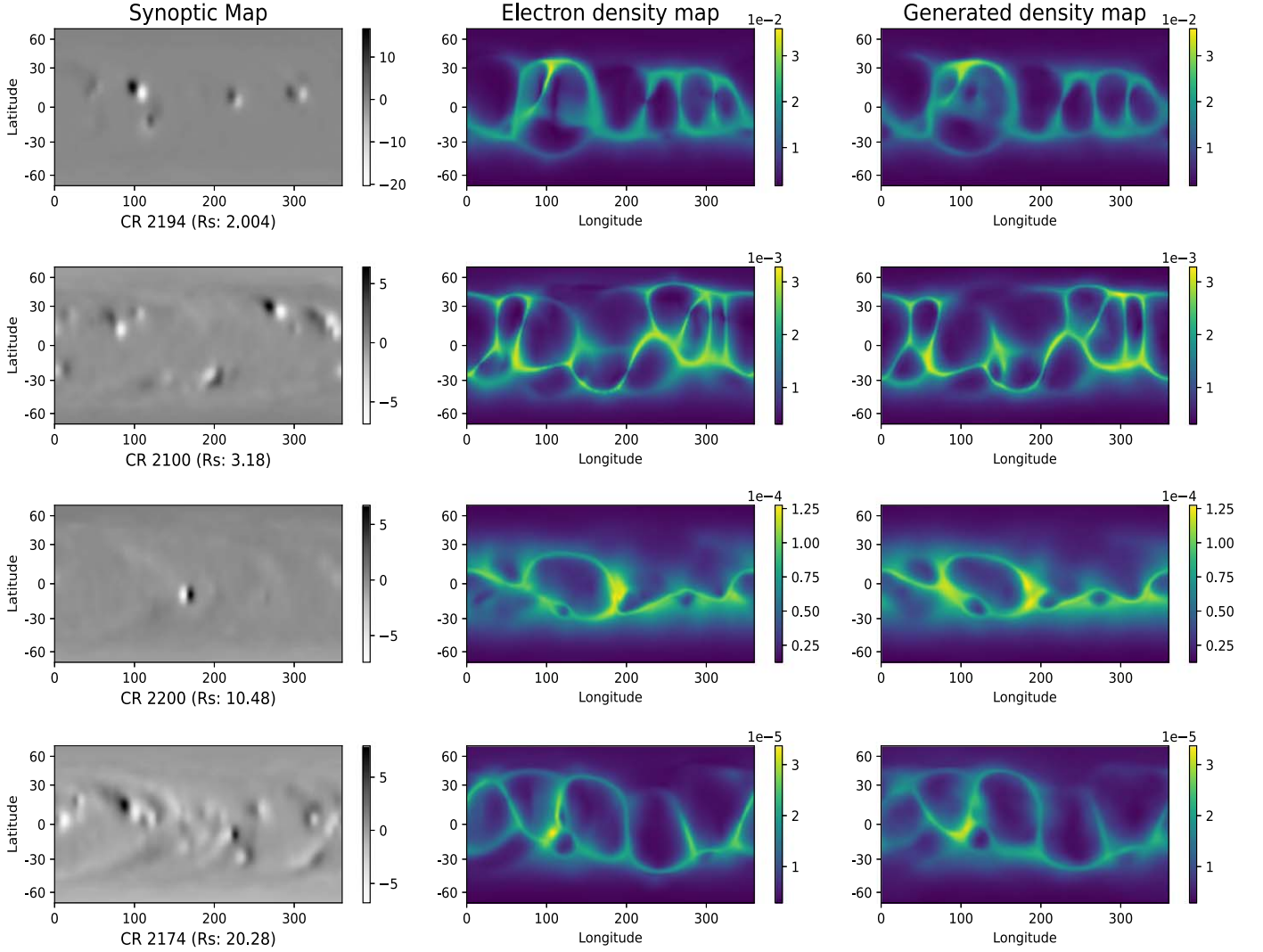


Figure 2. Qualitative comparisons between electron density distribution from MAS simulation and AI-generated electron density distribution with deep-learning models of CR 2194, CR 2100, CR 2200, and CR 2174 at 2.004, 3.18, 10.48, and 20.28 solar radii, respectively. Left column: synoptic map of the photospheric magnetic field; middle column: electron density distribution from MAS simulation model; right column: AI-generated electron density distribution.

The objective function of LSGAN is as follows:

$$\mathcal{L}_{\text{LSGAN}}^D(G, D) = \frac{1}{2}[(D(x, y) - y)^2] + \frac{1}{2}[(D(x, G(x)))^2], \quad (3)$$

$$\mathcal{L}_{\text{LSGAN}}^G(G, D) = \frac{1}{2}[(D(x, G(x)) - y)^2], \quad (4)$$

where x and y denote input (photospheric magnetic field) and target (coronal electron density) data, respectively. $G(x)$ means

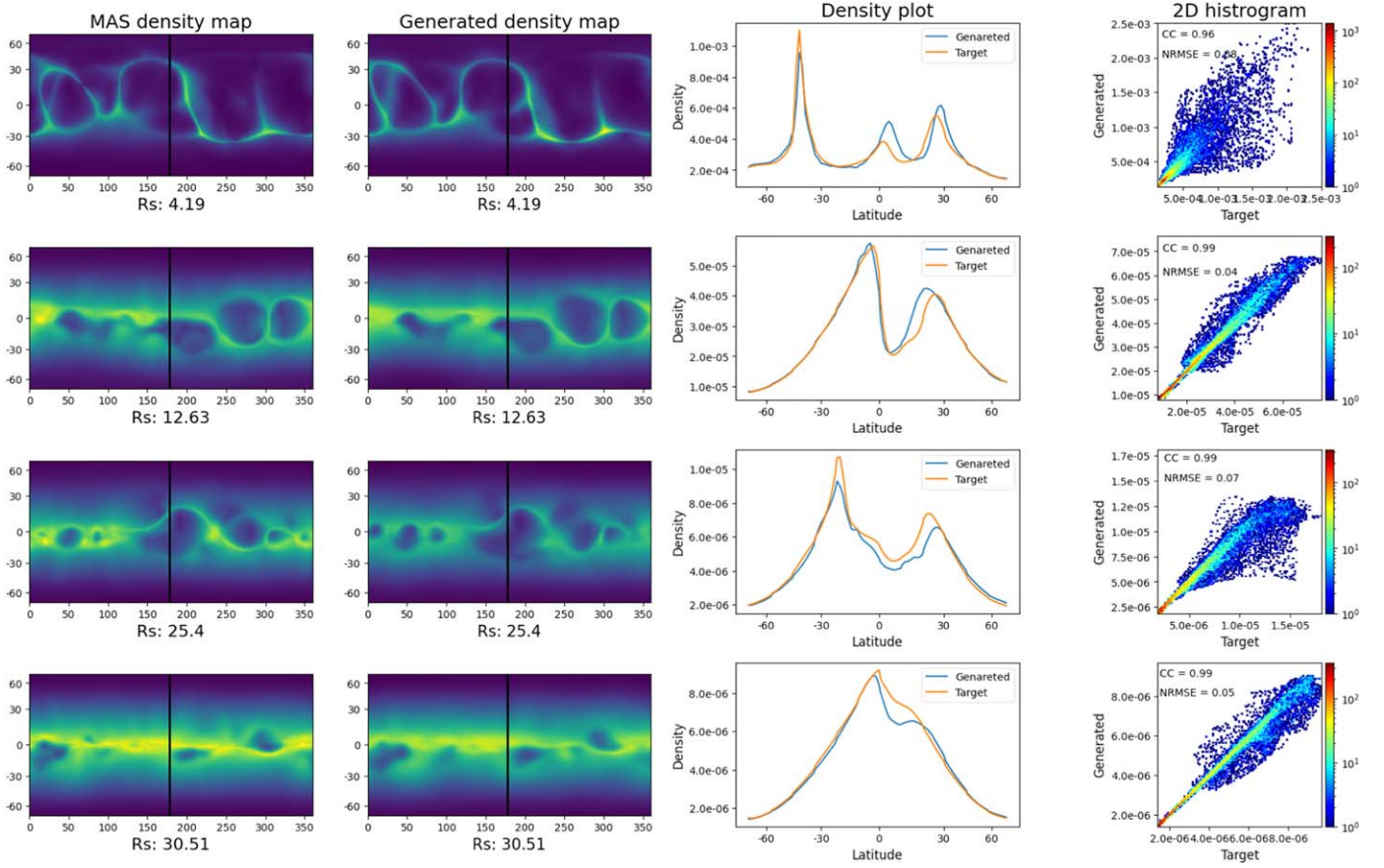


Figure 3. Two descriptive comparisons of electron density map, density graph in 180° longitude, and 2D histogram plots between the target and the AI-generated one at 4.19, 12.63, 25.40, and 30.51 radii solar radii of CR 2178, CR 2220, CR 2205, and CR 2228, respectively. Left to right: MAS density map, AI-generated density map, density graph in 0° longitude, and 2D histograms.

an output from the generator for a given input x . $D(x, y)$ and $D(x, G(x))$ are probabilities of the target pairs and AI-generated pairs, respectively.

To enable better learning, an additional loss function FM, known as “feature matching loss,” is added to the generator (Wang et al. 2018). FM loss is more effective for large dynamic range data because it reduces the absolute difference between the feature maps of the target data and AI-generated data (Rana et al. 2019; Marnierides et al. 2021). The FM loss function is expressed as follows:

$$\mathcal{L}_{\text{FM}}(G, D) = \sum_{i=1}^T \frac{1}{N_i} \| [D^{(i)}(x, y) - D^{(i)}(x, G(x))] \|, \quad (5)$$

where T denotes the total number of layers in the discriminator and N_i is the number of pixels in output feature maps of each layer.

The final objective of the deep-learning model is as follows:

$$\min_G (\mathcal{L}_{\text{LSGAN}}^G(G, D) + \lambda \mathcal{L}_{\text{FM}}(G, D)), \quad (6)$$

$$\min_D \mathcal{L}_{\text{LSGAN}}^D(G, D), \quad (7)$$

where G and D denote a generator and a discriminator, respectively, and $\mathcal{L}_{\text{GAN}}(G, D)$ and $\mathcal{L}_{\text{FM}}(G, D)$ are GAN loss

and FM loss, respectively. Here λ is a relative weight that controls $\mathcal{L}_{\text{LSGAN}}^G$ and \mathcal{L}_{FM} following Wang et al. (2018).

During the learning process, the generator tries to minimize the FM loss, while the discriminator tries to maximize the LSGAN loss. The goal of the loss function is to optimize the deep-learning model parameters. For training and testing, we consider the photospheric magnetic field (x) as input and 3D coronal density distribution (y) as output from the MAS simulation as shown in Figure 1. To cover whole coronal regions from 2.004 to 30 solar radii, we consider 152 deep-learning models separately. We compare the AI-generated ones with the MAS-simulated data, as well as SOHO (Domingo et al. 1995)/LASCO (Brueckner et al. 1995) C3 coronagraph data. We employ the Adam (Kingma & Ba 2014) optimizer with momentum $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The initial learning rate is 2×10^{-4} . In every 2×10^5 iterations of back-propagation, the learning rate is decreased to half from the previous label.

4. Results and Discussion

4.1. Qualitative and Quantitative Evaluation

A comparison between the results of our deep-learning models and the MAS simulation at 2.004, 3.18, 10.48, and 20.28 solar radii is shown in Figure 2. It seems from the results that our deep-learning models can reliably generate the electron density map of the solar minimum cases (CR 2100 and

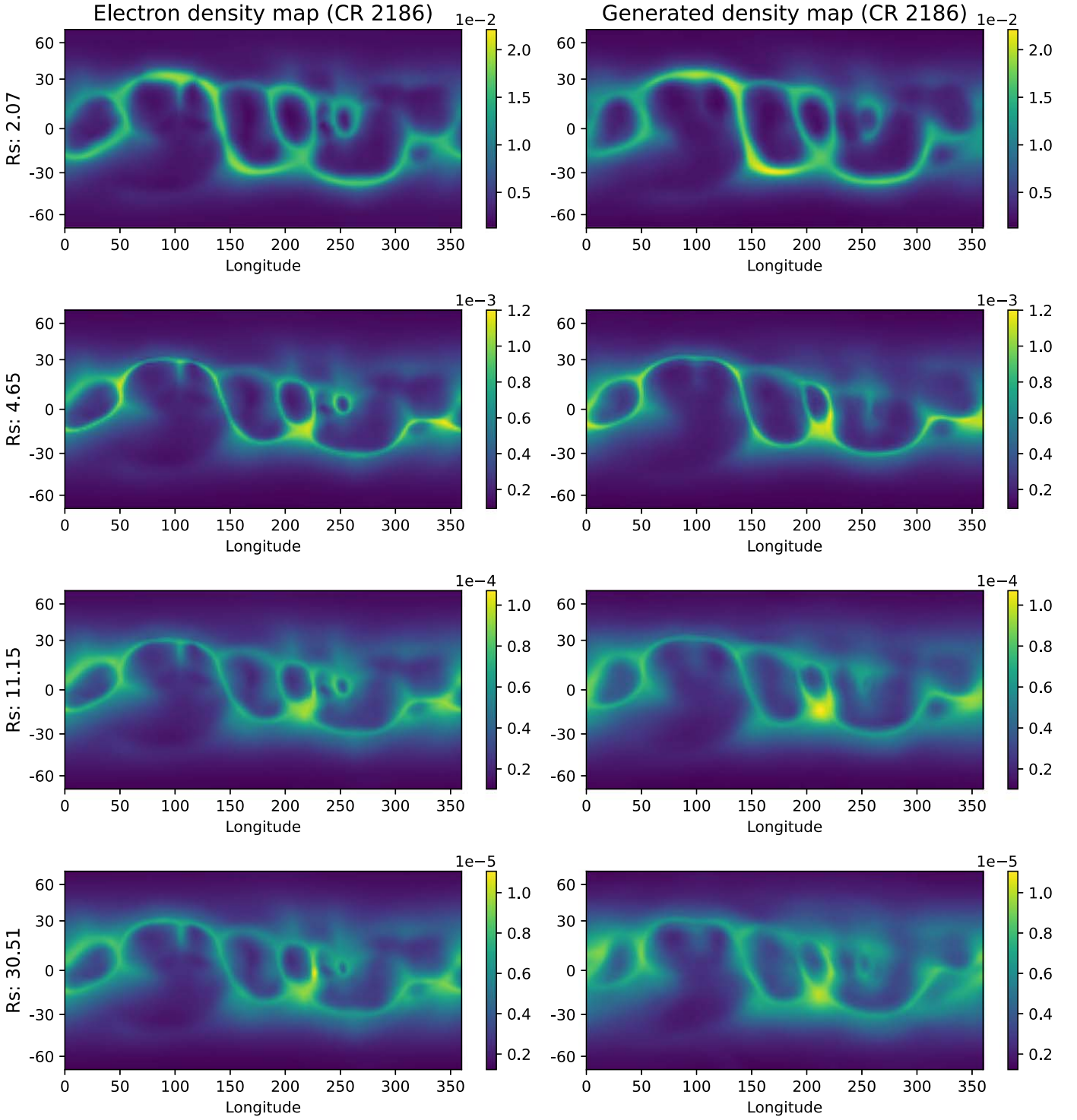


Figure 4. Comparisons of electron density distribution from the MAS simulation and AI-generated electron density map of CR 2186 at 2.07, 4.65, 25.40, and 30.51 solar radii. The left column represents the MAS density map, and the right column represents the AI-generated one.

CR 2174), as well as the solar maximum cases (CR 2194 and CR 2200).

Figure 3 shows a comparison of electron density maps, density graph in 180° longitude, and 2D histogram plots between the target and the AI-generated ones at 4.19, 12.63, 25.40, and 30.51 radii, respectively. As shown in Figure 3, both the target and AI-generated density are quite consistent with each other. Even at higher solar radii, the histograms accurately depict pixel distributions around the diagonal. We notice that

just a small fraction of points are deviated from the line, while the rest of the points are near the diagonal line. Next, we compare the MAS density maps with the AI-generated density maps at 2.07, 4.65, 25.40, and 30.51 radii of CR 2186 in Figure 4. From the color bar of both the columns, we can see that the electron density varies with radius. It is impressive that our deep-learning models are able to reproduce the coronal electron density variation over radius. Although the data have a fairly complex structure, the deep-learning models can

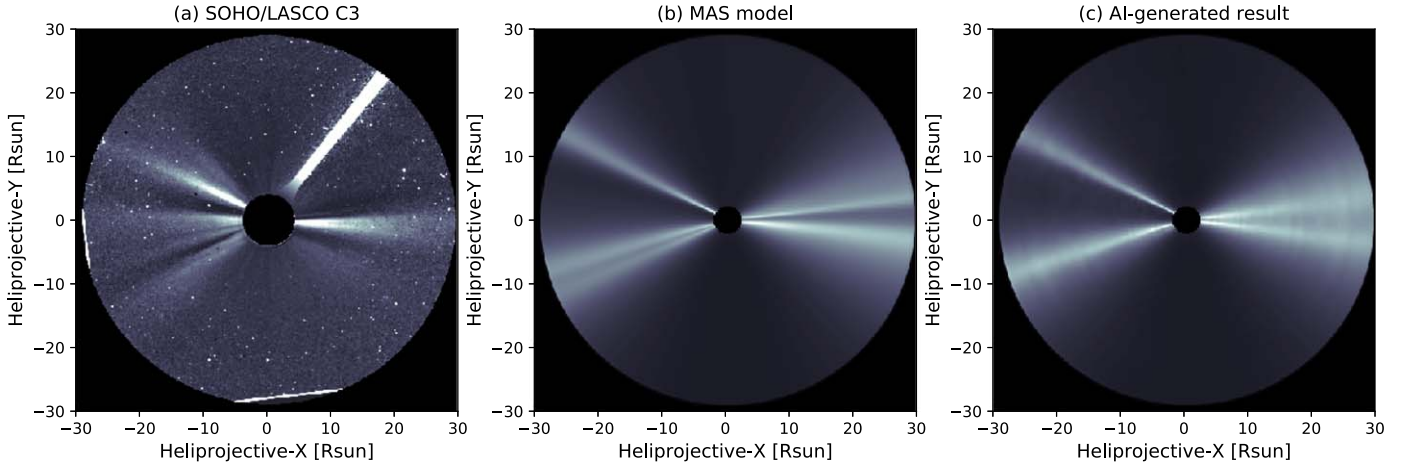


Figure 5. Comparison of (a) SOHO/LASCO C3 coronagraph image observed on 2017 May 13, (b) the synthetic image based on the MAS simulation model, and (c) the AI-generated one.

Table 1

The Averaged Values of Pearson’s Correlation Coefficient (CC) and Normalized rms Error (NRMSE) of the Solar Minimum and the Solar Maximum Period at 2.004, 3.12, 10.48, and 20.28 Solar Radii for the Testing Data Set

Radius	Solar Minimum		Solar Maximum	
	Pixel-to-pixel CC	NRMSE	Pixel-to-pixel CC	NRMSE
2.004	0.99	0.080	0.97	0.10
3.12	0.98	0.085	0.97	0.12
10.48	0.99	0.078	0.98	0.13
20.28	0.99	0.076	0.98	0.12

qualitatively reproduce the overall 3D density structure. Additionally, our AI-generated results can be employed to infer the global coronal magnetic field topology since the density structures, like the heliospheric current sheet, manifest the magnetic structures.

In order to evaluate the testing data sets quantitatively, we took two types of objective metrics for solar maximum and solar minimum period separately. The first metric is a pixel-to-pixel Pearson’s correlation coefficient between the target and the generated electron density map. The second metric is the normalized rms error (NRMSE). It is given by

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y - y')^2}, \quad (8)$$

$$\text{NRMSE} = \frac{\text{RMSE}}{y_{\max} - y_{\min}}, \quad (9)$$

where N , y , and y' are the number of pixels in a single testing data, target, and AI-generated coronal electron density distribution, respectively. Here y_{\max} and y_{\min} are the maximum and minimum electron density value from the MAS model of the corresponding solar radius, respectively.

Table 1 represents the averaged pixel-to-pixel Pearson’s correlation coefficient and NRMSE values between the density distribution obtained from the MAS model and the AI-generated

ones at 2.004, 3.18, 10.48, and 20.28 solar radii for the testing data set. As shown in Table 1, the pixel-to-pixel correlation of our deep-learning model is higher during the solar minimum than the solar maximum. However, due to the more complex structure during the solar maximum period, the solar minimum period has better metrics values.

4.2. Computational Time

To predict adverse space weather events and their impact on the geospace environment in near real time, high-performance computational and shorter-running time models are needed (Feng et al. 2013). In our study, we use a single NVIDIA TITAN XP graphics-processing unit (GPU), CUDA 11.4, and 12 GB. For the computing time evaluation, an AMD Ryzen 5 3600 6-Core Processor with 16.0 GB RAM CPU is used for loading and preprocessing of inputs, while a single GPU is used for forwarding inference. The most impressive result is that it takes approximately 45.2 s to compute 152 deep-learning models for the test data set with a resolution of $182 \times 96 \times 152$ up to 30 solar radii, which is much faster than a usual computational time on the MAS model (Caplan & Linker 2019). On the other hand, the computing time of the MAS model with a resolution of $101 \times 101 \times 128$ is about several days on a single-CPU machine (Mikić et al. 2018).

4.3. Comparison with SOHO/LASCO C3 Data

Figure 5 shows the comparison among the SOHO/LASCO C3 coronagraph image on 2015 May 15 (solar minimum case), the corresponding synthetic image based on the MAS model, and the AI-generated result of our deep-learning models. We construct the synthetic images using the densities near the limb of the solar frontside from 2.004 to 30 solar radii as seen by SOHO/LASCO. The LASCO C3 and synthetic images are processed using the normalizing radial graded filter, as described by Morgan et al. (2006), in order to reveal faint coronal structures. It seems that the AI-generated ones are quite similar to the MAS simulation and the observation in view of the position of polar regions and other structures.

Figure 6 shows the comparison among the SOHO/LASCO C3 coronagraph image on 2012 March 17, the synthetic image based on the MAS model, and the one AI-generated by our deep-learning models. The figure shows that the global

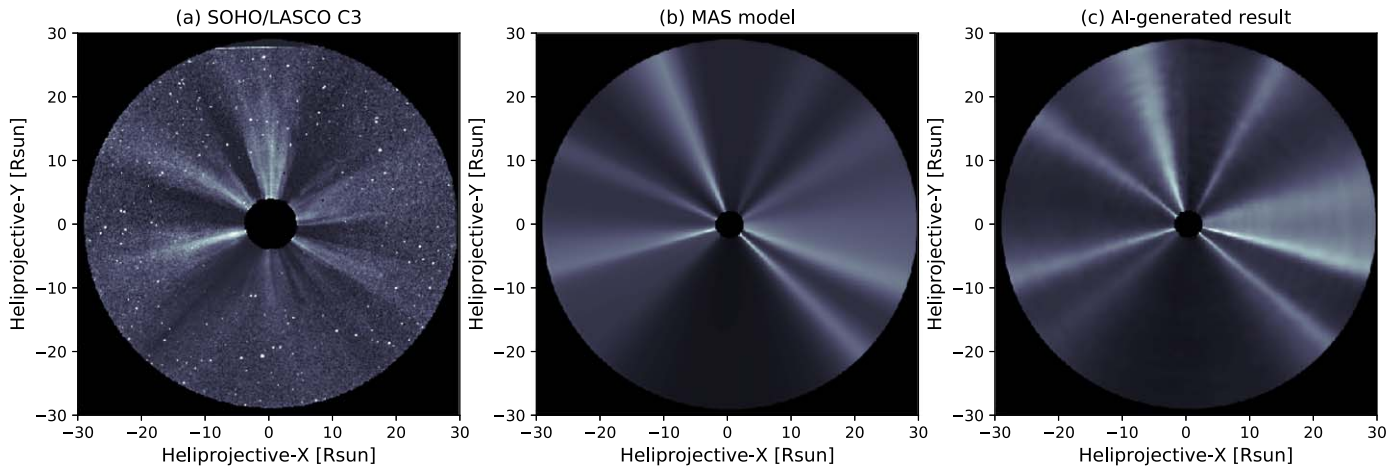


Figure 6. Comparison result with (a) SOHO/LASCO C3 coronagraph observed on 2012 March 17, (b) the synthetic image based on the MAS simulation model, and (c) the one AI generated by our deep-learning models during the solar maximum period.

configurations are similar but not exactly the same. We found few differences between the observed SOHO/LASCO C3 coronagraph image and AI-generated one. Such differences may be caused by synoptic magnetic fields used as input data of the MAS model. It is noted that these fields had been taken for one solar rotation period.

5. Conclusion and Summary

Our study is the first attempt to apply 152 deep-learning models to generate the 3D solar coronal electron density distribution up to 30 solar radii, whose computing times are much shorter than the MAS simulation one. For this work, we obtained the input photospheric magnetic field map and target solar coronal density distribution from the MAS simulation model collected from Predictive Science Inc. We used 155 pairs of data from CR 2097 to CR 2256 for training and testing our deep-learning models, which cover the time period from 2010 June to 2022 April.

Based on the outcomes of this study, we conclude that advanced deep-learning models can successfully generate the 3D solar coronal electron density distribution. Those results can be potentially used for specifying the condition on global magnetohydrodynamic models to reproduce large-scale coronal, heliospheric structures and solar wind modeling, such as the MAS heliospheric models (Lionello & Linker 2009) and the Wang–Sheeley–Arge model (Arge et al. 2004). Furthermore, deep-learning models can generate even small-scale density structures for both solar minima and maxima periods. We demonstrate that both solar maxima and minima AI-generated results are strongly correlated (the average pixel to pixel CC 0.97) with the target MAS-simulated electron density distribution map. In addition, the NRMSE values show that the deep-learning models are quite effective at reconstructing the electron density distribution map. Moreover, the intensity map and the histograms seem to be consistent not only in lower solar radii but also in higher solar radii. The impressive advantage of this work is that the computational time of the deep-learning models is much faster than the MAS simulation one. Finally, the synthetic coronagraphic data estimated from the models matched with the real coronagraph image of SOHO/LASCO data during the solar minimum period.

The deep-learning algorithm can generate complex and higher-radius electron density structures. The obtained AI-generated results can provide valuable insight for investigating and comparing physical aspects of the data from spacecraft in the inner heliosphere like Solar Orbiter (Müller et al. 2020) and Parker Solar Probe (Vourlidas et al. 2016). Furthermore, on a near-real-time basis, the AI-generated coronal density distribution can be used for reliable input for future space weather simulations. We intend to use more improved synchronic magnetic field data from the Solar Dynamics Observatory (SDO; Pesnell et al. 2012)/Helioseismic and Magnetic Imager (HMI; Schou et al. 2012) and AI-generated farside magnetograms as input data in the future to overcome the limitation of the MAS for the solar maximum period Jeong et al. (2022). In the future, we have plan to generate other parameters of the MAS simulation model, like magnetic fields, temperature, and velocities which can be reproduced using the deep learning algorithm.

This research was supported by the Korea Astronomy and Space Science Institute under the R&D program (Project No. 2023-1-850-07) supervised by the Ministry of Science and ICT and NRF-2020R1C1C1003892 project. We thank the numerous team members who have contributed to the MAS simulation. We acknowledge the community effort devoted to developing the following open-source packages used in this work. We thank the numerous team members who have contributed to the success of the SDO mission. The SDO data were (partly) provided by the Korea Data Center (KDC) for SDO in cooperation with NASA, Stanford University (JSOC), and KISTI (KREONET), which is supported by the "Next Generation Space Weather Observation Network" project of the Korea Astronomy and Space Science Institute (KASI).

Software: SunPy (The SunPy Community et al. 2020), PyTorch (Paszke et al. 2019), NumPy (Harris et al. 2020).

ORCID iDs

Seunghoon Shin <https://orcid.org/0000-0002-5144-8230>
 Hyun-Jin Jeong <https://orcid.org/0000-0003-4616-947X>
 Ashraf Siddique <https://orcid.org/0000-0003-2186-5735>
 Yong-Jae Moon <https://orcid.org/0000-0001-6216-6944>

Eunsu Park  <https://orcid.org/0000-0003-0969-286X>
 Jihye Kang  <https://orcid.org/0000-0001-6213-4088>
 Sung-Ho Bae  <https://orcid.org/0000-0003-2677-3186>

References

- Arge, C., Luhmann, J., Odstreil, D., et al. 2004, *JASTP*, **66**, 1295
 Brueckner, G., Howard, R., Koomen, M., et al. 1995, *SoPh*, **162**, 357
 Caplan, R. M., Linker, J. A., Mikić, Z., et al. 2019, *JPhCS*, **1225**, 012012
 de Patoul, J., Foullon, C., & Riley, P. 2015, *ApJ*, **814**, 68
 Detman, T., Smith, Z., Dryer, M., et al. 2006, *JGRA*, **111**, A07102
 Domingo, V., Fleck, B., & Poland, A. I. 1995, *SoPh*, **162**, 1
 Feng, L., Inhester, B., & Gan, W. 2015, *ApJ*, **805**, 113
 Feng, X., Wang, H., Xiang, C., et al. 2021, *ApJS*, **257**, 34
 Feng, X., Zhong, D., Xiang, C., & Zhang, Y. 2013, *ScChD*, **56**, 1864
 Gómez, J. R., Vieira, L., & Dal, L. 2018, *ApJ*, **852**, 137
 Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al. 2014, *Commun. ACM*, **63**, 139
 Harris, C. R., Millman, K. J., Van Der Walt, S. J., et al. 2020, *Natur*, **585**, 357
 Hayashi, K. 2005, *ApJS*, **161**, 480
 Isola, P., Zhu, J.-Y., Tinghui, E., & Efros, A. A. 2017, in Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 2017 (Piscataway, NJ: IEEE), 1125
 Jang, S., Kwon, R.-Y., Linker, J. A., et al. 2021, *ApJL*, **920**, L30
 Jeong, H.-J., Moon, Y.-J., Park, E., et al. 2020, *ApJL*, **903**, L25
 Jeong, H.-J., Moon, Y.-J., Park, E., et al. 2022, *ApJS*, **262**, 50
 Kim, T., Park, E., Lee, H., et al. 2019, *NatAs*, **3**, 397
 Kingma, D. P., & Ba, J. 2014, arXiv:1412.6980
 Linker, J. A., Lionello, R., Mikić, Z., et al. 2001, *JGR*, **106**, 25165
 Linker, J. A., Mikić, Z., & Biesecker, D. A. 1999, *JGR*, **104**, 9809
 Lionello, R., Linker, J. A., & Mikić, Z. 2001, *ApJ*, **546**, 542
 Lionello, R., Linker, J. A., & Mikić, Z. 2009, *ApJ*, **690**, 902
 Lionello, R., Mikić, Z., & Linker, J. A. 1999, *JCoPh*, **152**, 346
 Mao, X., Li, Q., Xie, H., et al. 2017, in Proc. of the IEEE Int. Conf. on Computer Vision (Piscataway, NJ: IEEE), 2794
 Marnierides, D., Bashford-Rogers, T., & Debattista, K. 2021, *Senso*, **21**, 4032
 Mikić, Z., Downs, C., Linker, J., et al. 2018, *NatAs*, **2**, 913
 Mikić, Z., & Linker, J. A. 1994, *ApJ*, **430**, 898
 Mikić, Z., Linker, J. A., & Schnack, D. D. 1999, *PhPl*, **6**, 2217
 Morgan, H., Habbal, S. R., & Woo, R. 2006, *SoPh*, **236**, 263
 Müller, D., Cyr, O. S., Zouganelis, I., et al. 2020, *A&A*, **642**, A1
 Park, E., Moon, Y.-J., Lee, J.-Y., et al. 2019, *ApJL*, **884**, L23
 Park, E., Moon, Y.-J., Lim, D., et al. 2019, *ApJL*, **891**, L4
 Paszke, A., Gross, S., Massa, F., et al. 2019, Advances in Neural Information Processing Systems 32 (Red Hook, NY: Curran Associates), <https://papers.nips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>
 Pesnell, W. D., Thompson, B. J., & Chamberlin, P. C. 2012, *SoPh*, **275**, 3
 Rana, A., Singh, P., Valenzise, G., et al. 2019, *ITIP*, **29**, 1285
 Riley, P., Linker, J., Lionello, R., et al. 2009, Proc. Solar Heliospheric and Interplanetary Environment Conf. (SHINE 2009), 38
 Riley, P., Linker, J., Lionello, R., et al. 2012, *JASTP*, **83**, 1
 Ronneberger, O., Fischer, P., & Brox, T. 2015, in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th Int. Conf. (Cham: Springer), 234
 Schou, J., Scherrer, P. H., Bush, R. I., et al. 2012, *SoPh*, **275**, 229
 Shin, G., Moon, Y.-J., Park, E., et al. 2020, *ApJL*, **895**, L16
 Shorten, C., & Khoshgoftaar, T. M. 2019, *J. Big Data*, **6**, 1
 The SunPy Community, Barnes, W. T., Bobra, M. G., et al. 2020, *ApJ*, **890**, 68
 Usmanov, A. 1998, *SoPh*, **146**, 377
 Usmanov, A., & Goldstein, M. 2003, in AIP Conf. Proc. 679, Solar Wind Ten (Melville, NY: AIP), 393
 Vourlidas, A., Howard, R., & Plunkett, A. 2016, *SSRv*, **204**, 83
 Vourlidas, A., Subramanian, P., Dere, K. P., et al. 2000, *ApJ*, **534**, 456
 Wang, T.-c., Liu, M.-Y., Jun-Yan, T., et al. 2018, in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (Piscataway, NJ: IEEE), 8798