



3D Solar Coronal Loop Reconstructions with Machine Learning

Iulia Chifu^{1,2} and Ricardo Gafeira^{3,4} ¹ Max-Planck-Institut für Sonnensystemforschung, Justus-von-Liebig-Weg 3, 37077 Göttingen, Germany; chifu@mps.mpg.de² Astronomical Institute of Romanian Academy, Căminul de Argint 5, Bucharest, Romania³ Univ Coimbra, IA, Department of Physics, Coimbra, Portugal, Rua do Observatório s/n, 3040-004 Coimbra, Portugal⁴ Instituto de Astrofísica de Andalucía (CSIC), Apartado de Correos 3004, E-18080 Granada, Spain

Received 2021 February 23; revised 2021 March 8; accepted 2021 March 10; published 2021 March 23

Abstract

The magnetic field plays an essential role in the initiation and evolution of different solar phenomena in the corona. The structure and evolution of the 3D coronal magnetic field are still not very well known. A way to ascertain the 3D structure of the coronal magnetic field is by performing magnetic field extrapolations from the photosphere to the corona. In previous work, it was shown that by prescribing the 3D-reconstructed loops' geometry, the magnetic field extrapolation produces a solution with a better agreement between the modeled field and the reconstructed loops. This also improves the quality of the field extrapolation. Stereoscopy, which uses at least two view directions, is the traditional method for performing 3D coronal loop reconstruction. When only one vantage point of the coronal loops is available, other 3D reconstruction methods must be applied. Within this work, we present a method for the 3D loop reconstruction based on machine learning. Our purpose for developing this method is to use as many observed coronal loops in space and time for the modeling of the coronal magnetic field. Our results show that we can build machine-learning models that can retrieve 3D loops based only on their projection information. Ultimately, the neural network model will be able to use only 2D information of the coronal loops, identified, traced, and extracted from the extreme-ultraviolet images, for the calculation of their 3D geometry.

Unified Astronomy Thesaurus concepts: [Solar coronal loops \(1485\)](#); [Convolutional neural networks \(1938\)](#)

1. Introduction

Coronal loops are 3D arch-like structures with footpoints located in the opposite polarity of the magnetic field. In the solar corona, the plasma pressure is dominated by the magnetic pressure (Gary 2001), which confines the coronal loop plasma to the 3D flux tubes. The coronal loops are mainly visible on the solar disk in the extreme-ultraviolet (EUV) images, which shows only the projection of the 3D structures. The reconstruction of the 3D shape of coronal loops has important implications for the magnetic field and topology (Reale 2010).

The magnetic field plays an important role in the initiation and evolution of different solar phenomena in the corona. While at the photospheric level full-disk vector magnetic field measurements are provided by several observatories, in the chromosphere and corona the vector magnetic field is only sometimes provided and only for small fields of view. Different approaches are used to obtain the coronal magnetic field. One of these approaches is based on magnetic field extrapolations into the corona having as bottom boundary photospheric magnetic field data (see Wiegmann & Sakurai 2012; Wiegmann et al. 2017, for reviews). Another approach is a stereoscopic reconstruction based on simultaneously recorded images from at least two view directions in the EUV wavelengths (chromosphere and low corona) and white light (high corona; see Aschwanden 2011 for a review). When applied to the 3D reconstruction of the coronal magnetic loops, the output of the two methods does not match (De Rosa et al. 2009). Chifu et al. (2017) showed that by constraining the nonlinear force-free field (NLFFF) extrapolations with the 3D

stereoscopically reconstructed loops, one can obtain a better field model.

The 3D stereoscopic reconstruction method requires images of the same object recorded from at least two view directions. This requirement limits the time range for finding the 3D information on solar coronal loops. One of the steps for stereoscopy is the identification and matching of the same object in the available images (Inhester 2006). This condition limits the possibility of selecting a large number of coronal loops.

With the method laid out in this work, we can make use of all of the epochs at which EUV images and vector magnetograms simultaneously exist. The 2D loops that one would be able to identify, visually or automatically, in an EUV image will be 3D-reconstructed by the neural network (NN) method. The resulting 3D loops can then be used as a constraint for the magnetic field extrapolation, as shown by Chifu et al. (2017). This method allows us to use many more loops as a constraint for extrapolation than had been previously possible. The method that we present here for the 3D coronal loop reconstruction constrains the coronal magnetic field models to fit the observations. These coronal magnetic field models can then be used by magnetohydrodynamic (MHD) models to investigate the heating mechanisms of coronal loops.

2. Methods

2.1. Convolution NN

NNs are designed to work like the processes inside a human brain. In this context, what we call a neuron in the brain is a mathematical function that infers quantities based on the input data according to the NN architecture.

Among the various different NN architectures, there is a subset called the convolutional neural network (CNN). These

Table 1
CNN Model Architecture Used in this Work

Layer	Size	Activation
1D convolutional	150	ReLU
1D MaxPooling	2	
1D convolutional	90	ReLU
1D MaxPooling	2	
1D convolutional	45	ReLU
Dropout	factor of 0.25	
Fully connected dense layer	4 times the loop length	ReLU
Fully connected dense layer	loop length length	linear

networks use convolutional layers where 1D or 2D filters are applied to the input data. The unidimensional (1D) CNN basic structure used in this work can be then resumed as a combination of the following layers: (1) layers that apply a convolution between a kernel and each set of data points (in our case, loop position), with the goal of enhancing particular features in the data; (2) a pooling layer that downsamples the data by reducing the data information but retains the more prominent features; (3) the so-called “fully connected layers,” which take the results of the previous operations and multiply them by a set of weights to estimate the final output. Most of the operations performed in these layers have an associated activation function, which are mathematical equations that determine the importance of a certain variation.

One key element of any NN is the training process in which an input data set and its respective output data are used. This allows the model to determine the weights that the fully connected layers will apply. The network employs an optimization procedure to minimize the differences, using a loss function, between the training set and the estimated output. It is common practice to use a fraction of the input data not used for the training and compare it to the model output. This is called validation loss, and it allows us to evaluate the evolution of the training and identify problems like overfitting. To optimize the minimization process, we can divide the data into small batches. During the training, the weights will be updated at the end of each batch, leading to a better and faster convergence. Overfitting can be avoided by introducing dropout elements that randomly remove a fraction of the points from the training process, introducing a level of randomness to the system and helping it to avoid false local minima. “Epoch” refers to each full iteration of the training process described above.

In this work, we follow the CNN architecture model presented on Gafeira et al. (2021). We optimized the hyperparameters by trial and error until we found a CNN model satisfying a loss function that was at least 10^3 of the normalization values to all considered cases. We trained the model using the optimizer AdaMax, which is a first-order gradient-based optimization of stochastic objective functions (Kingma & Ba 2014). We used a batch size of 64 and a validation split of 0.3. The convolutional layers and the first dense layer use a nonlinear function called the rectified linear activation unit (ReLU; Nair & Hinton 2010) and the last dense layer a linear function. The global structure of the CNN and the respective hyperparameters used in this work are summarized in Table 1.

The input parameters are the 2D position of the loops (X , Y), the length of the loop, the distance between the footpoints of

the loops, and the angle defined by the two footpoints with the top of the loops. The 2D positions X and Y are interpolated to all have a fixed size of 1500 points.

The output and training set of the CNN is the Z component of the corresponding 2D loop with a fixed size of 1500 points.

We applied a normalization factor that corresponds to the loop average height used in the training of CNN. In this work the normalization used is 8.75 ADU for Data type 1, and for Data type 2 Case 1 is 10.1 Mm, Case 2 is 77.54 Mm, and Case 3 is 73.15 Mm.

2.2. Nonlinear Force-free Field Extrapolation

To produce a data set that is large enough for training the CNN we used the NLFFF extrapolation method, proposed initially by Wheatland et al. (2000) and implemented by Wiegelmann (2004). The NLFFF optimization method is based on the minimization of a functional L_{tot} as a sum of the following terms:

$$L_1 = \int_V w_f \frac{|\nabla \times \mathbf{B}|^2}{B^2} d^3r, \quad (1)$$

$$L_2 = \int_V w_f |\nabla \cdot \mathbf{B}|^2 dr^3, \quad (2)$$

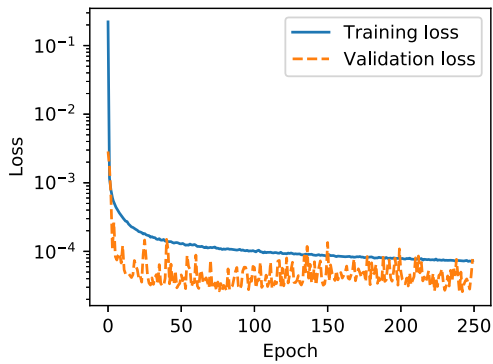
$$L_3 = \int_S (\mathbf{B} - \mathbf{B}_{\text{obs}}) \cdot \text{diag}(\sigma_\alpha^{-2}) \cdot (\mathbf{B} - \mathbf{B}_{\text{obs}}) d^2r, \quad (3)$$

where the minimization of L_1 satisfies the force-free condition and the L_2 satisfies the Gauss Law. The minimization of the L_3 ensures that at the photospheric level the modeled field (\mathbf{B}) is as close as possible to the observed one (\mathbf{B}_{obs}). w_f is the boundary weight function (more details in Wiegelmann 2004), $\text{diag}(\sigma_\alpha^{-2})$ represents the estimated measurement errors for the three field component (see Tadesse et al. 2011, for more details). The method was implemented in the Cartesian (Wiegelmann 2004) and spherical (Tadesse et al. 2011) coordinate system. Both of the implementations use as a bottom boundary the vector magnetic field. The lateral and top boundaries are fixed from an initial calculated potential field.

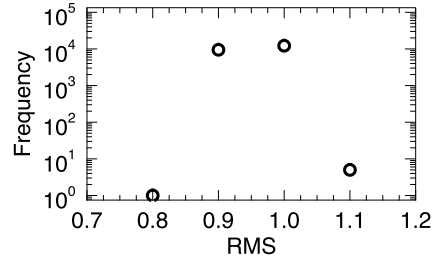
3. Data Preparation

We tested the CNN method in four different cases by using input loops obtained from different solutions of the NLFFF extrapolation. The first case uses synthetic data, and the last three are based on observational data.

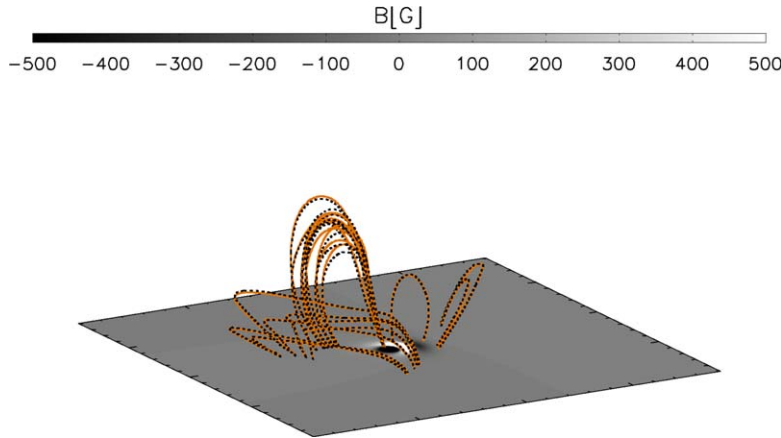
1. Data type 1. The NLFFF extrapolation uses as bottom boundary a semi-analytical force-free field solution proposed by Low & Lou (1990). The solution is obtained by solving the Grad–Shafranov equation for an axisymmetric nonlinear force-free field. The results obtained based on the synthetic data are defined as Case 0.
2. Data type 2. In this category are the data obtained by the Heliospheric Magnetic Imager (HMI) instrument (Couvidat et al. 2012) on board Solar Dynamic Observatory (SDO) spacecraft (Pesnell et al. 2012). We defined three Cases:
 - i. Case 1. We use as input for the NLFFF a field of view of 1280×400 from a full-disk HMI vector magnetogram.



(a) Display of the training (blue solid line) and validation (orange dashed line) loss function during the CNN training.



(b) Distribution of the root mean square for Case 0 loops.



(c) 3D CNN reconstructed loops with solid orange lines and original 3D loops with black dashed lines overplotted on the Low & Lou bottom boundary.

Figure 1. CNN results for Data type 1, Case 0.

- ii. Case 2. For the NLFFF extrapolation method in spherical coordinates, we used as bottom boundary a full-disk vector magnetogram recorded on 2010 August 2.
- iii. Case 3. For the last case, we use synoptic vector magnetograms, which are HMI data products suitable for the full Sun extrapolation (Liu et al. 2017). These maps are constructed from the daily HMI vector magnetograms. The size of the HMI synoptic magnetograms is 3600×1440 pixels and the pixel size is 0.1 degrees in longitude and 0.001 in sin latitude.

From the solutions of the extrapolations, using a fourth-order Runge–Kutta method, we traced 3D magnetic field lines and we projected them on the solar surface.

4. Results

4.1. Data Type 1

Case 0. We tested the CNN method described in Section 2.1 with 3D and 2D loops obtained from an NLFFF extrapolation, which used as a bottom boundary a Low & Lou semi-analytical force-free field solution. We used the Cartesian coordinate system for a computational box in the X , Y , and Z -direction of $192 \times 192 \times 96$ grid points. For the training of the CNN, we used 21,704 loops traced from the NLFFF solution.

We trained the CNN initially using 250 epochs. After the training, we evaluated the result of the training and validation

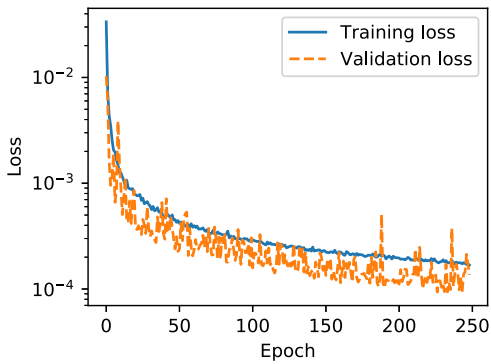
loss function. Since the model has converged after 250 epochs, we stopped the training process. In the presented model we did not find any indication of overfitting. That would be visible as a sudden increase of the validation loss function. In Figure 1(a) we display the validation and training loss evolution obtained from the CNN model for Case 0.

For the evaluation of the CNN solution, we calculated the root mean square (rms)

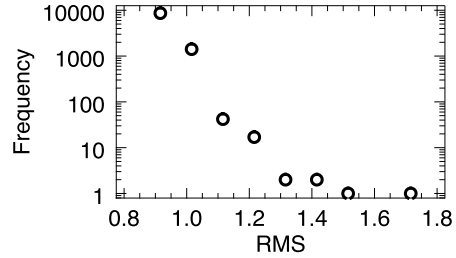
$$\text{RMS} = \sqrt{\frac{\sum \mathfrak{R}^2}{N}}, \quad (4)$$

where \mathfrak{R} is the ratio between the Z coordinate of the reconstructed and the original loops and N is the number of elements in a loop.

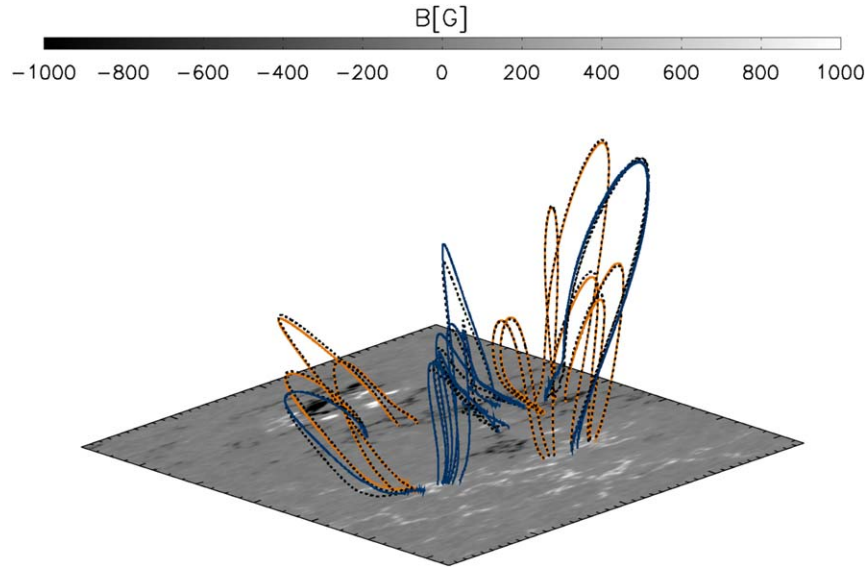
In Figure 1(b) we show the rms distribution for Case 0. The rms values spanning 0.9–1.1 represent 99.96% from the entire data set of loops. An rms value of unity means that the CNN reconstructed loop and the original ones are perfectly matched. The CNN could obtain almost perfectly the 3D position for 99.96% of loops. In Figure 1(c) we display the Low & Lou magnetic field solution used for the NLFFF extrapolation overplotted with a randomly selected subset (for better visibility) of the original traced and the reconstructed 3D loops.



(a) Display of the training (blue solid line) and validation (orange dashed line) loss function during the CNN training.



(b) Distribution of the root mean square for Case 1 loops.



(c) Randomly selected 3D loops overplotted on the HMI photospheric radial magnetic field. The black dashed lines shows the original 3d loops. The CNN reconstructed 3D loops with an RMS smaller than 1.1 are displayed with solid orange lines. In blue solid line are shown the 3D loops with an RMS larger than 1.1.

Figure 2. CNN results for Data type 2, Case 1.

4.2. Data Type 2

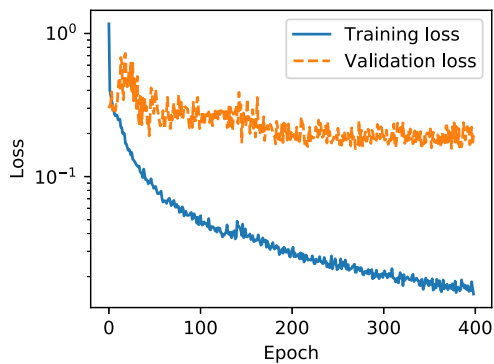
Case 1. We applied the NLFFF extrapolation in the Cartesian geometry to an HMI vector magnetic field recorded on 2010 July 15. From the full-disk 4096×4096 pixels we selected a 1280×400 field of view. The height of the box is 320 pixels. From the solution of the extrapolation, we traced 10,127 loops. The CNN network reached a reasonable solution after 250 epochs similar to Case 0. In Figure 2(a) we plot the evolution of the training and validation losses with the epochs. Figure 2(b) shows the distribution of the rms (Equation (4)). The rms values between 0.9 and 1.1 represent a percentage of 99.06 of all loops. In Figure 2(c) we present the HMI vector magnetogram used as input for extrapolation together with few randomly selected 3D original and reconstructed loops (orange and blue solid line). The two different categories of reconstructed loops are made based on the rms values. The orange loops are the ones with rms smaller than 1.1 and the blue loops with rms larger than 1.1.

Case 2. For the spherical NLFFF optimization code we use full-disk HMI vector magnetic field recorded on 2010 August 2. The computational domain of $256 \times 372 \times 512$ grid points

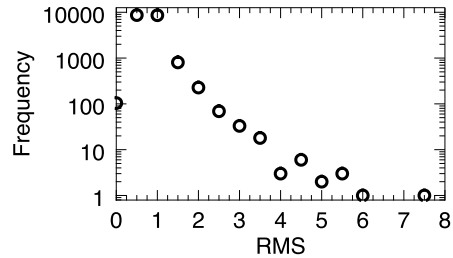
extends over $r = [1, 2.5] R_{\odot}$, $\theta = [-70^{\circ}, 70^{\circ}]$ in latitude and $\phi = [90^{\circ}, 270^{\circ}]$ in longitude. The latitudinal boundaries exclude the polar areas because the surface data in polar latitudes have poor quality and the numerical finite-difference representation used for the L_{tot} functional expressed in the spherical coordinates becomes singular at the poles. From the solution of the extrapolation, we traced and used for the CNN 10368 3D loops. 98.31% of the loops had an rms (Equation (4)) between 0.5 and 1.5.

Case 3. We used as boundary data full Sun synoptic vector magnetogram.

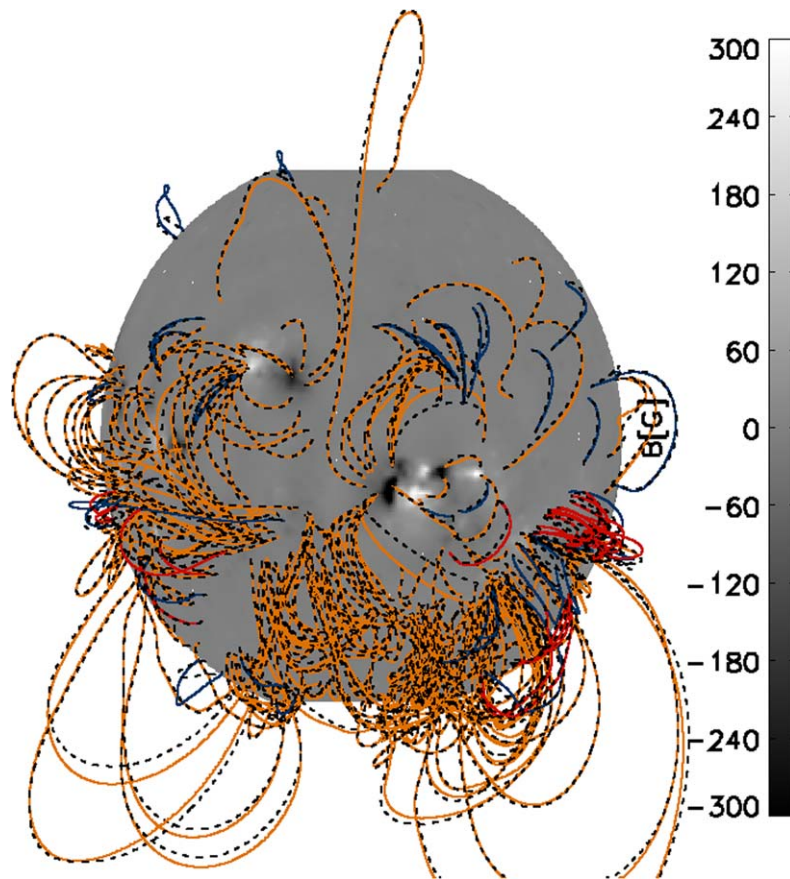
For the NLFFF extrapolation we used the computational domain of $180 \times 280 \times 720$ grid points extending over $r = [1, 2.5] R_{\odot}$, $\theta = [-70^{\circ}, 70^{\circ}]$ in latitude and $\phi = [0^{\circ}, 360^{\circ}]$ in longitude. From the solution of the extrapolation, we traced loops higher than $r = 1.05 R_{\odot}$. For the training, we used a total of 18,555 loops. We applied to the loops reversible transformations that are useful for the CNN method and which are: coordinate transformations, rotation to ensure z is positive, and transformations that x and y components are in the positive domain. We also filtered the loops based on their inflection points to exclude



(a) Display of the training (blue solid line) and validation (orange dashed line) loss function during the CNN training.



(b) Distribution of the root mean square for Case 3 loops.



(c) Randomly selected 3D loops overplotted on the HMI photospheric radial magnetic field. The black dashed lines shows the original 3D loops. The CNN reconstructed 3D loops with an RMS (Eq. 4) between 0.5 and 1.5 are displayed with solid orange lines. In blue solid line are shown the 3D loops with an RMS between 0. and 0.5 and between 1.5 and 2. In red solid lines are presented the 3D loops with an RMS larger than 2.

Figure 3. CNN results for Data type 2, Case 3.

the over-wiggling loops, which in general are not visible in the EUV images. Compared with the other cases, the Case 3 loops have a wider range in the values of length, distance between footpoints, and heights; therefore, the CNN method did not perform as well as in the previous cases. In Figure 3(a) we show the evolution of the training and validation loss function. For this case, the CNN model converged after 400 epochs. Figure 3(b) displays the distribution of the rms ratios of the loops. From all of the loops, 94.5% is in the rms range of 0.5–1.5. The performance is not as good as in the previous cases. In

Figure 3(c) we show the radial magnetic field of the synoptic map at the photospheric level with the disk center corresponding to 180° longitude.

5. Discussions and Conclusions

Stereoscopy is a suitable method for deriving the 3D geometry of solar phenomena. The study, evolution, and kinematics of these phenomena are essential for understanding their physical processes and space-weather forecasting models. This analysis is

constrained by the data availability of at least two view directions of the same object, reducing the reconstruction possibilities. On the other hand, solar observations from one view direction are more abundant than two simultaneous views. When only one vantage point is available, the CNN method can be used and applied to coronal loops and other curve-like structures.

In conclusion, the new 3D loop reconstruction method will allow the user to employ only one image from a single vantage point. One would select as many loops as possible from the EUV image by visual inspection or automatic identification. The 2D curve-like structure parameters mentioned in the Section 2.1 can be used as input to the CNN models that have already been trained (see Case 0 to Case 3 from Section 4). The output of the used CNN model will be the third component of the 2D curve-like structures. In the end, one will have all the three components of the curve-like structures.

The NN model presented in this Letter can only perform the 3D geometrical reconstruction of loop types used for the training. It does not have any direct contribution to the magnetic field modeling. Indirectly, the 3D coordinates of the CNN reconstructed loops can be used, for example, as a constraint for the magnetic field modeling as shown by Chifu et al. (2017).

Data are courtesy of NASA/SDO and the HMI science teams. I.C. acknowledges DFG-grant WI 3211/5-1. The HMI data are provided courtesy of NASA/SDO and the HMI science team. I.C. would like to thank M. Madjarska for helpful discussions. R.G. acknowledges financial support by the Portuguese Government

through the Foundation for Science and Technology—FCT FEDER—European Regional Development Fund through COMPETE 2020—Operational Programme Competitiveness and Internationalization.

ORCID iDs

Iulia Chifu  <https://orcid.org/0000-0002-5481-9228>

Ricardo Gafeira  <https://orcid.org/0000-0003-4920-0153>

References

- Aschwanden, M. J. 2011, *LRSP*, 8, 5
- Chifu, I., Wiegmann, T., & Inhester, B. 2017, *ApJ*, 837, 10
- Couvidat, S., Schou, J., Shine, R. A., et al. 2012, *SoPh*, 275, 285
- De Rosa, M. L., Schrijver, C. J., Barnes, G., et al. 2009, *ApJ*, 696, 1780
- Gafeira, R., Suárez, D. O., Milic, I., et al. 2021, arXiv:2103.09651
- Gary, G. A. 2001, *SoPh*, 203, 71
- Inhester, B. 2006, arXiv:0612649
- Kingma, D. P., & Ba, J. 2014, arXiv:1412.6980
- Liu, Y., Hoeksema, J. T., Sun, X., & Hayashi, K. 2017, *SoPh*, 292, 29
- Low, B. C., & Lou, Y. Q. 1990, *ApJ*, 352, 343
- Nair, V., & Hinton, G. E. 2010, in Proc. 27th Int. Conf. Machine Learning, ICML'10, ed. J. Fürnkranz & T. Joachims (Madison, WI: Omnipress), 807, <http://dl.acm.org/citation.cfm?id=3104322.3104425>
- Pesnell, W. D., Thompson, B. J., & Chamberlin, P. C. 2012, *SoPh*, 275, 3
- Reale, F. 2010, *LRSP*, 7, 5
- Tadesse, T., Wiegmann, T., Inhester, B., & Pevtsov, A. 2011, *A&A*, 527, A30
- Wheatland, M. S., Sturrock, P. A., & Roumeliotis, G. 2000, *ApJ*, 540, 1150
- Wiegmann, T. 2004, *SoPh*, 219, 87
- Wiegmann, T., Petrie, G. J. D., & Riley, P. 2017, *SSRv*, 210, 249
- Wiegmann, T., & Sakurai, T. 2012, *LRSP*, 9, 5