The Astrophysical Journal, 940:147 (16pp), 2022 December 1 https://doi.org/10.3847/1538-4357/ac9c60

Deep Learning–based Fast Spectral Inversion of Hα and Ca II 8542 Line Spectra

Kyoung-Sun Lee[1], Jongchul Chae[1], Eunsu Park[2], Yong-Jae Moon[3], Hannah Kwak[1,2], and Kyuhyoun Cho[1,4,5]
[1] Astronomy Program, Department of Physics and Astronomy, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Republic of Korea
lksun@astro.snu.ac.kr, rolrinsun@gmail.com
[2] Space Science Division, Korea Astronomy and Space Science Institute, Daejeon 305-348, Republic of Korea
[3] School of Space Research, Kyung Hee University, 1732, Deogyeongdae-ro, Giheung-gu, Yongin-si Gyunggi-do, 17104, Republic of Korea
[4] Bay Area Environmental Research Institute, NASA Research Park, Moffett Field, CA 94035, USA
[5] Lockheed Martin Solar & Astrophysics Laboratory, 3251 Hanover Street, Palo Alto, CA 94304, USA

Received 2022 January 1; revised 2022 October 10; accepted 2022 October 16; published 2022 November 30

Abstract

A multilayer spectral inversion (MLSI) model has recently been proposed for inferring the physical parameters of plasmas in the solar chromosphere from strong absorption lines taken by the Fast Imaging Solar Spectrograph (FISS). We apply a deep neural network (DNN) technique in order to produce the MLSI outputs with reduced computational costs. We train the model using two absorption lines, Hα and Ca II 8542 Å, taken by FISS, and 13 physical parameters obtained from the application of MLSI to 49 raster scans (~2,000,000 spectra). We use a fully connected network with skip connections and multi-branch architecture to avoid the problem of vanishing gradients and to improve the model's performance. Our test shows that the DNN successfully reproduces the physical parameters for each line with high accuracy and a computing time of about 0.3–0.4 ms per line, which is about 250 times faster than the direct application of MLSI. We also confirm that the DNN reliably reproduces the temporal variations of the physical parameters generated by the MLSI inversion. By taking advantage of the high performance of the DNN, we plan to provide physical parameter maps for all the FISS observations, in order to understand the chromospheric plasma conditions in various solar features.

Unified Astronomy Thesaurus concepts: Solar chromosphere (1479); Neural networks (1933); Spectroscopy (1558)

1. Introduction

The solar chromosphere is the interface atmospheric layer between the photosphere and the corona, which is highly dynamic, finely structured, and complex. In order to understand the phenomena that occur in the chromosphere and their varied physical drivers, it is necessary to quantitatively diagnose the physical properties of chromospheric plasmas. We can infer the physical conditions in the chromosphere through the analysis of strong lines, such as those of H, Ca II, and Mg II, in the visible, near-IR, and UV spectral regimes (Carlsson et al. 2019). These chromospheric lines are optically thick and formed under conditions of non–local thermodynamic equilibrium (NLTE), which requires NLTE radiative transfer modeling. The forward modeling makes use of simulations, a realistic 3D radiative magnetohydrodynamic model (Bifrost; Gudiksen et al. 2011), and NLTE radiative transfer (RH—Uitenbroek 2001; RADYN—Allred et al. 2005). These forward models provide powerful tools for inferring the physical conditions of the chromosphere, but are computationally expensive and complex, in terms of their calculations of atomic level populations from NLTE radiative transfer.

A practical way of inferring physical parameters from observed chromospheric lines involves an inversion process, which minimizes the deviation between the synthetic line profile—derived from the physically based radiative transfer calculations—and the observed line profile. There are two kinds of inversion process. One is inversion that is based on forward modeling, as described above (Hazel—Asensio Ramos et al. 2008; NICOLE—Socas-Navarro et al. 2015; SNAPI—Milić & van Noort 2018; STiC—de la Cruz Rodríguez et al. 2016, 2019). For example, the Non-LTE Inversion Code based on the Lorien Engine (NICOLE) derives stratified physical properties from chromospheric spectral line profiles, by synthesizing the profiles based on the NLTE calculation. Recently, de la Cruz Rodríguez et al. (2016, 2019) have developed the STockholm Inversion Code (STiC), which gives the inversion results of NLTE spectral lines, based on the RH forward synthesis code, allowing inversion with multiple atoms in NLTE and partial redistribution effects in terms of angle and frequency. These inversion codes that use forward models make it possible to infer the stratified physical properties in the solar chromosphere, by considering the complex underlying physical conditions. However, NLTE spectral synthesis is time-consuming and it needs to be generated multiple times for each spectrum, as part of the inversion process.

The second kind of inversion process uses parameterized models, in which the line profiles are calculated analytically, based on the model parameters (such as line-shape parameters or source functions determined from the radiative transfer calculation), to fit the observed spectral lines. The parametric models tune the model parameters until a good fit to the observed spectral line is obtained, making it relatively simple to extract the relevant information. For example, cloud model inversions (Beckers 1964; Tziotziou 2007; Chae 2014) treat the source function as a free model parameter over optical depth to infer the physical parameters of cloud-like features lying above the solar surface.

Recently, Chae et al. (2020, 2021a) have proposed a multilayer spectral inversion (MLSI) and applied it to the strong absorption spectral lines Hα and Ca II 8542 Å, as taken by the Fast Imaging Solar Spectrograph (FISS; Chae et al. 2013). MLSI successfully provides height-varying physical quantities in a simplified three-layer atmosphere, including the source function, Doppler velocity, and Doppler width. MLSI is much faster than forward modeling that requires NLTE calculations. Nevertheless, its speed is not yet quick enough, especially when larger numbers of line profiles have to be analyzed. With the higher spatial, temporal, and spectral resolutions of current observations, the amounts of data become larger and larger. To study spatiotemporal variations of physical parameters in the chromosphere, a huge number of line profiles should be analyzed based on spectral inversion. Even though MLSI represents a relatively fast inversion, it still requires unbearably long computing times for these kinds of data (big data). For example, the fast scan capability of FISS means that it can produce raster scan spectral data every 20 s, with a $30'' \times 40''$ field of view (∼46,000 pixels). It then takes about an hour to obtain the inverted physical parameters for a single instance of raster scan data using MLSI, meaning that it would take 180 hr to obtain the inversion results for an hour of observations (180 raster scans). Therefore, a much faster implementation of the inversion process is strongly required. This is the motivation of the present work, which aims to apply a deep learning technique to MLSI.

Various "deep learning" algorithms—such as deep neural networks (DNNs; LeCun et al. 2015), convolutional neural networks (CNNs; Lecun & Bengio 1995), or generative adversarial networks (Goodfellow et al. 2014)—have become popular means of dealing with big data in solar research. For instance, the Atmospheric Imaging Assembly (Lemen et al. 2012) on board the Solar Dynamics Observatory (Pesnell et al. 2012) continuously provides a 4096 × 4096 full-Sun image every 12 s, with multiple wavelength channels. This vast data set has been utilized by several deep learning applications to generate new observables, using EUV images at different wavelengths, as well as magnetograms (Kim et al. 2019; Park et al. 2019; Lee et al. 2021; Lim et al. 2021), or the calculations of differential emission measures (DeepEM; Cheung et al. 2018). Our approach is similar to DeepEM, which is a DNN implementation of differential emission measure (DEM) inversion, achieved by training the DNN on a set of imaging observations and their inverted DEM solutions, in order to reduce the calculation time.

Deep learning has also been applied to spectral inversions in solar physics (Carroll & Staude 2001; Asensio Ramos & Díaz Baso 2019; Osborne et al. 2019; Sainz Dalda et al. 2019). Sainz Dalda et al. (2019) applied a DNN to the STiC inversion on a large collection of observed IRIS Mg II h & k spectra, in order to construct a database of profiles and their underlying atmospheres. The trained model then estimated the thermodynamic parameters (temperature, line-of-sight velocity, electron density, etc.) in the chromosphere and upper photosphere, by using a lookup approach to the database, in a fast way. Asensio Ramos & Díaz Baso (2019) used the SIR inversion code to train a model on the Stokes parameters from 3D MHD simulations, in order to infer physical properties from observations of Stokes profiles. In another approach, Osborne et al. (2019) used an invertible neural network (INN; Ardizzone et al. 2018), trained with the radiation hydrodynamic model (RADYN) and synthesized spectra of Hα and Ca II line profiles, to produce inverted atmospheres with physical information from the observed spectra of flaring chromosphere.

In this study, we present a DNN implementation of the MLSI model developed by Chae et al. (2020, 2021a). We apply a fully connected network to model and determine the physical parameters as a regression problem. The DNN is able to infer the possible underlying physical parameters from an observed spectral line much more quickly, with comparable precision to MLSI, thereby allowing us to investigate the physical processes in various phenomena in the solar photosphere and chromosphere. This study is organized as follows. The basic concept and the applied DNN are described in Section 2, while the training and test data sets are described in Section 3. In Section 4, we present the results from our DNN and discuss its performance compared to the original inversion model. Finally, a summary is given in Section 5.

2. Idea and Model

2.1. Basic Idea

Our deep learning model involves a kind of supervised learning. Supervised learning means that the data used for training have a defined structure. Each example comprises a pair consisting of an input object and the desired (labeled) output value. We want the algorithm to learn a functional approximation, by understanding the predefined structure between the input and output. Once the algorithm has learned the proper relation between the predefined examples of inputs and outputs by means of its training, it should then correctly predict the values or class labels for new inputs.

A DNN is a deep learning algorithm that consists of multiple hidden layers between the input and output layers. Each hidden layer has nodes (artificial neurons) that transform the input data by means of an inner product with a different weight vector and a linear transformation. A nonlinear function is then applied to the node, which will determine whether the output signal of the node is activated and passed to another node. Training the network with many data sets reduces the difference between the output of the network and the desired output value, by determining the proper weights of each node and establishing a complex nonlinear function between the input and output data. This means that the network is able to learn functionally important relationships.

For the supervised learning, we used pairs—an observed spectrum from FISS and the physical parameters for that spectrum as inferred from the updated MLSI (Chae et al. 2021a)—comprising the input and target data, respectively (Figure 1). We expect the DNN to learn the relationship between the input and output pairs in the training of the network. The details of the input and output data are described in Section 2.2, and our model architecture and hyperparameters are presented in Section 2.3.

2.2. Input and Output

We use spectral line profiles taken with FISS on the Goode Solar Telescope (GST) at the Big Bear Solar Observatory as the model inputs. FISS simultaneously records two main spectral bands, Hα and Ca II 8542 Å, using a 32 μm slit and two cameras. The spectral coverages of the Hα and Ca II 8542 bands are 9.7 Å and 12.9 Å, respectively. The spectral samplings for Hα and Ca II 8542 Å are 0.019 Å and 0.025 Å, respectively. The spectral data were calibrated—corrected for flat fields, dark currents, and stray light—and compressed by the Principal Component Analysis method (Chae et al. 2013).

We also conducted a data reduction for the wavelength calibration and intensity normalization, as described in Chae et al. (2020). The absolute wavelength calibration was done by

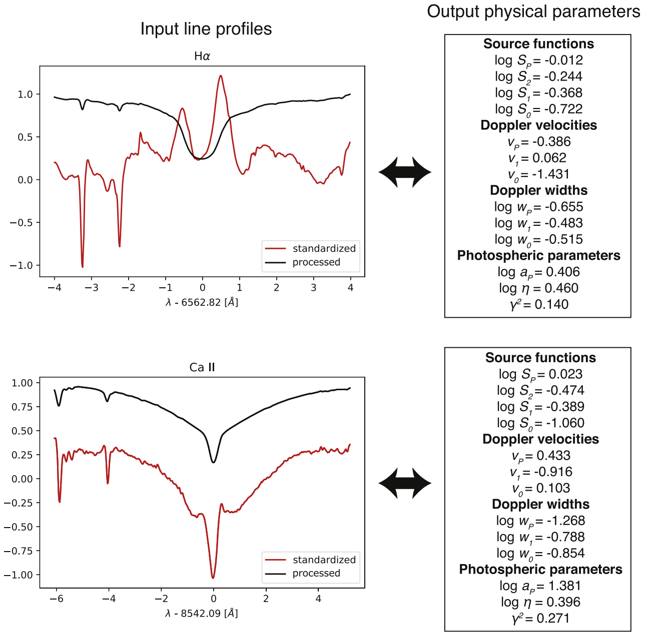


Figure 1. Example pairs—the input object and the desired output value—for supervised learning. The left panels display the inputs for the observed line profiles of Hα and Ca II 8542 Å (in black) and their shapes after the standardization processes described in Section 2.2 (in red). Details of the preprocessing are provided in the text. The right panels show the physical parameters calculated from the MLSI model using the input spectra.

using the flat/calibration data of the quiet region (QR), which are taken more than once every observing day. We decide the wavelengths using telluric lines from the Earth's atmosphere or metal lines from the solar photosphere. We can then calculate the positions of the line centers at Hα and Ca II, the wavelength per pixel, and the wavelength coverage of the data. We normalize the spectral profiles to a mean continuum intensity. First, the average spectral profile for each data set (i.e., the raster scan) is taken as a reference profile for diverse data sets. Then, we normalize all the spectral profiles using the maximum intensity of this reference profile, which is a proxy of the mean continuum intensity. In addition, we subtract the terrestrial absorption lines from the spectra, as described by Chae et al. (2021a).

To arrive at the homogeneous input parameters for the training, the data were preprocessed. First, we note that the wavelength ranges and scales of the spectral lines vary slightly from day to day, so we restrict the input wavelength ranges for all the data sets—6558.82 Å–6566.82 Å for Hα and 8535.99 Å–8547.29 Å for Ca II—to ensure that all the data have the same wavelength ranges. We then interpolate all the spectra using the same absolute wavelength scale, with samplings of 0.02 Å and 0.025 Å for Hα and Ca II, respectively. Second, we perform input data scaling, to avoid unstable or slow learning processes. Since the ranges of the input data values vary widely, if the data are not scaled, the learning algorithm may become confused and place more importance on the pixels, only because they have higher intensity values, which would make the learning process unstable. In addition, for the optimization of our model, we use gradient descent–based algorithms. The input data will affect the step size of the gradient descent, and the differences in the ranges of the input values will result in different step sizes for each of the input values. Therefore, we need to scale the data, before training the model to update the steps for the gradient descent at the same rate for all the input spectra, resulting in the gradient descent smoothly converging to the minima much faster. We apply the data scaling method, normalization, and standardization to our input data, then compare the model's performance. Standardization, which makes the mean of the training input data values zero, with the standard deviation set to 1, shows better performance. The standardization is conducted using all the training input data sets, but separately, along each wavelength. Examples of the input spectra for Hα and Ca II are displayed in the left panels of Figure 1. Each intensity along the wavelength is compared to the intensity of the entire input data set at the same wavelength. The solid black and red lines indicate the observed line profiles and the standardized profiles of the whole training input data set, respectively. If the wing value is close to the mean value of all the input data, then the profile value is near zero. But for the standardized (red) profile of Hα at the center, the intensity value of the example is larger than the mean value of the intensity of the whole input data set, so it is larger than zero.

We use the physical parameters inferred from the updated MLSI (Chae et al. 2021a) as the target data. The model solves the radiative transfer equation with two strong absorption profiles, the Hα and Ca II lines, assuming a three-layer atmosphere, consisting of the photosphere, the lower chromosphere, and the upper chromosphere (see Figure 3 in Chae et al. 2021a). The modeled line profile of MLSI is fully specified by 15 parameters. The two parameters of optical thickness, in the low chromosphere ($\tau_{p,2}$) and upper chromosphere ($\tau_{p,1}$), are fixed values. The 10 physical quantities as free parameters are determined from the model fitting, the source functions at the boundaries between each layer (S_p, S_2, S_1, and S_0), the Doppler velocities, the Doppler widths at the boundaries of each layer in the chromosphere (v_1, v_0, w_1, and w_0), and the parameters for the absorption profiles of the photosphere (the dimensionless damping parameter (a_p) and the ratio of peak line absorption to continuum absorption (η)). The Doppler velocity and Doppler width in the photosphere (v_p and w_p) are determined from analysis of the photospheric lines in the wings of the chromospheric profiles, while the parameter for collisional damping in the low chromosphere (γ_2) is determined from the shapes of the broad wings of the strong lines. Among the inferred physical quantities from MLSI, we use 13 parameters, except for the two fixed parameters ($\tau_{p,2}$, $\tau_{p,1}$). Examples of the output values for the physical quantities are listed in the right panels of Figure 1.

2.3. DNN Model

After defining the pair of input and output layers, we create a DNN of hidden layers. Figure 2 presents the architecture of our model. We use a fully connected network, adopting two concepts: skip (shortcut) connection and branching architecture.

The skip connection, or shortcut layer, is a widely used technique for improving the performance and the convergence of deep learning models, which is used in residual networks (He et al. 2015) or U-Net networks (Ronneberger et al. 2015). Generally, if we add more layers and deepen the neural network models, the accuracy of the results or the success rate of the prediction increases. However, if the depth further increases, the gradients used during the training become

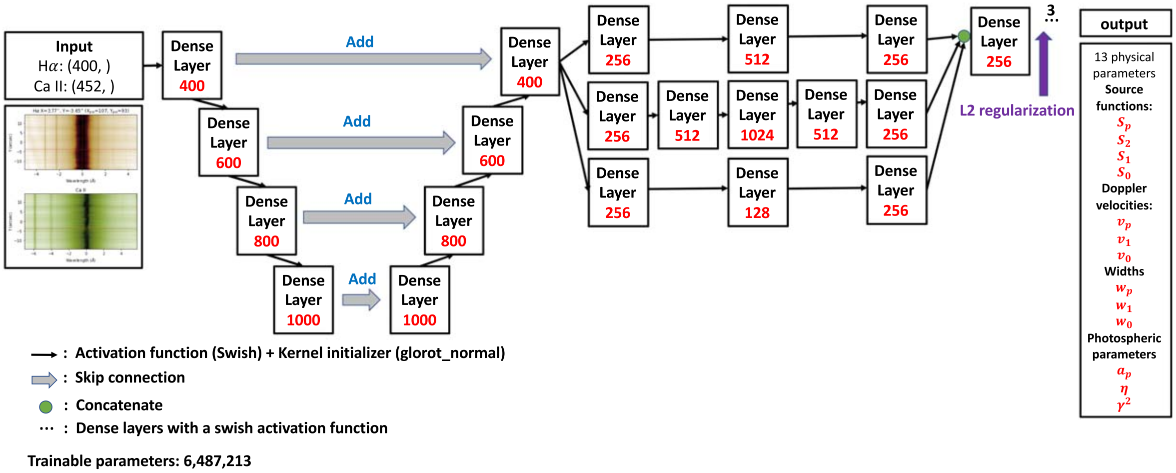


Figure 2. Flowchart of our DNN with skip connection and branching architecture. The red numbers for each of the layers indicate the numbers of nodes. The black number above the ellipsis denotes the number of layers that have the same number of nodes as the previous layer.

exponentially small, and the information cannot pass through the deeper layers, which is called the vanishing gradient problem. We utilize skip connections (the gray arrows in our model architecture) to allow gradient information to pass through the layers, with the previous layer's output being added to the output of a deeper layer. This connection, therefore, enables the signal to propagate in deeper networks, while accelerating the training by avoiding the effect of vanishing gradients.

After the shortcut layers, we create multiple parallel branches for processing the data independently, using different parameters with different weights. The branching architecture has been used in CNNs, such as inception networks (Szegedy et al. 2016) and Xception (Chollet 2017). The multiple branches increase the success of the model by reducing the duality gap of empirical risk, thereby making it easier to optimize the network (Zhang et al. 2018). Here, we introduce three parallel branches, consisting of a different number of layers with a different number of nodes. Following these multiple branches, the outputs are concatenated, before they pass through the fully connected dense layers. The final output layer is a dense layer, without an activation function to give the regression values. The outputs are 13 physical parameters for specifying the modeled line profiles.

In this network, we use a Swish activation function (Ramachandran et al. 2017), which is defined as

$$f(x) = x \times \text{sigmoid}(x) = \frac{x}{1 + e^{-x}} \tag{1}$$

at each layer, and which determines how the weighted sum of the input is transformed into an output from a node or nodes, making the network nonlinear. Since the Swish function helps to alleviate the vanishing gradient problem during backpropagation, it is known to have better performance on deeper models than the Rectified Linear Unit, ReLU, which is the most widely used activation function. In addition, we use Xavier initialization (glorot_normal: Glorot & Bengio 2010) at each layer, in order to determine the proper initial weight suitable for deeper networks, by avoiding the saturation of the weight.

We train the network using the Adam (Kingma & Ba 2014) optimizer, over 2000 epochs, with a batch size of 10,000. For the other hyperparameters, the initial learning rate is 0.2825×10^{-3}, which is then reduced exponentially by a factor of 0.45 every 50 epochs, until the running rate becomes 10^{-5}. To avoid overfitting, we introduce L2 regularization, with the lambda value of 5×10^{-5}, after the multiple branch layers. In addition, we set early stopping by monitoring the validation loss, with the value of patience being 30. Our model then stops early, at around 450 and 650 epochs for the Hα and Ca II parameters, respectively. To train our DNN, we use the mean squared error (MSE) as a scalar loss function and the mean absolute error (MAE) as the metric that is used to estimate the accuracy of the thirteen predicted physical parameters. We finally obtain the model that has the best minimum loss function, 0.003 and 0.009 of the MSE values for the Hα and Ca II parameters, respectively. Our DNN is implemented and trained in Tensorflow, using Graphical Processing Units (GPUs) to accelerate the calculations. We used an NVIDIA GeForce RTX 3090 for training, during the exploration of the hyperparameters.

3. Data Sets: Training and Test

For the supervised learning, the essential element is a well-prepared data set, covering as much of the parameter space as possible, with reliable mappings between the observed spectra and the physical parameters derived from MLSI. We prepare the training and validation data sets by choosing a scan raster from each of the different observing targets or pointings from the FISS/GST observations, covering both the Hα and Ca II 8542 lines in the period from 2013 to 2015. Table 1 presents a list of the data sets. We used 49 raster scans, consisting of about 2,500,000 spatial pixels, with observed spectral profiles and inverted physical parameters. Here, we note that the selected observing targets of FISS are mostly active regions (ARs), emerging flux regions, or pores, as well as regions around the disk center ($\mu > 0.5$), rather than QRs, due to the limitations of the seeing conditions. GST uses an adaptive optics (AO) system to improve its seeing. The AO system needs to choose a stable (slowly evolving) structure with high contrast, to continuously track the structure, and to correct the seeing by moving the tip-tilt mirror. It is easy to select the structures inside a sunspot (AR), pore, or emerging flux region that have highly contrasting features. However, if the atmospheric conditions are not good, with high turbulence, we cannot follow the QRs or coronal holes, which do not have

Table 1
The Data Sets for the DNN: Training and Validation

Number (1)	Date (2)	Time (UT) (3)	Observing Target (4)	NOAA Number (5)	Location (*x*, *y*) (6)
1	2013-Jul-16	19:17:25	Sunspot	11791	(62″, −30″)
2	2013-Jul-16	21:44:43	QR	…	(447″, 87″)
3	2013-Jul-17	17:02:38	Trailing pore	11791	(247″, −278″)
4	2013-Jul-17	18:41:07	QR	…	(−15″, −126″)
5	2013-Jul-17	20:48:15	Emerging flux region, pore	…	(556″, −291″)
6	2013-Jul-18	17:47:50	Sunspot	11793	(−332″, 272″)
7	2013-Jul-18	21:04:43	Pore	11793	(−365″, 259″)
8	2013-Jul-19	17:06:29	Pore	11793	(−17″, 302″)
9	2013-Jul-28	18:19:30	AR, pore	11801	(43″, 391″)
10	2013-Jul-29	17:20:57	Pore	11801	(194″, 381″)
11	2013-Jul-29	17:45:21	AR near east limb	11808	(−741″, 196″)
12	2013-Jul-30	17:53:46	AR	11808	(−640″, 134″)
13	2013-Jul-30	19:35:57	Pore	11806	(−357″, −373″)
14	2013-Jul-31	18:27:29	Sunspot	11801	(520″, 228″)
15	2013-Aug-16	17:04:16	Sunspot	11818	(222″, −202″)
16	2013-Aug-16	18:26:39	AR	11817	(593″, −433″)
17	2013-Aug-16	19:45:16	Sunspot	11818	(216″, −220″)
18	2013-Aug-16	21:11:10	Sunspot	11820	(−392″, −318″)
19	2013-Aug-17	18:08:29	Flaring AR (before flare)	11818	(472″, −220″)
20	2013-Aug-17	18:21:41	Flaring AR (flare peak)	11818	(472″, −220″)
21	2013-Aug-17	18:43:10	Flaring AR (after flare)	11818	(472″, −220″)
22	2013-Aug-18	16:26:35	Emerging flux region, pore	11825[a]	(131″, 140″)
23	2013-Aug-18	17:39:08	Emerging flux region, pore	11826[a]	(142″, 137″)
24	2013-Aug-23	16:30:40	Trailing spot	11827	(−152″, −402″)
25	2013-Aug-23	19:18:45	Trailing spot	11827	(−152″, −402″)
26	2013-Aug-24	16:40:53	Sunspot	11828	(−59″, 162″)
27	2013-Aug-24	19:05:35	Pore	11828	(−177″, 189″)
28	2013-Aug-25	16:27:49	Trailing pore	11828	(116″, 168″)
29	2013-Aug-25	17:53:37	Leading pore	11828	(167″, 150″)
30	2013-Aug-25	20:04:57	Sunspot near west limb	11823	(818″, −157″)
31	2014-Jun-3	16:52:01	Pore	12078	(134″, −318″)
32	2014-Jun-3	18:17:35	Trailing pore	12077	(−329″, −106″)
33	2014-Jun-3	19:57:57	Sunspot	12077	(−344″, −77″)
34	2014-Jun-3	21:44:07	Emerging flux region, pore	12079	(−696″, 51″)
35	2014-Jun-5	16:47:05	Sunspot	12080	(−527″, −204″)
36	2014-Jun-5	18:34:01	Filament	…	(−420″, −104″)
37	2014-Jun-5	18:58:32	AR	12082[a]	(−605″, 236″)
38	2014-Jun-6	17:02:45	Sunspot	12079	(−74″, 188″)
39	2014-Jun-6	18:42:17	AR	12085[a]	(−394″, −343″)
40	2014-Jun-6	20:08:19	Light bridge	12082	(−436″, 222″)
41	2015-Jun-15	16:55:49	Leading sunspot	12367	(−310″, −330″)
42	2015-Jun-15	17:49:38	QR	…	(−215″, −378″)
43	2015-Jun-15	17:54:34	QR	…	(−215″, −378″)
44	2015-Jun-15	18:46:58	Pore near AR	12367	(−341″, −321″)
45	2015-Jun-15	20:16:12	Leading pore	12370	(−190″, 271″)
46	2015-Jul-24	20:15:52	AR	12387	(595″, 219″)
47	2015-Jul-24	22:15:21	Sunspot near east limb	12389[a]	(−768″, −256″)
48	2015-Jul-28	20:30:52	Filament	…	(−561″, −25″)
49	2015-Jul-28	22:06:56	Filament	…	(−554″, −18″)

Note.
[a] Newly emerging ARs. At the observing time, the NOAA number had not yet been assigned.

high-contrast structures. Limb observations are also tricky, for the same reason. Since most training data sets are biased to ARs or near the disk center, the application of this model to QRs or observations near limbs may be less reliable.

We note that the updated MLSI introduced several physical constraints for inferring physically reasonable parameters, the details of which can be found in Chae et al. (2021a). The least squares fit minimizes the difference between the model and the data, considering the physical constraints. Then, the two dimensionless parameters, ϵ_D and ϵ_P, as defined in Chae et al. (2021a), provide estimates of the goodness of fit for the spectral profile and for the physical constraints, respectively. We examine the values of ϵ_D and ϵ_P for all the training data sets. The majority of the spectra that are acceptably fitted by MLSI have values of ϵ_D and ϵ_P that are less than 3. Therefore, we only use input and target data that meet the criteria $\epsilon_D < 3$ and $\epsilon_P < 3$. Among the training data sets in Table 1, the number of line profiles and inverted output parameters that meet the

Table 2
The Data Sets for the DNN: Testing

Date (1)	Time (UT) (2)	Target (3)	NOAA (4)
2020-July-30	16:48:21-17:57:43	QR	…
2013-July-17	18:32:24-19:56:18	AR	11791

criteria is about 2,000,000, or 80% of the data sets. For the other 20% of the data sets, ϵ_D and ϵ_P are larger than 3, with some of them even reaching over 10, showing profiles with emission for flares or profiles with strong secondary shifted components. We note that the features with a high goodness of fit parameter (outliers) are not evenly spread across all the target data sets, but rather a large number are to be found in the flare data sets. We exclude any poorly fitting pairs of spectral profiles and their inverted physical parameters from our DNN model training. We shuffle the selected data sets and randomly use 80% of the data for training and 20% for validation.

To verify the DNN's performance, we need to test the DNN on separate data sets that are not included in the training or validation data sets. The test data sets are listed in Table 2. One set is a time series observation of a QR on 2020 July 30. The other set is a time series observation of an AR on 2013 July 17, which is described in detail in Chae et al. (2014, 2021b). A set of monochromatic maps, constructed at some of the wavelengths from the raster scans for the QR and the AR, are presented in Figure 3.

4. Results and Discussion

Our model provides physical parameter values, given the input of the observed spectra. First, we compare each parameter value from the DNN to the parameters that are derived directly from MLSI, which are the target parameters (the ground truth), to evaluate the accuracy of the DNN parameter predictions. Second, if the model is able to learn the relationship between the pairs reasonably well, we plan to use the stand-alone DNN to infer the parameters, without performing the full MLSI calculation. To directly confirm that the obtained physical parameters are explainable in the observed spectra, we reproduce the line profiles using the parameters from our DNN and compare them to the observed spectra. Third, we also check the temporal variations of the parameters, to confirm that the temporal variations of the parameters recovered by the DNN are physically reasonable.

4.1. Comparison of Physical Parameters

To test our model, we use spectral line profiles in a QR (2020 July 30) and an AR (2013 July 17), taken from FISS (Table 2). Figure 3 shows the sample images of the observed scan rasters of the QR and the AR. The raster images of the QR have 150 × 250 pixels—that is, 37,500 line profiles. The raster image of the AR consists of 64,000 line profiles. We determine the physical parameters for each line profile of the data points using MLSI and the trained DNN, and reproduce the physical parameter maps.

Figures 4 and 5 show the sample images of the parameter maps, the source functions at each layer (S_p, S_2, S_1, and S_0), the Doppler velocities (v_p, v_1, and v_0), and the Doppler widths (w_p, w_1, and w_0), from Ca II in the QR and the Hα in the AR. The upper three rows display the inverted physical parameters from MLSI that are used as ground truth. The lower three rows present the target parameters produced by our trained model. Compared to the parameter maps from MLSI, the parameter maps from our DNN are successfully reproductions. In addition, we present the values of ϵ_D and radiative loss at the lower chromosphere, calculated by both the inverted and the predicted parameters, in Figures 4 and 5. ϵ_D gives the goodness of fit, by comparing the observed and synthesized spectra. The radiative losses for each layer of each line have been calculated from the values of the model parameters, and the values of the radiative losses from the inverted and from the predicted parameters are consistent.

We measure the means and standard deviations of the parameter maps from the DNN, along with four other types of metric, to compare the physical parameters from MLSI with the predicted ones from our DNN: (1) the MAE; (2) the normalized rms error (NRMSE); (3) the correlation coefficient (CC); and (4) the coefficient of determination (R^2 score). The values for each of these parameters are listed in Table 3.

First, the mean and standard deviation values of the physical parameters from our DNN display similar trends to the values of the parameters from MLSI, as discussed in Chae et al. (2021a). For example, the source function of Hα monotonically decreases with height, while that of Ca II has a local maximum in the chromosphere, which is physically expected.

Second, MAE and NRMSE are measured to evaluate the accuracy of the predicted parameters from the DNN compared to the values derived from MLSI. NRMSE is RMSE normalized by the standard deviation of the MLSI parameters. The smaller the values of MAE and NRMSE, the higher the accuracy of the predictions. Both values are pretty small for most parameters, implying that the model performance is quite good. The MAE values range from 0.001 to 0.01 for the source functions and Doppler widths, while those for the Doppler velocity are less than 250 m s^{-1}. The values of NRMSE are less than 30% for most of the parameters, except for the Doppler widths in the photosphere (w_p).

Third, we calculate the CC and R^2 score for each parameter, to examine the linearity of the relationship of each model parameter and the percentage of correct predictions returned by our DNN compared to the parameters from MLSI. If these values are 1, the regression model predictions perfectly fit the data. In relation to the CC, we display 2D density plots for the physical parameters from both spectral lines for the QR and AR in Figures 6–9. The 2D density plots display a tight correlation between the physical parameters derived from MLSI (ground truth) and the DNN (predicted). Here, even though we assume that the parameters derived from MLSI are the ground truth, MLSI itself is an inversion model, which results in some of the observed spectral lines not being well fitted; the cases of ϵ_D and ϵ_P are larger than 1. These data points are denoted by the red dots, which amount to less than 1.5% of the total data. The CCs are mostly close to 1 (larger than 0.8) for each of the physical parameters. Also, the R^2 scores are larger than 0.6, except for the Doppler width in the photosphere (w_p) of the Ca II line in the QR.

Overall, the DNN predicts the target parameters well, except for w_p. Regarding the parameter w_p, the NRMSEs are larger than the those of the other parameters. This is primarily due to the small standard deviation of w_p. The smaller standard deviation makes the NRMSE larger, even though the RMSE itself is not significant. Although the NRMSE has

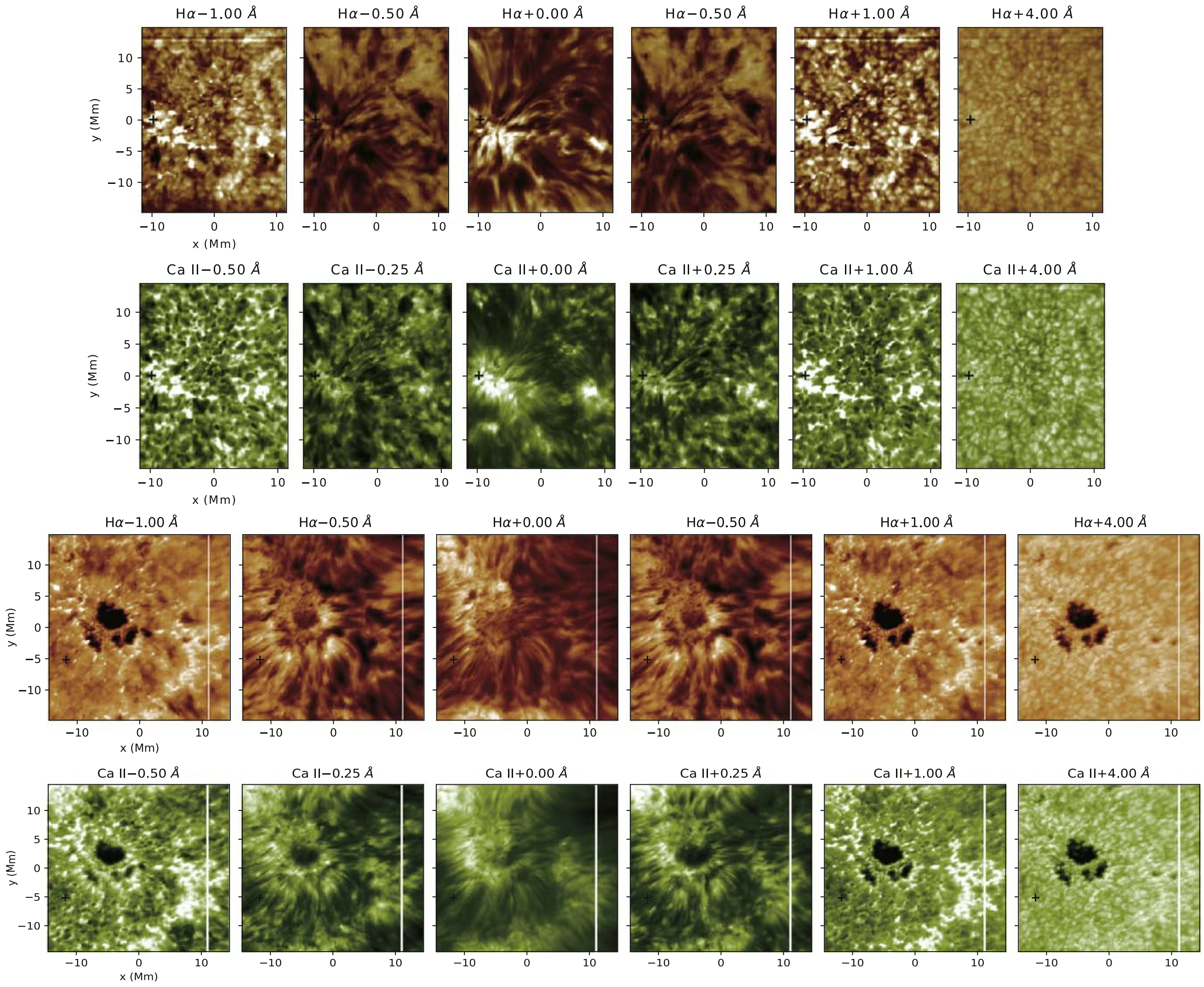


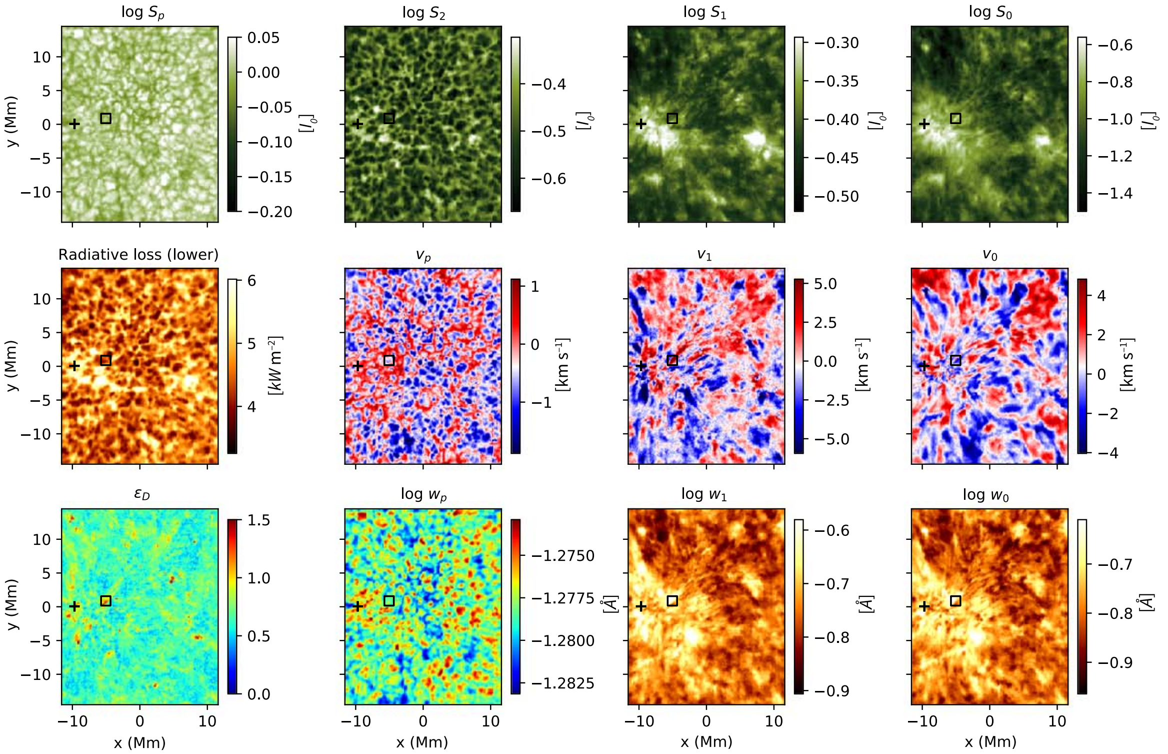
Figure 3. Monochromatic images constructed at the wavelengths of the Hα and the Ca II 8542 Å lines, obtained from FISS. The upper (first and second rows) and lower (third and fourth rows) panels show sample images of a QR and an AR, respectively. The cross symbols in the QR and AR images mark the positions of a network feature and a superpenumbral fibril that we selected to compare the observed line profiles with the modeled spectra, using the physical parameters inferred from the DNN in Figures 10 and 11. The white vertical slits in the scanned raster images of the AR indicate that data are missing.

large values, the high values of the CC (>0.8) and the R^2 score (>0.6) imply that the distributions of the w_p are well matched between the inverted and predicted ones. The w_p maps in Figures 4 and 5 also confirm that their distributions are consistent.

We note that the value of the NRMSE of the w_p of the Ca II line in the QR is extremely large (593%). Figure 7 shows that the values predicted from the DNN are systematically larger than those from MLSI, resulting in the NRMSE (593%) and R^2 score ($\ll$1) being poor. However, the CC is 0.826, which shows that their distributions are well correlated. Also, the w_p parameter map in Figure 4 shows that their distributions are consistent. By applying the DNN to the different quiet Sun observations, we also found systematic differences for the predicted w_p parameter of the Ca II line, which are not shown here. It seems that our trained DNN has a limit on low values of w_p since it is not able to reproduce the significantly low values of w_p from the Ca II line in the QRs derived by MLSI. Even so, the DNN seems to predict the spatial distributions. This discrepancy may result from our training data sets being biased to the ARs, emerging flux regions, or pores, rather than to QRs due to the limitations of the seeing conditions. As we have mentioned, the w_p are small values, and their deviations are very small. The predicted value converges to the mean value of all the training data sets, which are biased to ARs or pores with larger values of w_p. This resulted in the DNN predicting the larger values of w_p for the QR than those measured by MLSI.

Actually, the Doppler width in the photosphere (w_p) is a pretty tiny value, considering the thermal width from the radiative intensity. In the MLSI calculations of Chae et al. (2020), therefore, the w_p was fixed to a small value for all the regions, which made the MLSI model fit well. To know the spatial variation, even though the w_p is tiny, the updated MLSI model of Chae et al. (2021a) indirectly inferred the w_p, by combining the thermal width converted from the far-wing intensity of each spatial location with the assumed nonthermal speed (1 km s^{-1}). Practically, the systematic differences in the tiny value of w_p do not significantly affect either the inversion or the prediction of the physical parameters in the chromosphere.

(a) MLSI (Ca II, QR)



(b) DNN model (Ca II, QR)

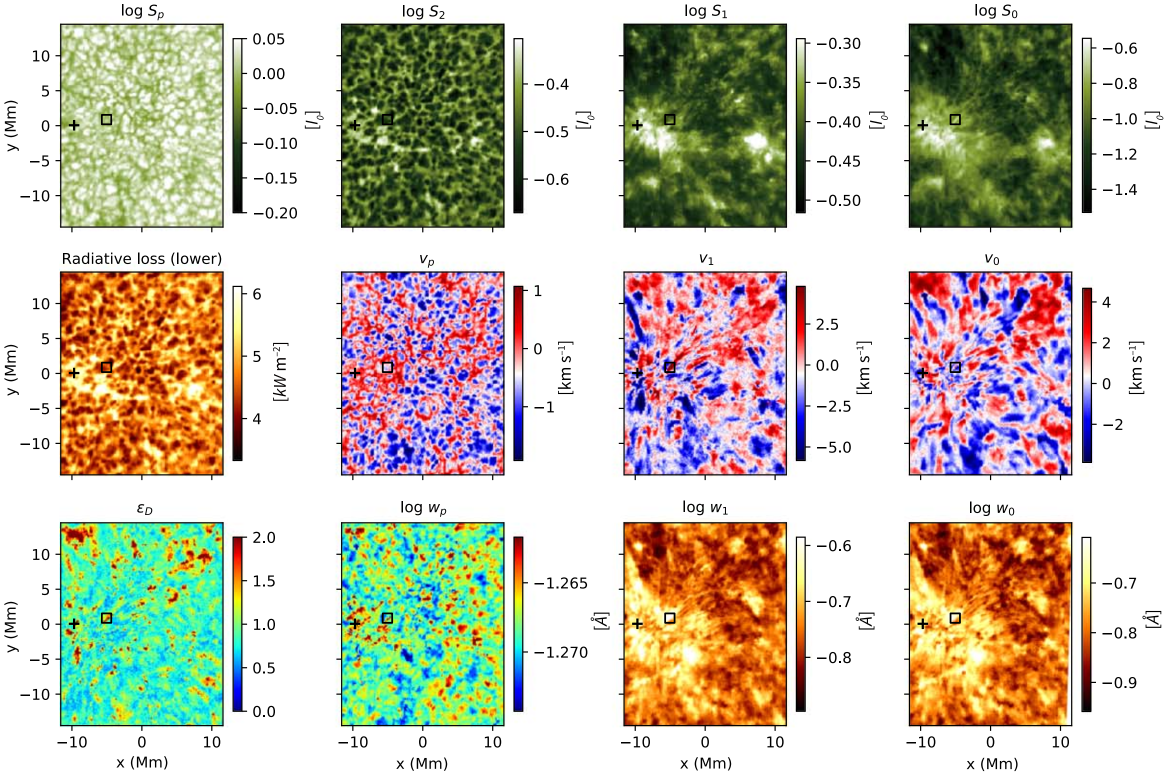
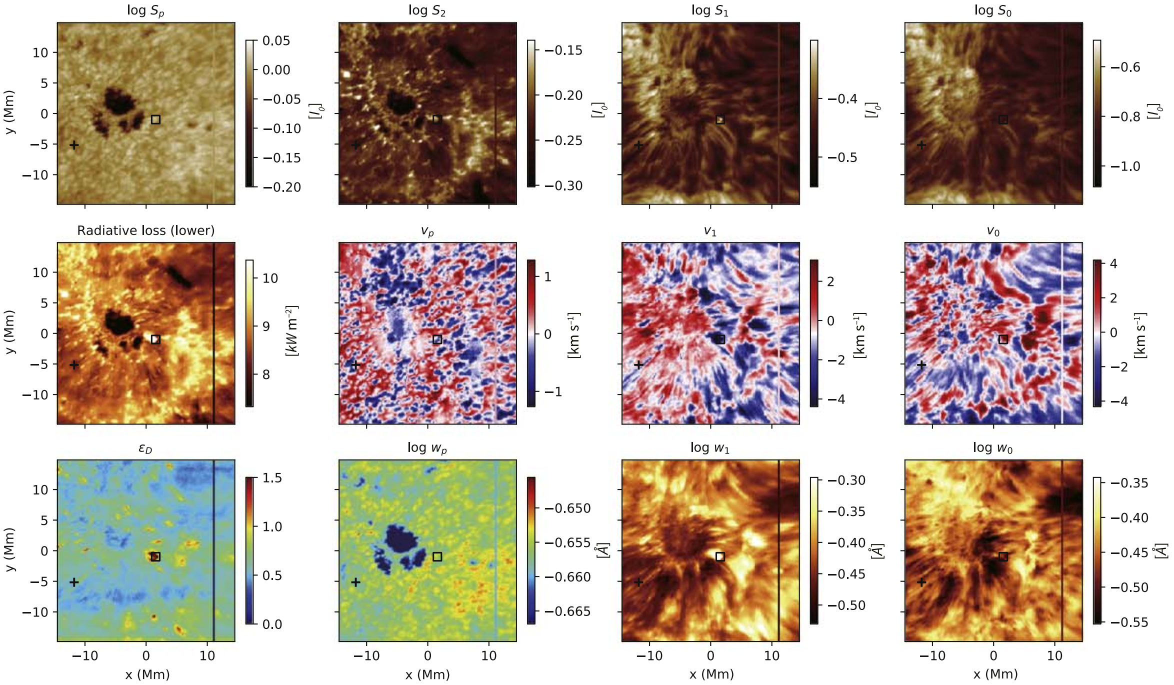


Figure 4. Physical parameter maps of the Ca II 8542 Å line using MLSI (upper three rows) and the DNN (lower three rows) for the QR. The cross and square symbols mark the locations of a network feature and a region with a large standard error, respectively.

4.2. Comparison of Line Profiles

We reproduce the spectral line profiles based on the three-layer atmosphere model using the predicted physical parameters derived from our DNN as the input. We then compare the observed spectral lines of Hα and Ca II 8542 Å with the synthesized line profiles using the physical parameters from the DNN. Figure 10 presents the intensity profiles of the network feature observed in the QR compared with the modeled intensity profiles from the DNN parameters for Hα (left panel)

(a) MLSI (Hα, AR)



(b) DNN model (Hα, AR)

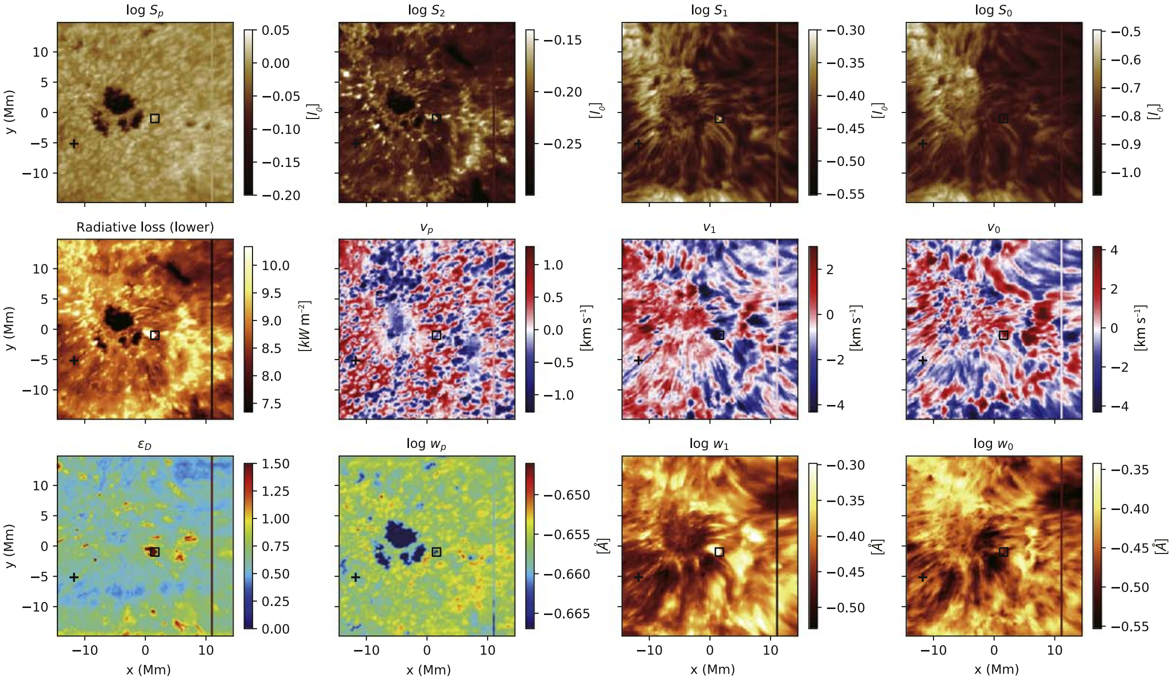


Figure 5. Physical parameter maps of Hα using MLSI (upper three rows) and the DNN (lower three rows) for the AR. The cross and square symbols mark the locations of a superpenumbral fibril and a region with a large standard error, respectively.

and Ca II 8542 Å (right panel). The synthesized intensity profile at the top of the upper chromosphere ($I_0(p)$) using the parameters ($p = S_p$, a_p, η, S_2, S_1, v_1, w_1, S_0, v_0, w_0) from the DNN is well matched with the observed spectral line profiles. Figure 11 displays the intensity profiles of the superpenumbral fibril observed in the AR, again showing the good agreement of the observed spectral line profile and the modeled intensity profile using the parameters predicted by the DNN.

Comparing the observed and the synthesized intensity profiles directly, we measure the parameter of the goodness of fit for the synthesized spectra reproduced by the physical parameters from the DNN, ϵ_D, which is defined in Equations (18–20) in Chae et al. (2021a):

$$\epsilon_D \equiv \left[\frac{1}{N}\sum_{k=1}^{N} f_k^2(p)\right]^{\frac{1}{2}}. \tag{2}$$

The f_k is the difference at each wavelength point (λ_k) between the observed intensity data and the synthesized intensity data using the DNN predicted parameters against the control parameters of the tightness of the constraint at each

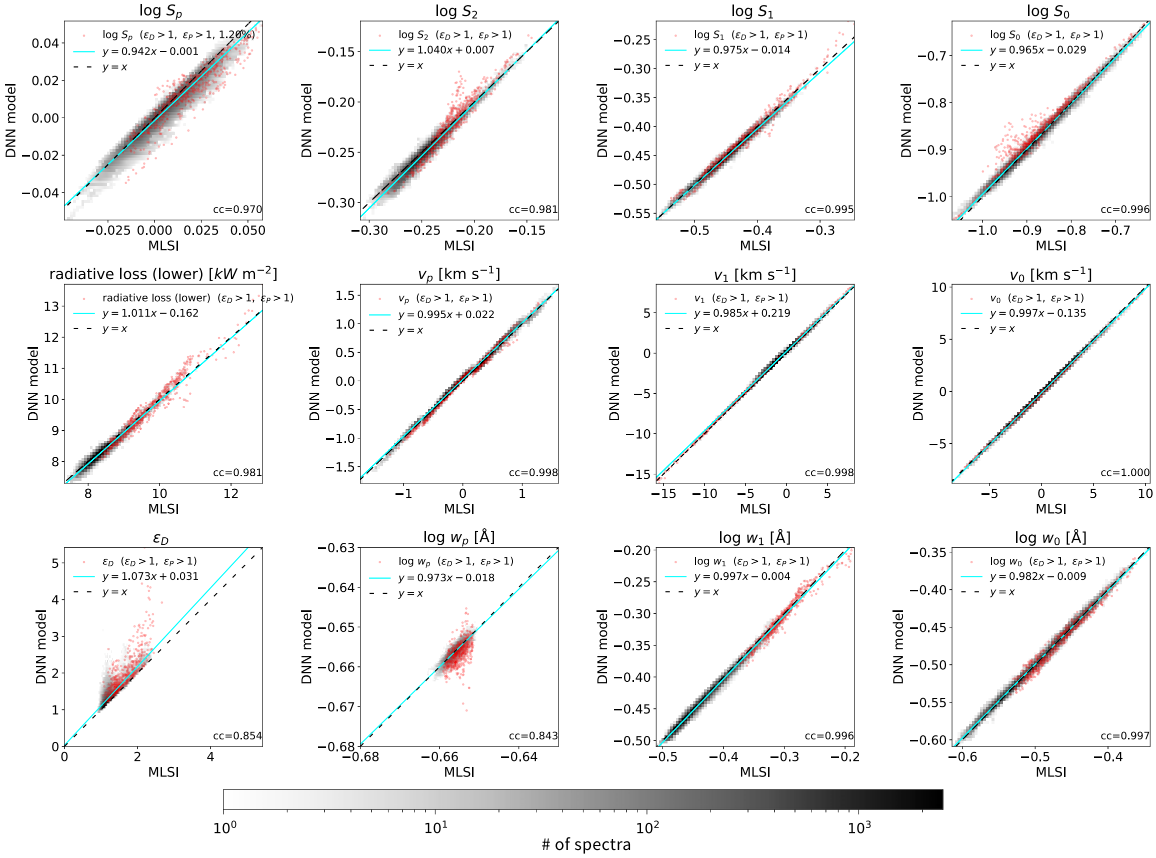


Figure 6. 2D density plots for each model parameter calculated from MLSI and our DNN for the Hα line profiles observed in the QR. The red dots denote the parameters with large errors of MLSI model fitting ($\epsilon_P > 1$ and $\epsilon_D > 1$). The dashed line indicates the equality of both model parameters, while the solid cyan line shows a linear fit to all the data points.

Table 3

Means and Standard Deviations of the Physical Parameters Derived from the DNN, with the MAE, NRMSE, CC, and Coefficient of Determination (R^2 Score) Metrics, for Evaluating the Accuracy of the DNN Parameter Predictions as Compared to Direct MLSI Calculations

Region	Physical Parameter	Mean ± Standard Deviation		MAE		NRMSE		CC [%]		R^2 Score	
		Hα	Ca II	Hα	Ca II	Hα	Ca II	Hα	Ca II	Hα	Ca II
QR	log S_p [I_0]	0.003 ± 0.014	0.017 ± 0.020	0.003	0.008	26.8	47.2	0.970	0.979	0.928	0.777
	log S_2 [I_0]	−0.251 ± 0.020	−0.487 ± 0.061	0.004	0.004	28.0	8.8	0.981	0.996	0.921	0.992
	log S_1 [I_0]	−0.448 ± 0.033	−0.405 ± 0.037	0.003	0.003	12.9	12.0	0.995	0.994	0.983	0.985
	log S_0 [I_0]	−0.881 ± 0.064	−1.038 ± 0.165	0.005	0.012	9.7	10.5	0.996	0.997	0.991	0.989
	v_p [km s^{-1}]	0.083 ± 0.469	−0.425 ± 0.501	0.032	0.055	8.1	14.6	0.998	0.993	0.993	0.979
	v_1 [km s^{-1}]	−0.217 ± 1.574	−0.504 ± 1.830	0.226	0.173	15.4	11.6	0.998	0.997	0.976	0.987
	v_0 [km s^{-1}]	0.119 ± 2.139	0.425 ± 1.465	0.136	0.046	7.0	4.2	1.000	0.999	0.995	0.998
	log w_p [Å]	−0.656 ± 0.002	−1.268 ± 0.002	0.001	0.010	62.2	593.2	0.843	0.826	0.613	−34.189
	log w_1 [Å]	−0.496 ± 0.040	−0.741 ± 0.053	0.003	0.004	11.5	10.1	0.996	0.996	0.987	0.990
	log w_0 [Å]	−0.441 ± 0.046	−0.784 ± 0.059	0.005	0.005	7.3	10.3	0.997	0.996	0.995	0.989
AR	log S_p [I_0]	−0.011 ± 0.027	0.003 ± 0.029	0.001	0.002	4.7	10.8	0.999	0.994	0.998	0.988
	log S_2 [I_0]	−0.220 ± 0.027	−0.443 ± 0.050	0.002	0.002	9.7	6.4	0.996	0.998	0.991	0.996
	log S_1 [I_0]	−0.426 ± 0.042	−0.335 ± 0.057	0.001	0.002	4.5	4.8	0.999	0.999	0.998	0.998
	log S_0 [I_0]	−0.788 ± 0.098	−0.749 ± 0.182	0.002	0.003	2.5	2.7	1.000	1.000	0.999	0.999
	v_p [km s^{-1}]	0.007 ± 0.423	0.001 ± 0.498	0.020	0.039	5.5	12.0	0.998	0.993	0.997	0.986
	v_1 [km s^{-1}]	−0.663 ± 1.241	−0.483 ± 1.663	0.087	0.047	9.6	4.6	0.995	0.999	0.991	0.998
	v_0 [km s^{-1}]	−0.071 ± 1.412	0.352 ± 1.029	0.041	0.030	3.9	4.7	0.999	0.999	0.998	0.998
	log w_p [Å]	−0.656 ± 0.003	−1.268 ± 0.003	0.001	0.001	26.7	40.5	0.966	0.934	0.928	0.836
	log w_1 [Å]	−0.414 ± 0.039	−0.705 ± 0.046	0.001	0.003	5.2	8.0	0.999	0.997	0.997	0.994
	log w_0 [Å]	−0.448 ± 0.035	−0.739 ± 0.055	0.001	0.002	5.1	6.2	0.999	0.998	0.997	0.996

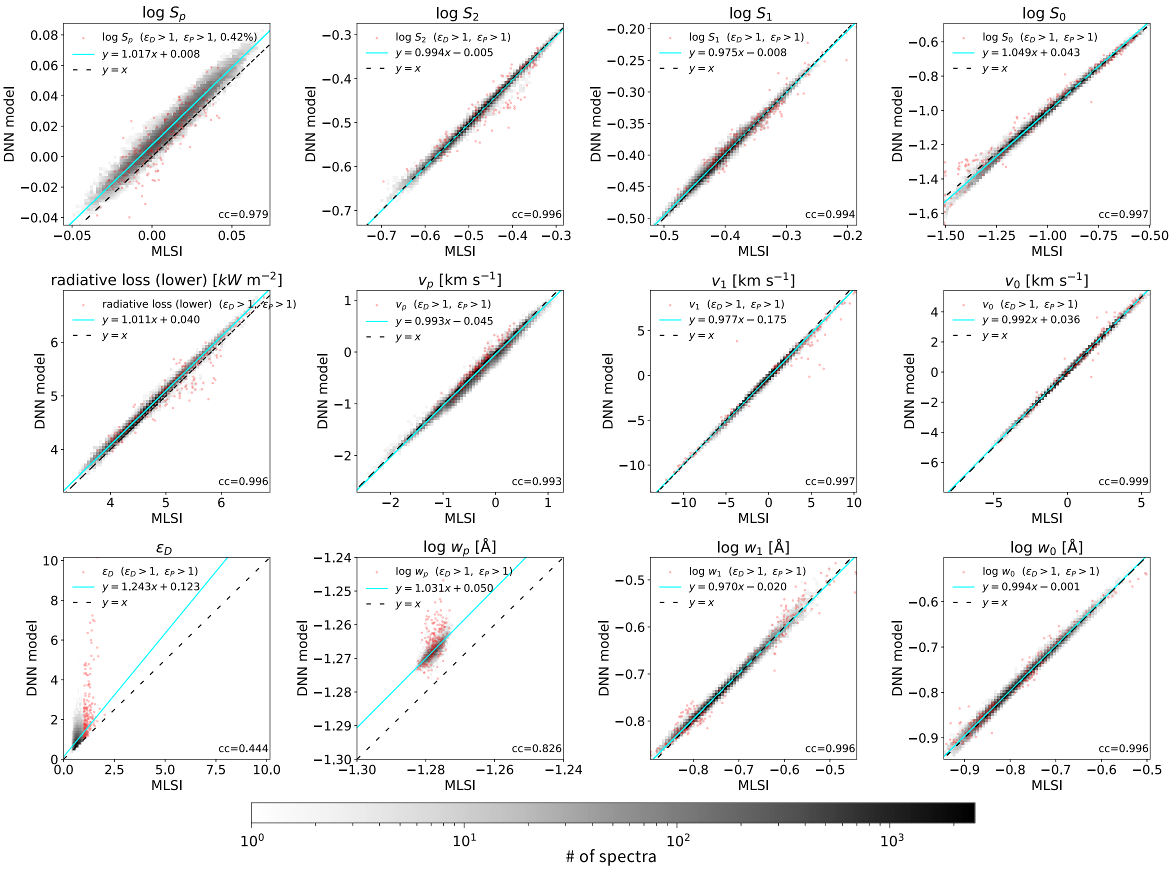


Figure 7. The same as Figure 6, but for the Ca II line profiles observed in the QR.

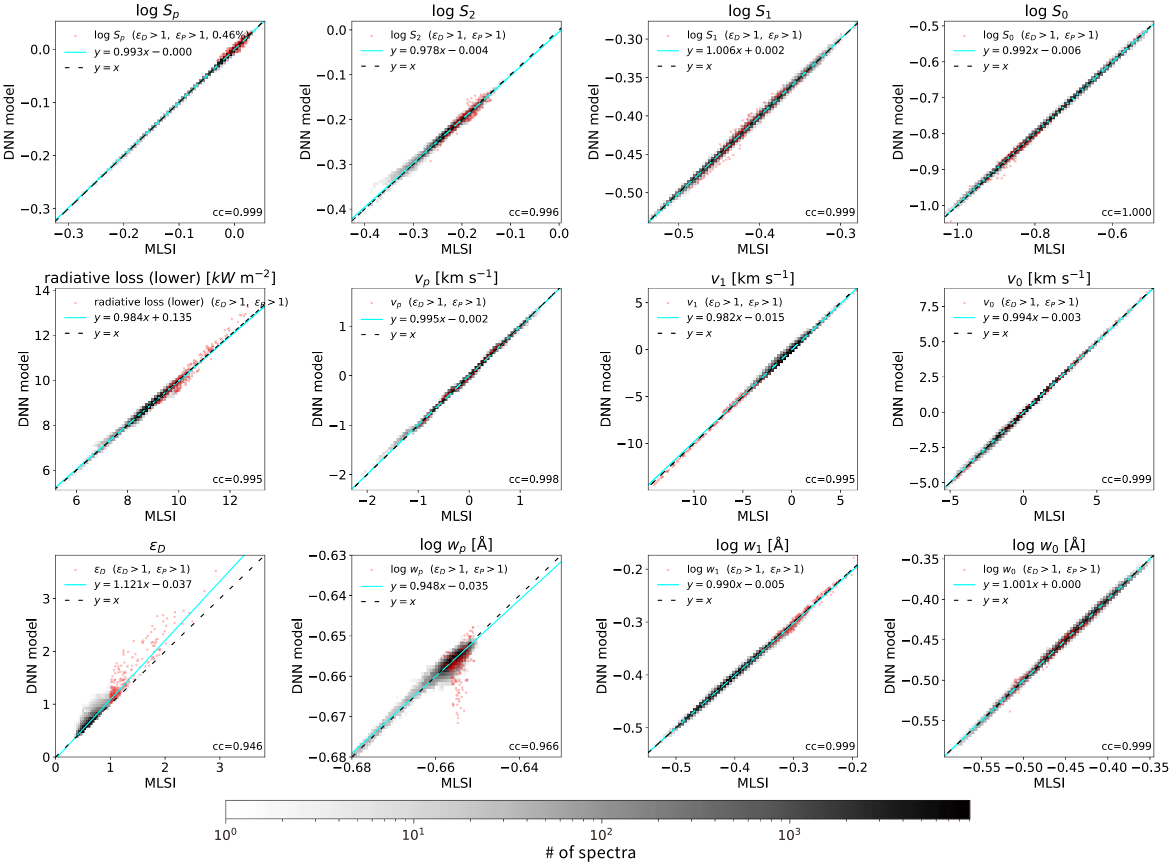


Figure 8. The same as Figure 6, but for the Hα line profiles observed in the AR.

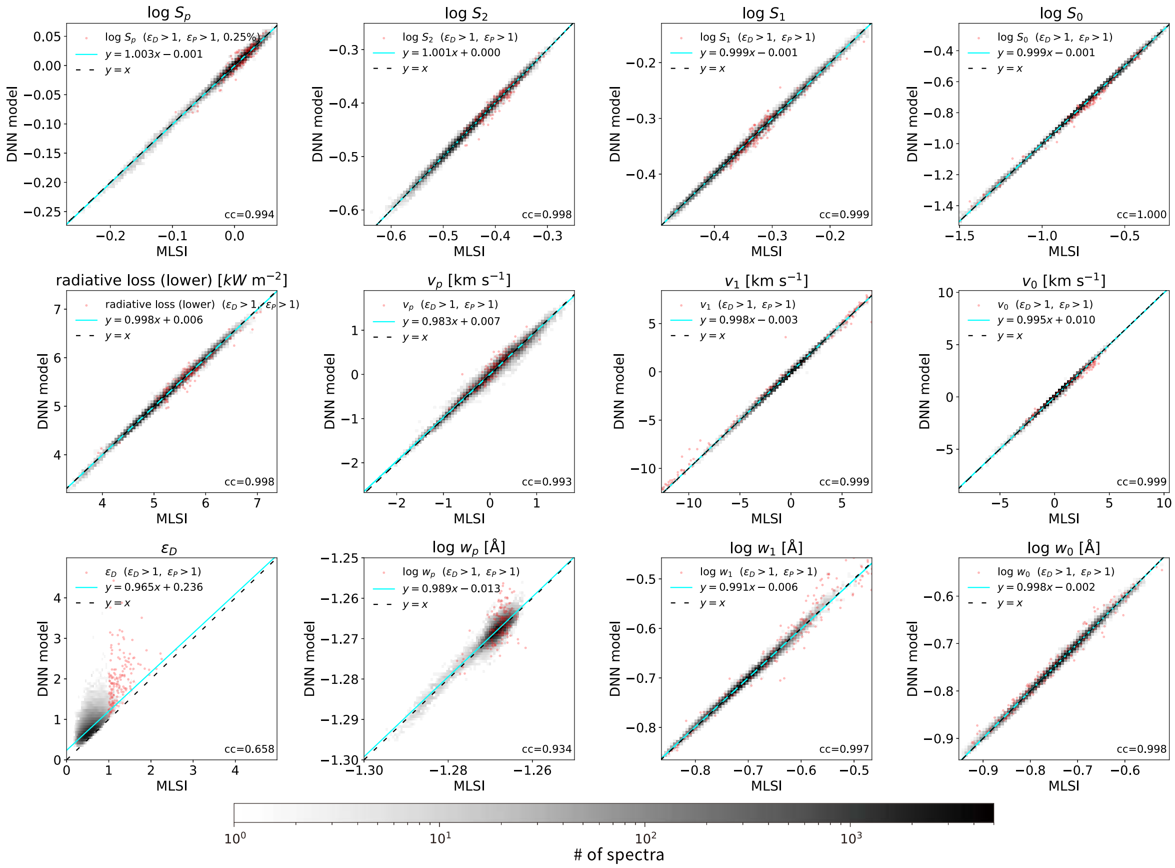


Figure 9. The same as Figure 6, but for the Ca II line profiles observed in the AR.

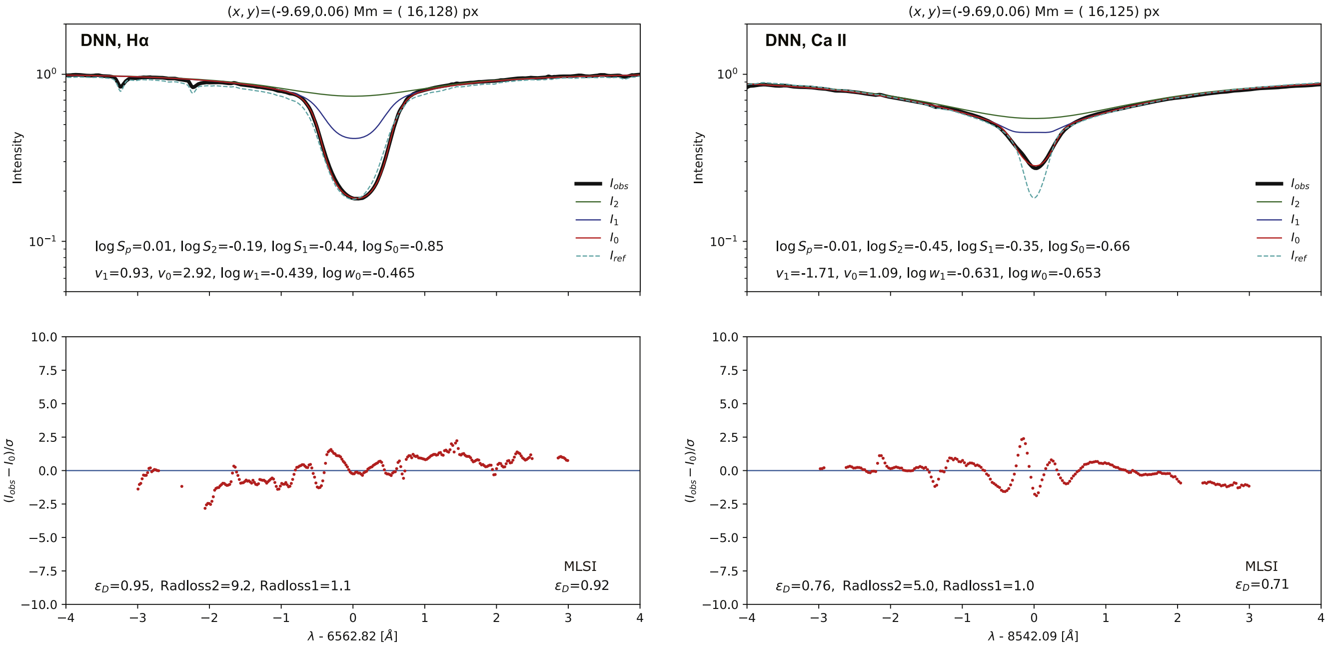


Figure 10. Three-layer model fitting of the Hα (left) and Ca II 8542 Å (right) line profiles taken from a network feature in the QR, marked by the cross symbol in Figure 4, using the physical parameters derived from the DNN. The parameter values are listed at the bottom of the figure. The thick solid black line indicates the observed line profile, while the solid green, blue, and red lines represent the modeled emergent intensity profiles at different layers—the top of the photosphere (layer 2), the lower chromosphere (layer 1), and the upper chromosphere (layer 0), respectively. The dashed cyan line indicates each reference profile obtained by taking the spatial average of the line spectra over the observed scan raster. The lower panels present the plots of the residuals between the observed and modeled intensity profiles. For comparison, the ϵ_D calculated from MLSI fitting for the observed spectra is also given.

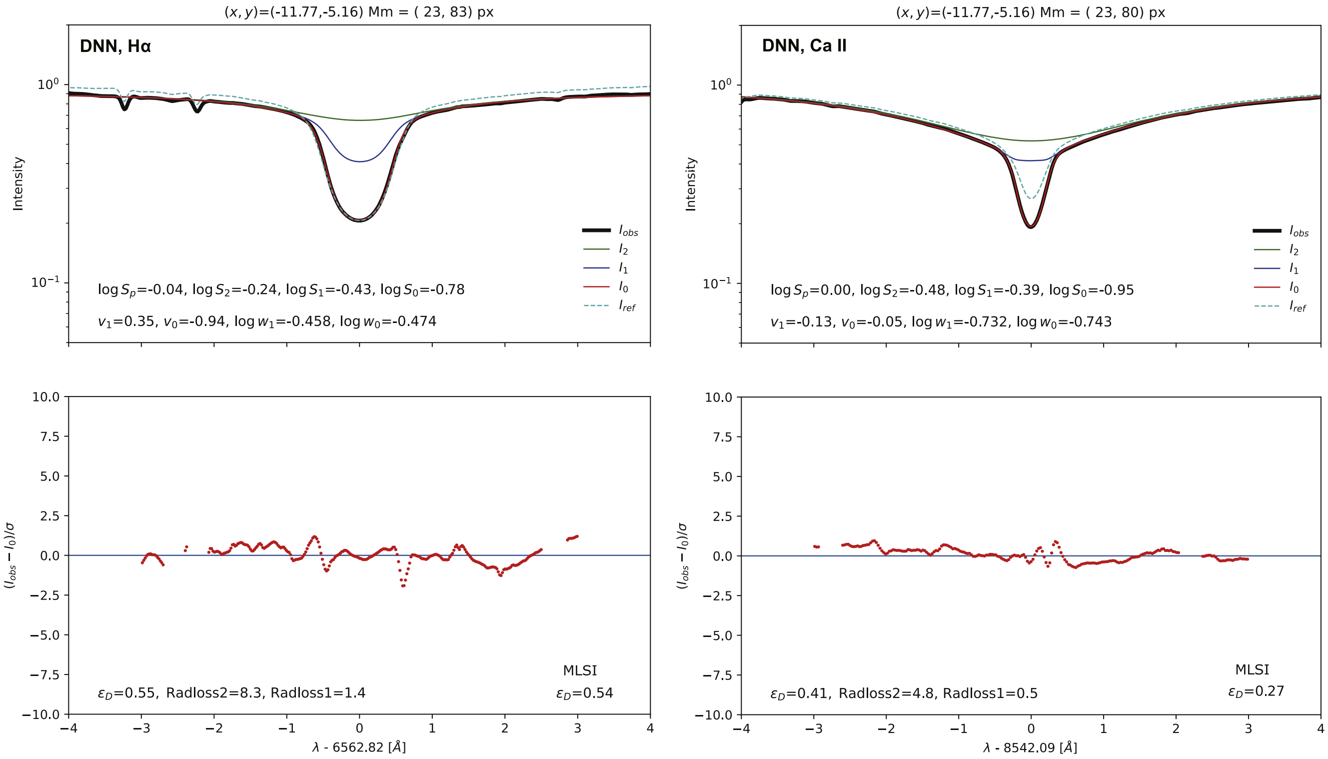


Figure 11. The same as Figure 10, but for a superpenumbral fibril in the AR, marked by the cross symbol in Figure 5.

data point, σ_k:

$$f_k(p) \equiv \frac{I_{\lambda_k,\mathrm{obs}} - I_{\lambda_k,0}(p)}{\sigma_k}. \tag{3}$$

Here, the control parameters for Hα and Ca II are set as $\sigma_k = \sigma_c\sqrt{I_{\lambda_k,\mathrm{obs}}}$, which is adopted from the photon noise form. Then, σ_k is equal to σ_c in the continuum $I_{\lambda_k,\mathrm{obs}} = 1$, and to $0.5\sigma_c$ in the core, with $I_{\lambda_k,\mathrm{obs}} = 0.25$. The noise level value of σ_c is set to 0.01, the same as the σ_c value in the MLSI model fitting (Chae et al. 2021a).

When we look at the scatter plots of ϵ_D in Figures 6–9, the ϵ_D values from the DNN for the well-fitting data sets with MLSI ($\epsilon_D < 1$ and $\epsilon P < 1$) have slightly larger values than the ϵ_D measured by MLSI fitting. Even so, the values of ϵ_D are mostly less than 2, which indicates that the predicted parameters from the DNN are satisfactory for those data sets. On the other hand, the acceptably fitted profiles ($1 < \epsilon_D < 3$ and $1 < \epsilon P < 3$ for MLSI; the red dots in the scatter plots) present large deviations between the observed and synthesized spectra. This implies that the regions where ϵ_D has large values in MLSI also have large ϵ_D values in the DNN. It is natural for our DNN not to be able to reproduce the specific features that are not well fitted by MLSI since the DNN is trained on the MLSI results. We take a look at the line profiles in the regions with large ϵ_D in Figures 12 and 13. The spectral profiles have significantly shifted components, which may be attributed to the presence of fast-moving plasma or cloud-like plasma, such as chromospheric jets.

4.3. Temporal Variations

We examine whether the physical parameters from the DNN are consistent with those MLSI over time. Figures 14 and 15 present the temporal variations of the Doppler velocity (v_0), the hydrogen temperature (T_{H}), and the nonthermal velocity (ξ) that were taken from the network feature in the QR and the superpenumbral fibril in the AR. The estimates of T_{H} and ξ are determined by the combination of the inferred Doppler widths at the upper chromosphere (w_0) from two absorption lines, the method for which is explained in Chae et al. (2020, 2021a). The temporal variations show that the Doppler velocity, temperature, and nonthermal velocity fluctuate over time. We find that the values inferred from the DNN are comparable to the values from MLSI. These values are physically reasonable for the features, as the T_{H} (ξ) for the network and the superpenumbral fibril are 11,400 K (9.6 km s^{-1}) and 10,200 K (8.4 km s^{-1}), on average, indicating that the DNN reliably predicts even the temporal variations of the physical parameters derived by MLSI.

4.4. Computing Time

Our investigation aims to reduce the cost of the MSLI calculations for inferring the chromospheric plasma properties from the enormous number of line profiles over position and time. The MLSI proposed in Chae et al. (2021a) is slower in fitting the observed absorption profiles. It takes 95 ms for each Hα line profile, and 81 ms for each Ca II line profile, with Python 3.8 software on a computer server with a 3.8 GHz AMD Ryzen Threadripper 3960X CPU. Our server is equipped with GPUs, but the MLSI calculation is only conducted using the CPU. It therefore takes about 2.5 hr for Hα and 2 hr for Ca II when reproducing the same field of view ($40'' \times 40''$) of the sample image of the AR.

By comparison, using the same system, the DNN reproduces the physical parameter maps for the field of view of the AR within 20 s, using the CPU. It takes 0.3–0.4 ms on average to reproduce the physical parameters for each absorption line profile. To synthesize the spectral profiles using the physical parameters recovered from the DNN takes 3.4 ms for each Hα

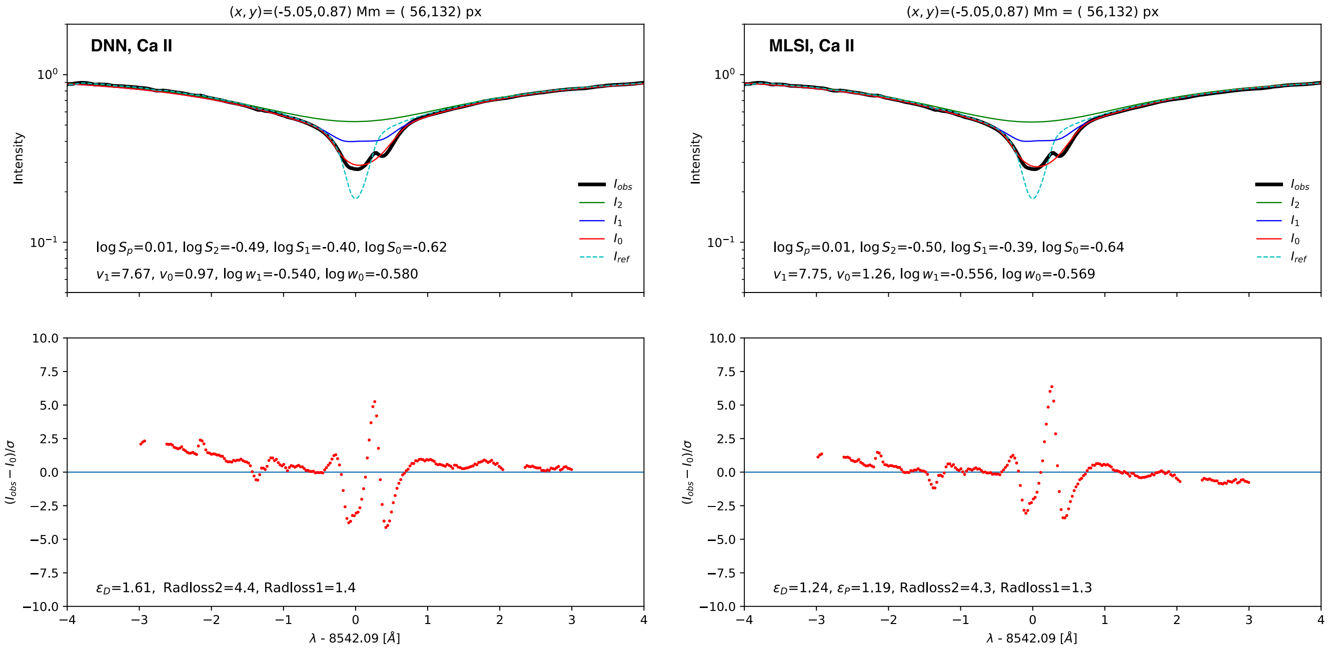


Figure 12. Three-layer model fitting of the Ca II 8542 Å line profiles taken from the region with a large value of ϵ_D, marked by the square symbol in Figure 4 (QR), using the different models (left: our DNN; right: MLSI).

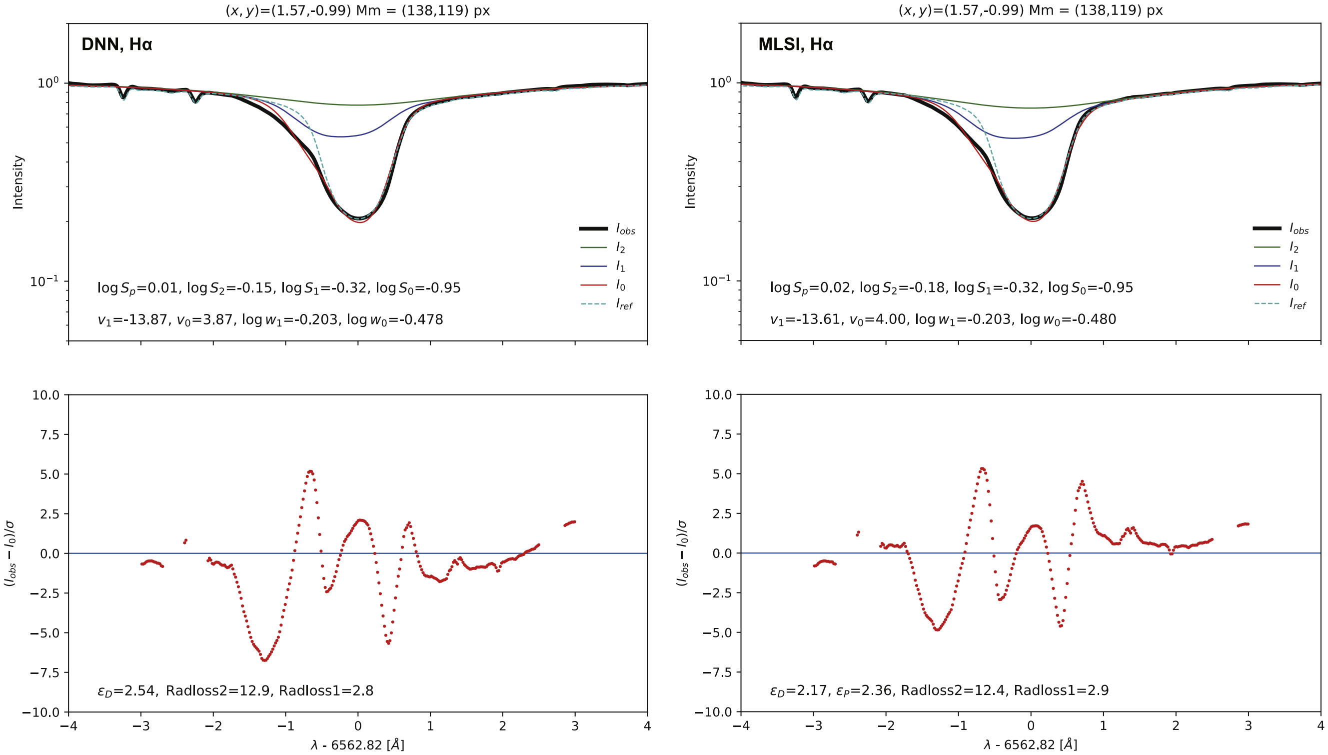


Figure 13. Three-layer model fitting of the Hα line profiles taken from the region with a large value of ϵ_D, marked by the square symbol in Figure 5 (AR), using the different models (left: our DNN; right: MLSI).

and Ca II line profile, on average. The computing time of the DNN for the parameters (line profiles) is 250 (25) times faster than MLSI. Moreover, if we use the GPUs that this system is equipped with, for acceleration, it takes 0.03 ms to predict the physical parameters and 2.84 ms to synthesize the spectra. This advantage of the fast inversion model allows us to investigate spatial distributions and temporal evolutions of chromospheric plasma over a reasonably short period of time.

5. Summary

We have produced a DNN application for estimating the physical parameter outputs from MLSI, which was developed by Chae et al. (2020, 2021a), in order to make applications to large data sets feasible. Our DNN is extremely fast, producing physical parameters in the photosphere and chromosphere, as well as synthesized intensity profiles of both absorption lines, in about 2.84 ms for each line profile, using GPUs. The test

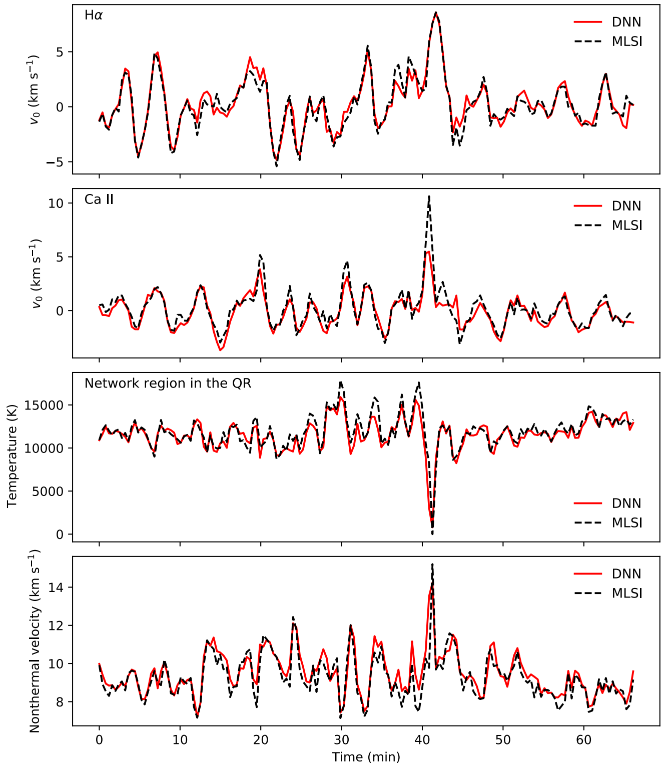


Figure 14. Temporal variations of the Doppler velocity (v_0), the temperature of hydrogen, and the nonthermal velocity at the upper chromosphere of the network feature in the QR, determined from the DNN and MLSI.

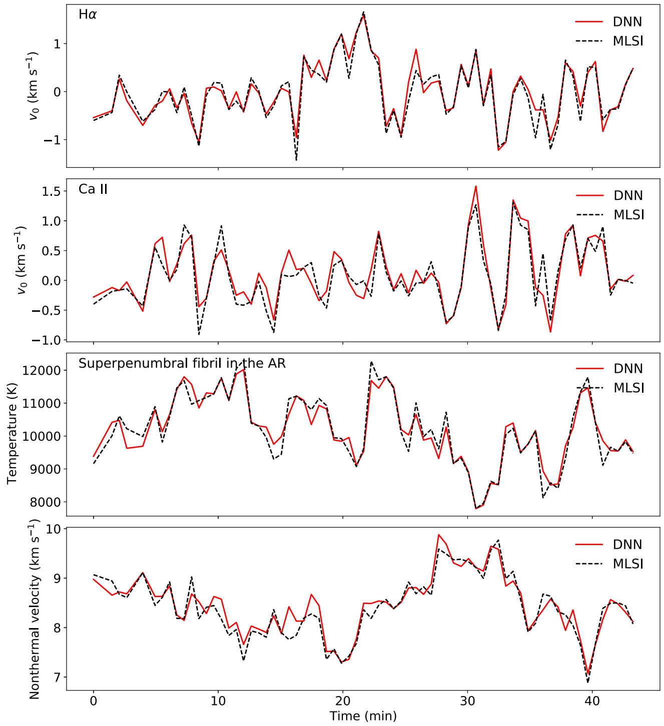


Figure 15. Temporal variation of the Doppler velocity (v_0), the temperature of hydrogen, and the nonthermal velocity at the upper chromosphere of the superpenumbral fibril in the AR, determined from the DNN and MLSI.

results of the DNN described in Section 4 prove that this model is able to accurately generate the physical parameters of chromosphere plasma in both QRs and ARs. Moreover, the variations of hydrogen temperature and nonthermal velocity in the upper chromosphere can also be investigated, by virtue of the high precision of the Doppler width measurements, as with MLSI.

Despite the fact that the high performance of the DNN has been demonstrated, there are still caveats to consider. Our approach, using a deep learning tool, is a kind of supervised learning, which is significantly affected by the training data, the observed spectra, and the derived parameters, based on MLSI. If the MLSI model is not able to make a good fit to an observed spectrum, the trained DNN will not well reproduce the physical parameters or the modeled intensity profile. Therefore, first, the model is applicable to QRs and ARs, but not to flares that have abnormal profiles, such as emission profiles due to strongly heated plasma or sudden density enhancement. Those flare spectra could possibly be analyzed by a sophisticated model, adding more atmospheric layers or combining a specific model for flare spectra, such as an INN using RADYN with Hα and Ca II, as proposed by Osborne et al. (2019). Second, most of the training data sets for the DNN from FISS observations consist of sunspots, pores, or emerging flux regions near the disk center. Therefore, applying our DNN to QRs or coronal holes may be less reliable, and it may not be appropriate for observations closer to the limb. Third, atypical profiles—e.g., spectra with largely shifted components or very broad profiles, which imply fast-moving or cloud-like features in the chromosphere—are fairly fitted ($1 < \epsilon_D < 3$) compared to the flaring spectra. Still, when the predicted physical parameters result in the synthesized spectra being significantly different from the observations, and hence having large values of ϵ_D, the DNN results should be interpreted with caution. In the same way, however, it may be possible to utilize the value of ϵ_D as an identifier of anomalous profiles, sometimes the result of fast-moving or strongly heated plasma, which would need a more detailed analysis.

We are planning to adapt our DNN to the FISS observation database. FISS provides good-quality imaging data and spectra with a high temporal resolution, which is appropriate for investigating oscillations, waves, or small flaring events (Yang et al. 2014; Cho et al. 2016; Kang et al. 2019; Kwak et al. 2020; Chae et al. 2021b). The DNN will allow us to analyze the huge amount of FISS data much faster than by using MLSI. For example, in 2013, FISS observed about 7400 scan rasters for both Hα and Ca II ($\sim 8.8 \times 10^8$ spectra). It could take 2.5 yr to obtain the inverted parameters and synthesized spectra using MLSI, but our DNN reproduced the parameters and synthesized spectra in about a month. We can be much faster in applying the DNN to FISS observations. Moreover, the measured model parameters—such as source functions over position and time, with their high precision—make it possible to investigate the height and temporal variations of temperature or radiative losses in the photosphere and chromosphere, which is important for understanding the heating in the solar chromosphere. We also note that our DNN could possibly be applied to other observations or simulations of Hα and Ca II, for instruments or models covering a similar wavelength range as FISS, by preprocessing the data set. For instance, by optimizing their wavelength scales to FISS spectra, we have been able to apply our DNN to other data sets. The trained model and code will be added to the FISSPy[6] python package

---

[6] http://fiss.snu.ac.kr/fisspy/

soon. We also plan to provide physical parameter maps from the DNN, based on MLSI, via the FISS web page.[7]

This research was supported by the National Research Foundation of Korea (NRF-2021R1A2C1010881 and NRF-2020R1A2C2004616), funded by the Korean Ministry of Science and ICT. Y.J.M. acknowledges support from the Basic Science Research Program through the National Research Foundation, funded by the Ministry of Education (NRF-2019R1A2C1002634). K.C. was supported by the Basic Science Research Program through the National Research Foundation of Korea, funded by the Ministry of Education (NRF-2020R1I1A1A01068789). The GST operation is partly supported by the Korea Astronomy and Space Science Institute, the Seoul National University, the Key Laboratory of Solar Activities of the Chinese Academy of Sciences (CAS), and the Operation, Maintenance and Upgrading Fund of CAS for Astronomical Telescopes and Facility Instruments.

Facility: Big Bear Solar Observatory (Fast Imaging Solar Spectrograph).

Software: Astropy (Astropy Collaboration et al. 2013, 2018), Fisspy (http://fiss.snu.ac.kr/fisspy), Keras (Chollet et al. 2015), Matplotlib (Hunter 2007), Numpy (Harris et al. 2020), Scikit-learn (Pedregosa et al. 2011), Scipy (Virtanen et al. 2020), TensorFlow (Abadi et al. 2015).

ORCID iDs

Kyoung-Sun Lee https://orcid.org/0000-0002-4329-9546
Jongchul Chae https://orcid.org/0000-0002-7073-868X
Eunsu Park https://orcid.org/0000-0003-0969-286X
Yong-Jae Moon https://orcid.org/0000-0001-6216-6944
Hannah Kwak https://orcid.org/0000-0001-8619-9345
Kyuhyoun Cho https://orcid.org/0000-0001-7460-725X

References

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, https://www.tensorflow.org/
Allred, J. C., Hawley, S. L., Abbett, W. P., & Carlsson, M. 2005, ApJ, 630, 573
Ardizzone, L., Kruse, J., Wirkert, S., et al. 2018, arXiv:1808.04730
Asensio Ramos, A., & Díaz Baso, C. J. 2019, A&A, 626, A102
Asensio Ramos, A., Trujillo Bueno, J., & Landi Degl Innocenti, E. 2008, ApJ, 683, 542
Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, AJ, 156, 123
Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33
Beckers, J. M. 1964, PhD thesis, Sacramento Peak Observatory, Air Force Cambridge Research Laboratories, Mass., USA
Carlsson, M., De Pontieu, B., & Hansteen, V. H. 2019, ARA&A, 57, 189
Carroll, T. A., & Staude, J. 2001, A&A, 378, 316
Chae, J. 2014, ApJ, 780, 109
Chae, J., Cho, K., Kang, J., et al. 2021a, JKAS, 54, 139
Chae, J., Cho, K., Nakariakov, V. M., Cho, K.-S., & Kwon, R.-Y. 2021b, ApJL, 914, L16
Chae, J., Madjarska, M. S., Kwak, H., & Cho, K. 2020, A&A, 640, A45
Chae, J., Park, H.-M., Ahn, K., et al. 2013, SoPh, 288, 1
Chae, J., Yang, H., Park, H., et al. 2014, ApJ, 789, 108
Cheung, C. M. M., Wright, P. J., Galvez, R., et al. 2018, AGUFM, 2018, SM31D–3536
Cho, K., Lee, J., Chae, J., et al. 2016, SoPh, 291, 2391
Chollet, F. 2015, Keras, https://keras.io
Chollet, F. 2017, in 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (Piscataway, NJ: IEEE)
de la Cruz Rodríguez, J., Leenaarts, J., & Ramos, A. A. 2016, ApJL, 830, L30
de la Cruz Rodríguez, J., Leenaarts, J., Danilovic, S., & Uitenbroek, H. 2019, A&A, 623, A74
Glorot, X., & Bengio, Y. 2010, PMLR, 9, 249
Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., et al. 2014, arXiv:1406.2661
Gudiksen, B. V., Carlsson, M., Hansteen, V. H., et al. 2011, A&A, 531, A154
Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, Natur, 585, 357
He, K., Zhang, X., Ren, S., & Sun, J. 2015, arXiv:1512.03385
Hunter, J. D. 2007, CSE, 9, 90
Kang, J., Chae, J., Nakariakov, V. M., et al. 2019, ApJL, 877, L9
Kim, T., Park, E., Lee, H., et al. 2019, NatAs, 3, 397
Kingma, D. P., & Ba, J. 2014, arXiv:1412.6980
Kwak, H., Chae, J., Madjarska, M. S., Cho, K., & Song, D. 2020, A&A, 642, A154
Lecun, Y., & Bengio, Y. 1995, in Convolutional Networks for Images, Speech, and Time-series, ed. M. Arbib (Cambridge, MA: MIT Press)
LeCun, Y., Bengio, Y., & Hinton, G. 2015, Natur, 521, 436
Lee, H., Park, E., & Moon, Y.-J. 2021, ApJ, 907, 118
Lemen, J. R., Title, A. M., Akin, D. J., et al. 2012, SoPh, 275, 17
Lim, D., Moon, Y.-J., Park, E., & Lee, J.-Y. 2021, ApJL, 915, L31
Milić, I., & van Noort, M. 2018, A&A, 617, A24
Osborne, C. M. J., Armstrong, J. A., & Fletcher, L. 2019, ApJ, 873, 128
Park, E., Moon, Y.-J., Lee, J.-Y., et al. 2019, ApJ, 884, L23
Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, JMLR, 12, 2825
Pesnell, W. D., Thompson, B. J., & Chamberlin, P. C. 2012, SoPh, 275, 3
Ramachandran, P., Zoph, B., & Le, Q. V. 2017, arXiv:1710.05941
Ronneberger, O., Fischer, P., & Brox, T. 2015, arXiv:1505.04597
Sainz Dalda, A., de la Cruz Rodríguez, J., De Pontieu, B., & Gošić, M. 2019, ApJL, 875, L18
Socas-Navarro, H., de la Cruz Rodríguez, J., Asensio Ramos, A., Trujillo Bueno, J., & Ruiz Cobo, B. 2015, A&A, 577, A7
Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. 2016, arXiv:1602.07261
Tziotziou, K. 2007, in ASP Conf. Ser. 368, The Physics of Chromospheric Plasmas, ed. P. Heinzel, I Dorotovič, & R. J. Rutten (San Francisco, CA: ASP), 217
Uitenbroek, H. 2001, ApJ, 557, 389
Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, NatMe, 17, 261
Yang, H., Chae, J., Lim, E.-K., et al. 2014, ApJL, 790, L4
Zhang, H., Shao, J., & Salakhutdinov, R. 2018, arXiv:1806.01845

---

[7] http://fiss.snu.ac.kr