Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

NJIT New Jersey's Science & Technology University

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

ABSTRACT

MACHINE LEARNING AND COMPUTER VISION IN SOLAR PHYSICS

by
Haodi Jiang

In the recent decades, the difficult task of understanding and predicting violent solar eruptions and their terrestrial impacts has become a strategic national priority, as it affects the life of human beings, including communication, transportation, the power grid, national defense, space travel, and more. This dissertation explores new machine learning and computer vision techniques to tackle this difficult task. Specifically, the dissertation addresses four interrelated problems in solar physics: magnetic flux tracking, fibril tracing, Stokes inversion and vector magnetogram generation.

First, the dissertation presents a new deep learning method, named SolarUnet, to identify and track solar magnetic flux elements in observed vector magnetograms. The method consists of a data preprocessing component that prepares training data from a physics-based tool, a deep learning model implemented as a U-shaped convolutional neural network for fast and accurate image segmentation, and a postprocessing component that prepares tracking results. The tracking results can be used in deriving statistical parameters of the local and global solar dynamo, allowing for sophisticated analyses of solar activities in the solar corona and solar wind.

Second, the dissertation presents another new deep learning method, named FibrilNet, for tracing chromospheric fibrils in Hα images of solar observations. FibrilNet is a Bayesian convolutional neural network, which adopts the Monte Carlo dropout sampling technique for probabilistic image segmentation with uncertainty quantification capable of handling both aleatoric uncertainty and epistemic uncertainty. The traced Hα fibril structures provide the direction of magnetic fields, where

the orientations of the fibrils can be used as a constraint to improve the nonlinear force-free extrapolation of coronal fields.

Third, the dissertation presents a stacked deep neural network (SDNN) for inferring line-of-sight (LOS) velocities and Doppler widths from Stokes profiles collected by GST/NIRIS at Big Bear Solar Observatory. Experimental results show that SDNN is faster, while producing smoother and cleaner LOS velocity and Doppler width maps, than a widely used physics-based method. Furthermore, the results demonstrate the better learning capability of SDNN than several related machine learning algorithms. The high-quality velocity fields obtained through Stokes inversion can be used to understand solar activity and predict solar eruptions.

Fourth, the dissertation presents a generative adversarial network, named MagNet, for generating vector components to create synthetic vector magnetograms of solar active regions. MagNet allows us to expand the availability of photospheric vector magnetograms to the period from 1996 to present, covering solar cycles 23 and 24, where photospheric vector magnetograms were not available prior to 2010. The synthetic vector magnetograms can be used as input of physics-based models to derive important physical parameters for studying the triggering mechanisms of solar eruptions and for forecasting eruptive events.

Finally, implementations of some of the deep learning-based methods using Jupyter notebooks and Google Colab with GitHub are presented and discussed.

MACHINE LEARNING AND COMPUTER VISION IN SOLAR PHYSICS

by
Haodi Jiang

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Science

Department of Computer Science

December 2021

Copyright © 2021 by Haodi Jiang
ALL RIGHTS RESERVED

APPROVAL PAGE

MACHINE LEARNING AND COMPUTER VISION IN SOLAR PHYSICS

Haodi Jiang

Dr. Jason T. L. Wang, Dissertation Advisor — Date
Professor of Computer Science, New Jersey Institute of Technology

---

Dr. James Calvin, Committee Member — Date
Professor of Computer Science, New Jersey Institute of Technology

---

Dr. Xiaoning Ding, Committee Member — Date
Associate Professor of Computer Science, New Jersey Institute of Technology

---

Dr. Haimin Wang, Committee Member — Date
Distinguished Professor of Physics, New Jersey Institute of Technology

---

Dr. Xiaoli Bai, Committee Member — Date
Associate Professor of Mechanical and Aerospace Engineering, Rutgers University

BIOGRAPHICAL SKETCH

Author: Haodi Jiang

Degree: Doctor of Philosophy

Date: December 2021

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Science,
  New Jersey Institute of Technology, Newark, NJ, 2021
- Master of Science in Computer Engineering,
  Florida International University, Miami, FL, 2014
- Bachelor of Science in Network Engineering,
  Southwest University, Chongqing, China, 2010

Major: Computer Science

Presentations and Publications:

H. Jiang, Q. Li, Z. Hu, N. Liu, Y. Abduallah, J. Jing, G. Zhang, Y. Xu, W. Hsu, J. T. L. Wang and H. Wang, "Creating Photospheric Vector Magnetograms of Solar Active Regions Using Generative Adversarial Networks," Submitted.

H. Jiang, Q. Li, Y. Xu, W. Hsu, K. Ahn, W. Cao, J. T. L. Wang and H. Wang, "Inferring Line-of-Sight Velocities and Doppler Widths from Stokes Profiles of GST/NIRIS," Submitted.

H. Jiang, J. Jing, J. Wang, C. Liu, Q. Li, Y. Xu, J. T. L. Wang and H. Wang, "Tracing Hα Fibrils through Bayesian Deep Learning," *The Astrophysical Journal Supplement Series*, 256:20, 2021.

H. Jiang, J. Wang, C. Liu, J. Jing, H. Liu, J. T. L. Wang and H. Wang, "Identifying and Tracking Solar Magnetic Flux Elements with Deep Learning," *The Astrophysical Journal Supplement Series*, 250:5, 2020.

H. Jiang, T. Turki and J. T. L. Wang, "DLGraph: Malware Detection Using Deep Learning and Graph Embedding," *Proceedings of the 17th IEEE International Conference on Machine Learning and Applications*, Orlando, Florida, USA, December 2018, pp. 1029-1033.

S. Zhang, Z. Du, J. T. L. Wang and H. Jiang, "Discovering Frequent Induced Subgraphs from Directed Networks," *Intelligent Data Analysis*, Vol. 22, No. 6, 2018, pp. 1279-1296.

H. Jiang, T. Turki and J. T. L. Wang, "Reverse Engineering Gene Regulatory Networks Using Graph Mining," *Proceedings of the 14th International Conference on Machine Learning and Data Mining in Pattern Recognition, Part I*, New York, USA, July 2018, pp. 335-349.

H. Jiang, T. Turki and J. T. L. Wang, "Reverse Engineering Regulatory Networks in Cells Using a Dynamic Bayesian Network and Mutual Information Scoring Function," *Proceedings of the 16th IEEE International Conference on Machine Learning and Applications*, Orlando, Cancun, Mexico, December 2017, pp. 761-764.

H. Jiang, "Stokes Inversion with Stacked Deep Neural Networks," *2021 American Geophysical Union Fall Meeting*. New Orleans, LA, USA, December 16, 2021, Poster Presentation.

H. Jiang, "Using Deep Learning to Detect and Trace H-alpha Fibrils," *238th Virtual Meeting of the American Astronomical Society,* June 7, 2021, Poster Presentation.

H. Jiang, "Big Data-Enabled AI-Powered Space Weather Analytics with Community-Driven Cyberinfrastructure," *8th Annual New Jersey Big Data Alliance Symposium — Smart State: Big Data for Community Impact*, April 30, 2021, Oral Presentation.

H. Jiang, "Using Deep Learning to Track Solar Magnetic Flux Elements," *50th Anniversary Meeting of the American Astronomical Society Solar Physics Division*, August 19, 2020, Poster Presentation.

H. Jiang, "DLGraph: Malware Detection Using Deep Learning and Graph Embedding," *17th IEEE International Conference on Machine Learning and Applications*, December 17, 2018, Oral Presentation.

H. Jiang, "Reverse Engineering Gene Regulatory Networks Using Graph Mining," *14th International Conference on Machine Learning and Data Mining in Pattern Recognition*, July 2018, Oral Presentation.

To My Beloved Parents, Shenglong Jiang and Xiaohua He;
Sister, Yiyi Jiang; Spouse, Yuexin Liu; and Son, Ethan Jiang.

ACKNOWLEDGMENT

First and foremost, I would like to express my sincere gratitude and appreciation to my dissertation advisor, Dr. Jason T. L. Wang, for his invaluable guidance, support, and encouragement of my study and research. His expertise and professional dedication to scientific research has had a profound effect upon me. The journey of being his PhD student is a truly precious experience and a lifelong benefit.

Second, I am extremely grateful to Dr. Haimin Wang, Dr. Chang Liu, Dr. Ju Jing and Dr. Yan Xu for providing insightful feedback and knowledge on solar physics. I also thank Dr. James Calvin, Dr. Xiaoning Ding and Dr. Xiaoli Bai for their comments and feedback on my research and serving on my dissertation committee.

It is worthwhile to note that without the generous support from the Department of Computer Science, it would be impossible for me to complete the PhD degree. The research projects in this dissertation were supported by the U.S. National Science Foundation under Grant No. AGS-1927578.

Besides, I would like to thank my labmates Dr. Hao Liu, Dr. Zhihang Hu and Yasser Abduallah, and collaborators from Department of Physics: Dr. Qin Li, Dr. Jiasheng Wang and Nian Liu, for their contributions to the work in the dissertation. I would like to extend special thanks to Dr. Ruogu Fang for inspiring me to pursue a career in research.

Most importantly, I truly appreciate my family for being my emotional anchor. I would like to thank my family members, Shenglong Jiang, Xiaohua He, Yiyi Jiang, Ethan Jiang, Mingli Liu and Pan Li for their support and understanding throughout my life. In particular, I am extremely grateful to my wife, Yuexin Liu, for her love, support and encouragement.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

Solar physics is the branch of astrophysics that specializes in the study of the Sun. It exploits and explains the detailed measurements, i.e., the observation data, that are possible only for our closest star. It intersects with many disciplines of pure physics, astrophysics, and computer science. Understanding the solar activities and predicting violent solar eruptions and their terrestrial impacts is a difficult task, which has become a national priority, as it affects the life of human beings such as power grids and communications networks, and affect spacecraft and satellites.

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, allowing patterns identification and decisions making with minimal human intervention. Computer vision is an interdisciplinary scientific field that seeks to develop techniques to help computers gain high-level understanding of the visual world, i.e., the images and videos. These techniques have drawn a lot of interest in recent years due to its effectiveness in processing big and complex observational data gathered from diverse instruments.

This dissertation explores new machine learning and computer vision techniques to addresses four interrelated problems in solar physics: magnetic flux tracking, fibril tracing, Stokes inversion and vector magnetogram generation.

First, this dissertation proposes a new deep learning method, called SolarUnet, to identify and track solar magnetic flux elements or features in observed vector magnetograms based on the Southwest Automatic Magnetic Identification Suite (SWAMIS). Our method consists of a data pre-processing component that prepares

training data from the SWAMIS tool, a deep learning model implemented as a U-shaped convolutional neural network for fast and accurate image segmentation, and a post-processing component that prepares tracking results. SolarUnet is applied to data from the 1.6 meter Goode Solar Telescope at the Big Bear Solar Observatory. When compared to the widely used SWAMIS tool, SolarUnet is faster while agreeing mostly with SWAMIS on feature size and flux distributions, and complementing SWAMIS in tracking long-lifetime features. Thus, the proposed physics-guided deep learning-based tool can be considered as an alternative method for solar magnetic tracking.

Second, this dissertation presents a new deep learning method, dubbed FibrilNet, for tracing chromospheric fibrils in Hα images of solar observations. Our method consists of a data pre-processing component that prepares training data from a threshold-based tool, a deep learning model implemented as a Bayesian convolutional neural network for probabilistic image segmentation with uncertainty quantification to predict fibrils, and a post-processing component containing a fibril-fitting algorithm to determine fibril orientations. The FibrilNet tool is applied to high-resolution Hα images from an active region (AR 12665) collected by the 1.6 m Goode Solar Telescope (GST) equipped with high-order adaptive optics at the Big Bear Solar Observatory (BBSO). We quantitatively assess the FibrilNet tool, comparing its image segmentation algorithm and fibril-fitting algorithm with those employed by the threshold-based tool. Our experimental results and major findings are summarized as follows. First, the image segmentation results (i.e., detected fibrils) of the two tools are quite similar, demonstrating the good learning capability of FibrilNet. Second, FibrilNet finds more accurate and smoother fibril orientation angles than the threshold-based tool. Third, FibrilNet is faster than the threshold-based tool and the uncertainty maps produced by FibrilNet not only provide a quantitative way to measure the confidence on each detected fibril, but

also help identify fibril structures that are not detected by the threshold-based tool but are inferred through machine learning. Finally, we apply FibrilNet to full-disk Hα images from other solar observatories and additional high-resolution Hα images collected by BBSO/GST, demonstrating the tool's usability in diverse datasets.

Third, as obtaining high-quality magnetic and velocity fields through Stokes inversion is crucial for understanding and predicting solar flares, this dissertation present a new deep learning method, named Stacked Deep Neural Networks (SDNN), for inferring line-of-sight (LOS) velocities and Doppler widths from Stokes profiles collected by the Near Infrared Imaging Spectropolarimeter (NIRIS) on the 1.6 m Goode Solar Telescope (GST) at the Big Bear Solar Observatory (BBSO). The training data of SDNN is prepared by the widely used MilneEddington (ME) method. We quantitatively assess the SDNN tool, comparing its inversion results with those obtained by the ME tool and related machine learning (ML) algorithms such as multiple support vector regression, multilayer perceptrons and a pixel-wise convolutional neural network. Major findings from our experimental study are summarized as follows. First, the SDNN-inferred LOS velocities are highly correlated to the ME-calculated ones with the Pearson product-moment correlation coefficient being close to 0.9 on average. Second, SDNN is faster, while producing smoother and cleaner LOS velocity and Doppler width maps, than the ME method. Third, the maps produced by SDNN are closer to ME's maps than those from the related ML algorithms, demonstrating the better learning capability of SDNN than the related ML algorithms. Finally, comparison between the inversion results of ME and SDNN based on GST/NIRIS and those from the Helioseismic and Magnetic Imager on board the Solar Dynamics Observatory in flare-prolific active region NOAA 12673 is presented and discussed.

Fourth, solar activities are usually caused by the evolution of solar magnetic fields. Magnetic field parameters derived from photospheric vector magnetograms of

solar active regions have been used to analyze and forecast solar flares. Unfortunately, the most recent solar cycle 24 was relatively weak with few large flares, though it is the only solar cycle in which consistent time-sequence vector magnetograms have been available through the Helioseismic and Magnetic Imager (HMI) on board the Solar Dynamics Observatory (SDO) since its launch in 2010. Here, we looks into another major instrument, namely the Michelson Doppler Imager (MDI) on board the Solar and Heliospheric Observatory (SOHO) from 1996 to 2010. The data archive of SOHO/MDI covers more active solar cycle 23 with many large flares. However, SOHO/MDI data only has line-of-sight (LOS) magnetograms. This dissertation proposes a new deep learning approach, specifically a generative adversarial network (GAN) model, to learn from combined LOS magnetograms, B_x and B_y taken by SDO/HMI with Hα observations collected by Big Bear Solar Observatory (BBSO), and to generate synthetic vector components B'_x and B'_y, which would form vector magnetograms with observed LOS data. This way we are able to expand the availability of vector magnetograms to the period from 1996 to present. Experimental results obtained by using Hα observations and HMI magnetograms in the period between 2014-01-01 and 2017-08-04 as training data demonstrated the good performance of our approach. Specifically, when using the trained model to predict vector components based on Hα and HMI LOS data and validated by HMI vector data in the period between 2017-08-05 and 2017-12-31, we obtained an average Pearson correlation coefficient (CC) of $\sim$0.9. When using the trained model to predict vector components based on Hα and MDI LOS data and validated by vector data from the overlapping period of MDI and HMI between 2010-05-01 and 2011-04-11, we obtained an average CC of $\sim$0.75. To our knowledge, this is the first time that deep learning has been used to create photospheric vector magnetograms of solar active regions.

Finally, this dissertation presents the implementations of some of the deep learning-based methods, SolarUnet and FibrilNet using Jupyter notebooks and Google Colab with GitHub.

The rest of this dissertation is organized in the following manner. Chapter 2 presents the SolarUnet model for identifying and tracking solar magnetic flux elements in observed vector magnetograms, Chapter 3 presents the FibrilNet model for tracing chromospheric fibrils in Hα images of solar observations with uncertainty quantification. Chapter 4 presents the SDNN model for inferring LOS velocities and Doppler widths from Stokes profiles collected by GST/NIRIS at Big Bear Solar Observatory, Chapter 5 presents the MagNet model for generating vector components to create synthetic vector magnetograms of solar active regions. Chapter 6 illustrates the implementations of some of the deep learning-based methods, Chapter 7 summarizes the dissertation.

CHAPTER 2

MAGNETIC FLUX TRACKING

2.1 Background and Related Work

Tracking magnetic flux elements is an important subject in heliophysics research [29, 96, 154].[1] Identifying and tracking the surface magnetic elements is useful in deriving statistical parameters of the local and global solar dynamo, allowing for sophisticated analyses of solar activity [29]. It not only helps scientists understand the distribution of magnetic fluxes [121], but also helps estimate the amount of energy in acoustic waves, which play an important part in the heating of the solar chromosphere and corona [38]. In addition, magnetic tracking is useful in deriving boundary conditions of magnetohydrodynamic (MHD) modeling of the solar corona and solar wind. In the past, many researchers have studied the behaviors and patterns of magnetic flux elements. For instance, Chen et al. [27] developed a technique to detect and classify small-scale magnetic flux cancellations and link them to chromospheric rapid blueshifted excursions. Giannattasio et al. [51] investigated the occurrence and persistence of magnetic elements in the quiet Sun to understand the scales of organization at which turbulent convection operates. Moreno-Insertis et al. [114] reported findings related to small-scale magnetic flux emergence in the quiet Sun.

In magnetic tracking, features are defined as a visually identifiable part of an image, such as a clump of magnetic flux or a blob in a magnetogram. One of the most popular software tools for magnetic feature tracking across multiple images/frames is the Southwest Automatic Magnetic Identification Suite (SWAMIS) [29]. SWAMIS takes five steps to track magnetic flux elements: (1) feature discrimination for each frame; (2) feature identification within a frame; (3) feature association across frames;

[1] In the study presented here, we focus on tracking signed, including positive and negative, magnetic flux elements.

(4) occasional noise filtering; and (5) event detection [27]. Magnetic events are broadly classified into two categories: death and birth [89]; the former refers to the end of a magnetic features existence while the latter refers to the start of a magnetic features existence.

In this chapter, we present a new tool, called SolarUnet, to track magnetic flux elements. Our tool is built using deep learning [94]. The tool can detect three different types of events in each category, namely (i) disappearance and appearance, (ii) merging and splitting, and (iii) cancellation and emergence. The event "disappearance" is defined as the end of a single unipolar magnetic feature that "fades away" to nothing in the absence of nearby features across two frames; the opposite event "appearance" is defined as the origin of a single unipolar feature where the unipolar feature does not exist in the previous frame. The event "merging" is defined as the combination of two or more like-sign features into a single magnetic feature; the opposite event "splitting" is defined as the breakup of a single magnetic feature into at least two like-sign features, where the total flux of all child features is roughly the same as that of the parent feature. The event "cancellation" is defined as the demise of a magnetic feature that collides with one or more opposite-sign features, resulting in the demise of these features or an alive feature carrying the remaining flux; the event "emergence" is defined as the appearance of opposite-sign features with approximately the same magnitude or a new feature adjacent to previously existing opposite-sign features in a nearly flux-conserving manner.

Deep learning, which is a subfield of machine learning, has drawn a lot of interest in recent years [94]. Inspired by its success in computer vision, speech recognition and natural language processing, researchers have started to use deep learning in astronomy and astrophysics [69, 98, 81, 99, 103, 163]. In contrast to the existing methods for magnetic tracking [88, 90, 27], our SolarUnet tool is built using deep learning. Compared to the most closely related magnetic tracking tool, SWAMIS,

which uses hysteresis as the discrimination scheme and a gradient-based "downhill" method to identify features in a frame, SolarUnet runs faster while producing similar or complementary results.

2.2 Observations and Data Preparation

We adopted two collections of observations in this study. The first collection was conducted by the Near InfraRed Imaging Spectropolarimeter (NIRIS) [24] of the 1.6 m Goode Solar Telescope (GST) at the Big Bear Solar Observatory (BBSO) [25, 56, 54, 149]. This collection contained observations of the magnetic polarity inversion region in NOAA AR 12665 (431″, −131″) during ∼20:16–22:42 UT on 2017 July 13. The obtained data included spectro-polarimetric observations of a full set of Stokes measurements at the Fe I 1564.8 nm line (0.25 Åbandpass) by NIRIS with a FOV (field of view) of 80″ at 0″.24 resolution and 56 s cadence. Vector magnetic field products in local coordinates were constructed after removing azimuth ambiguity [97].

The second collection of observations was conducted with a clear seeing condition; BBSO/GST achieved diffraction-limited imaging during ∼16:17–22:17 UT on 2018 June 07. The obtained multi-wavelength observations revealed detailed structural and evolutionary properties of small scale magnetic polarities in quiescent solar regions at north of the disk center (−32″, 294″). The essential data included in this collection were the images taken by the GST's NIRIS using a 2048×2048 pixels Teledyne camera with a ∼80″ FOV. The spatial resolution (at diffraction limit $\theta = \lambda/D$) of the NIRIS images was 0″.2, and the temporal cadence was 56 s. The magnetograms were then aligned based on sunspot and plage features, with an alignment accuracy within 0″.3, which was the best accuracy by using interpolation.

We prepared our training and testing sets by using the magnetograms taken from the two collections of observations described above. Because the magnetograms

Table 2.1 Numbers of Images Used in Our Study

Number of Training Images	Number of Testing Images
196 (from the second collection)	147 (from the first collection)

taken on 2018 June 07 had higher quality than the observations conducted on 2017 July 13, we used the higher-quality magnetograms to prepare our training data so as to obtain a better magnetic tracking model. Specifically, we gathered all 202 frames from the second collection of observations, and excluded six images with poor quality (these excluded images were very noisy). The remaining 196 frames were used as training data for magnetic tracking. The testing set contained all 147 magnetograms from the first collection of observations. Table 2.1 summarizes the numbers of training and testing images used in this study.

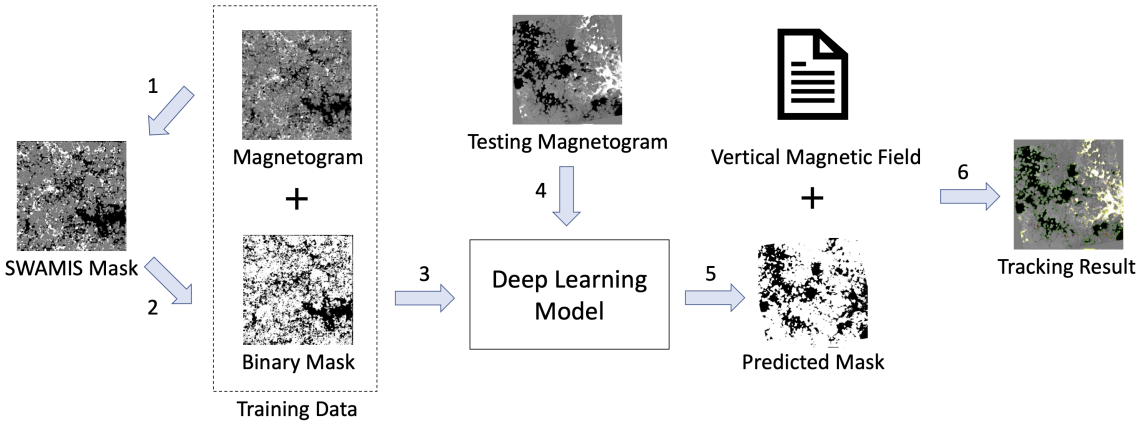

Figure 2.1 Illustration of the proposed method (SolarUnet) for identifying and tracking solar magnetic flux elements. SolarUnet employs a deep learning model for image segmentation. The training data used to train the deep learning model are highlighted in the dashed box.

2.3 Methodology

2.3.1 Overview of SolarUnet

Figure 2.1 explains how SolarUnet works. Training magnetograms are pre-processed in steps 1 and 2, and then used to train the deep learning model for image segmentation (step 3). The trained model takes a testing magnetogram (step 4) and produces a predicted mask (step 5). Through post-processing of the predicted mask, SolarUnet produces magnetic tracking results (step 6).

Specifically, in step 1, we apply SWAMIS with the downhill option to the 196 training magnetograms to get 196 masks. These images, including the magnetograms and masks, are converted to 8-bit grayscale images of 720×720 pixels, suitable for our deep learning model. Pixels in the masks belong to three classes represented by three colors/labels, respectively: positive magnetic flux with a label of 1 (white), negative magnetic flux with a label of -1 (black), and non-significant flux with a label of 0 (gray). During pre-processing, we convert the 196 three-class masks obtained

from SWAMIS to 196 two-class (binary) masks by (i) changing the label of the non-significant flux regions from 0 to 1; and (ii) changing both the positive magnetic flux regions and negative magnetic flux regions to significant flux regions with label -1 (step 2).

The 196 magnetograms (images) and two-class (binary) masks are then used to train the deep learning model, implemented in TensorFlow and Keras, for image segmentation (step 3). Because our deep learning model needs a large amount of data in order to train successfully, the model invokes the ImageDataGenerator. in Keras to perform data augmentation, expanding the training set by shifting, rotating, flipping and scaling the training images during the model training process. Shifting an image is to move all pixels of the image horizontally or vertically while keeping the dimensions of the image the same. Rotating an image is to rotate the image clockwise by a given number of degrees from 0 to 360. Flipping an image is to reverse the rows or columns of pixels in the image. Scaling an image is to randomly zoom the image in and either add new pixel values around the image or interpolate pixel values in the image. We train the deep learning model using 1 epoch with 10,000 iterations/epoch. In each iteration, the model randomly selects one of the 196 training magnetograms and its binary mask, feeds them to the ImageDataGenerator to generate a synthetic magnetogram and binary mask, and uses the synthetic magnetogram and binary mask to train the model. There are 10,000 iterations and hence 10,000 synthetic magnetograms and binary masks are generated through the data augmentation process, where the 10,000 generated magnetograms and binary masks are used for model training.[2] We have chosen to use data augmentation as opposed to acquiring more training data because the quality of ground-based observations is subject to many factors such as seeing conditions and observing time limits. Obtaining large volumes of high-quality

[2]Notice that SWAMIS is applied only to the 196 training magnetograms mentioned in Table 2.1; SWAMIS is never run on the 10,000 generated (synthetic) magnetograms.

training data requires further observations. Nevertheless, using the synthetic training images produces reasonably good results as shown in Section 2.4.

When a testing magnetogram is submitted, it is converted to a 8-bit grayscale image of 720×720 pixels, and fed to the trained deep learning model (step 4). The trained deep learning model predicts a two-class (binary) mask, containing non-significant flux regions with label 1 and significant flux regions with label -1 (step 5). We convert the predicted two-class (binary) mask back to a three-class mask via post-processing as follows. For the non-significant flux regions with label 1, we change their label from 1 to 0. For the significant flux regions with label -1, we use the information of radial components in the vertical magnetic field image shown in Figure 2.1, where the radial components are perpendicular to the plane of the Sun, to reconstruct positive and negative magnetic flux regions. Specifically, for each pixel x in the significant flux regions in the predicted binary mask, we check the magnetic strength of the pixel, y, at x's corresponding location in the vertical magnetic field image. If y's magnetic strength is greater than 150 G, we set x as positive magnetic flux and change the label of this pixel from -1 to 1. If y's magnetic strength is smaller than -150 G, we set x as negative magnetic flux and the label of this pixel remains -1. If y's magnetic strength is between -150 G and 150 G, we set x as non-significant flux and change the label of this pixel from -1 to 0. This yields a three-class mask with the polarity information.

Finally, we apply our magnetic tracking algorithms described in Subection 2.3.3 to the testing magnetogram and masks to get tracking results (step 6). Magnetic tracking is often involved with more than one testing magnetogram, and we output the tracking results in all of the testing magnetograms.

2.3.2 Implementation of the Deep Learning Model in SolarUnet

Figure 2.2 illustrates the deep learning model used in SolarUnet, which is a U-shaped convolutional neural network. We adapt U-Net [35] to our work, enhancing it to obtain our model. The model has an encoder, a bottleneck, a decoder, followed by a pixelwise binary classification layer.[3] The encoder consists of 4 blocks: E1, E2, E3, E4. Each block has two 3×3 convolution layers, represented by blue arrows, followed by a 2×2 max pooling layer, represented by a red arrow. In each convolution layer, we adopt batch normalization (BN) [70] after convolution, followed by a rectified linear unit (ReLU) activation function. Furthermore, we add a dropout layer [142] after each max pooling layer. The four encoder blocks E1, E2, E3, E4 have 32, 64, 128 and 256 kernels, respectively.

[3]Please see Section 2.6 for more detailed descriptions of the technical terms used here.

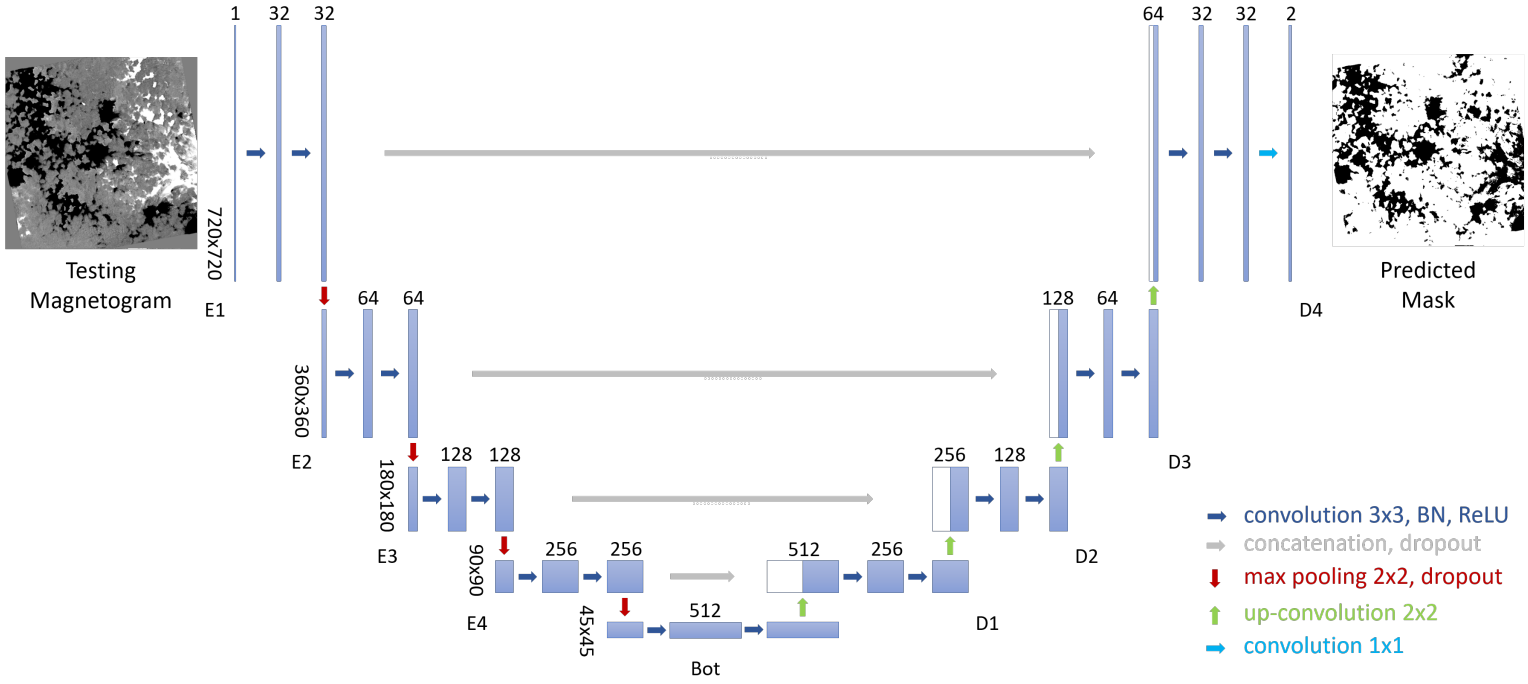

Figure 2.2 Illustration of the deep learning model used in SolarUnet. This model is a U-shaped convolutional neural network, consisting of an encoder, a bottleneck, a decoder, followed by a pixelwise binary classification layer. The encoder is comprised of 4 blocks: E1, E2, E3, E4. Each block has two 3×3 convolution layers, represented by blue arrows, followed by a 2×2 max pooling layer, represented by a red arrow. The decoder is also comprised of 4 blocks: D1, D2, D3, D4. Each block has two 3×3 convolution layers followed by a 2×2 up-convolution layer, represented by a green arrow. The bottleneck, denoted Bot, mediates between the encoder and the decoder. It uses two 3×3 convolution layers followed by a 2×2 up-convolution layer.

The bottleneck, denoted Bot, mediates between the encoder and the decoder. It uses two 3×3 convolution layers followed by a 2×2 up-convolution layer, represented by a green arrow. The bottleneck has 512 kernels. Similar to the encoder, the decoder consists of 4 blocks: D1, D2, D3, D4. Each block has two 3×3 convolution layers followed by a 2×2 up-convolution layer. The four decoder blocks D1, D2, D3, D4 have 256, 128, 64 and 32 kernels, respectively. The input of each decoder block is concatenated by the output of the corresponding encoder block where the concatenation is represented by a gray arrow. A dropout layer is added after each concatenation. Finally, a 1×1 convolution layer, represented by a turquoise arrow, with 2 kernels followed by a softmax activation function, is used to produce a segmentation mask. During testing, the deep learning model takes a testing magnetogram as input and produces a two-class mask as output.

The input resolution of the encoder block E1 is set to 720×720 pixels to match the size of the testing magnetogram. Each max pooling layer reduces the size by a factor of 2. Hence, the input resolution of the encoder block E2 (E3, E4, respectively) is 360×360 (180×180, 90×90, respectively) pixels. The input resolution of the bottleneck, Bot, is 45×45 pixels. Each up-convolution layer increases the size by a factor of 2. Thus, the input resolution of the decoder block D1 (D2, D3, D4, respectively) is 90×90 (180×180, 360×360, 720×720, respectively) pixels.

The loss function, L, used by the deep learning model is the binary cross-entropy function defined below:

$$L = -\sum_{x} \log y_c(x, W). \tag{2.1}$$

Here, W are the parameters of the convolutional neural network, $y_c(x, W)$ is the output of the softmax layer of the convolutional neural network, and c is the class label (1 vs. -1) of each pixel x.

In training the deep learning model, we adopt adaptive moment estimation (Adam) to find the optimal parameters of the model. The learning rate of Adam is set to 0.0001. Adam combines the advantages of two popular methods: AdaGrad and RMSProp [57]. In most cases, Adam achieves better performance than other stochastic optimization methods including the stochastic gradient descent (SGD) with momentum employed by U-Net [57].

Although both our deep learning model and U-Net [35] have the same U-shaped architecture, they differ in several ways. First, U-Net used SGD with a momentum of 0.99 to train and optimize its model. By contrast, we choose Adam because it achieves better performance in our case where the training process would be trapped in a poor local minimum if SGD were used. Second, U-Net focused on imbalanced datasets and used a weighted cross-entropy loss function to tackle the imbalanced classification problem. By contrast, because our training set is relatively balanced in the sense that non-significant flux regions roughly have the same number of pixels as significant flux regions, we use the binary cross-entropy loss function as defined in Equation. (2.1). Third, we adopt batch normalization and dropout layers, which were not used by U-Net. Batch normalization improves model learning, stabilizes the learning process, reduces the learning (training) time and improves prediction accuracy [70]. Dropout prevents neural networks from overfitting [142], where overfitting means that a trained model fits training data too well, and cannot generalize to make predictions on unseen testing data. Finally, we reduce the numbers of kernels of the encoder, bottleneck and decoder blocks by a factor of 2 compared to U-Net to speed up the training process and reduce GPU memory usage.

2.3.3 Algorithms for Magnetic Tracking and Event Detection

After describing the deep learning model used in SolarUnet, we now turn to the magnetic tracking algorithms employed by SolarUnet. Based on the positive magnetic flux regions and negative magnetic flux regions found in Subsection 2.3.1, we identify signed magnetic flux elements or features in a magnetogram (image/frame) by utilizing a connected-component labeling algorithm [66] to group all adjacent segments in the positive magnetic flux regions and negative magnetic flux regions, respectively, if their pixels in edges or corners touch each other. We filter out those magnetic features whose sizes are smaller than a user-determined threshold. The features eliminated from consideration are treated as noise. Then, we assign each of the remaining features a label number and highlight the features with different bordering colors. Finally, we consider the association of features (magnetic flux elements) across different frames to perform event detection.

Based on the observational data and instruments used, we calculate the moving distance D (number of pixels) of a magnetic flux element X as follows:

$$D = \frac{C \times \text{cadence}}{725\ km/arcsec \times \Delta s} \tag{2.2}$$

where C is the transverse speed (km/s) on the photosphere according to the observational environment and Sun's activity, and Δs is the pixel scale. In this study, C is set to 4 km/s. For the NIRIS magnetograms used here, Δs = $0.083''$/pixel. We then calculate the radius of the region of interest (ROI) with respect to the location or position of the magnetic flux element X, denoted $\text{ROI}_{p(X)}$, as follows:

$$radius(\text{ROI}_{p(X)}) = 2 \times D + r \tag{2.3}$$

where r is the radius of the smallest region that covers the magnetic flux element. The $\text{ROI}_{p(X)}$ defines the region which the magnetic flux element X can not move beyond between two contiguous frames.

The magnetic flux $\Phi(X)$ is calculated by the surface integral of the normal component of magnetic field passing through X, as follows:

$$\Phi(X) = \int_S B_z \, dS \tag{2.4}$$

where B_z is the magnitude of the magnetic field from the vertical magnetic field image of the testing magnetogram and S is the area of the surface of X.

For any two features or magnetic flux elements X and Y in a frame, we define the distance between X and Y, denoted $Dist(X, Y)$, as follows:

$$Dist(X, Y) = \min_{x \in X, y \in Y} d(x, y) \tag{2.5}$$

where x and y are pixels in X and Y, respectively; and $d(x, y)$ is the Euclidean distance between x and y. For a given magnetic flux element X in a frame, the adjacent features of X are defined as the k-nearest neighboring features of X in the frame. (In the study presented here, k is set to 10.)

Let X_i be a magnetic flux element in the current frame F_1. Let Y_i be a magnetic flux element in the next frame F_2 where Y_i occurs in the $\mathrm{ROI}_{p(X_i)}$ in F_2. We say X_i is approximately equal to Y_i, denoted $X_i \approx Y_i$, if X_i and Y_i have the same sign, and

$$\left|\frac{\Phi(X_i) - \Phi(Y_i)}{\Phi(X_i)}\right| \leq \epsilon_1 \tag{2.6}$$

where ϵ_1 is a user-determined threshold based on the observation setting and tracking task requirement. (In the study presented here, ϵ_1 is set to 0.33.)

With the above terms and definitions, we are now ready to describe our algorithms for magnetic tracking and event detection. For each magnetic flux element or feature X_i in the current frame F_1, the algorithms below determine and indicate whether X_i exists in the next frame F_2, or X_i is involved in a merging or cancellation event, or X_i disappears.

[Main Algorithm]

(i) If there exists a magnetic feature Y_i in the $\text{ROI}_{p(X_i)}$ in the next frame F_2 such that $X_i \approx Y_i$, then indicate X_i exists in F_2 (more precisely, X_i becomes Y_i in F_2) and go to (ii); otherwise go to (iii).

(ii) Check the sign of X_i, highlighting X_i by yellow bordering color if X_i is positive and by green bordering color if X_i is negative. Exit the Main Algorithm.

(iii) Find all magnetic features in the $\text{ROI}_{p(X_i)}$ in the current frame F_1. Group those features in the $\text{ROI}_{p(X_i)}$ whose signs are the same as the sign of X_i into $G_{same}(X_i)$ and group those features in the $\text{ROI}_{p(X_i)}$ whose signs are opposite to the sign of X_i into $G_{opposite}(X_i)$.

(iv) Go to the Merging Algorithm to check whether X_i and the features in $G_{same}(X_i)$ meet the merging criterion. If yes, perform the merging using the Merging Algorithm and then exit the Main Algorithm; otherwise indicate X_i is not involved in a merging event.

(v) Go to the Cancellation Algorithm to check whether X_i and the features in $G_{opposite}(X_i)$ meet the cancellation criterion. If yes, perform the cancellation using the Cancellation Algorithm and then exit the Main Algorithm; otherwise indicate X_i is not involved in a cancellation event.

(vi) If X_i is not involved in a merging event according to (iv) and X_i is not involved in a cancellation event based on (v), indicate X_i disappears and highlight X_i by purple bordering color.[4] Exit the Main Algorithm.

[Merging Algorithm]

[4]For the events belonging to the death category, namely disappearance, merging and cancellation, magnetic features involved in the events are highlighted by different bordering colors (purple for disappearance, amber for merging and pink for cancellation) in the current frame F_1. For the events belonging to the birth category, namely appearance, splitting and emergence, magnetic features involved in the events are highlighted by different bordering colors (blue for appearance, aqua for splitting and red for emergence) in the next frame F_2.

(i) For each feature X_j in $G_{same}(X_i)$, check whether there exists a feature Y_j in the $\text{ROI}_{p(X_j)}$ in the next frame F_2 such that $X_j \approx Y_j$, and if yes, delete X_j from $G_{same}(X_i)$. Call the remaining set, $G'_{same}(X_i)$. If there are too many features in $G'_{same}(X_i)$, only keep those adjacent features of X_i in $G'_{same}(X_i)$.

(ii) If there exist a combination C_s of features in $G'_{same}(X_i)$ and a magnetic feature Y_i in the $\text{ROI}_{p(X_i)}$ in the next frame F_2 where Y_i and X_i have the same sign, such that Equation. (2.7) below is satisfied, then we say X_i and the features in C_s are merged into Y_i:

$$|\frac{(\Phi(X_i) + \sum_{X \in C_s} \Phi(X)) - \Phi(Y_i)}{\Phi(Y_i)}| \leq \epsilon_2 \tag{2.7}$$

where $\Phi(X_i)$ and $\Phi(X)$ have the same sign, ϵ_2 is a user-determined threshold (which is set to 0.5). Indicate X_i and the features in C_s are merged into Y_i by highlighting X_i and the features in C_s using amber bordering color. Exit the Merging Algorithm.

(iii) If there does not exist Y_i or a combination of features satisfying Equation. (2.7) (i.e., the condition in (ii) is not satisfied), indicate X_i and the features in $G_{same}(X_i)$ do not meet the merging criterion. Exit the Merging Algorithm.

[Cancellation Algorithm]

(i) For each feature X_j in $G_{opposite}(X_i)$, check whether there exists a feature Y_j in the $\text{ROI}_{p(X_j)}$ in the next frame F_2 such that $X_j \approx Y_j$, and if yes, delete X_j from $G_{opposite}(X_i)$. Call the remaining set, $G'_{opposite}(X_i)$. If there are too many features in $G'_{opposite}(X_i)$, only keep those adjacent features of X_i in $G'_{opposite}(X_i)$.

(ii) If there exists a combination C_o of features in $G'_{opposite}(X_i)$ such that Equation. (2.8) below is satisfied, then we say X_i and the features in C_o cancel each other (referred to as balanced cancellation in [29]):

$$|\frac{\Phi(X_i) + \sum_{X \in C_o} \Phi(X)}{\Phi(X_i)}| \leq \epsilon_2 \tag{2.8}$$

where $\Phi(X_i)$ and $\Phi(X)$ have opposite signs. Indicate X_i and the features in C_o cancel each other by highlighting X_i and the features in C_o using pink bordering color. Exit the Cancellation Algorithm.

(iii) If there exist a combination C_o of features in $G'_{opposite}(X_i)$ and a magnetic feature Y_i in the $\mathrm{ROI}_{p(X_i)}$ in the next frame F_2, such that Equation. (2.9) below is satisfied, then we say X_i and the features in C_o are canceled to yield a feature Y_i carrying the remaining flux (referred to as unbalanced cancellation in [29]):

$$\left| \frac{|\Phi(X_i) + \sum_{X \in C_o} \Phi(X)| - |\Phi(Y_i)|}{\Phi(Y_i)} \right| \leq \epsilon_2 \tag{2.9}$$

where $\Phi(X_i)$ and $\Phi(X)$ have opposite signs. Indicate X_i and the features in C_o are canceled by highlighting X_i and the features in C_o using pink bordering color. Exit the Cancellation Algorithm.

(iv) If neither the condition in (ii) nor the condition in (iii) is satisfied, indicate X_i and the features in $G_{opposite}(X_i)$ do not meet the cancellation criterion. Exit the Cancellation Algorithm.

To determine whether the magnetic feature X_i appears or is involved in a splitting or emergence event, we use the same algorithms as described above except that we treat the next frame F_2 as the current frame and the current frame F_1 as the next frame.

2.4 Results

SolarUnet is implemented in Python. Our deep learning model is coded with TensorFlow and Keras libraries. The data processed by SolarUnet, with the aid of Astropy [10], include FITS files containing vector magnetic fields, PNG images of the observational data described in Section 2.2, and image masks obtained from the SWAMIS tool presented in DeForest et al. [29]. SWAMIS, written in PDL (Perl

Data Language) and available via SolarSoft [41], was run with the downhill option. Figures in this section were produced with the aid of matplotlib. Statistical tests were performed by SciPy [150]. All experiments were conducted on a Dell PC with i7-8700k CPU, 32GB RAM and a NVIDIA GeForce RTX 2080 GPU for training and testing the deep learning model.

2.4.1 Magnetic Tracking and Event Detection Results

In this series of experiments, we used the 196 magnetograms mentioned in Table 2.1 and the corresponding masks obtained from SWAMIS to train SolarUnet as described in Subsection 2.3.1, and then we performed testing on the set of 147 magnetograms mentioned in Table 2.1. The testing set contained observations in NOAA AR 12665 ($431''$, $-131''$) during $\sim$20:16–22:42 UT on 2017 July 13. The filter size threshold was fixed at 10 pixels. Thus, in the experiments we considered features or magnetic flux elements having at least 10 pixels.

We present figures to illustrate the six events studied here. Frames taken at 20:15:49 UT and 20:16:45 UT are used to illustrate a disappearance event. Frames taken at 21:00:48 UT and 21:01:45 UT are used to illustrate an appearance event. Frames taken at 20:18:38 UT and 20:19:34 UT are used to illustrate a merging event. Frames taken at 20:19:34 UT and 20:20:30 UT are used to illustrate a splitting event and a cancellation event. Frames taken at 20:17:41 UT and 20:18:38 UT are used to illustrate an emergence event. We present enlarged FOV results in these figures with a FOV of $7.5''$ where each figure has two axes: E-W (x-axis) and S-N (y-axis).

Figure 2.3 shows a disappearance event. In Figure 2.3(A), a magnetic feature highlighted by purple bordering color exists at E-W = $421''$ and S-N = $-191''$, which is pointed to by a red arrow in the frame from 20:15:49 UT. This feature disappears in the next frame from 20:16:45 UT as shown in Figure 2.3(B). Figure 2.4 illustrates an appearance event. In Figure 2.4(A), there exists no feature at E-W = $420''$ and S-N

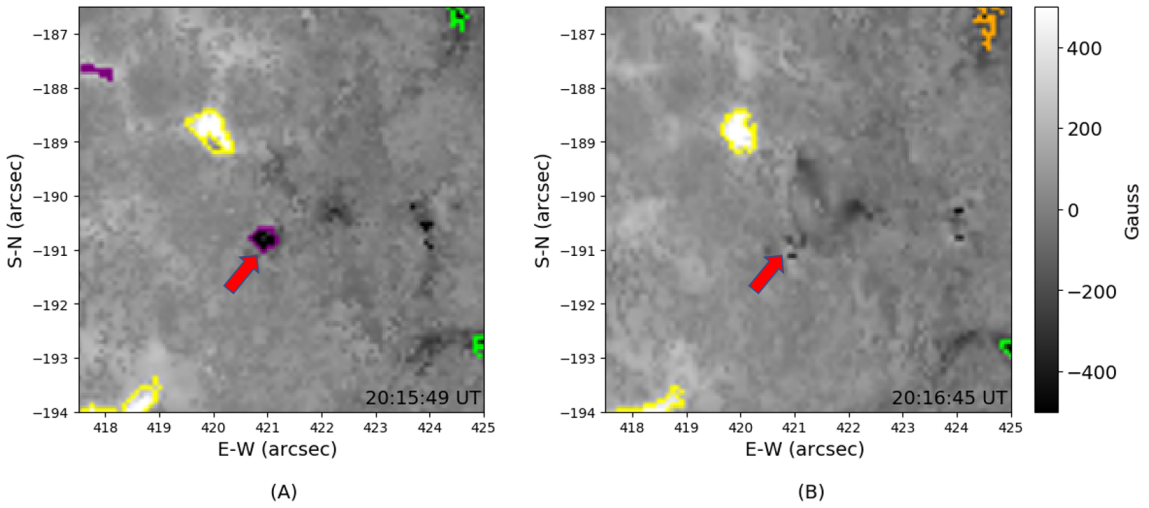

Figure 2.3 Example of BBSO/GST images of a disappearance event. The negative magnetic flux element highlighted by purple bordering color in (A) disappears in (B). Time in UT is at the bottom right of each image.

$= -192''$ in the frame from 21:00:48 UT. However, a new feature appears in the next frame from 21:01:45 UT, which is highlighted by blue bordering color and pointed to by a red arrow as shown in Figure 2.4(B).

Figure 2.5 shows a merging event. In Figure 2.5(A), two separate positive polarity magnetic features highlighted by amber bordering color and pointed to by a red arrow in the frame from 20:18:38 UT are merged into a single positive polarity feature at E-W $= 453''$ and S-N $= -180''$ in the frame from 20:19:34 UT as shown in Figure 2.5(B). Figure 2.6 illustrates a splitting event. In Figure 2.6(A), a negative polarity feature exists at E-W $= 448''$ and S-N $= -179''$ in the frame from 20:19:34 UT. This negative polarity feature is split into two negative polarity features highlighted by aqua bordering color and pointed to by a red arrow in the frame from 20:20:30 UT as shown in Figure 2.6(B).

Figure 2.7 shows an unbalanced cancellation event. In Figure 2.7(A), there exist two magnetic features with opposite signs around E-W $= 437''$ and S-N $= -158.5''$ in the frame from 20:19:34 UT. The two magnetic features with opposite signs, highlighted by pink bordering color, are canceled to yield a negative polarity

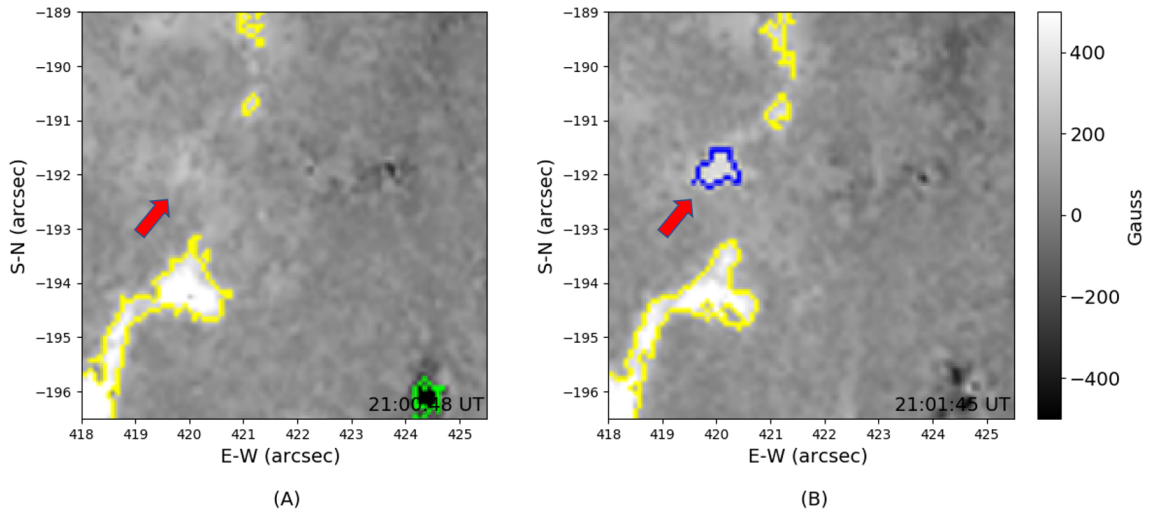

Figure 2.4 Example of BBSO/GST images of an appearance event. The positive magnetic flux element highlighted in blue bordering color in (B) does not exist in (A), and hence an appearance event is detected. Time in UT is at the bottom right of each image.

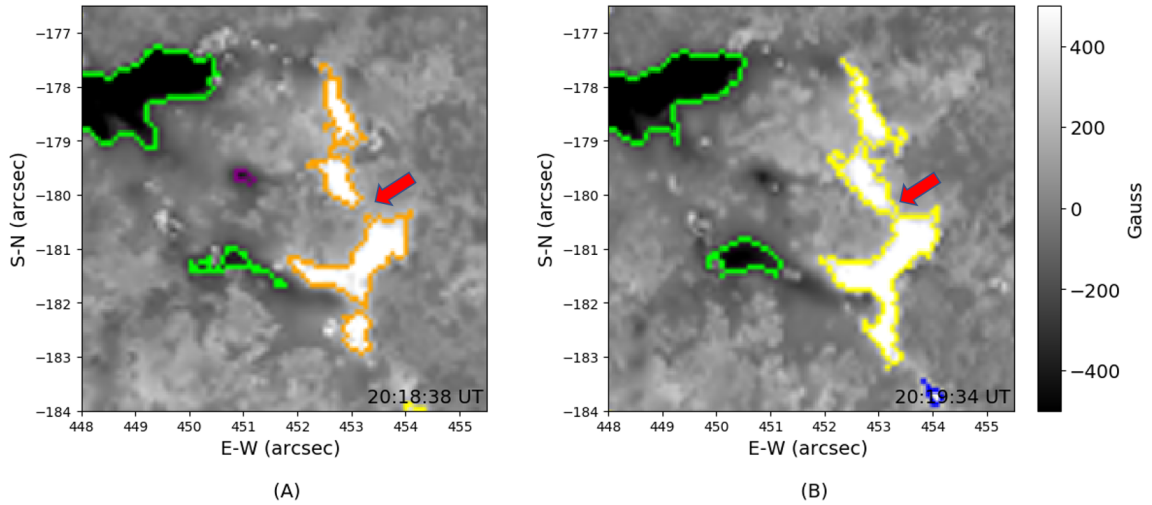

Figure 2.5 Example of BBSO/GST images of a merging event. Two positive magnetic flux elements highlighted by amber bordering color in (A) are merged into a single positive magnetic flux element in (B). Time in UT is at the bottom right of each image.

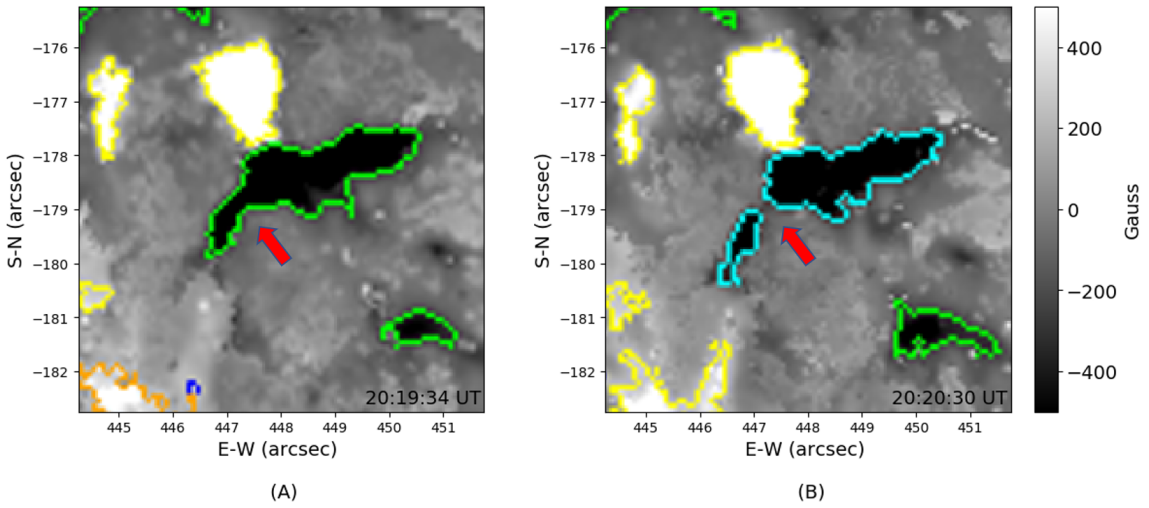

Figure 2.6 Example of BBSO/GST images of a splitting event. A negative magnetic flux element in (A) is split into two negative magnetic flux elements highlighted by aqua bordering color in (B). Time in UT is at the bottom right of each image.

magnetic feature carrying the remaining flux, pointed to by a red arrow, in the frame from 20:20:30 UT as shown in Figure 2.7(B). Figure 2.8 illustrates an unbalanced emergence event. A new negative polarity feature emerges, next to a pre-existing positive polarity feature, in the frame from 20:18:38 UT as shown in Figure 2.8(B). The flux of the positive polarity feature pointed to by a red arrow in Figure 2.8(A) is approximately equal to the total flux of the two features with opposite signs, highlighted by red bordering color and pointed to by a red arrow, in Figure 2.8(B).

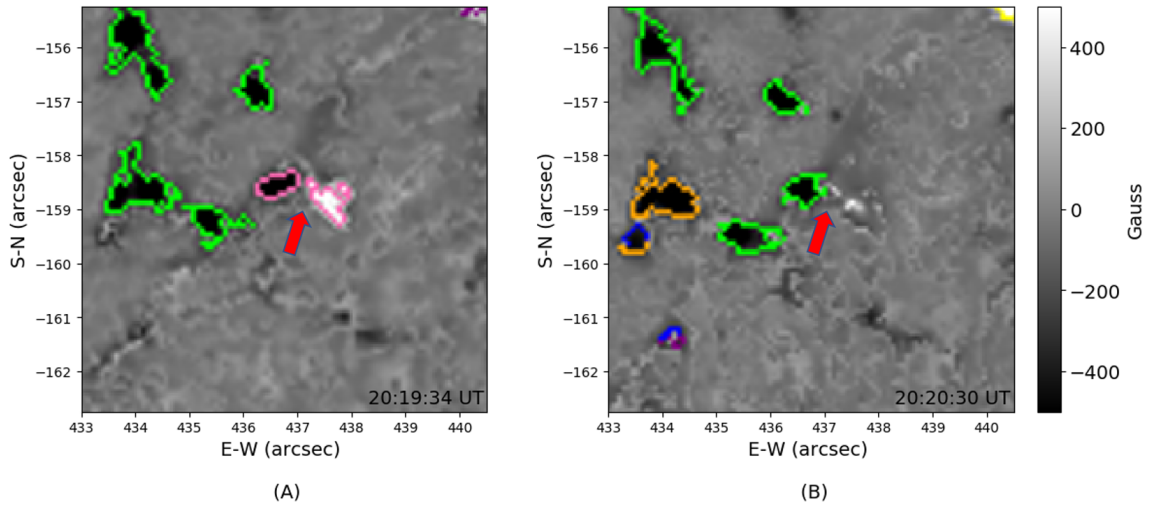

Figure 2.7 Example of BBSO/GST images of an unbalanced cancellation event. A positive magnetic flux element and a negative magnetic flux element, both of which are highlighted by pink bordering color in (A), are canceled to yield a negative magnetic flux element carrying the remaining flux, which is pointed to by a red arrow in (B). Time in UT is at the bottom right of each image.

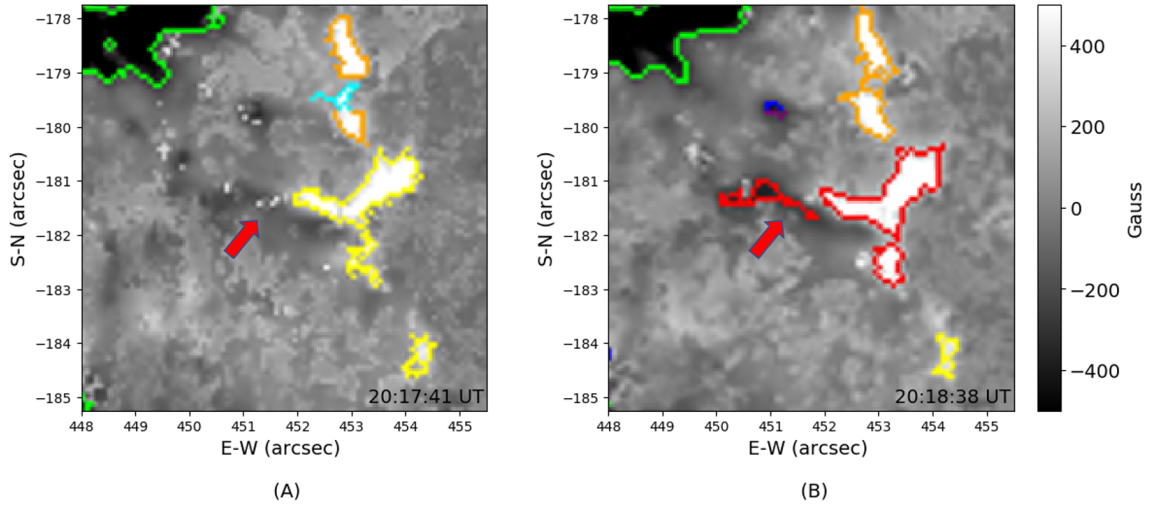

Figure 2.8 Example of BBSO/GST images of an unbalanced emergence event. A new negative magnetic flux element emerges, next to a pre-existing positive magnetic flux element in (A), in a nearly flux-conserving manner where the two magnetic flux elements with opposite signs are highlighted by red bordering color in (B). Time in UT is at the bottom right of each image.

2.4.2 Comparison with SWAMIS

While both SolarUnet and SWAMIS [29] aim to track magnetic features and detect magnetic events, they differ in two ways.

1. Their feature discrimination and identification algorithms are different. SWAMIS used hysteresis and a threshold-based method to separate non-significant flux regions, positive magnetic flux regions and negative magnetic flux regions. Then it used direct clumping and a gradient based ("downhill") method to identify magnetic features in these regions. By contrast, SolarUnet employs a U-shaped convolutional neural network to gain knowledge from training data, and then predicts a binary (two-class) mask containing non-significant flux regions and significant flux regions. Next, SolarUnet separates the significant flux regions into positive magnetic flux regions and negative magnetic flux regions through post-processing of the binary mask. Finally, SolarUnet uses a connected-component labeling algorithm [66] to group all adjacent segments in the positive magnetic flux regions and negative magnetic flux regions, respectively, if their pixels in edges or corners touch each other to identify positive and negative magnetic flux elements.

2. Their feature tracking and event detection algorithms are different. SWAMIS used a dual-maximum-overlap criterion to find persistent features across frames. In contrast, SolarUnet defines the region of interest (ROI) of a magnetic feature and traces the flux changes of the magnetic features in the ROI to find the association of features across frames.

It should be pointed out that, although the U-shaped network (i.e., the deep learning model) in SolarUnet gains knowledge from the training data prepared by SWAMIS, the model is able to generalize learned features to more generic forms. In our work, the model gains knowledge from the training images in quiescent solar regions collected on 2018 June 07 and uses the acquired knowledge to make predictions on unseen testing images from an active region (NOAA AR 12665) collected on 2017 July 13. With the generalization and inference capability, the model may discover new magnetic flux elements not found by the SWAMIS method. For example, with the filter size threshold of SolarUnet fixed at 10 pixels, SolarUnet detected two opposite-sign features not found by SWAMIS on the testing image (magnetogram) from AR

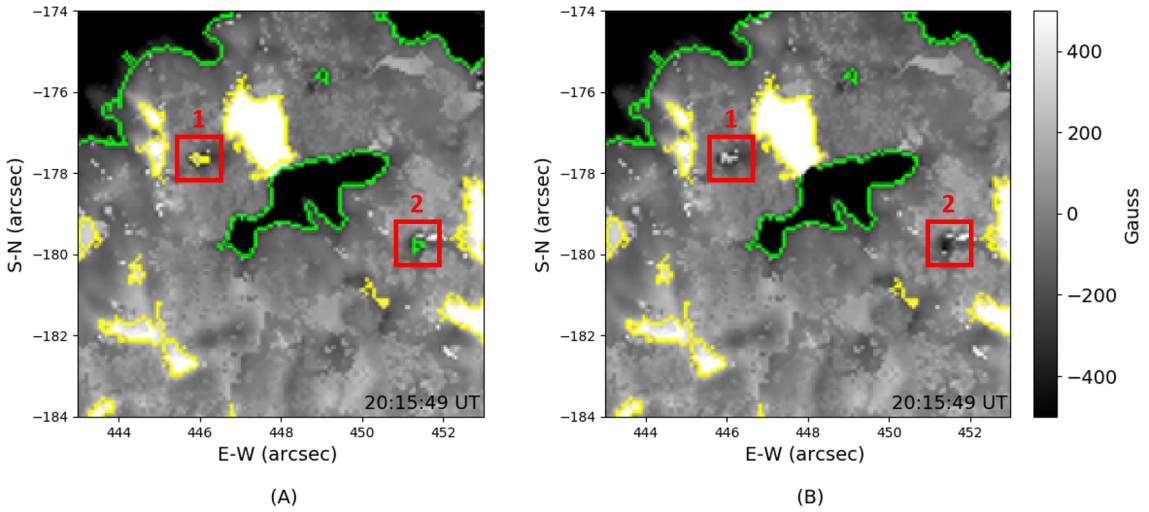

Figure 2.9 Illustration of the magnetic flux elements detected by SolarUnet but not found by SWAMIS on the testing magnetogram from AR 12665 collected on 2017 July 13 20:15:49 UT. (A) SolarUnet identifies a positive feature highlighted by yellow bordering color and a negative feature highlighted by green bordering color where the two highlighted features are enclosed by red square boxes numbered by 1 and 2, respectively. (B) SWAMIS does not find the two features as no bordering color is shown inside the red square boxes numbered by 1 and 2, respectively. Time in UT is at the bottom right of each image.

12665 collected on 2017 July 13 20:15:49 UT. Figure 2.9(A) highlights these two features; Figure 2.9(B) shows that the two features were not found by SWAMIS.

Figure 2.10 compares the feature size distributions of SWAMIS and SolarUnet on the testing image (magnetogram) where the features had at least 2 pixels (0.007242 Mm^2).[5] The feature sizes of SWAMIS are represented by blue color and those of SolarUnet are represented by orange color. It can be seen from Figure 2.10 that SolarUnet agrees mostly with SWAMIS on the feature size distributions. To quantify this finding, we conducted the Epps-Singleton two-sample test [33, 53, 52]. According to the test, the results of SolarUnet and SWAMIS have a significant difference when $p \leq 0.05$. In our case $p = 0.858 > 0.05$, and hence we conclude that the results of the two tools are similar. Table 2.2 shows the minimum, maximum, median, mean

[5]In this and subsequent experiments, features with 1 pixel were considered as noise and excluded.

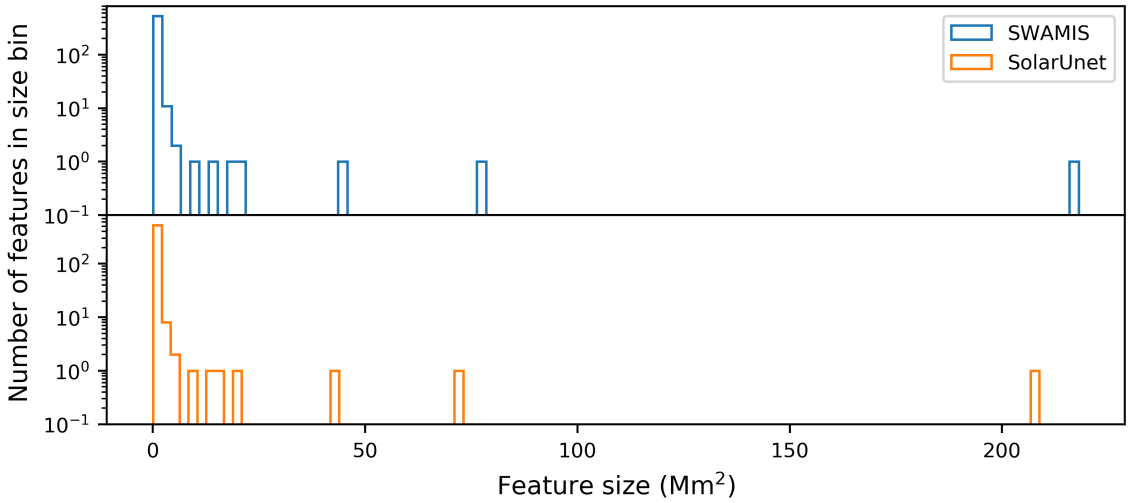

Figure 2.10 Magnetic feature size distributions as derived by SWAMIS (represented by blue color) and SolarUnet (represented by orange color) on the testing magnetogram from AR 12665 collected on 2017 July 13 20:15:49 UT. SolarUnet agrees mostly with SWAMIS on the Magnetic feature size distributions.

and standard deviation (SD) of the feature sizes found by SWAMIS and SolarUnet, respectively. SWAMIS detected 548 features while SolarUnet identified 543 features. The largest magnetic feature, which was a negative feature, found by SWAMIS had 60213 pixels (218.03 Mm^2). This feature was also detected by SolarUnet, with a smaller size of 57662 pixels (208.80 Mm^2). This size difference occurs due to the different feature identification and tracking algorithms used by the two tools.

Table 2.2 Summary Statistics of Feature Size and Flux Distributions as Derived by SWAMIS and SolarUnet

	Method	Minimum	Maximum	Median	Mean	SD
Feature Size	SWAMIS	0.007242	218.03	0.029	0.94	10.14
(Mm^2)	SolarUnet	0.007242	208.80	0.022	0.89	9.72
Feature Flux	SWAMIS	0.009507	1805.13	0.048	6.70	81.59
(10^{18}Mx)	SolarUnet	0.011051	1780.96	0.048	6.61	80.89

Notes.

[a] The data presented in this table are based on the testing magnetogram from AR 12665 collected on 2017 July 13 20:15:49 UT.

[b] SWAMIS detected 548 features in the testing magnetogram.

[c] SolarUnet identified 543 features in the testing magnetogram.

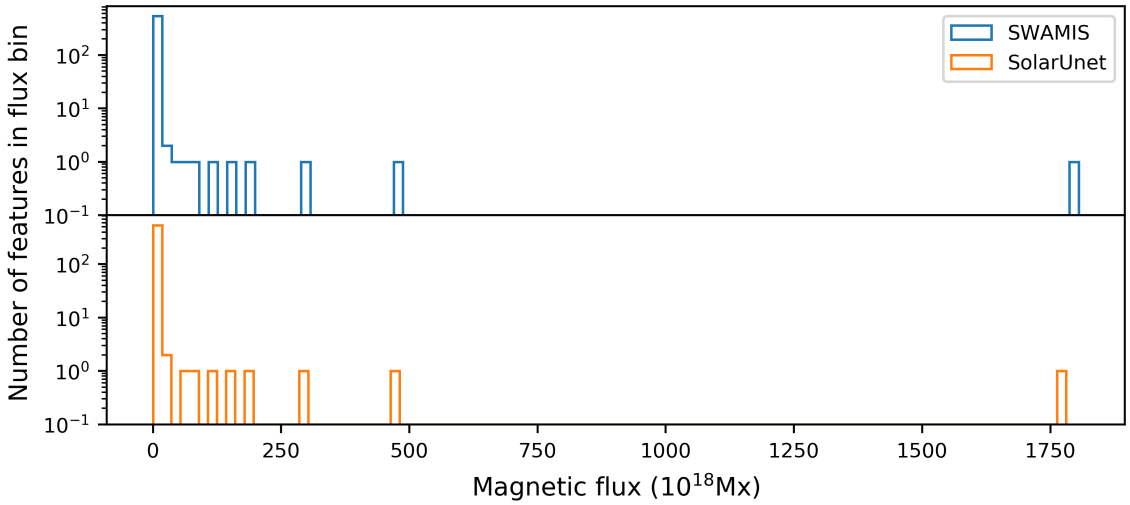

Figure 2.11 Magnetic feature flux distributions as derived by SWAMIS (represented by blue color) and SolarUnet (represented by orange color) on the testing magnetogram from AR 12665 collected on 2017 July 13 20:15:49 UT. SolarUnet agrees mostly with SWAMIS on the Magnetic feature flux distributions.

Next, for each feature detected by the tools, we calculated its flux using the formula in Equation. (2.4). Figure 2.11 compares the feature flux distributions of SWAMIS and SolarUnet. The results in Figure 2.11 are consistent with those in Figure 2.10; SolarUnet agrees mostly with SWAMIS on the feature flux distributions. According to the Epps-Singleton two-sample test, the feature flux distributions of SolarUnet and SWAMIS have a significant difference when $p \leq 0.05$. In our case $p = 0.983 > 0.05$, and consequently we conclude that the feature flux distributions of SolarUnet and SWAMIS are similar. Table 2.2 shows the minimum, maximum, median, mean and standard deviation (SD) of the feature fluxes found by SWAMIS and SolarUnet, respectively. The feature fluxes detected by SWAMIS ranged from 0.009507×10^{18} Mx to 1805.13×10^{18} Mx. The feature fluxes detected by SolarUnet ranged from 0.011051×10^{18} Mx to 1780.96×10^{18} Mx. Some of the small fluxes could be noise whiles others might be involved in small-scale magnetic flux emergence [114] or small-scale magnetic flux cancellation [27]. Similar results on feature size and flux distributions were obtained from the other magnetograms in the testing set.

To further understand the behavior of SolarUnet and compare it with SWAMIS, we performed additional experiments to examine the lifetimes of the features identified and tracked by the two tools. We applied SolarUnet and SWAMIS to all of the 147 testing magnetograms mentioned in Table 2.1. The lifetime of a feature X is defined as X's disappearance time minus X's appearance time. More precisely, assuming X appears in the mth frame and disappears after the nth frame (i.e., X is not shown in the $(n+1)$th frame), the lifetime of X is defined to be $n-m+1$ frames. Feature lifetime is strongly dependent on the feature identification and tracking algorithms employed by a tool [29], and can be used to measure flux turnover rate [62].

Figure 2.12 compares the lifetimes of features found by SWAMIS and SolarUnet. SWAMIS tracked 48145 features across the 147 testing magnetograms while SolarUnet tracked 42470 features. The lifetimes of features found by SWAMIS ranged from 1 frame (56 seconds) to 138 frames (128.8 minutes). The lifetimes of features detected by SolarUnet ranged from 1 frame to 147 frames (137.2 minutes). SWAMIS tracked more short-lifetime features than SolarUnet while SolarUnet tracked more long-lifetime features than SWAMIS. Specifically, among the 48145 features tracked by SWAMIS, 37110 features had a lifetime of 1 frame while SolarUnet only identified and tracked 22657 such features. On the other hand, SolarUnet tracked 19813 features whose lifetimes lasted more than 1 frame while SWAMIS only identified and tracked 11035 such features. SolarUnet complements SWAMIS in tracking long-lifetime features. We note that the training data of SolarUnet are from SWAMIS. For those features with short lifetime in the training images, our deep learning model may not acquire enough knowledge about them, and hence, may miss similar features in the testing images. This may explain why SolarUnet detects fewer short-lifetime features than SWAMIS.

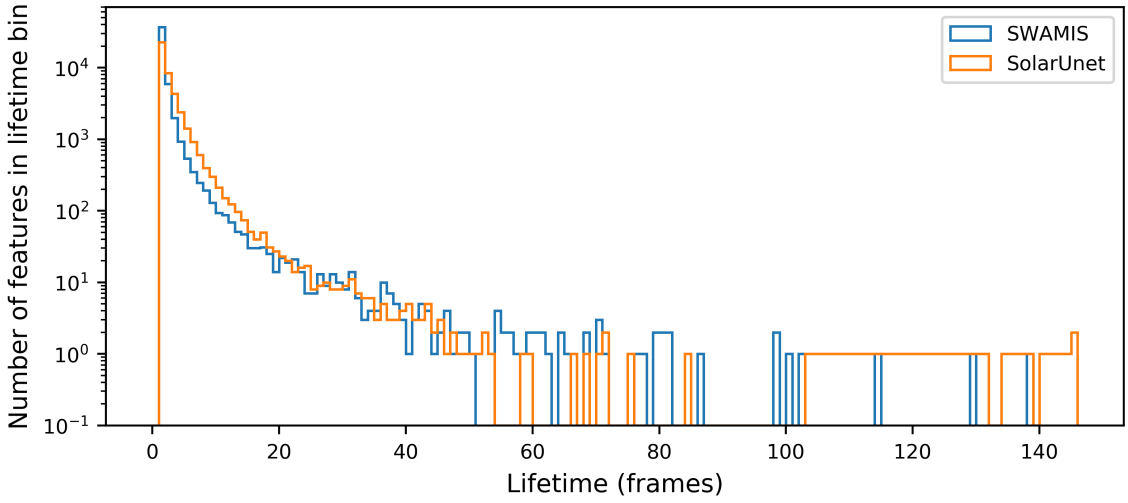

Figure 2.12 Feature lifetime histograms derived from SWAMIS and SolarUnet based on the 147 testing magnetograms (frames) from AR 12665 collected on 2017 July 13. SWAMIS tracks 48145 features, among which 37110 features have a lifetime of 1 frame. SolarUnet tracks 42470 features, among which 22657 features have a lifetime of 1 frame. On the other hand, SolarUnet tracks 19813 features whose lifetimes last more than 1 frame while SWAMIS only tracks 11035 such features. SolarUnet complements SWAMIS in tracking long-lifetime features.

2.5 Summary

We develop a deep learning method, SolarUnet, for tracking signed magnetic flux elements (features) and detecting magnetic events in observed vector magnetograms. We apply the SolarUnet tool to data from the 1.6 meter Goode Solar Telescope (GST) at the Big Bear Solar Observatory (BBSO). The tool is able to identify the magnetic features and detect three types of events, namely disappearance, merging and cancellation, in the death category and three types of events, namely appearance, splitting and emergence, in the birth category. We use the BBSO/GST images to illustrate how our tool works on feature identification and event detection, and compares with the widely used SWAMIS tool [29].

Our main results are summarized as follows:

1. For the testing data considered, SolarUnet agrees mostly with SWAMIS on feature size (area) and flux distributions, and complements SWAMIS in tracking long-lifetime features. It is worth noting that because SolarUnet performs magnetic tracking through making predictions, it is faster than the current version of SWAMIS. In general, SolarUnet runs in seconds on a testing magnetogram while the current version of SWAMIS runs in minutes on the same testing magnetogram.

2. SolarUnet is a physics-guided tool in the sense that it incorporates physics knowledge into its model and algorithms in several ways. First, the training data of SolarUnet are from the physics-based SWAMIS tool. Second, in designing the loss function for the deep learning model used by SolarUnet, based on the observation that non-significant flux regions roughly have the same number of pixels as significant flux regions in the training set, we adopt a binary cross-entropy loss function as defined in Equation. (2.1) instead of the weighted cross-entropy loss function used by the related U-Net model [35]. Third, in converting the binary (two-class) mask predicted by our deep learning model for a testing magnetogram to a three-class mask with polarity information, we use the information of radial components in the vertical magnetic field image of the testing magnetogram to reconstruct positive and negative magnetic flux regions in the predicted mask. Lastly, by exploiting physics knowledge and based on the observational data and instruments used, we introduce the moving distance as defined in Equation. (2.2) and region of interest (ROI) as defined in Equation. (2.3) of a magnetic flux element to find the association of features across frames so as to track these features.

3. Although SolarUnet gets training data from SWAMIS, our tool may discover new features not found by the SWAMIS method. For example, refer to Figure 2.9. SolarUnet may detect smaller opposite-polarity features, as shown and highlighted in Figure 2.9(A), near larger magnetic flux elements. Small-scale energy release phenomena, ranging from coronal jets down to spicules, may be responsible for providing the upward flux of energy and momentum for the observed heatings and flows in the corona, and may plausibly drive the small transients in the solar wind recently discovered by the Parker Solar Probe [120]. There is mounting evidence that these events are generated via small-scale magnetic reconnection [130], the photospheric signature of which is flux cancellation involving opposite magnetic polarities [169]. The ability of SolarUnet in detecting smaller opposite-polarity features near larger magnetic flux elements in a faster manner can result in an improved determination of magnetic reconnection rate, thus contributing to the understanding of the mechanisms of solar coronal heating and the acceleration of the solar wind.

4. The deep learning model in SolarUnet performs binary (two-class) classification, i.e., predicting a two-class mask, rather than three-class classification, i.e., predicting a three-class mask, during image segmentation. SolarUnet produces a three-class mask through post-processing of the predicted two-class mask as described in item 2 above and in Subsection 2.3.1. As indicated in the machine learning literature, multiclass classification including three-class classification often adds more noise to the loss function [61], and it is easier to devise algorithms for binary classification [3]. We conducted additional experiments to compare SolarUnet with a three-class classification method. This method trained its deep learning model using the three-class masks obtained directly from SWAMIS, and predicted three-class masks. Its model was the same as SolarUnet's model except that (i) its loss function was changed from the binary cross-entropy function defined in Equation. (2.1) to a categorical cross-entropy loss with three class labels (1, 0, -1); (ii) its softmax activation function was modified to output three-class masks. The three-class classification method used the same tracking algorithms as described in Subsection 2.3.3 for magnetic tracking and event detection. The results of the three-class classification method were not as good as those of SolarUnet. For example, the feature size distribution obtained from the three-class classification method was significantly different from the feature size distribution obtained from SWAMIS with $p = 0.025 \leq 0.05$ according to the Epps-Singleton two-sample test on the testing magnetogram from AR 12665 collected on 2017 July 13 20:15:49 UT.

Based on our experimental results, we conclude that the proposed SolarUnet should be considered a novel and alternative method for identifying and tracking magnetic flux elements. More testing of the method, using different training and test data, should be performed. With the advent of big and complex observational data gathered from diverse instruments such as BBSO/GST and the upcoming Daniel K. Inouye Solar Telescope (DKIST), it is expected that the physics-guided deep learning-based SolarUnet tool will be a useful utility for processing and analyzing the data.

2.6 Related Technical Terms

Here we explain the technical terms used in describing our deep learning model (i.e., the U-shaped convolutional neural network).

Encoder is a neural network, which takes an input image and generates a high-dimensional vector that is an abstract representation of the image (see Chapter 8.5.2 in Aggarwal 2018 [2]). By using the encoder, our model can better understand the content and context of the image.

Decoder is a neural network, which takes a high-dimensional vector and generates a segmentation mask (see Chapter 8.5.2 in Aggarwal 2018 [2]). By using the decoder, our model can recover the spatial information in the input image.

Bottleneck, also known as the "compressed code" (see Chapter 8.5.2 in Aggarwal 2018 [2]), is a layer with less neurons than the layer below or above it [50]. In general, it can be used to obtain a representation of the input with reduced size (dimensionality). In our model, bottleneck mediates between the encoder and the decoder.

Convolution layer contains multiple kernels where a kernel is a matrix whose elements (weights) need to be learned from training data (see Chapter 9 in Goodfellow et al. 2016 [57]). Each kernel is multiplied with an image vector X (via element-wise multiplications) to produce a new image vector that contains only the important information in X (see Chapter 8 in Aggarwal 2018 [2]).

Max pooling layer reduces the size of an image vector X while retaining only the important information in X (see Chapter 8.2 in Aggarwal 2018 [2]).

Up-convolution layer, containing learnable parameters (weights), increases the size of an image vector X. This layer, also called an upsampling [139] or deconvolution layer, can recover the spatial information in X (see Chapter 8.5.2 in Aggarwal 2018

[2]).

Softmax activation function converts a vector of k real values to a vector of k real values that sum to 1 (see page 14 in Aggarwal 2018 [2]). Softmax is useful because it converts the scores in the vector to a normalized probability distribution, which can be displayed to a user. In our model, softmax is used to output the class label (1 vs. -1 or non-significant flux vs. significant flux) of each pixel.

Rectified linear unit (ReLU) employs an activation function $f(x)$, defined as $f(x) = \max(0, x)$, where x is the input to a neuron, $f(x) = x$ if $x \geq 0$ and $f(x) = 0$ otherwise (see Chapter 1.2 in Aggarwal 2018 [2]). It is easy to train a model that uses ReLUs, which often achieves good performance.

CHAPTER 3

FIBRILS TRACING

3.1 Background and Related Work

Fibrils are thin threadlike absorption features ubiquitously observed in the solar chromosphere. Depending on their location and dynamic behavior, they may have different names, e.g., threads of filaments [109, 153], the superpenumbra of sunspots [108, 76], mottles in quiet-Sun rosette structures [67], etc. Fibrils are often observed with narrowband solar filtergrams in the chromospheric spectral lines such as Hα, where they are denser than their surroundings [113]. Physically speaking, fibrils represent the cool gas "frozen" in magnetic field lines and protected by magnetic fields from diffusing out [124, 93, 128]. For this reason, fibrils have been traditionally assumed to be aligned with the direction of the chromospheric magnetic field [39, 40].

Tracing chromospheric fibrils in Hα is an important subject in heliophysics research [77, 96], and has attracted much attention in the heliophysics community. The comparison between fibrils and the potential magnetic field may provide a quick way to examine the nonpotentiality of active regions (ARs) [77]. The orientation of fibrils could be used as a constraint to improve the non-linear force-free modeling of coronal fields [158, 5, 36]. Tracing fibrils also helps estimate the amount of energy in acoustic waves [38] and the free magnetic energy in the chromosphere [5].

Many fibril tracing methods have been developed in recent years. Leenaarts et al. [96] conducted three-dimensional magnetohydrodynamic simulations to investigate the relation between chromospheric fibrils and magnetic field lines. Aschwanden et al. [5] performed nonpotential field modeling of chromospheric structures and coronal loops with the VCA-NLFFF code. Jafarzadeh et al. [73] adopted the CRISPEX tool for visual inspection and identification of isolated slender fibrils.

Gafeira et al. [44] used image processing and contrast enhancement techniques to identify these fibrils. Asensio Ramos et al. [7] employed the rolling Hough transform (RHT) for fibril detection and a Bayesian hierarchical model to analyze the pixels in spectro-polarimetric chromospheric images of penumbrae and fibrils. The authors concluded that fibrils are often well aligned with magnetic azimuth. This RHT technique has also been used by Schad [132] to analyze fibrils and coronal rain. Jing et al. [77] developed a threshold-based algorithm to automatically segment chromospheric fibrils from Hα observations and extracted direction information along the fibrils with a fibril-fitting algorithm. The authors further quantitatively measured the nonpotentiality of the fibrils by the magnetic shear angle. In contrast to the above methods, our deep learning-based tool (FibrilNet) presented here can automatically predict fibrils and measure the uncertainties in the predicted results simultaneously.

Deep learning is a branch of machine learning where neural networks are designed to learn from large amounts of data [94]. It has been used extensively in computer vision and natural language processing, and more recently in astronomy and astrophysics for flare prediction, spectroscopic analysis, solar image segmentation, among others [69, 98, 81, 99, 103, 163, 75]. Different from the previous solar image segmentation techniques, which focus on predicting a value for each pixel, our FibrilNet employs a probabilistic segmentation model, specifically a Bayesian convolutional network, that predicts a value for each pixel accompanied with reliable uncertainty quantification. Such a model leads to a more informed decision, and improves the quality of prediction.

In general, there are two types of uncertainty in Bayesian modeling: aleatoric uncertainty and epistemic uncertainty [79]. Aleatoric uncertainty, also known as data uncertainty, measures the noise inherent in observations. Epistemic uncertainty, on the other hand, measures the uncertainty in the parameters of a model; this uncertainty is also known as model uncertainty. Quantifying uncertainties

with machine learning finds many applications ranging from computer vision [79], natural language processing [164], medical image analysis [86] to geomagnetic storm forecasting [60, 165]. Here we present a new application of uncertainty quantification with machine learning in fibril tracing.

3.2 Observations and Data Preparation

The Goode Solar Telescope (GST) is a 1.6 m clear aperture, off-axis telescope at BBSO, which is located in Big Bear Lake, California [25, 56, 55, 149]. GST is equipped with a high-order adaptive optics system, AO-308, which provides high-order correction of atmospheric seeing within an isoplanatic patch (about 6″ at 500 nm in summer), with a gradual roll-off of correction at larger distances [140]. Under a stable seeing condition, BBSO/GST observed AR 12665 at (W27°, S4°) on July 13, 2017, in which the data taken during ∼20:16-22:42 UT were used in the study presented here.

The Visible Imaging Spectrometer(VIS) [25] of GST utilizes a telecentric mount of the Fabry-Prot etalon. This imaging system was used for observing the Hα line. It scanned the target area at 0.6, 0.4 and 0.0 Å (0.08 Å bandpass) from the Hα line center 6563 Å with a 70″ circular field of view (FOV). At each wavelength step, the 25 frames, out of 60 frames taken in succession, with the best contrast were saved. These frames, with exposure time ranging from 7 to 20 ms and an image scale of 0″.03 per pixel, were processed by the high-order AO system and post-facto speckle image reconstruction algorithms [162], which improved the quality of the images by correcting the wavefront deformation caused by atmospheric distortion. An Hα line scan was performed over the FOV, and the position with the minimum intensity was defined as the Hα line center. It should be pointed out that the GST narrowband Hα data do not contain the full spectral information, which restricts the

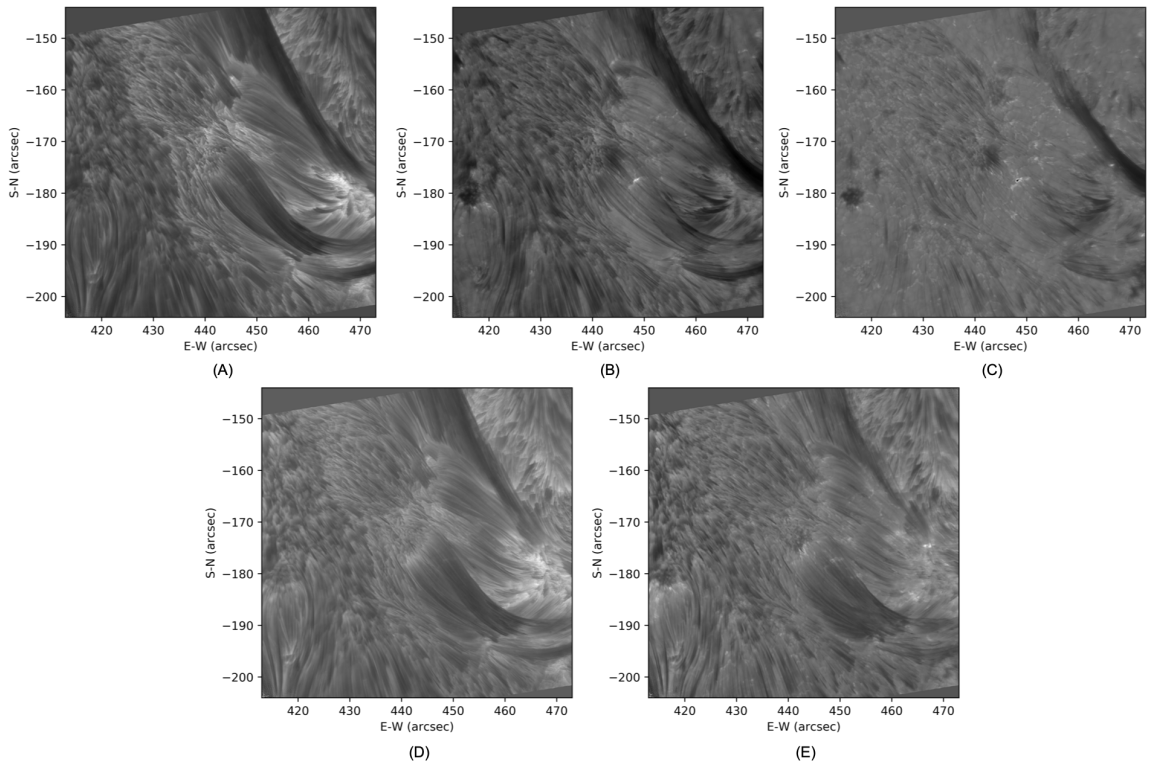

Figure 3.1 Five test images at (A) 0.0 Å, (B) +0.4 Å, (C) +0.6 Å, (D) −0.4 Å, (E) −0.6 Å, respectively, from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT. Enormous amounts of fibrils exist in these Hα images.

full characterization of fibrils in three dimensions. Therefore, our study of fibrils is all based on their projected morphology on the observational image plane.

Our dataset contained the GST Hα observations in AR 12665 from 20:16:32 UT to 22:41:30 UT on July 13, 2017 where the observed region was located at (W27°, S4°). During this period of time, 241 Hα line center images (i.e., those at 0.0 Å from the Hα line center 6563 Å with a 70″ circular FOV) were used as training data since features in these images were abundant. The test set contained five Hα images taken from AR 12665 at 20:15:58 UT on the same day (see Figure 3.1). Thus, there were 241 training Hα images and 5 test Hα images where the size of each image was 720×720 pixels. The training and test sets were disjoint, as the training observations and test observations were taken at different time points. Please note that the five test images

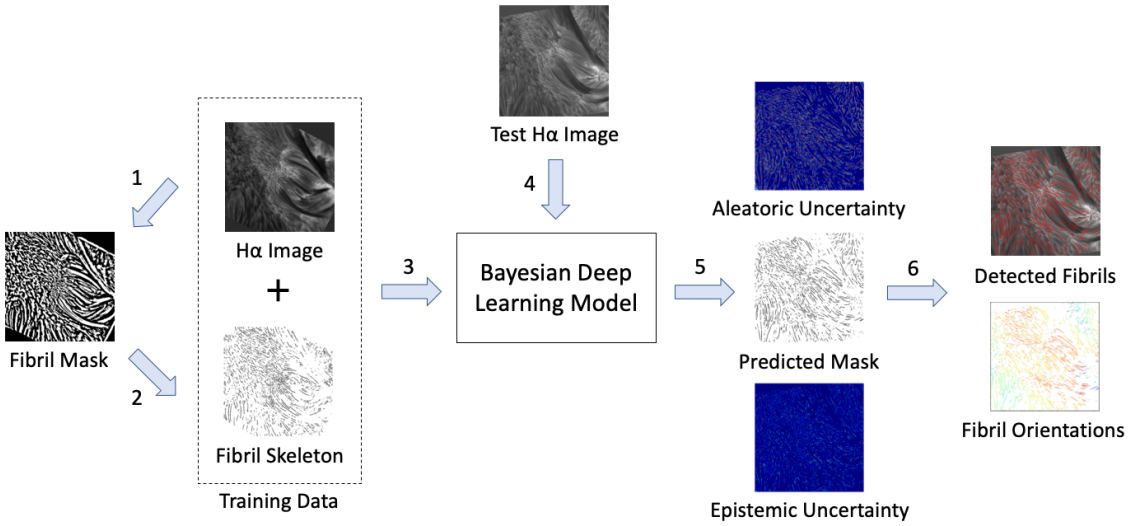

Figure 3.2 Illustration of the proposed method (FibrilNet) for fibril tracing. FibrilNet employs a Bayesian deep learning model for probabilistic image segmentation with uncertainty quantification to predict fibrils and a fibril-fitting algorithm to determine fibril orientations. The training data used to train the Bayesian deep learning model are highlighted in the dashed box. The tracing results for the test Hα image include predicted/detected fibrils, their orientations, aleatoric uncertainty and epistemic uncertainty.

were chosen in such a way that they were on five different wavelength positions rather than at five different time points on the same wavelength position. The reason why we did not choose the test images equally distributed over the time series on the same wavelength position was because the features in the images on the same wavelength position did not change much across the images. By contrast, the features in the images on the five different wavelength positions appeared quite differently as shown in Figure 3.1.

3.3 Methodology

3.3.1 Overview of FibrilNet

Figure 3.2 explains how FibrilNet works. Training Hα images are pre-processed in steps 1 and 2, and then used to train the Bayesian deep learning model (step 3). The trained model takes as input a test Hα image (step 4) and produces as output

a predicted mask accompanied with results for quantifying aleatoric uncertainty and epistemic uncertainty (step 5). In the post-processing phase (step 6), based on the predicted mask, fibrils on the test Hα image are detected and highlighted by thin red curves. Furthermore, the orientations of the detected fibrils are determined based on a fibril-fitting algorithm where the orientations are shown by different colors.

Specifically, in step 1, we apply the threshold-based tool developed by Jing et al. [77] to each training Hα image described in Section 3.2 to obtain a corresponding fibril mask. Fibril patterns on this mask are very thick, which contains a lot of noise. In step 2, we refine the fibril mask via a skeletonization procedure to obtain a fibril skeleton in which fibrils are marked by black and regions without fibrils are marked by white. The skeletonization procedure works by extracting a region-based shape feature representing the general form of fibrils. This skeletonization procedure results in better and cleaner images suitable for model training [147].

The training Hα images and fibril skeletons are then used to train the Bayesian deep learning model for probabilistic image segmentation and uncertainty quantification (step 3). During training, in order to obtain a robust model, we use the data augmentation technique described in Jiang et al. [75] to expand the training set by shifting, rotating, flipping and scaling the training images. In step 4, a test Hα image is fed to the trained Bayesian deep learning model. During testing, we use the Monte Carlo (MC) dropout sampling technique described in Subsection 3.3.2 to produce the predicted mask of the test Hα image accompanied with aleatoric uncertainty and epistemic uncertainty results (step 5). In step 6, by using the fibril-fitting algorithm based on the polynomial regression model described in Subsection 3.3.3, our FibrilNet tool outputs detected fibrils marked by red color on the test Hα image and their orientations represented by different colors.

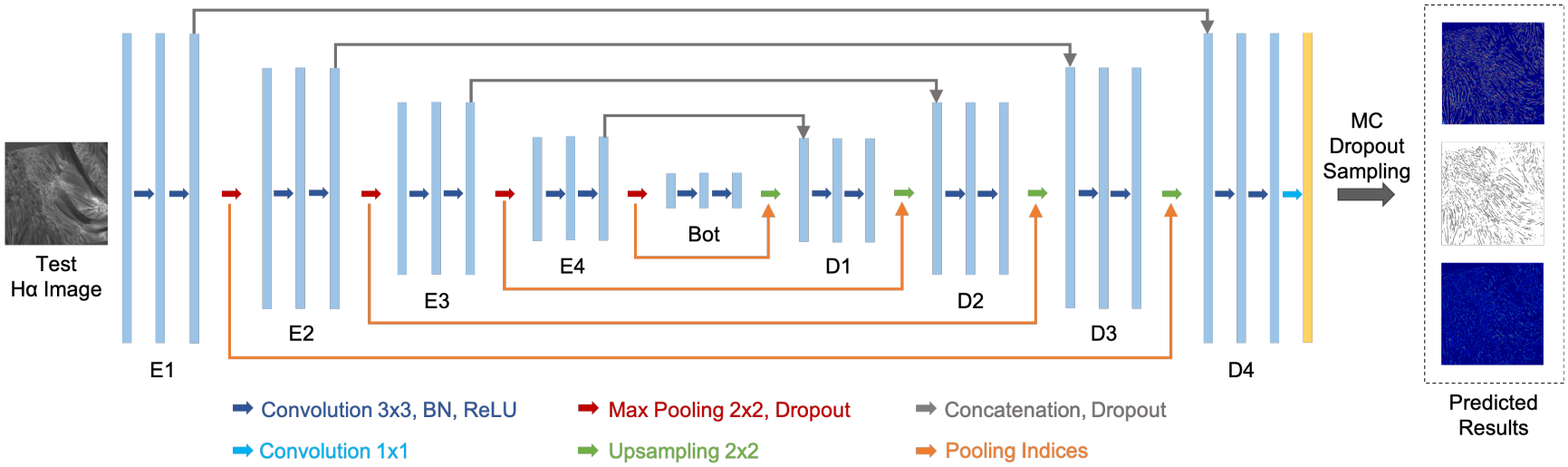

Figure 3.3 Architecture of the encoder-decoder convolutional neural network (i.e., the Bayesian deep learning model) used in FibrilNet. This network is similar to the one presented in [75]. See text for their differences.

3.3.2 Implementation of the Bayesian Deep Learning Model in FibrilNet

The Bayesian deep learning model used by FibrilNet is similar to the model used in SolarUnet [75] for tracking magnetic flux elements. Both models have 4 encoder blocks (E1, E2, E3, E4), 4 decoder blocks (D1, D2, D3, D4), mediated by a bottleneck (Bot). See Figure 3.3 and Jiang et al. [75] for the configuration and parameter settings of the models. While both models are based on an encoder-decoder convolutional neural network, they differ in three ways. First, in performing 2×2 max pooling, represented by a red arrow in Figure 3.3, the corresponding max pooling indices are stored. During decoding, the max pooling indices at the corresponding encoder layer are recalled, represented by an orange arrow, to upsample, represented by a green arrow, as done in Badrinarayanan et al. [12]. This upsampling technique used by FibrilNet, designed to reduce the number of trainable parameters in the model (network) and hence save memory, is different from the up-convolution layers used in SolarUnet. Second, since fibril patterns are relatively vague and harder to identify than magnetic flux elements, FibrilNet uses twice as many kernels as those in SolarUnet in all of the blocks in the encoder and decoder, as well as the bottleneck. Finally, during testing, instead of using the trained model (network) directly to produce segmentation results as done in SolarUnet, FibrilNet employs a Monte Carlo (MC) dropout sampling technique, detailed below, to produce, for a test Hα image, a predicted mask accompanied with aleatoric uncertainty and epistemic uncertainty results (see Figure 3.2). This MC dropout sampling technique allows FibrilNet to perform probabilistic image segmentation with uncertainty quantification, which is lacking in SolarUnet.

Specifically, to quantify uncertainty with the convolutional neural network, we use a prior probability, $P(\mathbf{W})$, over the network's weights, $\mathbf{W}$. During training, pairs of Hα images and their corresponding fibril skeletons, collectively referred to as $\mathbf{D}$,

are used to train the network. According to Bayes' theorem,

$$P(\mathbf{W} \mid \mathbf{D}) = \frac{P(\mathbf{D} \mid \mathbf{W})P(\mathbf{W})}{P(\mathbf{D})}. \tag{3.1}$$

Computing the exact posterior probability, $P(\mathbf{W} \mid \mathbf{D})$, is intractable [32]. Nevertheless, we can use variational inference [59] to learn the variational distribution over the network's weights parameterized by θ, $q_\theta(\mathbf{W})$, by minimizing the KullbackLeibler (KL) divergence of $q_\theta(\mathbf{W})$ and $P(\mathbf{W} \mid \mathbf{D})$ [17]. It is known that training a network with dropout is equivalent to a variational approximation on the network [45]. Furthermore, minimizing the cross-entropy loss of the network is equivalent to the minimization of the KL divergence [57]. Therefore, we use a binary cross-entropy loss function and the adaptive moment estimation (Adam) optimizer [57] with a learning rate of 0.0001 to train our model (network). Let $\hat{\theta}$ denote the optimized variational parameter obtained by training the model (network); we use $q_{\hat{\theta}}(\mathbf{W})$ to represent the optimized weight distribution.

In deep learning, dropout is mainly used to prevent over-fitting, where a trained model overfits training data and hence can not be generalized to make predictions on unseen test data. During training, dropout refers to ignoring or dropping out units (i.e., neurons) of certain set of neurons which is chosen randomly. During testing, dropout can be used to retrieve T Monte Carlo (MC) samples by processing the input test Hα image T times [45]. (In the study presented here, T is set to 50.) Each time a set of weights is randomly drawn from $q_{\hat{\theta}}(\mathbf{W})$. Each pixel in the predicted mask, shown in step 5 of Figure 3.2, gets a mean and variance over the T samples. If the mean is greater than or equal to a threshold, the pixel is marked by black indicating that the pixel is part of a fibril; otherwise the pixel is marked by white indicating that the pixel is not part of a fibril. (In the study presented here, the threshold is set to 0.5.) Following Kwon et al. [86], we decompose the variance into the aleatoric uncertainty and epistemic uncertainty at the pixel. The aleatoric uncertainty captures

the inherent randomness of the predicted result, which comes from the input test Hα image, while the epistemic uncertainty comes from the variability of $\mathbf{W}$, which accounts for the uncertainty in the model parameters (weights).

In the post-processing phase, we use a connected-component labeling algorithm [66] to group all adjacent black segments if their pixels in edges or corners touch each other. For each resulting group, which represents a fibril, we locate its pixels in the predicted mask and highlight their corresponding pixels in the test Hα image by red. (Resulting groups containing less than 10 pixels are considered as noise and filtered out.) We then output the detected fibrils highlighted by red color in the test Hα image, as shown in step 6 of Figure 3.2.

3.3.3 Implementation of the Fibril-Fitting Algorithm in FibrilNet

Most of the detected fibrils are lines or curves. In contrast to Jing et al. [77], which used a quadratic function to fit the detected fibrils, we adopt a polynomial regression model here. Specifically, our regression model is a polynomial function with varying degrees capable of fitting the detected fibrils with different curvatures. In general, regression analysis investigates the relationship between a dependent variable and an independent variable [16]. We model a detected fibril as an nth degree polynomial function as follows:

$$y = \gamma_0 + \gamma_1 x + \gamma_2 x^2 + \ldots + \gamma_n x^n + \epsilon, \tag{3.2}$$

where γ_i are coefficients and ϵ is a random error term. In Equation (3.2), the independent variable x represents the x coordinate of a pixel in the detected fibril and the dependent variable y represents the y coordinate of the same pixel, where the x-axis represents the E-W direction and the y-axis represents the S-N direction (see Figure 3.1). When the degree n equals 1, Equation (3.2) represents a linear regression

model, meaning that the detected fibril is represented by a straight line. In our work, n ranges from 1 to 10.

We then use the least squares method [117] to find the optimal γ_i values. There are 10 candidate polynomial functions for representing the detected fibril. We use the R-squared score [117] to assess the feasibility of these 10 candidate polynomial functions. Specifically, we choose the candidate polynomial function yielding the largest R-squared score, and use this polynomial function to represent the detected fibril. To determine the orientation of the detected fibril, we calculate the derivative of the chosen polynomial function. For each pixel on the detected fibril, we thus obtain the slope of the tangent at the pixel, leading to the orientation angle of the pixel, denoted θ_f, with respect to the x-axis. Notice that the orientation angle θ_f is in the $0^\circ - 180^\circ$ range, as two directions differing by 180° are indistinguishable here because the detected fibril in Hα does not carry information on the vertical dimension. Thus, θ_f represents the direction of the detected fibril with a 180° ambiguity [77].

3.4 Results

3.4.1 Tracing Results of FibrilNet Based on Data from AR 12665

In this series of experiments, we used the 241 Hα line center images from 20:16:32 UT to 22:41:30 UT on 2017 July 13 mentioned in Section 3.2 along with their corresponding fibril skeletons to train the FibrilNet tool as described in Section 3.3. We then used the trained tool to predict and trace fibrils on the five test images at 0.0 Å, +0.4 Å, +0.6 Å, −0.4 Å, −0.6 Å, respectively, from the Hα line center 6563 Å with a $70''$ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT (see Figure 3.1). Figure 3.4 presents tracing results on the test image at 0.0 Å; tracing results on the other four test images can be found in the Section 3.6. In all of the tracing results, fibrils containing 10 or fewer pixels were treated as noise and excluded.

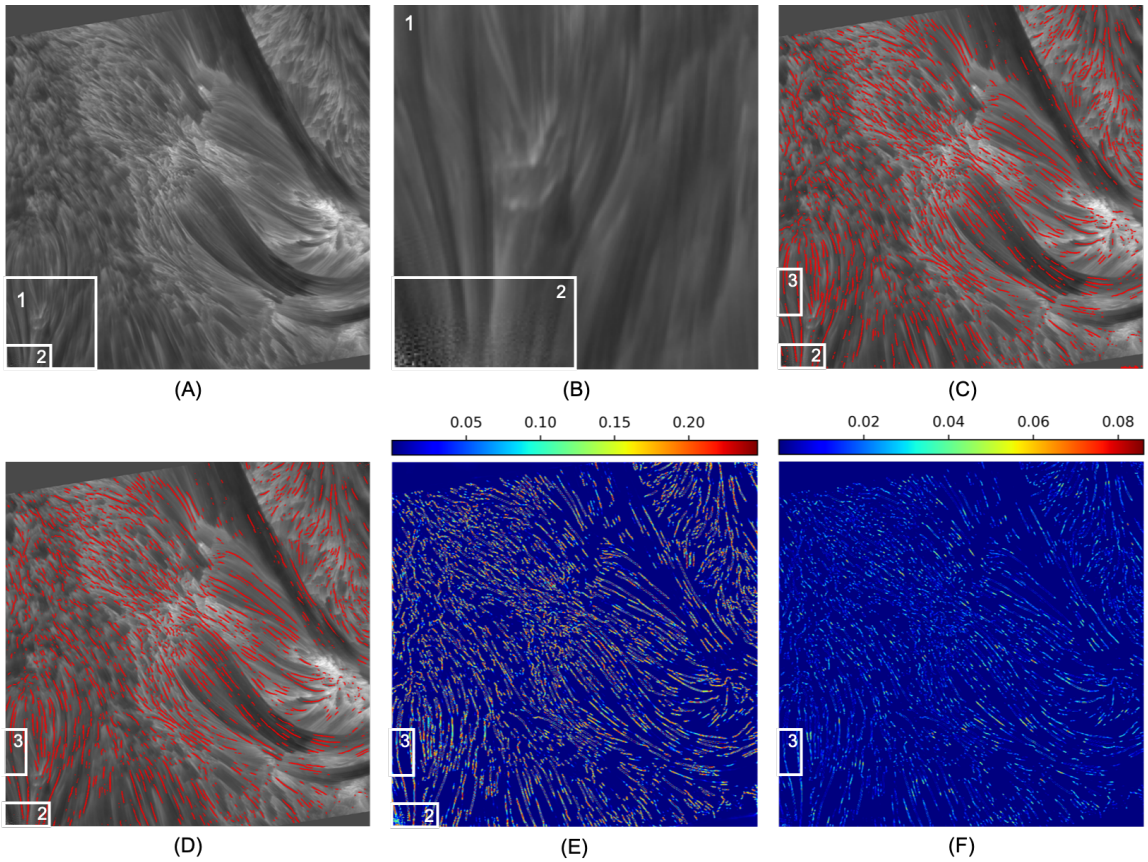

Figure 3.4 Fibril tracing results on the test image at 0.0 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT where training data were 241 Hα line center images taken from the same AR between 20:16:32 UT and 22:41:30 UT on the same day. (A) The original test Hα image. (B) The enlarged FOV of the region highlighted by the white box 1 in (A). (C) Fibrils (red curves) on the test Hα image detected by the tool in Jing et al. [77]. (D) Fibrils (red curves) on the test Hα image predicted by FibrilNet. (E) The aleatoric uncertainty (data uncertainty) map produced by FibrilNet. (F) The epistemic uncertainty (model uncertainty) map produced by FibrilNet.

Figure 3.4(A) shows the original test Hα image. Figure 3.4(B) shows the enlarged FOV of the region highlighted by the white box 1 in Figure 3.4(A). It can be seen from Figure 3.4(B) that there are salt-and-pepper noise pixels in the region highlighted by the white box 2, where the noise pixels are caused by image reconstruction limitations. Figure 3.4(C) shows the fibrils (red curves) on the test Hα image detected by the tool (after skeletonization) presented in Jing et al. [77]. Figure 3.4(D) shows the fibrils (red curves) predicted by FibrilNet. FibrilNet uses the images processed by the tool in Jing et al. [77] as training data. The results in Figures 3.4(C) and 3.4(D) are quite similar, demonstrating the good learning capability of FibrilNet.

Figures 3.4(E) and 3.4(F) show the aleatoric uncertainty (data uncertainty) and epistemic uncertainty (model uncertainty) maps, respectively, produced by FibrilNet. Regions predicted with less uncertainty and higher confidence are colored by blue. Regions predicted with more uncertainty and lower confidence are colored by red. We can see that the main source of uncertainty comes from the data rather than the model. Specifically, the values in the data uncertainty map in Figure 3.4(E) range from 0 to 0.246 while the values in the model uncertainty map in Figure 3.4(F) range from 0 to 0.086. Furthermore, we observe that the ends of a detected fibril are often associated with higher uncertainty. This happens because there is ambiguity surrounding the transition from the fibril body to the non-fibril background area, a finding consistent with that in object detection with uncertainty quantification reported in the literature [79, 86].

Notice also that the map in Figure 3.4(E) shows higher uncertainty in the noisy region inside the white box 2 compared to the region outside the white box 2. Specifically, in the noisy region inside the white box 2, 90% of the values are contained in the range [0.00014 (5%), 0.20528 (95%)]. By contrast, in the region outside the white box 2, 90% of the values are contained in the range [0 (5%), 0.18969 (95%)].

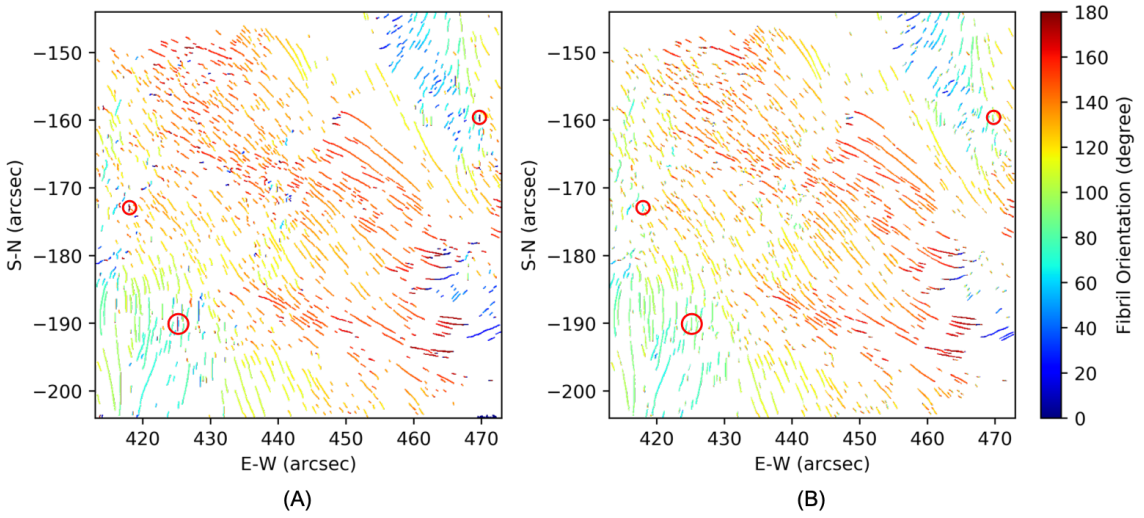

Figure 3.5 Orientation angles (colored curves) of the detected fibrils on the test image at 0.0 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT. (A) Fibril orientation angles calculated by the tool in Jing et al. [77]. (B) Fibril orientation angles determined by FibrilNet. Orientation angles of a number of fibrils, some of which are highlighted by small red circles here, are calculated wrongly by the tool in Jing et al. [77], but correctly by FibrilNet.

Furthermore, it is observed in Figure 3.4(C) that the tool in Jing et al. [77] misses some fibril structures with at least 15 pixels inside the white box 3. These fibril structures are not present in the mask predicted by FibrilNet either, as shown inside the white box 3 in Figure 3.4(D). Nevertheless, the uncertainty maps of FibrilNet are able to catch and display these missed fibril structures with higher uncertainty by Bayesian inference, as shown inside the white box 3 in Figures 3.4(E) and 3.4(F), respectively. This finding demonstrates the usefulness of the uncertainty maps, as they not only provide a quantitative way to measure the confidence on each predicted fibril, but also help identify fibril structures that are not detected by the tool in Jing et al. [77] but are inferred through machine learning. It should be pointed out that the previous fibril tracing tool in Jing et al. [77] does not have the capability of producing these uncertainty maps as described here.

Figure 3.5 compares the orientation angles of the fibrils found by the tool in Jing et al. [77] and by FibrilNet, respectively. The colors of angles between 0° and 90° range from dark blue to green. The colors of angles between 90° and 180° range from green to dark red. It can be seen from Figure 3.5 that the orientation angles found by the two tools mostly agree with each other, though the angles detected by FibrilNet tend to be smoother. This happens because FibrilNet uses polynomial functions of varying degrees, as opposed to the quadratic function employed by the tool in Jing et al. [77], to better fit the detected fibrils with different curvatures. Notice also that the quadratic function used by the tool in Jing et al. [77] may produce wrong angles, which are calculated correctly by the polynomial regression model of FibrilNet. For example, the orientation angle of the fibril at E-W = 425″ and S-N = −190″, which is highlighted by a small red circle in Figure 3.5, is roughly 90°. It is calculated incorrectly by the tool in Jing et al. [77], as shown in Figure 3.5(A). On the other hand, FibrilNet calculates the orientation angle of this fibril correctly, as shown in Figure 3.5(B). It should be pointed out that the smoother and more accurate orientation angles detected by FibrilNet are due to the better fibril-fitting algorithm used by the tool, as explained above. They are not caused by FibrilNet's Bayesian deep learning model, whose purpose is mainly for image segmentation (i.e., marking each pixel by black indicating the pixel is part of a fibril or white indicating the pixel is not part of a fibril as shown in the predicted mask in Figure 3.2) with uncertainty quantification (i.e., producing the uncertainty maps as shown in Figure 3.4).

3.4.2 Quantitative Assessment of FibrilNet Based on Data from AR 12665

As mentioned above, FibrilNet has two parts: (i) the Bayesian deep learning model for predicting fibrils with probabilistic image segmentation, and (ii) the fibril-fitting

algorithm for determining orientations of the predicted fibrils based on the polynomial regression function in Equation (3.2). Here, we adopt four measures, defined below, to quantitatively assess the first part, comparing the image segmentation algorithms employed by FibrilNet and the tool (after skeletonization) in Jing et al. [77], based on the same data from AR 12665 used in Subsection 3.4.1. Unlike FibrilNet, which employs deep learning for image segmentation, the tool in Jing et al. [77] used a threshold-based algorithm rather than machine learning for image segmentation.

Let A (B, respectively) denote the set of $720 \times 720 = 518,400$ pixels in the mask (skeleton, respectively) predicted by FibrilNet (calculated by the tool in Jing et al. [77], respectively) for a test image. Let $p \in A$ be a pixel in A and let $q \in B$ be p's corresponding pixel in B, i.e., q is at the same position as p. We use $A \cap_A B$ to represent the subset of pixels in A such that for each pixel p in $A \cap_A B$ and p's corresponding pixel q in B, p and q are marked by the same color. That is, p, q are both marked by black indicating p (q, respectively) is part of a fibril in A (B, respectively), or p, q are both marked by white indicating p (q, respectively) is not part of a fibril in A (B, respectively). Similarly, we use $A \cap_B B$ to represent the subset of pixels in B such that for each pixel q in $A \cap_B B$ and q's corresponding pixel p in A, q and p are marked by the same color. The first quantitative measure is the pixel similarity (PS), also known as global accuracy [12], which is defined as

$$\text{PS} = \frac{|A \cap_A B| + |A \cap_B B|}{|A| + |B|}, \tag{3.3}$$

where $|.|$ is the cardinality of the indicated set. PS is used to assess the pixel-level similarity between the mask A predicted by FibrilNet and the skeleton B calculated by the tool in Jing et al. [77] for the test image. The value of PS ranges from 0 to 1. The larger (i.e., closer to 1) the PS value, the higher the pixel-level similarity between the mask A and the skeleton B.

Let A_F (B_F, respectively) denote the set of pixels on the fibrils in A (B, respectively). Thus, in A, the pixels in A_F are marked by black while the pixels not in A_F are marked by white. Similarly, in B, the pixels in B_F are marked by black while the pixels not in B_F are marked by white. We use $A_F \cap_{A_F} B_F$ to represent the subset of pixels in A_F such that for each black pixel p in $A_F \cap_{A_F} B_F$, p's corresponding pixel q is also black, i.e., q is in B_F. Similarly, we use $A_F \cap_{B_F} B_F$ to represent the subset of pixels in B_F such that for each black pixel q in $A_F \cap_{B_F} B_F$, q's corresponding pixel p is also black, i.e., p is in A_F. The second quantitative measure is the fraction of common fibril pixels (FCFP), defined as

$$\text{FCFP} = \frac{|A_F \cap_{A_F} B_F| + |A_F \cap_{B_F} B_F|}{|A_F| + |B_F|}. \tag{3.4}$$

FCFP is used to measure the pixel-level similarity between the fibrils predicted by FibrilNet and those found by the tool in Jing et al. [77]. The value of FCFP ranges from 0 to 1. The larger (i.e., closer to 1) the FCFP value, the higher the pixel-level similarity between the fibrils predicted by FibrilNet and those found by the tool in Jing et al. [77].

The third quantitative measure is the fraction of disjunct fibril pixels (FDFP), defined as

$$\text{FDFP} = 1 - \text{FCFP}\,. \tag{3.5}$$

FDFP is used to measure the pixel-level dissimilarity (distance) between the fibrils predicted by FibrilNet and those found by the tool in Jing et al. [77]. The value of FDFP ranges from 0 to 1. The smaller (i.e., closer to 0) the FDFP value, the higher the pixel-level similarity between the fibrils predicted by FibrilNet and those found by the tool in Jing et al. [77].

The fourth quantitative measure is the Rand Index (RI) [126, 148], which calculates the ratio of pairs of pixels whose colors (black or white) are consistent

between the mask A predicted by FibrilNet and the skeleton B calculated by the tool in Jing et al. [77] for the test image. RI accommodates the inherent ambiguity in image segmentation, and provides region sensitivity and compensation for coloring errors near the ends of detected fibrils. For example, consider a wider fibril. FibrilNet may detect the portion to the left of the center of the fibril and highlight this portion by red. The tool in Jing et al. [77] may detect the portion to the right of the center of the fibril and highlight that portion by red. Under this circumstance, FCFP does not consider there are common pixels between the two red curves, though RI treats the two red curves as consistent curves. Visually the fibril is indeed found by both tools. As a consequence, RI is often used in comparing image segmentation algorithms. The value of RI also ranges from 0 to 1. The larger (i.e., closer to 1) the RI value, the higher the visual similarity between the fibrils predicted by FibrilNet and those found by the tool in Jing et al. [77].

Table 3.1 presents the quantitative measure values of FibrilNet based on the five test images in Figure 3.1. It can be seen from the table that the mask predicted by FibrilNet and the skeleton calculated by the tool in Jing et al. [77] are very similar at pixel level, with PS $\geq$ 95% on the test images. The fraction of common fibril pixels (FCFP) is about 80%. However, visually, the similarity/consistency between the fibrils predicted by FibrilNet and those found by the tool in Jing et al. [77] is much higher, where the similarity/consistency is quantitatively assessed with RI $\geq$ 91% on the test images. This finding is consistent with the results presented in Figures 3.4(C) and 3.4(D).

Next, we quantitatively assess the second part of FibrilNet, comparing the fibril-fitting algorithms employed by FibrilNet and the tool (after skeletonization) in Jing et al. [77], based on the same data from AR 12665 described in Subsection 3.4.1. The fibril-fitting algorithms are used to determine orientations of detected fibrils. Let θ_f represent the fibril orientation angle of a pixel calculated by the polynomial regression

Table 3.1 Comparison of the Image Segmentation Algorithms Used in FibrilNet and the Tool of Jing et al. [77] Based on Four Quantitative Measures and Five Test Images

Test Image	PS	FCFP	FDFP	RI
Hα 0.0 Å	0.9576	0.8038	0.1962	0.9188
Hα +0.4 Å	0.9571	0.8097	0.1903	0.9178
Hα +0.6 Å	0.9659	0.8079	0.1921	0.9340
Hα −0.4 Å	0.9546	0.7922	0.2078	0.9134
Hα −0.6 Å	0.9536	0.8022	0.1978	0.9115

function in FibrilNet, and let θ_j represent the fibril orientation angle of the same pixel calculated by the quadratic function in the tool of Jing et al. [77]. The acute angle difference between θ_f and θ_j, denoted $\delta(\theta_f, \theta_j)$, is defined as

$$\delta(\theta_f, \theta_j) = \begin{cases} |\theta_f - \theta_j| & \text{if } |\theta_f - \theta_j| \leq 90^\circ \\ 180^\circ - |\theta_f - \theta_j| & \text{otherwise} \end{cases} . \tag{3.6}$$

The angle difference is decided in favor of an acute or right angle, i.e., $0^\circ \leq \delta(\theta_f, \theta_j) \leq 90^\circ$.

Figure 3.6 quantitatively compares the orientation angles of common fibril pixels calculated by the fibril-fitting algorithms used in FibrilNet and the tool of Jing et al. [77] based on the test image at 0.0 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT. Figure 3.6(A) shows the 2D histogram of the orientation angles of common fibril pixels produced by the two tools where the x-axis (y-axis, respectively) represents the orientation angles calculated by FibrilNet (the tool of Jing et al. [77], respectively). The 2D histogram is computed by grouping common fibril pixels whose orientation angles are specified by their x and y coordinates into bins, and counting the common fibril pixels in a bin to compute the color of the tile representing the bin. The width of each bin equals 2 degrees. It

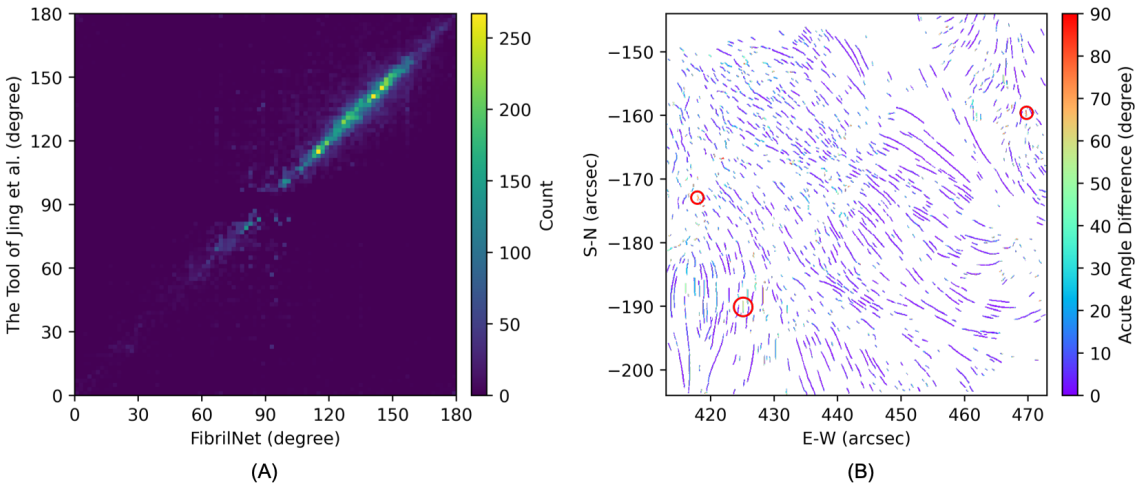

Figure 3.6 Quantitative comparison of the orientation angles of common fibril pixels calculated by the fibril-fitting algorithms used in FibrilNet and the tool of Jing et al. [77] based on the test image at 0.0 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT. (A) 2D histogram of the orientation angles of common fibril pixels produced by the two tools. (B) Differences of the orientation angles of common fibril pixels produced by the two tools. The orientation angles of the fibrils highlighted by small red circles are calculated wrongly by the tool of Jing et al. [77], but correctly by FibrilNet as indicated in Figure 3.5.

can be seen from Figure 3.6(A) that the orientation angles of common fibril pixels calculated by the two tools mostly agree with each other, which is consistent with the findings shown in Figure 3.5. Figure 3.6(B) shows differences of the orientation angles of common fibril pixels produced by the two tools. It can be seen from Figure 3.6(B) that most of the common fibril pixels have very small orientation angle differences, displayed by purple color. For the common fibril pixels with large orientation angle differences, the orientation angles calculated by the quadratic function used in the tool of Jing et al. [77] are often incorrect (see, for example, the fibrils highlighted by the small red circles in Figure 3.6(B) and Figure 3.5).

3.4.3 Application of FibrilNet to Other Data

In this series of experiments, we applied FibrilNet to other types of test images, including (i) a full-disk image from the Global Oscillation Network Group (GONG)

[63, 125] at the National Solar Observatory (NSO), (ii) a full-disk image from the Kanzelhhe Solar Observatory (KSO) [118, 119], (iii) high-resolution superpenumbral fibrils from BBSO [76], and (iv) two high-resolution quiet Sun regions from BBSO. The GONG full-disk Hα LH (Learmonth Reduced Hα) data in (i) was collected on 2015 September 28 00:01:34 UT. The KSO full-disk Hα Fi (Full-disk raw image) data in (ii) was collected on 2015 September 14 09:14:20 UT. The GONG and KSO full-disk images have relatively low resolution. The BBSO superpenumbra of sunspots in (iii) was collected at Hα -0.6 Å from AR 12661 (501E, 95N) on 2017 June 4 19:08:44 UT. The two BBSO quiet-Sun regions in (iv) were collected on 2018 July 29 16:33:12 UT and 2020 June 10 16:10:25 UT at Hα -0.6 Å from (604E, 125S) and Hα 0.0 Å from (283E, 789N), respectively. The FibrilNet tool was trained using the same 241 Hα line center images described in Section 3.2. Here we present results without uncertainty maps. Results with uncertainty maps can be generated similarly as done in Subsection 3.4.1.

Figure 3.7 shows fibrils (red curves) predicted by FibrilNet on the GONG and KSO test images. Figure 3.7(A) presents the GONG full-disk Hα image. Figure 3.7(B) shows the enlarged view of the region highlighted by the white box in Figure 3.7(A). In Figure 3.7(C), we see that FibrilNet detects many fibrils on the GONG image. Figure 3.7(D) presents the KSO full-disk Hα image. Figure 3.7(E) shows the enlarged view of the region highlighted by the white box in Figure 3.7(D). Figure 3.7(F) clearly demonstrates that FibrilNet detects the threads of filaments and fibrils on the KSO image.

Figure 3.8 presents fibril prediction results on the BBSO high-resolution test Hα images. Figure 3.8(A) shows the BBSO superpenumbra of sunspots image used in the study. It can be seen that there are superpernumbral fibrils around the sunspot in the center of the image. Figure 3.8(D) shows the predicted superpenumbral fibrils (red curves) produced by FibrilNet on the image in Figure 3.8(A). We see in Figure 3.8(D)

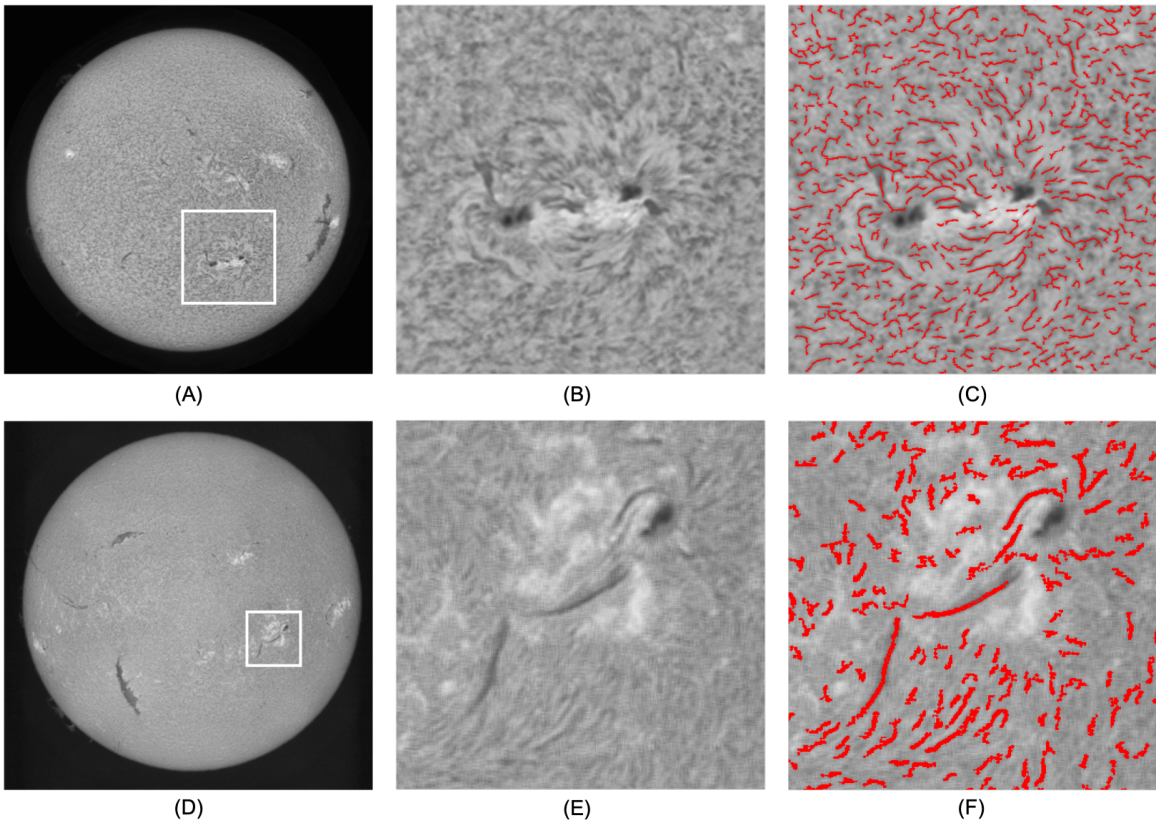

Figure 3.7 Fibrils (red curves) predicted by FibrilNet on the GONG and KSO full-disk Hα images collected on 2015 September 28 00:01:34 UT and 2015 September 14 09:14:20 UT, respectively. (A) The GONG full-disk Hα image. (B) The enlarged view of the region highlighted by the white box in (A). (C) Fibrils predicted by FibrilNet on the image in (B). (D) The KSO full-disk Hα image. (E) The enlarged view of the region highlighted by the white box in (D). (F) Fibrils predicted by FibrilNet on the image in (E).

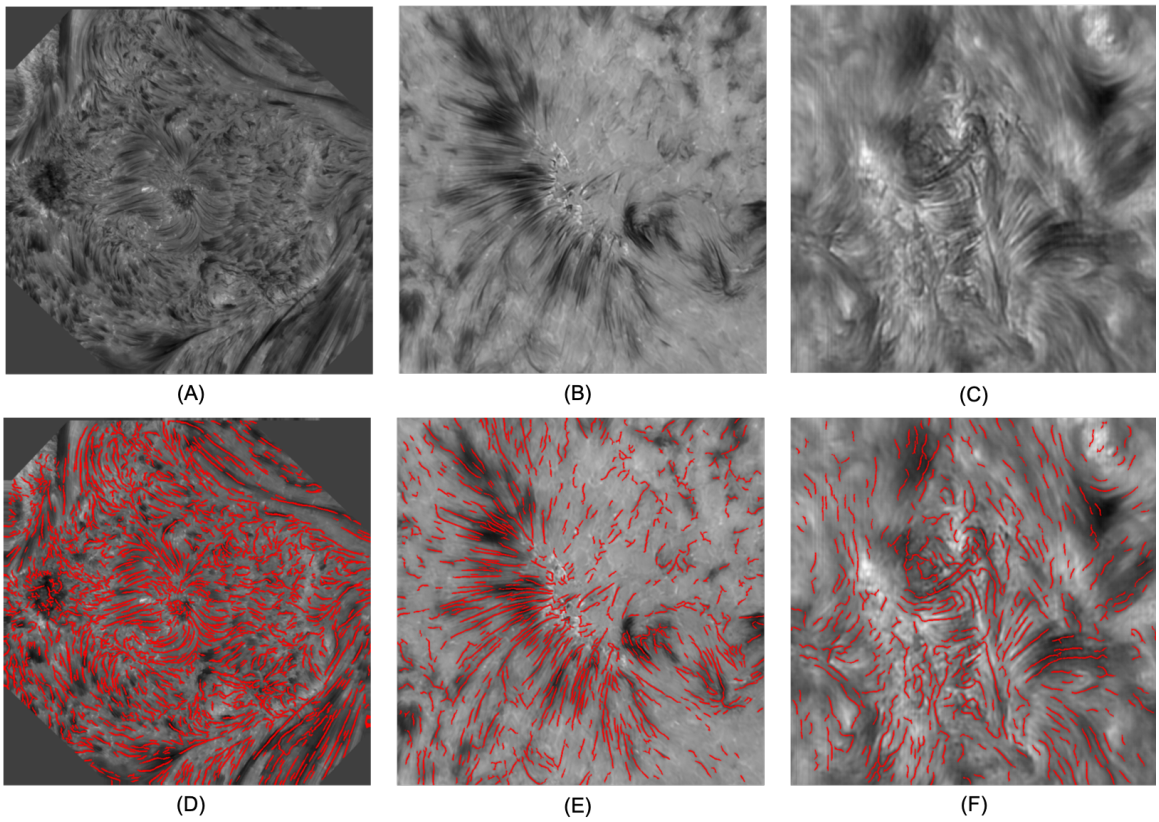

Figure 3.8 Fibrils (red curves) predicted by FibrilNet on additional high-resolution BBSO test Hα images. (A) The BBSO superpenumbra of sunspots image collected at Hα -0.6 Å from AR 12661 (501E, 95N) on 2017 June 4 19:08:44 UT. (B) The BBSO quiet-Sun image collected at Hα -0.6 Å from (604E, 125S) on 2018 July 29 16:33:12 UT. (C) The BBSO quiet-Sun image collected at Hα 0.0 Å from (283E, 789N) on 2020 June 10 16:10:25 UT. (D) Fibrils predicted by FibrilNet on the image in (A). (E) Fibrils predicted by FibrilNet on the image in (B). (F) Fibrils predicted by FibrilNet on the image in (C).

that FibrilNet can distinguish the superpenumbral fibrils from the clusters of spicules nearby. Figures 3.8(B) and 3.8(C) present the two BBSO quiet-Sun regions. Figures 3.8(E) and 3.8(F) show the predicted mottles in the quiet-Sun rosette structures in Figures 3.8(B) and 3.8(C), respectively. These high-resolution Hα images clearly demonstrate the good fibril prediction capability of our tool.

3.5 Summary

We develop a Bayesian deep learning method, FibrilNet, for tracing chromospheric fibrils in Hα images of solar observations. We apply the FibrilNet tool to high-resolution Hα images from an active region (AR 12665) collected by BBSO/GST on July 13, 2017. The tool performs well on these high-resolution Hα images, predicting fibrils with uncertainty quantification and determining the orientations of the predicted fibrils. We further apply FibrilNet to full-disk Hα images from other solar observatories and additional high-resolution Hα images collected by BBSO/GST, demonstrating the tool's usability in diverse datasets.

Our main results are summarized as follows:

1. The encoder-decoder convolutional neural network (i.e., the Bayesian deep learning model) used in FibrilNet, as illustrated in Figure 3.3, is an enhancement of two deep learning models, namely U-Net [34], based on which our SolarUnet [75] for magnetic tracking was developed, and SegNet [12]. FibrilNet predicts fibrils on a test Hα image through image segmentation (i.e., predicting each pixel in the test Hα image to be black indicating the pixel is part of a fibril or white indicating the pixel is not part of a fibril). In computer vision and image processing, U-Net and SegNet are two of the best image segmentation models. By combining these two models, FibrilNet produces good image segmentation (i.e., fibril prediction) results, as described in Section 3.4.

2. The training dataset used in this study comprises 241 high-resolution Hα line center images in AR 12665 collected by BBSO/GST from 20:16:32 UT to 22:41:30 UT on 2017 July 13. After FibrilNet is trained on this dataset, we apply the trained model to predict fibrils on five high-resolution test Hα images from the same active region (AR 12665) collected by BBSO/GST on 2017 July 13 20:15:58 UT as described in

Subsection 3.4.1, as well as an additional five test Hα images including two full-disk Hα images from GONG/KSO and three other high-resolution Hα images collected by BBSO/GST as described in Subsection 3.4.3. Our experimental results show that the Bayesian deep learning model employed by FibrilNet performs well not only on the five high-resolution test Hα images from AR 12665 that are not seen during training, but also on the additional five test Hα images. No further training is needed for FibrilNet to predict fibrils in the additional five test Hα images. This is achieved by the generalization and inference capabilities of the deep learning model used by FibrilNet. On the other hand, the threshold-based tool in Jing et al. [77] is tailored for the high-resolution Hα images collected by BBSO/GST. When applying the threshold-based tool in Jing et al. [77] to the GONG full-disk Hα image in Figure 3.7, the threshold-based tool performs poorly, missing many fibrils on the GONG Hα image.

3. FibrilNet obtains training data from the threshold-based tool in Jing et al. [77] where the training dataset contains 241 high-resolution Hα line center images from AR 12665 collected by BBSO/GST as described in item 2 above. When applying FibrilNet and the threshold-based tool to the five high-resolution test Hα images from the same active region (AR 12665) collected by BBSO/GST, the two tools agree well on the detected fibrils as described in Subsections 3.4.1 and 3.4.2. This demonstrates the good learning capability of FibrilNet. When predicting fibrils on a test Hα image, FibrilNet uses an uncertainty quantification technique (more precisely a Monte Carlo sampling technique) to process the test Hα image T times where $T = 50$ as described in Subsection 3.3.2. Unlike FibrilNet, which employs deep learning, the tool in Jing et al. [77] used a threshold-based algorithm, rather than machine learning, for image segmentation to detect fibrils on the test Hα image. It takes several seconds for the threshold-based tool to process the test Hα image. When the uncertainty quantification technique is turned off (i.e., T is set to 1), FibrilNet is ten times faster than the threshold-based tool in Jing et al. [77] due to the fact that FibrilNet detects fibrils through making predictions, while the two tools produce similar results. When the uncertainty quantification technique is turned on (i.e., T is set to 50), FibrilNet is as fast as the threshold-based tool while producing uncertainty maps that not only provide a quantitative way to measure the confidence on each detected fibril, but also help identify fibril structures that are not detected by the threshold-based tool (i.e., that do not exist in the training data) but are inferred through machine learning as described in Subsection 3.4.1. It is worth noting that the main source of uncertainty comes from the data rather than our deep learning model. Uncertainty values are higher in the noisy regions of the test Hα image. Furthermore, the ends of a predicted fibril are often associated with higher uncertainty,

due to the ambiguity surrounding the transition from the fibril body to the non-fibril background area. To the best of our knowledge, FibrilNet is the first tool capable of predicting fibrils with uncertainty quantification.

4. We conducted additional experiments to evaluate the effectiveness of the data augmentation technique used for training FibrilNet as described in Subsection 3.3.1. Our experimental results show that, without the data augmentation technique, the performance of FibrilNet degrades, particularly when the tool is applied to the GONG and KSO full-disk Hα images in Figure 3.7. This happens because the data augmentation technique can increase the generalization and inference capabilities of the Bayesian deep learning model used by FibrilNet. Our training dataset comprises 241 Hα line center images from AR 12665 collected on July 13, 2017 as described in Section 3.2. We also performed experiments where we split the training dataset into two parts based on image quality. The first part contained 12 Hα line center images with slightly lower quality. The second part contained the remaining 229 Hα line center images with higher quality. Since the first part contained too few Hα images, we expanded it by including 12 lower-quality images from the other four wavelength positions in AR 12665 studied here, yielding a total of 60 lower-quality Hα images. Our experimental results show that the deep learning models trained by all 241 Hα line center images and by the 229 higher-quality Hα line center images produce similar results. On the other hand, the performance of the deep learning model trained by the 60 lower-quality Hα images degrades, and becomes even worse in the absence of data augmentation, particularly when the model is applied to the GONG and KSO full-disk Hα images.

5. To further understand the behavior of FibrilNet, we trained the tool using all $241 \times 5 = 1205$ high-resolution Hα images from all five wavelength positions in AR 12665 studied here and applied the trained tool to the same test images described in Section 3.4. The results obtained are similar to those presented here, indicating our tool works equally well even with fewer training images. When the tool is trained by a much smaller dataset such as one with less than 100 Hα line center images from AR 12665 collected on July 13, 2017, the tool still performs well on the high-resolution test Hα images described in Section 3.4, but finds fragmented filaments and fibrils, rather than long, complete filaments and fibrils, on the KSO full-disk Hα image in Figure 3.7, even when the tool is trained by the data augmentation technique with higher-quality training images.

6. As mentioned above, the Bayesian deep learning model in FibrilNet performs image segmentation to predict fibrils with uncertainty quantification. On the other hand, the fibril-fitting algorithm in FibrilNet uses

a polynomial regression function with varying degrees to calculate the orientation angles of the predicted fibrils. This polynomial regression model produces more accurate and smoother fibril orientation angles than the quadratic function used by the tool in Jing et al. [77] as described in Subsections 3.4.1 and 3.4.2. However, if we replace the polynomial regression model by the quadratic function in FibrilNet, the two tools would produce the same orientation angles on common fibril pixels detected by the tools.

We conclude that FibrilNet is an effective and alternative method for fibril tracing. It is expected that this tool will be a useful utility for processing observations from diverse instruments including BBSO/GST and the new DKIST (Daniel K. Inouye Solar Telescope).

3.6 Additional Results

Figure 3.9 (Figure 3.10, Figure 3.11, Figure 3.12, respectively) compares fibril tracing results and fibril orientations obtained by FibrilNet and the tool in Jing et al. [77] on the test image at +0.4 Å (+0.6 Å, −0.4 Å, −0.6 Å, respectively) from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT where training data were 241 Hα line center images taken from the same AR between 20:16:32 UT and 22:41:30 UT on the same day.

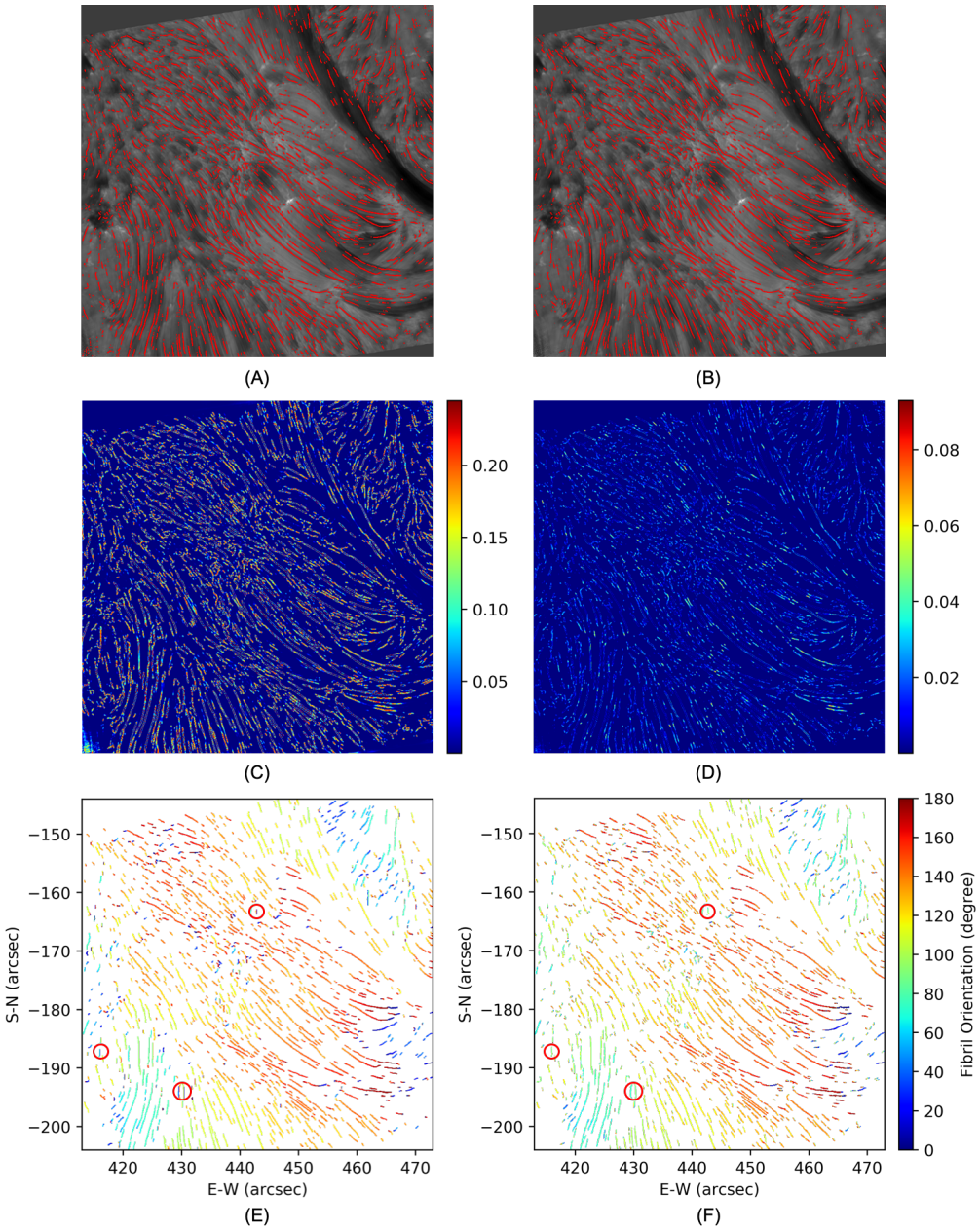

Figure 3.9 Fibril tracing results on the test image at +0.4 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT where training data were 241 Hα line center images taken from the same AR between 20:16:32 UT and 22:41:30 UT on the same day. (A) Fibrils on the test Hα image detected by the tool in Jing et al. [77]. (B) Fibrils on the test Hα image predicted by FibrilNet. (C) The aleatoric uncertainty (data uncertainty) map produced by FibrilNet. (D) The epistemic uncertainty (model uncertainty) map produced by FibrilNet. (E) Fibril orientation angles calculated by the tool in Jing et al. [77]. (F) Fibril orientation angles determined by FibrilNet. Orientation angles of a number of fibrils, some of which are highlighted by small red circles here, are calculated wrongly by the tool in Jing et al. [77], but correctly by FibrilNet.

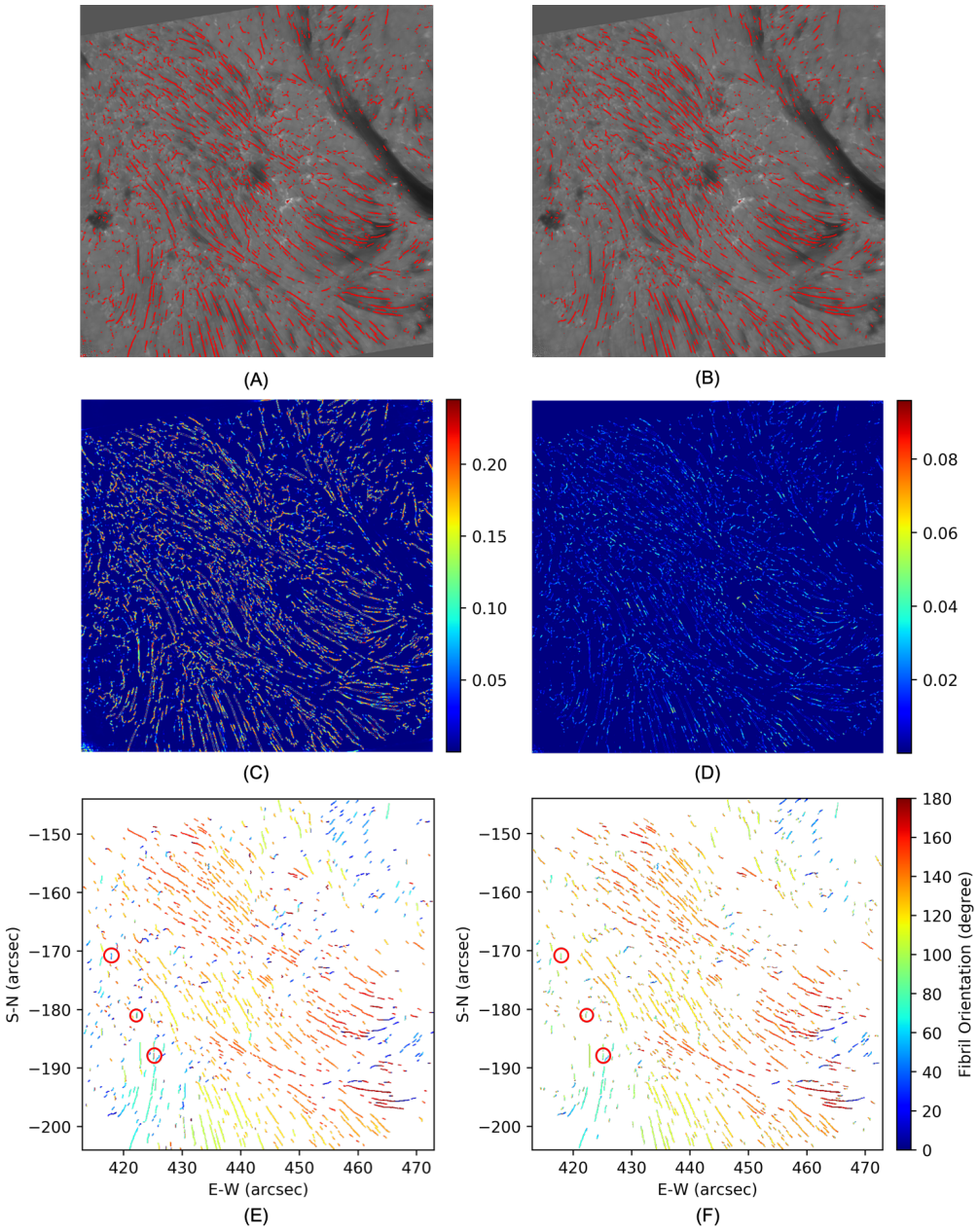

Figure 3.10 Fibril tracing results on the test image at +0.6 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT where training data were 241 Hα line center images taken from the same AR between 20:16:32 UT and 22:41:30 UT on the same day. (A) Fibrils on the test Hα image detected by the tool in Jing et al. [77]. (B) Fibrils on the test Hα image predicted by FibrilNet. (C) The aleatoric uncertainty (data uncertainty) map produced by FibrilNet. (D) The epistemic uncertainty (model uncertainty) map produced by FibrilNet. (E) Fibril orientation angles calculated by the tool in Jing et al. [77]. (F) Fibril orientation angles determined by FibrilNet. Orientation angles of a number of fibrils, some of which are highlighted by small red circles here, are calculated wrongly by the tool in Jing et al. [77], but correctly by FibrilNet.

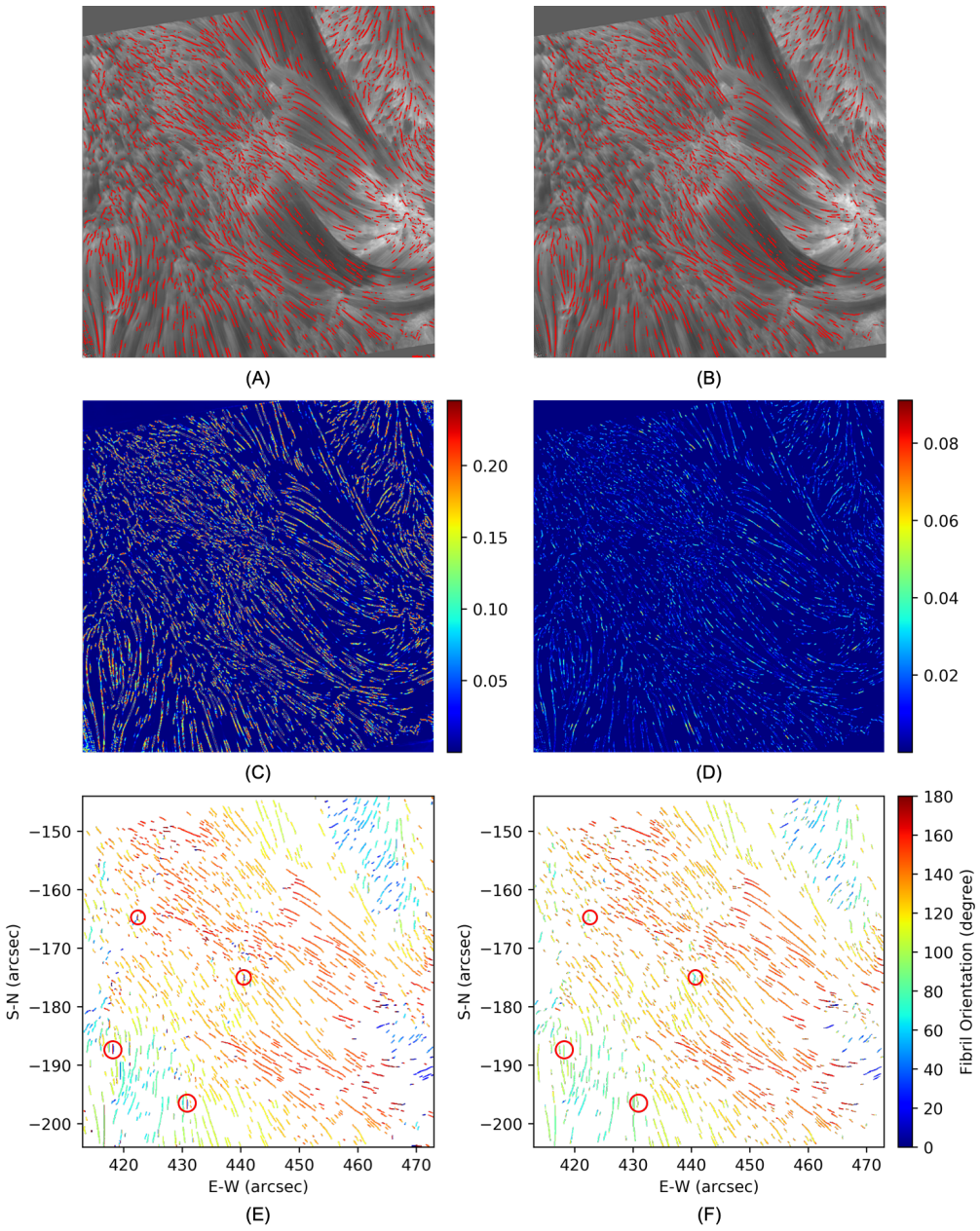

Figure 3.11 Fibril tracing results on the test image at −0.4 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT where training data were 241 Hα line center images taken from the same AR between 20:16:32 UT and 22:41:30 UT on the same day. (A) Fibrils on the test Hα image detected by the tool in Jing et al. [77]. (B) Fibrils on the test Hα image predicted by FibrilNet. (C) The aleatoric uncertainty (data uncertainty) map produced by FibrilNet. (D) The epistemic uncertainty (model uncertainty) map produced by FibrilNet. (E) Fibril orientation angles calculated by the tool in Jing et al. [77]. (F) Fibril orientation angles determined by FibrilNet. Orientation angles of a number of fibrils, some of which are highlighted by small red circles here, are calculated wrongly by the tool in Jing et al. [77], but correctly by FibrilNet.

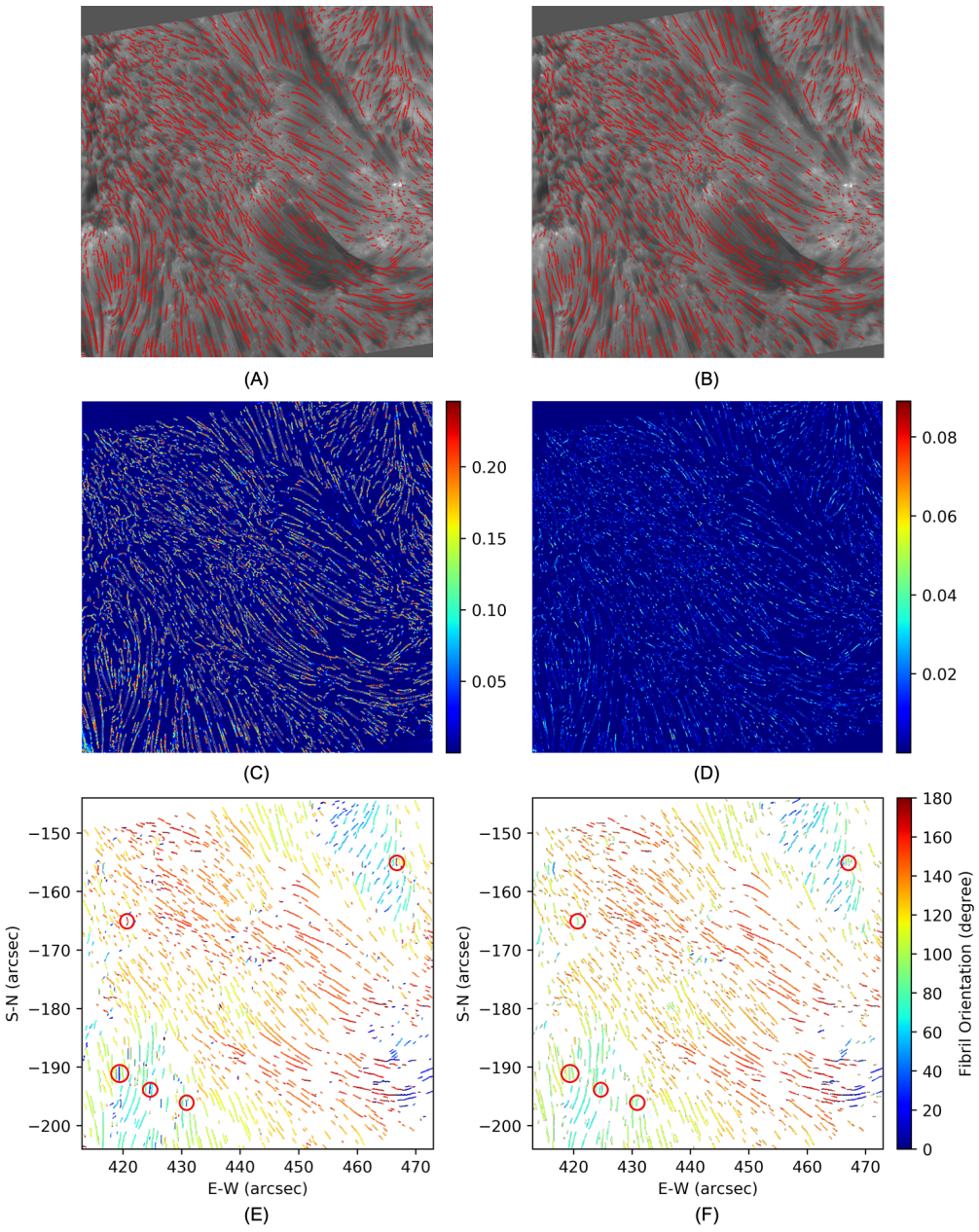

Figure 3.12 Fibril tracing results on the test image at −0.6 Å from the Hα line center 6563 Å with a 70″ circular FOV collected in AR 12665 on 2017 July 13 20:15:58 UT where training data were 241 Hα line center images taken from the same AR between 20:16:32 UT and 22:41:30 UT on the same day. (A) Fibrils on the test Hα image detected by the tool in Jing et al. [77]. (B) Fibrils on the test Hα image predicted by FibrilNet. (C) The aleatoric uncertainty (data uncertainty) map produced by FibrilNet. (D) The epistemic uncertainty (model uncertainty) map produced by FibrilNet. (E) Fibril orientation angles calculated by the tool in Jing et al. [77]. (F) Fibril orientation angles determined by FibrilNet. Orientation angles of a number of fibrils, some of which are highlighted by small red circles here, are calculated wrongly by the tool in Jing et al. [77], but correctly by FibrilNet.

CHAPTER 4

STOKES INVERSION

4.1 Background and Related Work

Stokes inversion is a method used to infer the physical conditions of the solar atmosphere based on the interpretation of observed Stokes profiles [9, 21]. Estimates of the physical magnitudes governing the state of the solar atmosphere can be obtained through the various inversion algorithms that try to achieve the best fit to the observed Stokes profiles [145]. One of the most widely used inversion algorithms is the MilneEddington (ME) [11, 91, 92] algorithm. ME assumes that all the physical quantities relevant to spectral line formation are constant with depth, and provides a solution to the radiative transfer equation. Extensions of the ME algorithm include Helix+ [87], MILOS [115], MERLIN [100] and VFISV [23, 22]. On the other hand, with the availability of high performance computing, algorithms based on the local thermodynamic equilibrium (LTE) and non-LTE conditions, which can solve the full radiative transfer equation, also become popular. Examples of such algorithms include SIR [129], SPINOR [42], and NICOLE [141].

Since Stokes inversion is a time-consuming task, there have been efforts of employing machine learning (ML) to accomplish the task. After an ML model is trained, one can use the trained model to perform Stokes inversion through making predictions, which reduces the inversion time significantly [105]. Examples of these ML models include multiple support vector regression (MSVR) [127] and multilayer perceptrons (MLP) [26]. In recent years, more powerful deep learning approaches have been developed to solve the Stokes inversion problem. For example, Asensio Ramos & D az Baso[8] presented two convolutional neural networks (CNN) [94] to perform Stokes inversion on the synthetic two-dimensional (2D) maps of Stokes profiles where

the authors exploited the 2D spatial coherence of the field of view. Liu et al. [105] designed a pixel-level CNN, referred to as PCNN, to perform Stokes inversion on the Near InfraRed Imaging Spectropolarimeter (NIRIS) data [24] from the 1.6 m Goode Solar Telescope (GST) at the Big Bear Solar Observatory (BBSO) [25, 56, 55, 149]. The authors used normalized Stokes Q, U, and V profiles as training data where the labels of the training data were obtained from the output of the ME method.

Typically the ME method can infer five atmospheric parameters, including three components of the magnetic field vector and two kinematic parameters, i.e., the line-of-sight (LOS) velocity and Doppler width, which are mostly used by researchers to understand the evolution of physical properties of the solar atmosphere [80]. In this paper we propose a new deep learning method, named Stacked Deep Neural Networks (SDNN), to infer the LOS velocities and Doppler widths from Stokes profiles of GST/NIRIS. Although both our SDNN and the previous PCNN developed in Liu et al. [105] perform Stokes inversion on the data from GST/NIRIS, the two tools differ in three ways. First, PCNN focuses on predicting the total magnetic field strength, inclination and azimuth angles. In contrast, our SDNN is designed to infer the LOS velocity and Doppler width in addition to the vector magnetic field. Second, when applying PCNN to infer the LOS velocity, it fails to infer the granular patterns in the LOS velocity image of sunspot data. By contrast, SDNN can infer all the convective granulation structures in the LOS velocity image [116]. Third, the architecture of SDNN, which is better suited for Stokes inversion, is totally different from that of PCNN. As demonstrated in our experimental study, SDNN outperforms PCNN on sevral different datasets.

4.2 Observations and Data Preparation

The GST/NIRIS is a Fabry-Pérot based imaging system, which provides high-resolution Stokes parameters I, Q, U and V of the Fe 1 1560 nm line within a ± 0.25

nm spectral window [24]. A typical field-of-view is about 85″ with an image scale of 0.″083 pixel^{-1} [151, 167, 101, 166]. The data used in this study were obtained from three active regions (ARs), namely AR 12371, AR 12665, and AR 12673. Our first dataset, denoted D1, is a 990×950 image from AR 12371 collected at 17:33:00 UT on 2015 June 22. The second dataset, denoted D2, is a 720×720 image from the same AR 12371 but collected approximately three days later at 16:55:13 UT on 2015 June 25. The third dataset, denoted D3, is a 720×720 image from AR 12665 collected at 16:20:12 UT on 2017 July 13. The fourth dataset, denoted D4, is a 720×720 image from AR 12673 collected at 19:17:53 UT on 2017 September 6.

Since D1 has the most pixels with the largest range of Stokes component values among the four datasets, we use D1 as the training set. This dataset has 940,500 pixels. Each pixel is treated as a training data sample containing Stokes component values and labeled by the LOS velocity and Doppler width calculated by the ME method. Thus, as in Liu et al. [105], we use the output of the ME method as the training labels. Notice that the number of spectral points scanned by GST/NIRIS is usually 60, but varied in some particular days. For instance, there were 56 spectral points for AR 12673 scanned on 2017 September 6. For consistency reasons, zeros are added so that the number of spectral points is unified and fixed at 60. There are four Stokes components I, Q, U, V at each spectral position, so the length of each training data sample, corresponding to each pixel, is 60×4. There are two labels, namely the LOS velocity and Doppler width, associated with the pixel. Therefore, the total length of the training data sample fed to our SDNN model is $60 \times 4 + 2 = 242$.

The remaining three datasets, D2, D3, D4, are used as test sets. The training set and test tests are disjoint. Hence, our SDNN model is tested on data that the model has never seen during training. Each test data sample in the test sets (D2, D3, D4) has the same format as the training data samples in D1 except that the test data

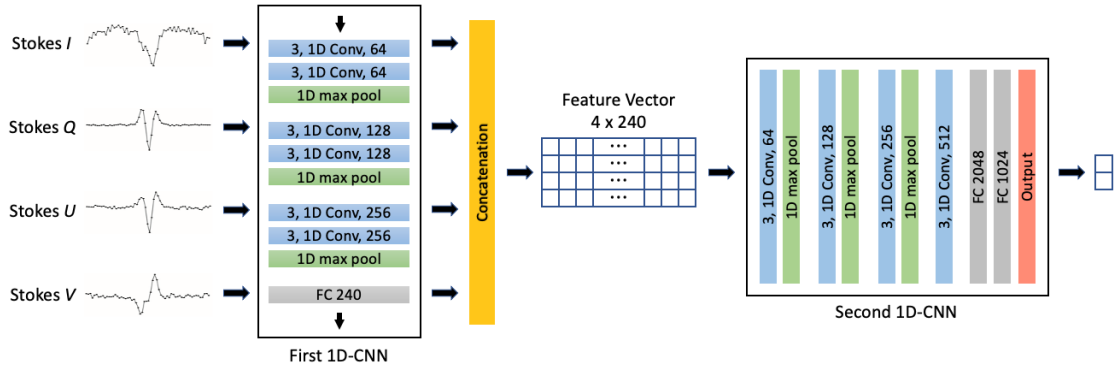

Figure 4.1 Architecture of our SDNN model. This model contains two 1D convolution neural networks (1D-CNNs) where the first 1D-CNN is stacked on top of the second 1D-CNN; hence the name SDNN (Stacked Deep Neural Networks) is used. The input of the SDNN is a sequence of four Stokes I, Q, U and V components where each component has 60 wavelength sampling points and the four Stokes components correspond to a pixel. The first 1D-CNN takes as input each Stokes component, and encodes and produces as output a 240-dimensional feature vector. There are four Stokes components so the first 1D-CNN outputs four 240-dimensional feature vectors, which are concatenated to form a 4×240 feature vector. The second 1D-CNN takes the 4×240 feature vector and produces two numbers representing the estimated Doppler shift and Doppler width of the input pixel.

sample does not have the two labels. Therefore, the length of the test data sample is $60 \times 4 = 240$. Because the values of the Stokes components vary, we normalize the Stokes Q, U, V profiles as done in Liu et al. [105] to facilitate machine learning. In addition, we normalize the Stokes I profile by dividing the measurements by 10000 since the mean of the measurements is around 10000. The Doppler shift values, which are obtained directly from the ME method and can be converted to the LOS velocities as we explain later, range from -0.5 to 0.5. We normalize the Doppler shift values by adding 0.5 to all the values, so that they range from 0 to 1. The Doppler width values calculated by the ME method already range from 0 to 1, and hence no normalization is done and the Doppler width values are used directly for model training.

4.3 Methodology

Figure 4.1 illustrates the architecture of our deep learning model, SDNN, used to infer LOS velocities and Doppler widths from Stokes profiles of GST/NIRIS. This model contains two 1D convolution neural networks (1D-CNN) [82]. The first 1D-CNN contains 3 convolution blocks followed by a fully connected layer with 240 neurons. Each convolution block contains two 1D convolution layers with kernels of size 3, activated by a ReLU (rectified linear unit) function, followed by a 1D max pooling layer with a kernel of size 2. Each 1D convolution layer in the first (second and third, respectively) convolution block has 64 (128 and 256, respectively) kernels.

The second 1D-CNN contains four convolution blocks followed by two fully connected layers activated by ReLU and having 2048 and 1024 neurons, respectively. Each of the first three convolution blocks contains a 1D convolution layer with a kernel of size 3 activated by ReLU, followed by a 1D max pooling layer with a kernel of size 2. The fourth convolution block only contains a 1D convolution layer with a kernel of size 3 activated by ReLU. The 1D convolution layer in the first (second, third and fourth, respectively) convolution block has 64 (128, 256 and 512, respectively) kernels. The output layer has 2 neurons, activated by the linear function, $f(x) = ax$ [57], which is suitable for predicting continuous numerical values and performs better than the Tanh function used in Liu et al. [105].

The input of our SDNN model is a sequence of four Stokes I, Q, U and V components where each component has 60 wavelength sampling points and the four Stokes components correspond to a pixel. The first 1D-CNN takes as input each Stokes component, and encodes and produces as output a 240-dimensional feature vector. There are four Stokes components so the 1D-CNN outputs four 240-dimensional feature vectors, which are concatenated to form a 4×240 feature vector. The second 1D-CNN takes as input the 4×240 feature vector and produces

as output two numbers representing the estimated Doppler shift and Doppler width of the input pixel.

The regression loss function used by our SDNN model is the L1 loss function [57], defined as follows:

$$\text{L1 loss} = \frac{1}{N}\sum_{i=1}^{N}(\left|y_i^{ds} - \hat{y}_i^{ds}\right| + \left|y_i^{dw} - \hat{y}_i^{dw}\right|) \tag{4.1}$$

where N is the total number of pixels in a test set, y_i^{ds} (y_i^{dw}, respectively) is the Doppler shift (Doppler width, respectively) of the ith pixel calculated by the ME method, $\hat{y}_i^{ds}$ ($\hat{y}_i^{dw}$, respectively) is the Doppler shift (Doppler width, respectively) of the ith pixel predicted by our SDNN model. We use the L1 loss function here because it is more robust to outliers, hence making the model more tolerant to noise in the training data [161].

We train the SDNN model using the Adam optimizer. The batch size is set to 1024, and the number of epochs is set to 40. During testing, the trained model takes as input the GST/NIRIS Stokes I, Q, U, and V profiles of each pixel in a test set, and predicts as output the Doppler shift and Doppler width of the pixel. The predicted values are denormalized so that they fall in the original (correct) range. We then convert the Doppler shift, denoted Δ_λ, to the LOS velocity, denoted v_{LOS}, as follows:

$$v_{LOS} = \frac{C \times \Delta_\lambda}{\lambda} \tag{4.2}$$

where λ is the GST/NIRIS magnetogram wavelength, which is set to 1.56 um, and C is the speed of light. The unit of v_{LOS} is km/s.

4.4 Results

4.4.1 Performance Metrics

For each test data sample, which corresponds to each pixel in a test set, we can use the proposed SDNN model to predict or infer its LOS velocity and Doppler

width. In addition, we can also use the ME method [11, 91, 92] to calculate its LOS velocity and Doppler width. We adopted four metrics to evaluate the performance of our SDNN model and compare it with related machine learning algorithms. We considered two quantities: LOS velocity and Doppler width. For each quantity, we compared its ME-calculated values with our SDNN-inferred values and computed the four performance metrics.

The first performance metric is the mean absolute error (MAE) [131]. MAE quantitatively assesses the difference between the ME-calculated and SDNN-inferred values for the test set (test image). The smaller the MAE is, the better performance an algorithm has. The second performance metric is the percent agreement (PA) [110]. Let y_i and $\hat{y}_i$ denote the ME-calculated and SDNN-inferred value, respectively, for the ith pixel in the test image. The ith pixel is an agreement pixel if $|y_i - \hat{y}_i|$ is smaller than a threshold. (The default threshold is set to 1 km/s for the LOS velocity and 0.1 Å for the Doppler width.) PA equals the number of agreement pixels divided by the total number of pixels in the test image multiplied by 100%. Thus PA quantitatively assesses the similarity between the ME-calculated and SDNN-inferred values for the test image. The closer to 100% the PA is, the better performance an algorithm has. The third performance metric is the R-squared value [112], which ranges from $-\infty$ to 1. R-squared measures the strength of the relationship between the ME-calculated and SDNN-inferred values for the test image. The larger (i.e., closer to 1) the R-squared value is, the stronger the relationship between the ME-calculated and SDNN-inferred values we have. The fourth performance metric is the Pearson product-moment correlation coefficient (PPMCC) [83], which ranges from -1 to 1. PPMCC quantifies how well the SDNN-inferred values agree with the ME-calculated values for the test image. The larger (i.e., closer to 1) the PPMCC is, the better performance an algorithm has.

4.4.2 Impact of Training Data on the Performance of the SDNN Method

Table 4.1 presents experimental results of using D1 as the training set to train SDNN and using D2, D3 and D4 as test sets to test SDNN as described in Section 4.2. SDNN works well on D2 and D3. However, the performance of SDNN degrades on D4 which contains pixels from AR 12673. We note that AR 12673 is the most flare-productive active region in solar cycle 24, which shows strong magnetic fields in the light bridge and apparent photospheric twist [152]. The training set D1 only contains pixels from AR 12371, and hence the trained SDNN model does not have sufficient knowledge about AR 12673. To assess and quantify the impact of training data on the performance of SDNN, we additionally selected a 720×720 image with 518,400 pixels (data samples) from AR 12673 collected at 16:18:41 UT on 2017 September 6. We referred to this additional dataset as D5. Thus, the image of D5 was taken approximately 3 hours before the image of D4. We then combined the 940,500 pixels in D1 and the 518,400 pixels in D5 to get a new training set, denoted D1 $\cup$ D5. D1 $\cup$ D5 contains 1,458,900 training data samples (pixels) in total. Results of using D1 $\cup$ D5 as the training set to train SDNN and using D2, D3 and D4 as test sets to test SDNN are also presented in Table 4.1. Notice that, due to the time difference, D1 $\cup$ D5 and D4 are disjoint even though the data samples in D5 and D4 are from the same AR 12673.

We see from Table 4.1 that the SDNN model trained by D1 $\cup$ D5 outperforms the SDNN model trained by D1 when the two models are tested on D4. This happens because the SDNN model trained by D1 $\cup$ D5 acquires more knowledge concerning D4 than the SDNN model trained by D1 due to the fact that D4 and D5 are from the same AR 12673 as indicated above. On the other hand, the two models have similar performance when tested on D2 and D3. Notice that D1, D2 and D3 are from normal active regions while D4 contains special pixels with extremely large magnetic field strengths (saturated at the value of 5000 G or even larger). This kind of special

Table 4.1 Performance Metric Values of SDNN Based on Two Training Sets and Test Images from Three ARs

		LOS Velocity		Doppler Width	
Test Image	Metric	D1	D1 ∪ D5	D1	D1 ∪ D5
D2	MAE	0.251	0.260	0.048	0.047
2015 June 25	PA	97.8%	97.5%	89.1%	90.3%
16:55:13 UT	R-squared	0.796	0.786	0.317	0.458
(AR 12371)	PPMCC	0.915	0.912	0.780	0.787
D3	MAE	0.289	0.302	0.039	0.040
2017 July 13	PA	97.6%	97.0%	93.2%	93.2%
16:20:12 UT	R-squared	0.768	0.728	0.224	0.278
(AR 12665)	PPMCC	0.912	0.896	0.716	0.723
D4	MAE	0.952	0.316	0.064	0.034
2017 September 6	PA	51.0%	94.5%	82.3%	95.3%
19:17:53 UT	R-squared	-0.402	0.749	-1.334	0.208
(AR 12673)	PPMCC	0.804	0.883	0.516	0.711

pixels (data samples) do not occur in D1, D2, D3. Thus, the model trained by D1 lacks knowledge of extremely large magnetic field strengths, and hence does not work well on D4. On the other hand, like D4, D5 also contains pixels with extremely large magnetic field strengths. As a consequence, the model trained by D1 ∪ D5 performs well on D4. These results indicate that when dealing with normal ARs, the training data samples in D1 are sufficient to produce good results. On the other hand, when dealing with special ARs such as AR 12673, the training set must be expanded to include data samples from the special ARs so that our SDNN model can acquire sufficient knowledge about the special ARs to produce good results. In view of the experimental results, we used D1 ∪ D5 as the training set in subsequent experiments.

4.4.3 Comparison between the SDNN and ME Methods

Here we compare the LOS velocity maps and Doppler width maps produced by the SDNN and ME methods. Figure 4.2 (Figure 4.3, Figure 4.4, respectively) presents the LOS velocity maps (top panels) and Doppler width maps (bottom panels) produced by the two methods based on the test set/image D2 (D3, D4, respectively) with data samples/pixels from AR 12371 (AR 12665, AR 12673, respectively) collected on 2015 June 25 16:55:13 UT (2017 July 13 16:20:12 UT, 2017 September 6 19:17:53 UT, respectively). The first columns in the figures show scatter plots, the second columns show maps produced by the ME method and the third columns show maps produced by the SDNN method. Pixels on the diagonal line in a scatter plot have identical ME-calculated and SDNN-inferred values. We see from Figures 4.2, 4.3 and 4.4 that the maps produced by the two methods are highly correlated. The high correlation can be seen particularly from the scatter plots of the LOS velocity maps and the corresponding PPMCC values of $\sim$0.9 as shown in Table 4.1. Since the training labels for the SDNN model are produced by the ME method, these results demonstrate the good learning capability of SDNN. Notice also that the maps produced by SDNN are smoother and cleaner than those produced by the ME method. We see many salt-and-pepper noise pixels in the maps of the ME method. The many noise pixels from the ME method are also reflected in the scatter plots in Figures 4.2, 4.3 and 4.4. For example, refer to the scatter plot of the LOS velocity maps in Figure 4.3 where there is a vertical line on which pixels have a LOS velocity of zero. Many of the pixels on the vertical line are noisy ones in the LOS velocity map produced by the ME method.

It is worth noting that the LOS velocity maps in Figure 4.3 contain granular patterns and a portion of a sunspot penumbra. For example, there are granular patterns located in the region whose E-W coordinates are between $-460''$ and $-450''$ and N-S coordinates are between $190''$ and $200''$. A partial sunspot penumbra

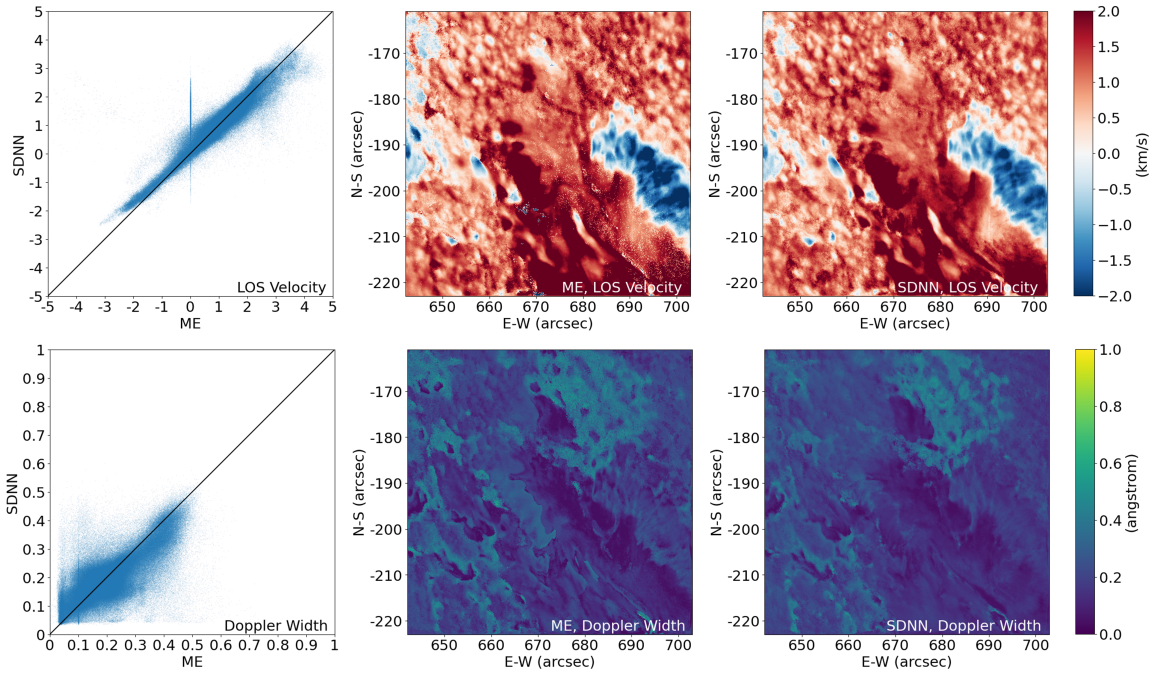

Figure 4.2 Comparison between the ME and SDNN methods for producing the LOS velocity maps (top panels) and Doppler width maps (bottom panels) based on the test image/dataset D2 from AR 12371 collected on 2015 June 25 16:55:13 UT, where training data were taken from D1 ∪ D5. The first column shows scatter plots where the X-axis and Y-axis represent the values obtained by the ME and SDNN methods, respectively. Pixels on the diagonal line in a scatter plot have identical ME-calculated and SDNN-inferred values. The second and third columns show the maps produced by the ME and SDNN methods, respectively.

is located in the region whose E-W coordinates are between $-450''$ and $-440''$ and N-S coordinates are between $150''$ and $160''$. Figure 4.5 presents an enlarged view of these two regions. Both ME and SDNN methods produce the granular patterns with a PPMCC of 0.982 and the partial sunspot penumbra with a PPMCC of 0.893. The similarity between the local maps produced by the two methods is also reflected in the scatter plots in Figure 4.5. In addition, we see from Figure 4.5 that the local maps produced by the SDNN method are smoother and cleaner than those produced by the ME method. There are many salt-and-pepper noise pixels in the maps of the ME method, particularly in the region of the partial sunspot penumbra.

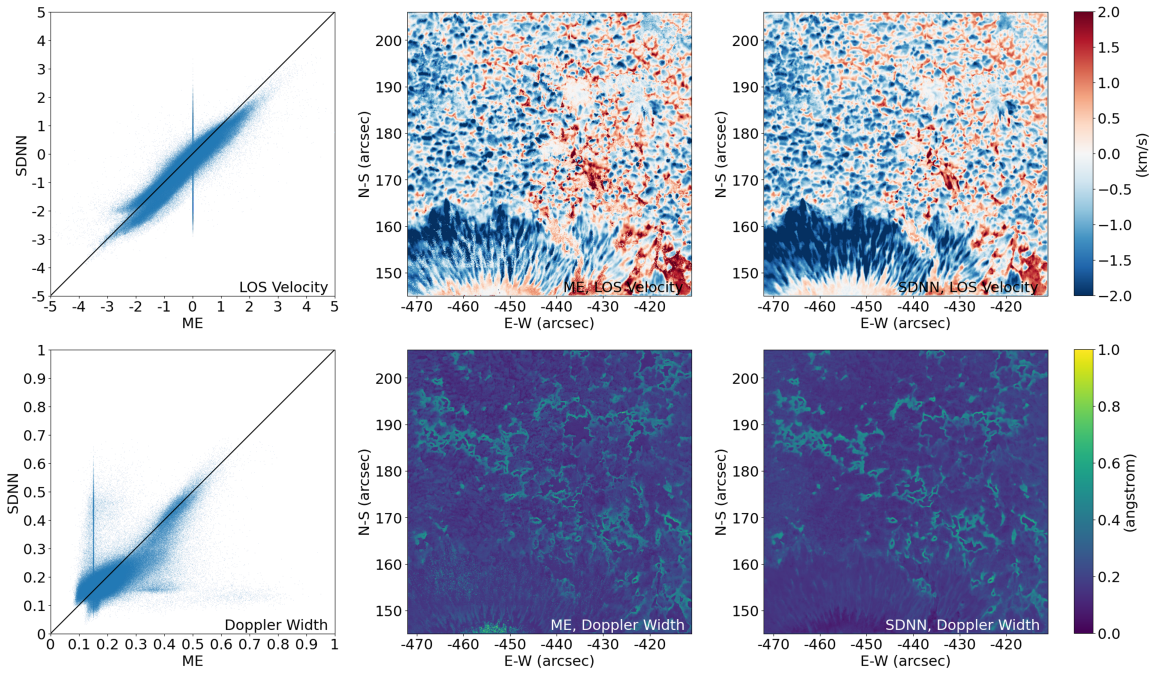

Figure 4.3 Comparison between the ME and SDNN methods for producing the LOS velocity maps (top panels) and Doppler width maps (bottom panels) based on the test image/dataset D3 from AR 12665 collected on 2017 July 13 16:20:12 UT, where training data were taken from D1 ∪ D5. The first column shows scatter plots where the X-axis and Y-axis represent the values obtained by the ME and SDNN methods, respectively. Pixels on the diagonal line in a scatter plot have identical ME-calculated and SDNN-inferred values. The second and third columns show the maps produced by the ME and SDNN methods, respectively.

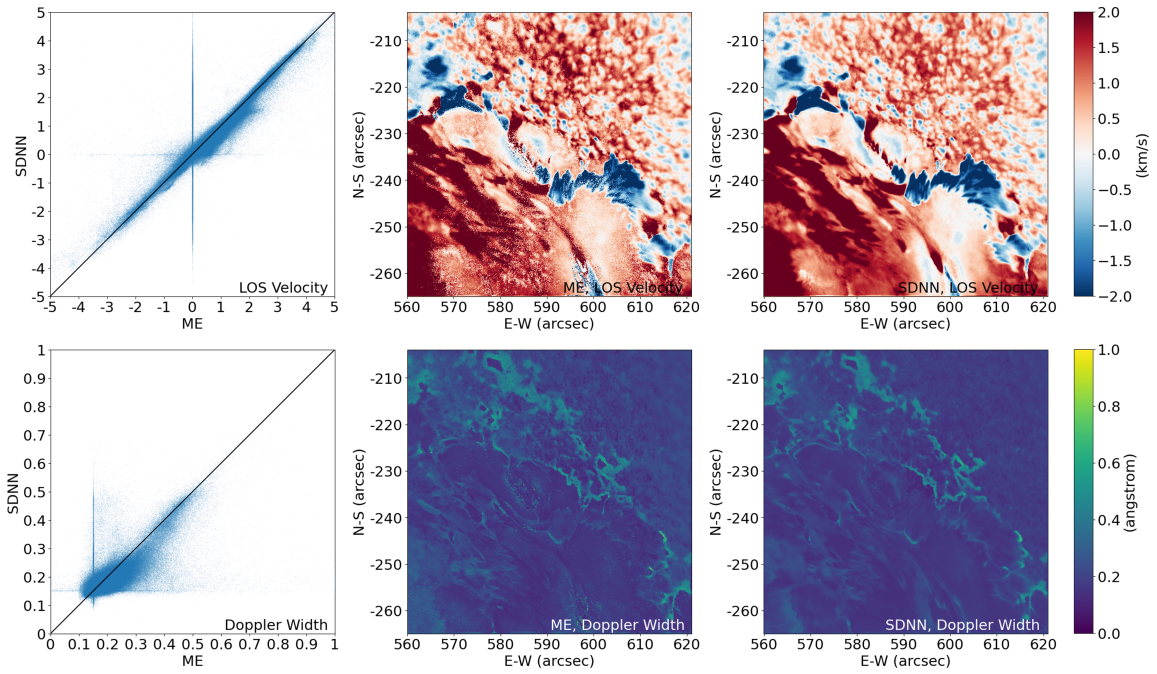

Figure 4.4 Comparison between the ME and SDNN methods for producing the LOS velocity maps (top panels) and Doppler width maps (bottom panels) based on the test image/dataset D4 from AR 12673 collected on 2017 September 6 19:17:53 UT, where training data were taken from D1 ∪ D5. The first column shows scatter plots where the X-axis and Y-axis represent the values obtained by the ME and SDNN methods, respectively. Pixels on the diagonal line in a scatter plot have identical ME-calculated and SDNN-inferred values. The second and third columns show the maps produced by the ME and SDNN methods, respectively.

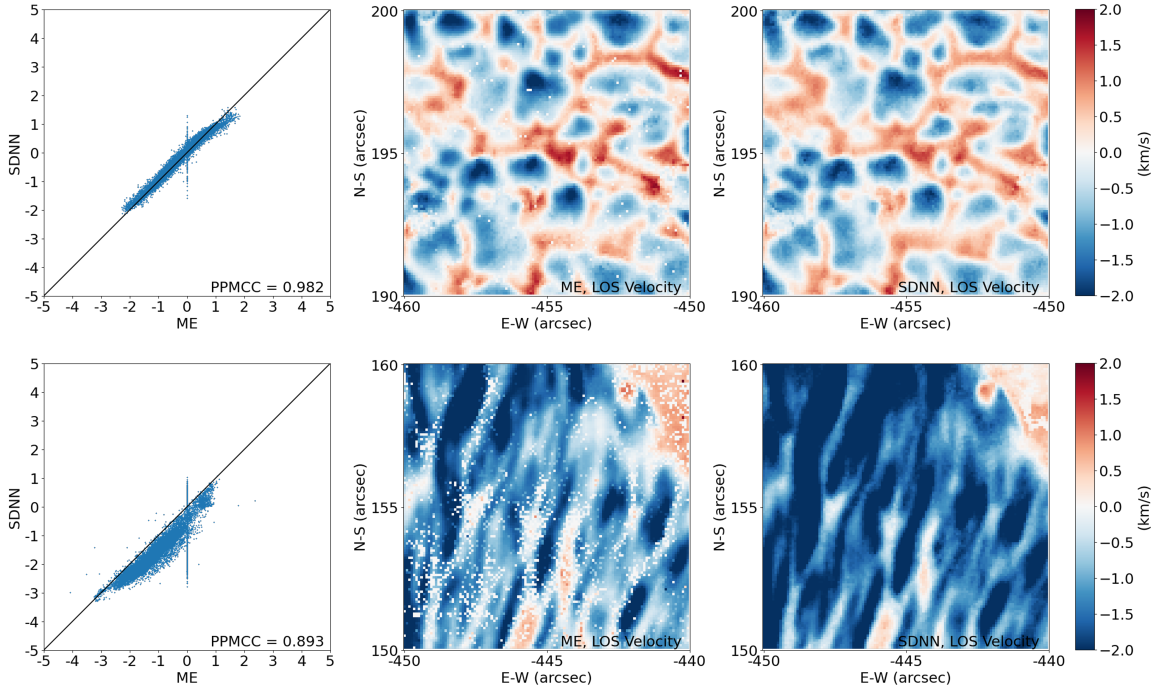

Figure 4.5 Comparison between the ME and SDNN methods for producing the local LOS velocity maps containing granular patterns (top panels) and a portion of a sunspot penumbra (bottom panels) based on the test image/dataset D3 from AR 12665 collected on 2017 July 13 16:20:12 UT, where training data were taken from D1 ∪ D5. The first column shows scatter plots where the X-axis and Y-axis represent the values obtained by the ME and SDNN methods, respectively. Pixels on the diagonal line in a scatter plot have identical ME-calculated and SDNN-inferred values. The second and third columns show the local LOS velocity maps produced by the ME and SDNN methods, respectively.

4.4.4 Comparison with Related Machine Learning Methods

Here we compare SDNN with three related machine learning (ML) methods including multiple support vector regression (MSVR) [127], multilayer perceptrons (MLP) [26] and a pixel-wise convolutional neural network (PCNN) [105]. The MSVR method uses the radial basis function kernel where the regularization parameter is set to 1 and the epsilon is set to 0.2. The MLP model consists of an input layer, an output layer, and two hidden layers each of which has 1024 neurons. The PCNN model, which was originally designed for inferring vector magnetic fields, is modified to output the LOS velocity and Doppler width, though the same model architecture and hyperparameter setting are used here.

Figure 4.6 compares the MAE, PA, R-squared and PPMCC values of the four ML methods based on the test image/dataset D2 (D3, D4, respectively) from AR 12371 (AR 12665, AR 12673, respectively) collected on 2015 June 25 16:55:13 UT (2017 July 13 16:20:12 UT, 2017 September 6 19:17:53 UT, respectively) where training data were taken from D1 $\cup$ D5. It can be seen from Figure 4.6 that SDNN outperforms the other three ML methods in terms of all the four performance metrics. When compared to the most closely related PCNN method, SDNN achieves an MAE of 0.260 (0.302, 0.316, respectively) while PCNN achieves an MAE of 0.455 (0.537, 0.542, respectively), showing an improvement of 42.9% (43.8%, 41.7%, respectively) by SDNN, on D2 (D3, D4, respectively) in producing the LOS velocity maps for the three datsets. Furthermore, in producing these LOS velocity maps, SDNN achieves a PPMCC of 0.912 (0.896, 0.883, respectively) while PCNN achieves a PPMCC of 0.849 (0.676, 0.832, respectively) on D2 (D3, D4, respectively); SDNN beats PCNN by 7.4%, 32.5% and 6%, respectively, on the three datasets. These results indicate that our SDNN model is more robust and has a better generalization and inference capability than the closely related PCNN model. It is worth pointing out that the granular patterns in the LOS velocity map of the SDNN method as shown in Figure

4.5, which exhibit important kinematic information on the photospheric surface [72], are missing from the LOS velocity maps produced by the other three ML methods, another sign indicating the superiority of the proposed SDNN model.

4.4.5 Comparison between the Inversion Results of SDO/HMI and GST/NIRIS

So far we have focused on Stokes inversion for GST/NIRIS. There are inversion results from the Helioseismic and Magnetic Imager (HMI) [134] on board the Solar Dynamics Observatory (SDO) [123]. Here we compare the inversion results of the space-borne observatory (SDO/HMI) and ground-based observatory (BBSO/GST). We selected an HMI Dopplergram from AR 12673 collected on 2017 September 6 19:00:00 UT, and a temporarily closest test image/dataset, denoted D6, from the same AR 12673 collected by GST/NIRIS on 2017 September 6 19:01:48 UT. We aligned the HMI and GST/NIRIS images, and applied our SDNN model trained by the dataset D1 $\cup$ D5 and the ME method to D6, respectively.

Figure 4.7 presents the LOS velocity map from the HMI Dopplergram and the LOS velocity maps produced by the ME and SDNN methods (top panels), and shows the corresponding scatter plots (bottom panels). It can be seen from the top panels that the maps of GST/NIRIS obtained by ME and SDNN are much clearer than the map from HMI. This happens due to the higher resolution imaging data offered by GST/NIRIS. On the other hand, the maps from HMI and SDNN are smoother than the map of ME which contains salt-and-pepper noise pixels. Furthermore, we see from the left and middle scatter plots in Figure 4.7 that the map produced by SDNN is closer to the map from HMI with a PPMCC of 0.827 than the map produced by ME which has a PPMCC of 0.745. The black regression lines in the scatter plots have a slope of $\sim$1.4, possibly caused by the offset in calibration, which occurs due to the intrinsic difference between the two instruments HMI and GST/NIRIS.

4.5 Summary

We develop a deep learning model (SDNN) to infer LOS velocities and Doppler widths from Stokes profiles of GST/NIRIS. The labels for training SDNN are prepared by the widely used MilneEddington (ME) method. We compare the LOS velocity and Doppler width maps produced by SDNN with those from ME and related machine learning (ML) algorithms including multiple support vector regression (MSVR), multilayer perceptrons (MLP) and a pixel-wise convolutional neural network (PCNN). We also compare the inversion results of ME and SDNN based on GST/NIRIS with those from SDO/HMI in a dataset taken from flare-prolific AR 12673.

Our main results are summarized as follows.

1. For the test datasets from GST/NIRIS considered in the paper, SDNN produces smoother and cleaner LOS velocity and Doppler width maps than the ME method. The same conclusion is obtained when comparing the inversion results of GST/NIRIS and SDO/HMI. Furthermore, SDNN performs Stokes inversion through making predictions, and hence is faster than the computation-based ME method. It takes $\sim$75 s for SDNN to process a test image here, which is approximately five times faster than the ME method.

2. The SDNN-inferred LOS velocities are highly correlated to the ME-calculated ones with PPMCC being close to 0.9 on average. The LOS velocity and Doppler width maps produced by SDNN are closer to MEs maps than those from the related ML methods (MSVR, MLP, PCNN). Furthermore, both ME and SDNN are able to infer granular patterns in LOS velocity maps. These patterns exhibit important kinematic information on the photospheric surface, which are missing from the LOS velocity maps produced by the other three ML methods. These results demonstrate the better learning capability of SDNN than the other ML methods.

3. Training data has a significant impact on the performance of SDNN. When SDNN is trained by data from normal active regions (ARs), it performs well on normal ARs, but suffers on special ARs such as AR 12673. AR 12673 contains special pixels with extremely large magnetic field strengths (saturated at the value of 5000 G or even larger). To acquire knowledge concerning these extremely large magnetic field strengths, we have to include some of the special pixels/data samples in the training set. This finding is consistent with the guidelines suggested in the machine

learning (ML) literature [16] where an iterative training process with increasing training data is often used to improve the performance of ML models.

Based on these results, we conclude that SDNN is an effective and alternative method for inferring LOS velocities and Doppler widths from Stokes profiles of GST/NIRIS. It is hoped that SDNN will be a useful tool in producing high-quality velocity fields that are crucial for understanding and predicting solar flares.

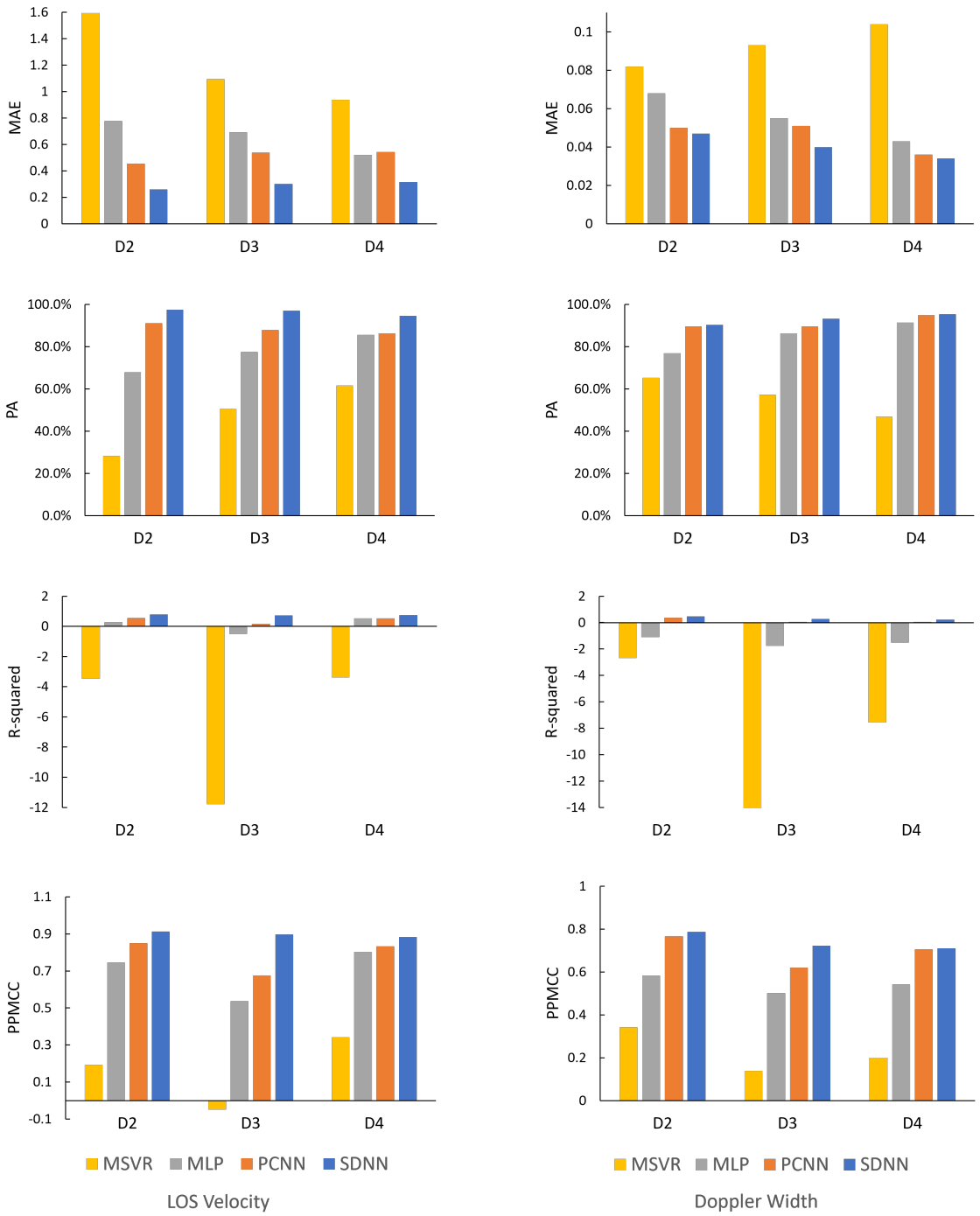

Figure 4.6 Performance metric values of MSVR, MLP, PCNN and SDNN based on the test image/dataset D2 (D3, D4, respectively) from AR 12371 (AR 12665, AR 12673, respectively) collected on 2015 June 25 16:55:13 UT (2017 July 13 16:20:12 UT, 2017 September 6 19:17:53 UT, respectively) where training data were taken from D1 ∪ D5. Left column: performance metric values, displayed by bar charts, that are obtained in predicting LOS velocities. Right column: performance metric values obtained in predicting Doppler widths.

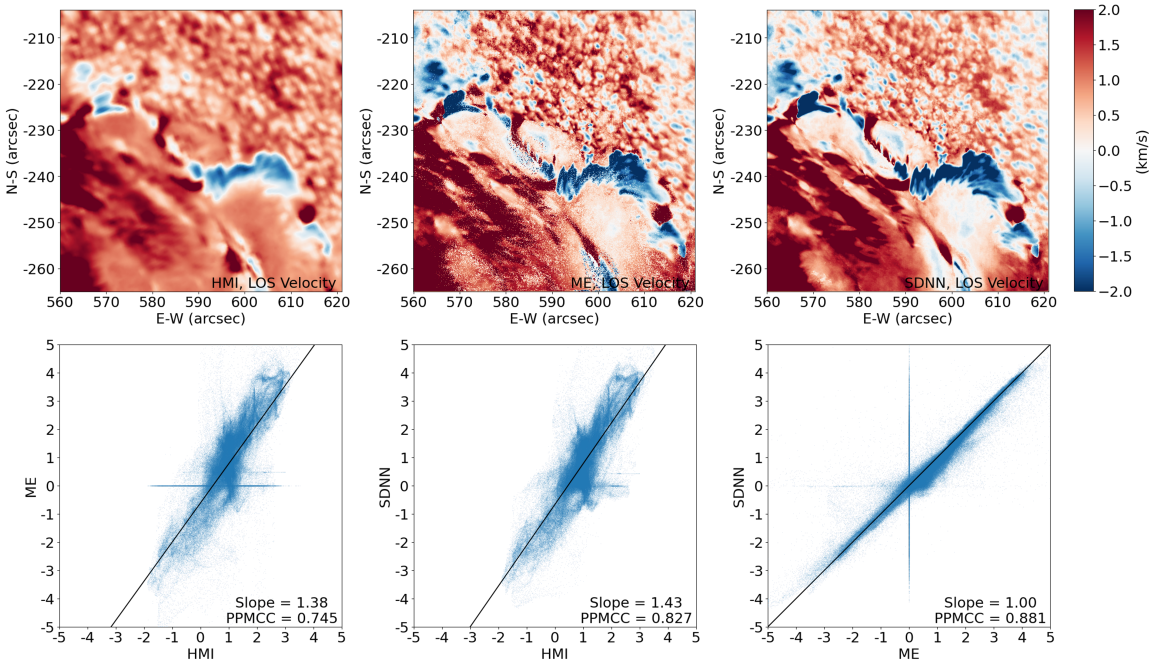

Figure 4.7 Comparison between the inversion results of SDO/HMI and GST/NIRIS in AR 12673. Top panels (from left to right): the LOS velocity map from the HMI Dopplergram collected on 2017 September 6 19:00:00 UT, the LOS velocity map produced by the ME method, and the LOS velocity map produced by the SDNN method with training data from D1 ∪ D5 where the ME and SDNN methods were applied to the test image/dataset D6 collected on 2017 September 6 19:01:48 UT. Bottom panels (from left to right): the scatter plot between ME and HMI, the scatter plot between SDNN and HMI, and the scatter plot between SDNN and ME.

CHAPTER 5

VECTOR MAGNETOGRAM GENERATION

5.1 Background and Related Work

The energy source to power solar eruptions is the magnetic energy stored in solar active regions (ARs). These eruptive events are the direct sources of disturbances in the near-Earth environment, the so-called space weather. Magnetic energy is divided into potential and non-potential components [37, 136]. The global coronal magnetic field and the extension to the heliosphere can be described by the potential field well [135]. From the solar corona down to the photosphere, the magnetic field becomes highly non-potential and is in association with the origin of solar activities. The quantitative measurement of nonpotentiality is usually referred to as free magnetic energy, which can be calculated using vector magnetograms by nonlinear force-free field (NLFFF) extrapolation methods [156, 157, 111, 20, 137, 159, 143]. With three-dimensional (3D) NLFFFs, magnetic topology and magnetic helicity injection can also be derived in addition to free magnetic energy. The studies of free magnetic energy and comparisons with energy release provide important information of an AR's capability in producing eruptive events including flares and coronal mass ejections (CMEs) [6], thereby leading to quantitative prediction of future events. The physical process of the flare or CME initiation can be explained by plasma instability such as torus and/or kink instabilities [146, 85, 84], or via magnetohydrodynamic (MHD) processes such as magnetic reconnection [4, 78]. Such processes can be revealed by numerical MHD simulations using, once again, vector magnetograms as boundary conditions. In addition, magnetic field parameters such as those of Space-weather HMI Active Region Patches (also known as SHARP parameters)[19] can be derived from the vector magnetograms taken by the Helioseismic and Magnetic Imager (HMI)

on board the NASA Solar Dynamics Observatory (SDO) [134] mission and are widely used in machine learning-based flare and CME forecasting [18, 28, 155, 1].

Since 2010, consistent time-sequence full disk photospheric vector magnetograms have been available through SDO/HMI. This data set covers a relatively weak solar cycle, namely cycle 24, which had fewer big eruptive events, and therefore the sampling is not sufficient to understand and predict solar eruptions, Prior to that, vector magnetograms were available sporadically for certain ARs from some observatories, while line-of-sight (LOS) magnetograms have been consistently available from another NASA source: the Michelson Doppler Imager (MDI) on board the Solar and Heliospheric Observatory (SOHO) [133].

With the importance of vector magnetograms and their unavailability prior to 2010 as described above, we propose a new deep learning approach, specifically a generative adversarial network (GAN) model, to generate consistent time-sequence photospheric vector magnetograms of all ARs from 1996 to 2010, which covered a more active solar cycle 23. Our GAN model is trained by LOS magnetograms, B_x, B_y from SDO/HMI combined with Hα observations from Big Bear Solar Observatory (BBSO) [31]. The validation of our approach uses the overlapping data from 2010-05-01 to 2011-04-11, when MDI and HMI obtained data simultaneously. There is a good physical reason to use Hα as the additional constraint for this research: in the solar atmosphere, magnetic fields and flows are at frozen-in condition, and therefore the Hα fibril structure can provide the direction of magnetic fields in X and Y dimensions [74].

Deep learning is a branch of machine learning where neural networks are designed to learn from large amounts of data [94]. It has been used extensively in computer vision, natural language processing, and lately in biology [35, 144], medicine [160, 48], heliophysics [47, 103, 104, 75, 74], and astronomy [81, 106, 15]. Deep learning employs various networks such as deep neural networks, deep belief networks,

convolutional neural networks (CNNs) and recurrent neural networks, among which GANs have drawn significant interest in recent years [58].fu2019dual A GAN model has two components: the generator that generates fake samples and the discriminator that tries to classify examples as either real or fake ones. The two components are trained together in an adversarial way. The contest operates in terms of data distributions. The generator learns to map from a latent space to a data distribution while the discriminator distinguishes the fake samples produced by the generator from the true data distribution. The generators objective is to fool the discriminator by producing fake samples that the discriminator thinks are part of the true data distribution. When this objective is accomplished, the training of the GAN model is completed. GANs have been used in video prediction [68], image enhancement [30], image-to-image translation [71], and image generation (synthesis) [81, 106].

To date, several deep learning approaches have been developed for estimating magnetic fields. Kim et al. generated farside solar magnetograms from STEREO/Extreme UltraViolet Imager (EUVI) 304-Å images using a deep learning model based on conditional GANs (cGANs) [81]. The authors trained their model using pairs of SDO/Atmospheric Imaging Assembly (AIA) 304-Å images and SDO/HMI magnetograms. They reported some preliminary results obtained from the cGAN model [81]. Liu et al. performed a detailed analysis of the cGAN model and concluded that more research needs to be performed to obtain scientifically reliable magnetograms [106]. More recently, Bai et al. presented a CNN to estimate the unsigned radial component of the magnetic filed and transverse field from photospheric continuum images [13].

In contrast to the previous work, our new GAN-based method, named MagNet, aims to generate photospheric vector magnetograms of solar active regions. We design a novel architecture and loss function tailored for vector magnetogram generation to take into account the magnetic field strength. Our trained MagNet takes as input

LOS magnetograms from SOHO/MDI as well as Hα images from BBSO, and produces as output synthetic magnetic field components B'_x and B'_y. The generated B'_x and B'_y components along with the line-of-sight (LOS) components of the magnetic field, which can be treated as B_z components, create synthetic vector magnetograms from 1996 to 2010. These generated vector magnetograms will be useful for studying the triggering mechanisms of solar eruptions and for forecasting eruptive events.

5.2 Observations and Data Preparation

MDI, which is part of the SOHO satellite, acquires a LOS magnetogram every 96 minutes during the period from 1995 to 2011, which covers more active solar cycle 23 with many large flares [133]. The spatial resolution is $4''$ and the full disk images are collected on a 1024×1024 detector. As a successor of MDI, the HMI instrument, which is part of the SDO mission, provides continuous coverage of full disk Doppler velocities, LOS magnetograms, and continuum proxy images [134]. HMI has been operational since May 1, 2010, covering solar cycle 24. HMI observes the full solar disk at 6173 Å every 12 minutes for a better signal-to-noise ratio. The spatial resolution is $1''$ and the full disk images are collected on a 4096×4096 detector. Photospheric vector magnetograms have been available since the lunch of SDO/HMI. BBSO, which is a ground-based observatory, has provided Hα observations (images) since 1970s [31]. BBSOs full disk Hα observations are taken every 1 minute, up to 9 hours for one observing day at the wavelength of 6563 Å. The spatial resolution is $2''$ and the full disk images are collected on a 2048×2048 detector. Unlike satellite-based instruments such as MDI and HMI, the ground-based full disk telescope at BBSO sometimes has seeing limitations due to unstable conditions of Earth's atmosphere and weather.

In our datasets, we excluded low-quality Hα images with an incomplete field of view (FOV) and cloud shades as well as other out-of-focus images. We selected

Hα images in the period from 2010-05-01 to 2017-12-31. We then collected their temporarily closest full disk MDI LOS magnetograms, HMI LOS magnetograms and HMI vector magnetograms, respectively. Specifically, for each selected Hα image, we collected all magnetograms available within 6 minutes of the time stamp of the Hα image. If there were more than one image available, we selected the one with the closest time stamp.

Our training set, contained selected 2874 Hα images from 2014-01-01 to 2017-08-04 and their temporarily closest HMI LOS magnetograms and HMI vector magnetograms. Our first test set, contained 226 HMI LOS magnetograms from 2017-08-05 to 2017-12-31 and their temporarily closest Hα images. Our second test set, contained 115 MDI LOS magnetograms from 2010-05-01 to 2011-04-11 and their temporarily closest Hα images. During this period, both MDI/HMI LOS magnetograms and HMI vector magnetograms were available. The true vector components B_x, B_y used to train our MagNet model and validate our predictions were obtained from the HMI vector magnetograms.

Because we are mainly interested in solar flares in solar active regions (ARs), we cut AR patches of 256×256 pixels that may produce flares from the full disk magnetograms collected above. We aligned the AR patches with the corresponding regions in their temporarily closest Hα images. As a result of the alignment process, we obtained 8442 AR pairs in the training set, denoted Train_HMI, 327 AR pairs in the first test set, denoted Test_HMI, and 159 AR pairs in the second test set, denoted Test_MDI. Furthermore, we removed some low-quality image pairs and obtained 250 AR pairs in Test_HMI. For Test_MDI, since the MDI and HMI images had a big resolution gap, in addition to removal of image pairs with low quality, we calculated the Pearson correlation coefficient (CC) between the HMI LOS magnetogram and MDI LOS magnetogram for each AR pair. We filtered out those AR pairs with CC values less than 0.8, obtaining 24 AR pairs in Test_MDI.

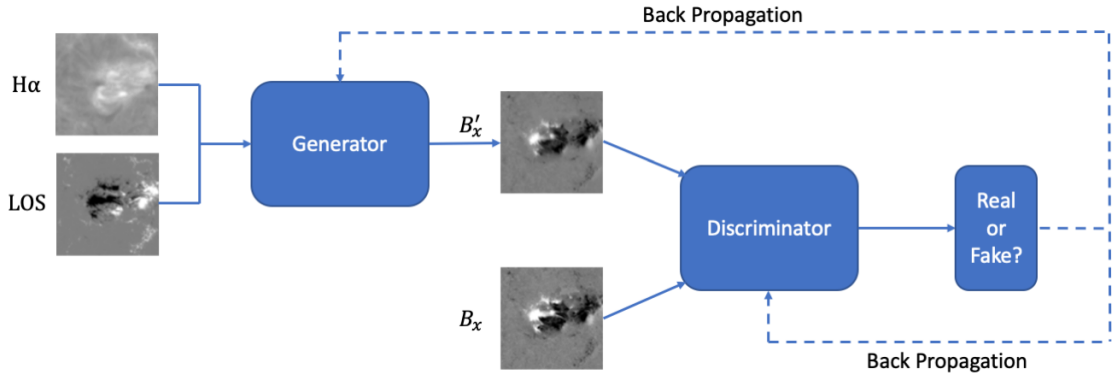

Figure 5.1 Overview of the training process of MagNet. LOS represents an active region of an HMI LOS magnetogram, Hα represents the corresponding region of the Hα observation (image) temporarily closest to the LOS, B_x is the ground truth and B'_x is the generated fake sample. Dashed lines represent back propagation in the neural network.

5.3 Methodology

5.3.1 The MagNet Model

Generating synthetic B'_x and B'_y amounts to solving a regression problem because the output of the generating procedure consists of real numbers, i.e., magnetic field strengths. We employ a novel generator architecture containing an advanced CNN with self-attention [168] to solve this regression problem. MagNet generates B'_x and B'_y separately. During training, MagNet takes as input pairs of aligned BBSO Hα images and HMI LOS magnetograms and uses corresponding HMI vector components B_x (B_y, respectively) as labels (ground truth). Fig. 5.1 shows the training process of MagNet for generating B'_x; the training process for generating B'_y is similar and omitted. During testing, MagNet takes a pair of aligned BBSO Hα image and MDI LOS magnetogram (or HMI LOS magnetogram) and generates a predicted B'_x (B'_y, respectively).

5.3.2 Model Training and Testing

Figure 5.1 presents an overview of the MagNet training process for generating/predicting B'_x where the discriminator, denoted D_x, and generator, denoted G_x, compete and learn in an adversarial way for optimizing the following loss function, denoted L^x_{adv}, to reach an equilibrium:

$$\begin{aligned} \min_{G_x} \max_{D_x} L^x_{adv}(G_x, D_x) &= E_{B_x \sim P_{DATA}(B_x)}[\log D_x(B_x)] \\ &+ E_{B'_x \sim P_{DATA}(B_x)}[\log(1 - D_x(B'_x))] \\ B'_x &= G_x(\text{LOS}, \text{H}\alpha) \end{aligned} \tag{5.1}$$

Here, E takes the expectation over $P_{DATA}(B_x)$, which is the true distribution of B_x. The generator G_x takes as input a pair of aligned HMI LOS magnetogram and Hα image and generates as output the vector component B'_x. The discriminator D_x takes as input the pair of B'_x (B_x, respectively) images and produces as output "Fake" features ("Real" features, respectively) for the fake sample B'_x (the ground truth B_x, respectively). Initially, $D_x(B_x) = 1$ and $D_x(B'_x) = 0$. When the equilibrium is reached, the training is completed where the discriminator cannot tell the difference between the generated fake sample and ground truth. We adopt the same training scheme for generating/predicting B'_y with the generator G_y, discriminator D_y, and loss L^y_{adv}.

Figure 5.2 illustrates the generator G_x; G_y is implemented similarly and its description is omitted. During training, each of the input HMI LOS magnetogram and Hα image is first processed by a convolutional block. Each convolutional block consists of a convolution layer with batch normalization, followed by a parametric ReLU (PReLU) activation function [64]. The feature maps produced by the convolutional blocks are concatenated and fed to a ResNet-like network [65]. The output of the ResNet-like network is then sent to two modules: the channel attention

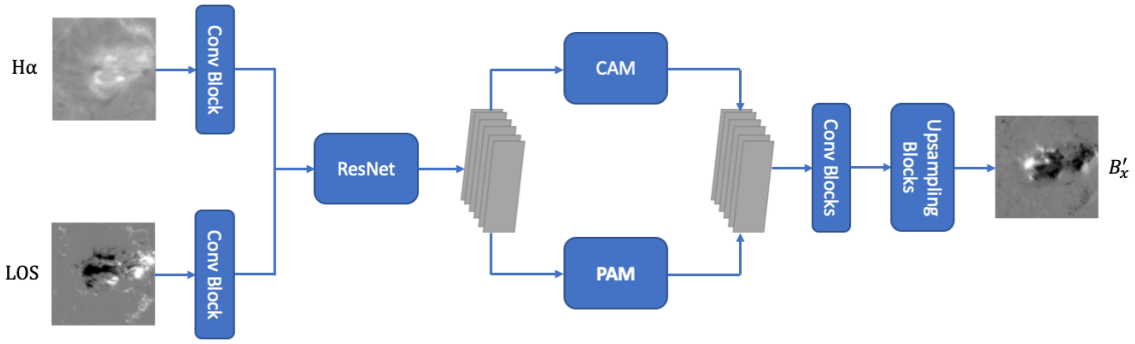

Figure 5.2 Illustration of the generator G_x of MagNet, which takes as input the pair of aligned HMI LOS and Hα images and produces as output the fake sample B'_x.

module (CAM) and position attention module (PAM), both of which were originally designed for DANet [43]. CAM and PAM leverage the self-attention mechanisms in the modules to better capture and transform a wider range of contextual information into local features, thus enhancing their representation capability. Both CAM and PAM are calculated in a similar way where CAM applies the self-attention mechanism to image channels while PAM focuses on the calculation of location information. The outputs of CAM and PAM are combined and sent to convolutional blocks, followed by upsampling blocks (i.e., convolutional blocks with upsampling), to produce the vector component B'_x.

Figure 5.3 illustrates the discriminator D_x, which employs the architecture developed by Ledig et al. [95], except that we replace their Leaky ReLU layer with the PReLU layer. D_x takes as input the pair of B'_x (B_x, respectively) images and produces as output "Fake" features ("Real" features, respectively) for the fake sample B'_x (the ground truth B_x, respectively). The discriminator D_y is implemented similarly.

During inference (testing), MagNet takes as input a pair of aligned MDI (or HMI) LOS magnetogram and Hα image and produces as output a fake sample B'_x (B'_y, respectively) through the trained generator G_x (G_y, respectively). These B'_x and B'_y are generated/predicted vector components corresponding to the input MDI (or HMI) LOS magnetogram and Hα image.

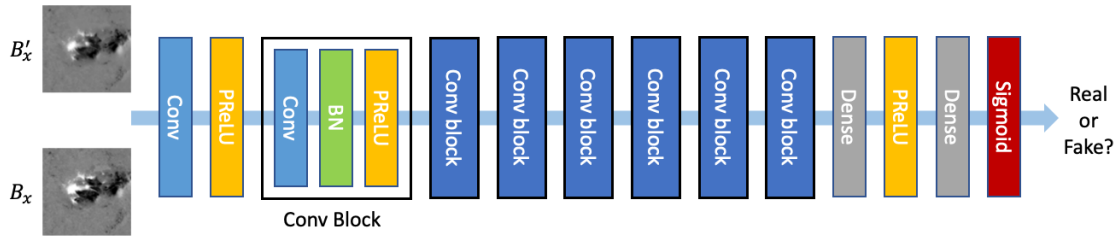

Figure 5.3 Illustration of the discriminator D_x of MagNet, which takes as input the pair of B'_x (B_x, respectively) images and produces as output "Fake" features ("Real" features, respectively) for the fake sample B'_x (the ground truth B_x, respectively).

For many of the AR patches used in our study, a large portion of each of them has small magnetic field strengths ($\leq$ 200 Gauss). Relatively few pixels in an AR patch have large magnetic field strengths ($>$ 200 Gauss). To tackle this imbalanced problem in our datasets, we employ a novel weighted L_w^p loss of a pixel p, defined as:

$$L_w^p(s', s) = |\frac{s}{c}||s' - s|. \tag{5.2}$$

Here, s' represents the MagNet-predicted magnetic field strength at p, s represents the true magnetic field strength at p, and c is a threshold. (In the study presented here, the threshold was set to 0.95.) The absolute difference between s' and s at p, usually reflected by the L_1 loss, is multiplied by a weight, $|\frac{s}{c}|$. This suggests that a pixel p with a larger (smaller, respectively) magnetic field strength yield a larger (smaller, respectively) L_w^p loss.

The weighted loss between the predicted B'_x and the true B_x, denoted $L_w(B'_x, B_x)$, is defined as:

$$L_w(B'_x, B_x) = \frac{1}{N} \sum_{p \in B_x} L_w^p(s', s), \tag{5.3}$$

where N is the total number of pixels in B_x. The total loss, denoted L_{MagNet}^x, is defined as the sum of the adversarial loss in Equation (5.1) and the weighted L_w loss

in Equation (5.3), as shown in Equation (5.4) below:

$$L^x_{MagNet} = \lambda L^x_{adv} + L_w(B'_x, B_x), \tag{5.4}$$

where λ was set to 0.001 [95]. The total loss, L^y_{MagNet}, for predicting B'_y is defined similarly. The training of MagNet was done by applying the Adam [14] optimizer to minimizing L^x_{MagNet} and L^y_{MagNet} individually on two separate NVIDIA A100 GPUs.

5.3.3 Performance Metrics

We adopt three metrics, namely the mean absolute error (MAE) [138], percent agreement (PA) [110], and Pearson correlation coefficient (CC) [46, 122], to quantitatively evaluate the performance of MagNet. The first performance metric is defined as:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |s'_i - s_i|, \tag{5.5}$$

where $N = 256 \times 256 = 65536$ is the total number of pixels in an AR patch (vector component), and s_i (s'_i, respectively) denotes the true (predicted, respectively) magnetic field strength for the ith pixel, $1 \leq i \leq 65536$, in the vector component. This metric is used to quantitatively assess the dissimilarity (distance) between the true magnetic field strengths and predicted magnetic field strengths in the vector component.

The second performance metric is defined as:

$$\text{PA} = \frac{M}{N} \times 100\%, \tag{5.6}$$

where M denotes the total number of agreement pixels in the vector component. We say the ith pixel in the vector component is an agreement pixel if $|s'_i - s_i|$ is smaller than a threshold. (In the study presented here, the threshold was set to 100 Gauss.)

This metric is used to quantitatively assess the similarity between the true magnetic field strengths and predicted magnetic field strengths in the vector component.

The third performance metric is defined as:

$$\mathrm{CC} = \frac{\mathrm{E}[(T - \mu_T)(G - \mu_G)]}{\sigma_T \sigma_G}, \tag{5.7}$$

where T and G represent the true magnetic field strengths and predicted magnetic field strengths, respectively, in the vector component, μ_T and μ_G are the mean of T and G, respectively; σ_T and σ_G are the standard deviation of T and G, respectively; and $E(\cdot)$ is the expectation. The value of CC ranges from -1 to 1. A value of 1 means that a linear equation describes the relationship between T and G perfectly where all data points lying on a line for which G increases as T increases. A value of -1 means that all data points lie on a line for which G decreases as T increases. A value of 0 means that there is no linear correlation between the variables T and G. CC measures the linear correlation between the true magnetic field strengths and predicted magnetic field strengths, quantifying how close these values are [46, 122, 138].

5.4 Results

5.4.1 Quantitative Evaluation of the MagNet Model on HMI and MDI Data

Our training set, Train_HMI, contains 8442 pairs of aligned BBSO Hα images and HMI LOS magnetograms of ARs in the period between 2014-01-01 and 2017-08-04 as well as corresponding 8442 vector components B_x (B_y, respectively) from HMI, which are used as labels. We employ two test sets to evaluate the performance of the trained MagNet model. The first test set, Test_HMI, contains 250 pairs of aligned BBSO Hα images and HMI LOS magnetograms of ARs in the period between 2017-08-05 and 2017-12-31. The second test set, Test_MDI, contains 24 pairs of aligned BBSO

Hα images and MDI LOS magnetograms of ARs in the overlapping period of MDI and HMI between 2010-05-01 and 2011-04-11 when MDI and HMI obtained data simultaneously. We use the $250 + 24 = 274$ corresponding true vector components B_x and B_y from HMI to verify whether the predictions made by the trained MagNet model are correct or not. All the images have 256×256 pixels. The data collection and processing procedures are described in Section 5.2 The training set and the two test sets are disjoint, so the trained MagNet model can make predictions on data that it has never seen before.

We adopt three performance metrics described in Subsection 5.3.3, to quantitatively evaluate the performance of MagNet. PA and CC do not have units, while MAE has a unit of Gauss (G). Table 5.1 presents the average performance metric values of MagNet based on the data in the two test sets. The row of Test_HMI(B'_x) (Test_HMI(B'_y), respectively) presents the average performance metric values of MagNet obtained by generating B'_x (B'_y, respectively) based on the test data in Test_HMI. The row of Test_MDI(B'_x) (Test_MDI(B'_y), respectively) presents the average performance metric values of MagNet obtained by generating B'_x (B'_y, respectively) based on the test data in Test_MDI. The training data were from Train_HMI.

Table 5.1 shows that the generated B'_x (B'_y, respectively) components are close to the true B_x (B_y, respectively) components with MAE < 100 G for both of the HMI and MDI test data. The CC is approximately 0.9 for the HMI test data and 0.75 for the MDI test data. MagNet performs better on the HMI test data than on the MDI test data. This happens due to several reasons. First, the MagNet model is trained by HMI data. There is a difference between HMI LOS magnetograms and MDI LOS magnetograms due to the spatially dependent calibration factor applied to the MDI magnetograms compared to the HMI magnetograms. The strong-field and weak-field pixels yield slightly different scaling factors [107]. In addition, the resolution of MDI

Table 5.1 Average Performance Metric Values of MagNet Based on the Data in Our Test Sets.

Test Sets	MAE	PA	CC
Test_HMI(B'_x)	62.70	86.52%	0.9097
Test_HMI(B'_y)	59.26	85.57%	0.8771
Test_MDI(B'_x)	75.86	81.06%	0.7410
Test_MDI(B'_y)	79.95	78.14%	0.7818

images is $4''$, which is lower than the resolution, $1''$, of HMI images. Finally, due to the larger cadence (96 minutes) of MDI compared to the cadence (12 minutes) of HMI, the time gaps between the MDI images and their aligned Hα images are often larger than the time gaps between the HMI images and their aligned Hα images. As a result, the quality of the aligned pairs for MDI is lower than the quality of the aligned pairs for HMI. In what follows we present some sample predictions for HMI and MDI.

5.4.2 Predicting Vector Components of AR 12683 Based on Hα and HMI LOS Data

Figure 5.4 presents predicted B'_x and B'_y components for AR 12683 on 2017 October 2 16:00:00 UT where training data were from Train_HMI. Figure 5.4(A) shows a pair of aligned BBSO Hα image and HMI LOS magnetogram of AR 12683. This pair of images is used as input to the trained MagNet model. Figure 5.4(B) shows the B'_x and B'_y components predicted by MagNet with respect to the input, corresponding true B_x and B_y components and scatter plots. AR 12683 is a relatively simple active region. It can be seen from Figure 5.4(B) that MagNet works extremely well on this simple AR, capable of producing the predicted vector components that are very close to the true components, with CC values being 0.97 for B_x and B_y.

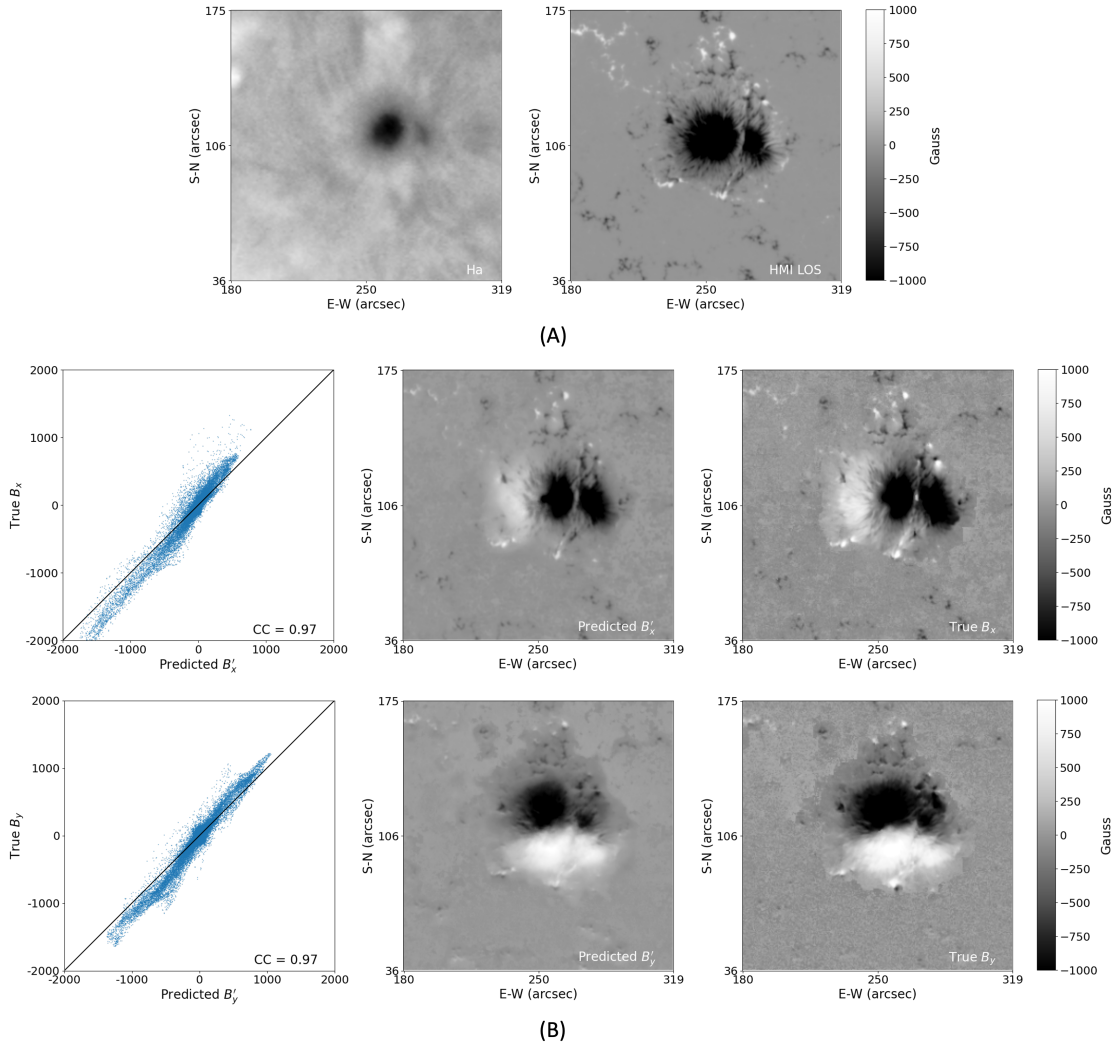

Figure 5.4 Comparison between MagNet-predicted B'_x, B'_y and true B_x, B_y based on Hα and HMI LOS data of AR 12683 on 2017 October 2 16:00:00 UT, where training data were taken from ARs in the period between 2014-01-01 and 2017-08-04. (A) The test data containing a pair of aligned BBSO Hα image (left) and HMI LOS magnetogram (right) of AR 12683. (B) The results predicted by MagNet. The first column shows scatter plots where the X-axis represents the predicted B'_x (B'_y, respectively) and the Y-axis represents the true B_x (B_y, respectively). The diagonal line in a scatter plot corresponds to pixels whose predicted B'_x (B'_y, respectively) values are identical to true B_x (B_y, respectively) values. The second column shows the predicted B'_x (top) and B'_y (bottom). The third column shows the true B_x (top) and B_y (bottom).

5.4.3 Predicting Vector Components of AR 12673 Based on Hα and HMI LOS Data

AR 12673 is a very complex active region on 2017 September 6. It is the most flare-productive AR in solar cycle 24, showing strong magnetic fields in the light bridge and apparent photospheric twist [152]. Furthermore, this active region contains many pixels with extremely large magnetic field strengths (saturated at the value of 5000 G or larger). The MagNet model trained by the 8442 pairs of images in the Train_HMI set described in the beginning of "Result" did not perform well on AR 12673, due to the lack of knowledge of very complex structures and extremely large magnetic field strengths such as those in AR 12673. To enhance the knowledge of MagNet and to tackle this extreme case, we created a new training set, denoted Train_HMI_New, by manually picking 835 Hα images from complex ARs in 2015 and 5 Hα images from AR 12673 that do not occur in the test sets. Thus, the new training set and the test sets are disjoint. This new training set contains 835 + 5 = 840 pairs of aligned BBSO Hα images and HMI LOS magnetograms as well as corresponding 840 vector components B_x (B_y, respectively) from HMI used as labels. Figure 5.5 presents the B'_x and B'_y components for AR 12673 on 2017 September 6 19:00:00 UT predicted by the new MagNet model trained by Train_HMI_New. Figure 5.5(A) shows a pair of aligned BBSO Hα image and HMI LOS magnetogram of AR 12673. This pair of images is used as input to the new MagNet model. Figure 5.5(B) shows the B'_x and B'_y components predicted by the new MagNet model with respect to the input, corresponding true B_x and B_y components and scatter plots. It can be seen from Figure 5.5(B) that the new MagNet model works well on the very complex AR 12673. capable of producing the predicted vector components that are close to the true components, with CC values being 0.94 for B_x and 0.84 for B_y.

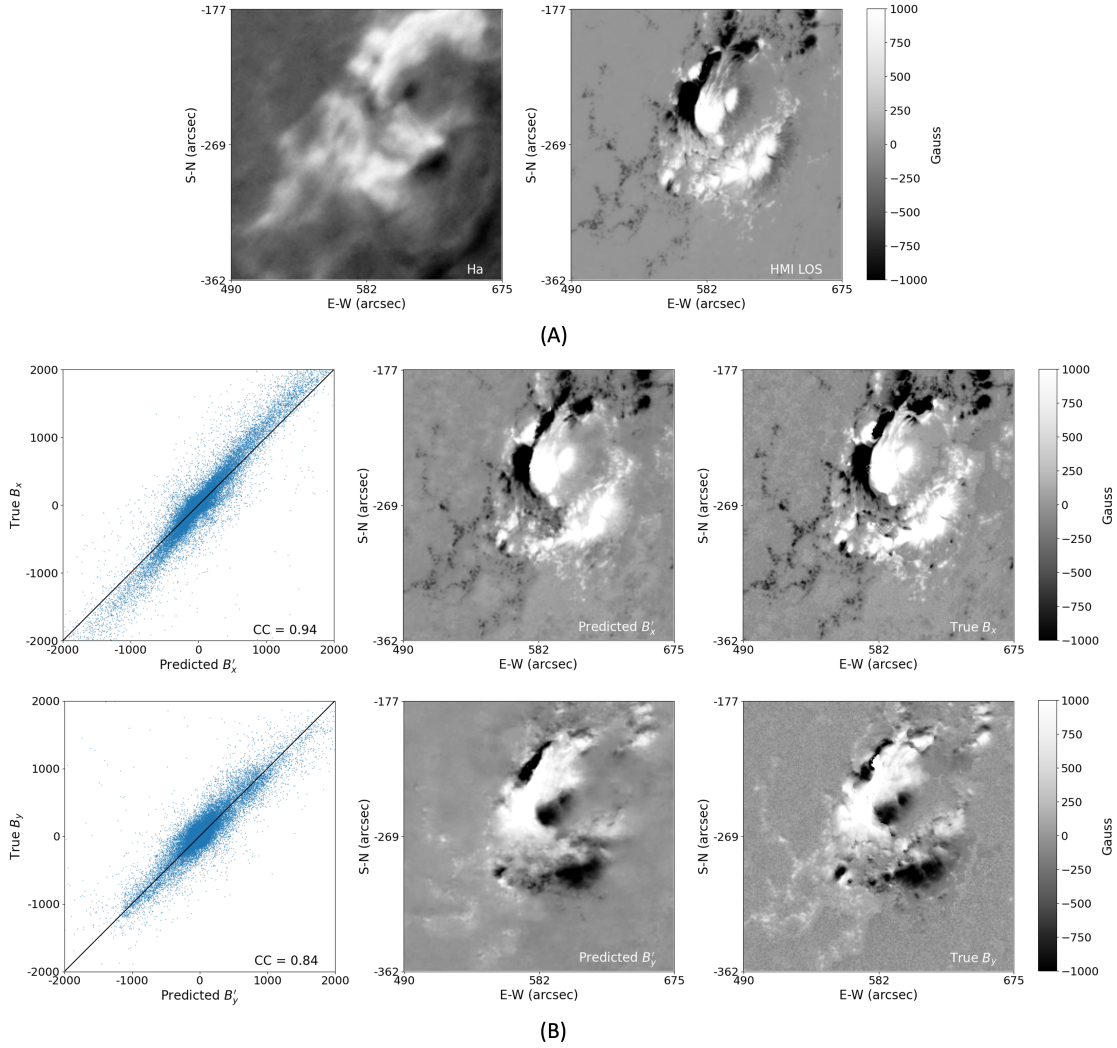

Figure 5.5 Comparison between MagNet-predicted B'_x, B'_y and true B_x, B_y based on Hα and HMI LOS data of AR 12673 on 2017 September 6 19:00:00 UT, where training data were taken from complex ARs in 2015 and from AR 12673 at time points different from 19:00:00 UT on 2017 September 6. (A) The test data containing a pair of aligned BBSO Hα image (left) and HMI LOS magnetogram (right) of AR 12673. (B) The results predicted by MagNet. The first column shows scatter plots where the X-axis represents the predicted B'_x (B'_y, respectively) and the Y-axis represents the true B_x (B_y, respectively). The diagonal line in a scatter plot corresponds to pixels whose predicted B'_x (B'_y, respectively) values are identical to true B_x (B_y, respectively) values. The second column shows the predicted B'_x (top) and B'_y (bottom). The third column shows the true B_x (top) and B_y (bottom).

5.4.4 Predicting Vector Components of AR 11101 Based on Hα and MDI LOS Data

Figure 5.6 presents predicted B'_x and B'_y components for AR 11101 on 2010 August 30 17:36:00 UT where training data were from Train_HMI. Figure 5.6(A) shows a pair of aligned BBSO Hα image and MDI LOS magnetogram of AR 11101. This pair of images is used as input to the trained MagNet model. Figure 5.6(B) shows the B'_x and B'_y components predicted by MagNet with respect to the input, corresponding true B_x and B_y components and scatter plots. AR 11101 is a relatively simple active region. It can be seen from Figure 5.6(B) that MagNet works pretty well on this simple AR, capable of producing the predicted vector components that are similar to the true components, with CC values being 0.87 for B_x and 0.88 for B_y. We note that, although both AR 11101 and the AR 12683 described above are relatively simple active regions, the CC values for AR 11101 are lower than those for AR 12683. This happens because MagNet is trained by HMI data and the vector components of AR 12683 are predicted based on test data also from HMI (though the training set and test set are collected in different time periods and hence are disjoint). However, the vector components of AR 11101 are predicted based on test data from MDI.

5.4.5 Predicting Vector Components of AR 11117 Based on Hα and MDI LOS Data

Figure 5.7 presents predicted B'_x and B'_y components for AR 11117 on 2010 October 27 22:24:00 UT where training data were from Train_HMI. Figure 5.7(A) shows a pair of aligned BBSO Hα image and MDI LOS magnetogram of AR 11117. This pair of images is used as input to the trained MagNet model. Figure 5.7(B) shows the B'_x and B'_y components predicted by MagNet with respect to the input, corresponding true B_x and B_y components and scatter plots. AR 11117 is a relatively complex active region. It can be seen from Figure 5.7(B) that MagNet works reasonably well

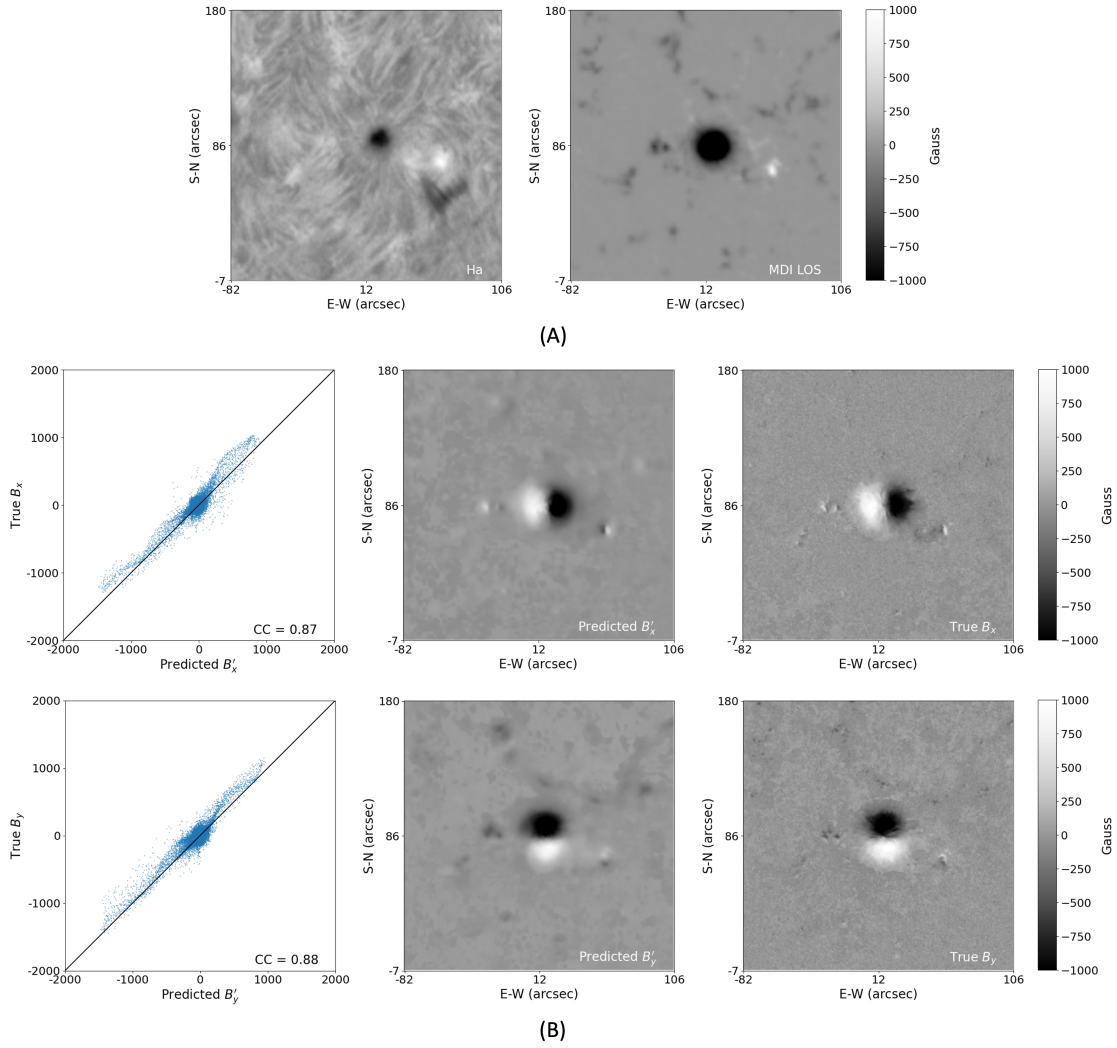

Figure 5.6 Comparison between MagNet-predicted B'_x, B'_y and true B_x, B_y based on Hα and MDI LOS data of AR 11101 on 2010 August 30 17:36:00 UT, where training data were taken from ARs in the period between 2014-01-01 and 2017-08-04. (A) The test data containing a pair of aligned BBSO Hα image (left) and MDI LOS magnetogram (right) of AR 11101. (B) The results predicted by MagNet. The first column shows scatter plots where the X-axis represents the predicted B'_x (B'_y, respectively) and the Y-axis represents the true B_x (B_y, respectively). The diagonal line in a scatter plot corresponds to pixels whose predicted B'_x (B'_y, respectively) values are identical to true B_x (B_y, respectively) values. The second column shows the predicted B'_x (top) and B'_y (bottom). The third column shows the true B_x (top) and B_y (bottom).

on this complex AR, capable of producing the predicted vector components that are similar to the true components, with CC values being 0.78 for B_x and 0.79 for B_y. Compared to the CC values for the relatively simple AR 11101 described above, the CC values for the more complex AR 11117 are lower.

5.5 Summary

We present a new deep learning approach (MagNet) for generating vector components B'_x and B'_y to create synthetic vector magnetograms of solar active regions. This approach allows us to expand the availability of photospheric vector magnetograms to the period from 1996 to present, covering solar cycles 23 and 24. The vector magnetograms can be used as input of NLFFF extrapolation methods to calculate free magnetic energy, magnetic topology and magnetic helicity injection. They can also be used as boundary conditions for MHD simulations. Furthermore, one can derive useful magnetic field parameters from the vector magnetograms to forecast solar flare activity [102, 103, 104]. Our experimental results obtained by using Hα observations and HMI magnetograms in the period between 2014-01-01 and 2017-08-04 as training data demonstrated the good performance of the proposed approach. Specifically, when using the trained model to predict vector components based on Hα and HMI LOS data and validated by HMI vector data in the period between 2017-08-05 and 2017-12-31, we obtained an average Pearson correlation coefficient (CC) of $\sim$0.9. When using the trained model to predict vector components based on Hα and MDI LOS data and validated by vector data from the overlapping period of MDI and HMI between 2010-05-01 and 2011-04-11, we obtained an average CC of $\sim$0.75.

Like other machine learning models, the performance of MagNet depends on training data. For example, in handling the very complex active region AR 12673, our original model trained by 8442 pairs of images does not perform well. We have to include complex images in the training set in order to generate satisfactory vector

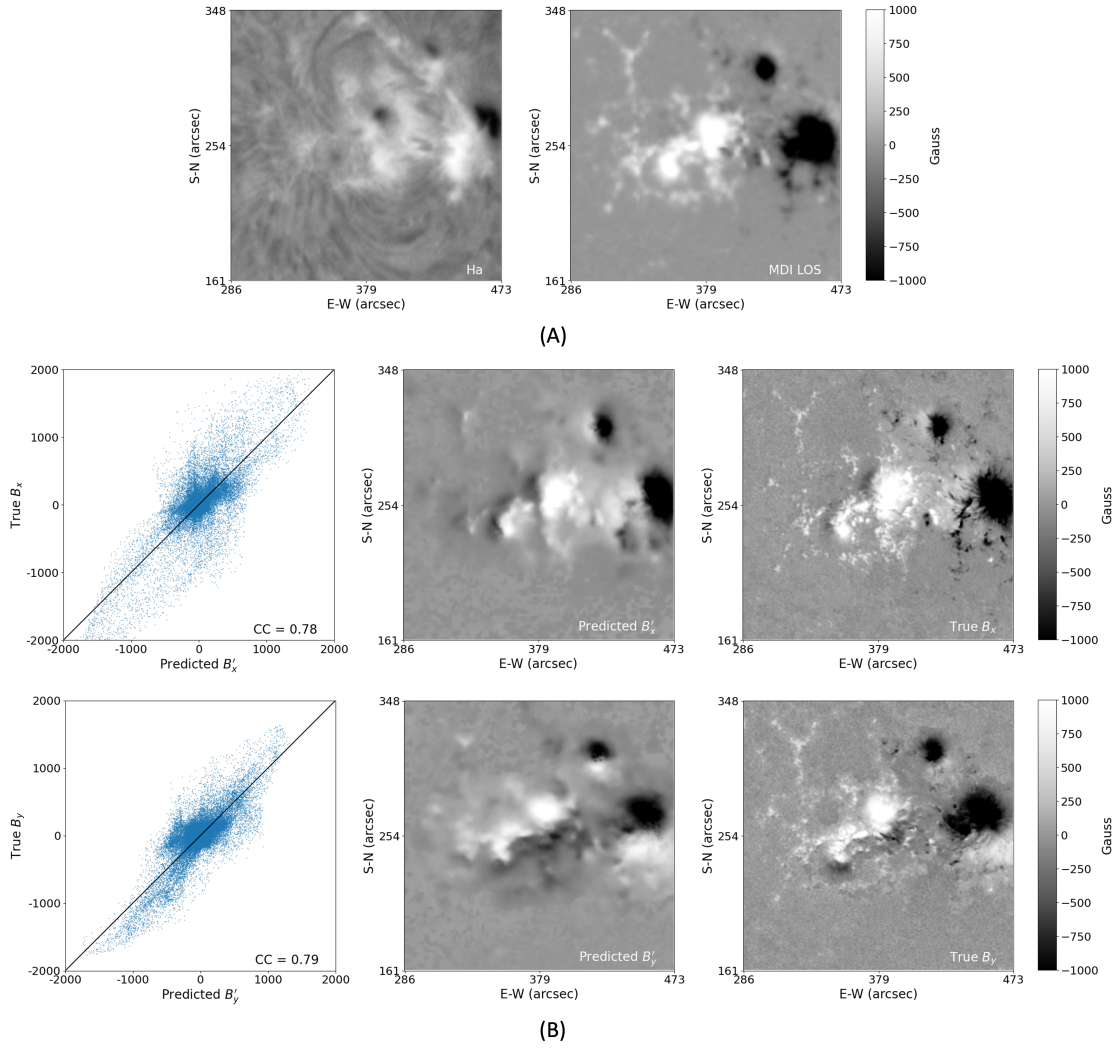

Figure 5.7 Comparison between MagNet-predicted B'_x, B'_y and true B_x, B_y based on Hα and MDI LOS data of AR 11117 on 2010 October 27 22:24:00 UT, where training data were taken from ARs in the period between 2014-01-01 and 2017-08-04. (A) The test data containing a pair of aligned BBSO Hα image (left) and MDI LOS magnetogram (right) of AR 11117. (B) The results predicted by MagNet. The first column shows scatter plots where the X-axis represents the predicted B'_x (B'_y, respectively) and the Y-axis represents the true B_x (B_y, respectively). The diagonal line in a scatter plot corresponds to pixels whose predicted B'_x (B'_y, respectively) values are identical to true B_x (B_y, respectively) values. The second column shows the predicted B'_x (top) and B'_y (bottom). The third column shows the true B_x (top) and B_y (bottom).

magnetograms. Most regions in the 8442 pairs of images have small magnetic field strengths, with the maximum magnetic field strength being much smaller than 5000 G. As a consequence, the model trained by these 8442 pairs of images performs well in predicting vector components with small magnetic field strengths. It suffers when predicting vector components having complex structures with very large magnetic field strengths.

To our knowledge, MagNet is the first tool capable of generating photospheric vector magnetograms of solar active regions. To further improve the performance of MagNet, one could use image super-resolution techniques [95] to enhance the quality of MDI LOS magnetograms before feeding them to the MagNet model. In the current study, MagNet is trained by HMI data and tested on MDI data. However, HMI and MDI are two different instruments on different observatories. The performance of MagNet could be further improved by using transfer learning techniques [49].

CHAPTER 6

DEEP LEARNING TOOLS FOR SOLAR PHYSICS

This chapter presents the implementation of some of deep learning-methods using Jupyter notebooks and Google Colab with Github, e.g. SolarUnet for magnetic flux tracking and FibrilNet for fibril tracing. The rest two tools will be public available once the paper are accepted.

Figure 6.1 shows the Github repository of the SolarUnet implemented based on Jiang et al. [75]. This tool has also been Binder enabled and published at Zenodo. The following Figures 6.2, 6.3 , 6.4 , 6.5 , 6.6 and 6.7 demo the details of how SolarUnet tool works on Google Colab.

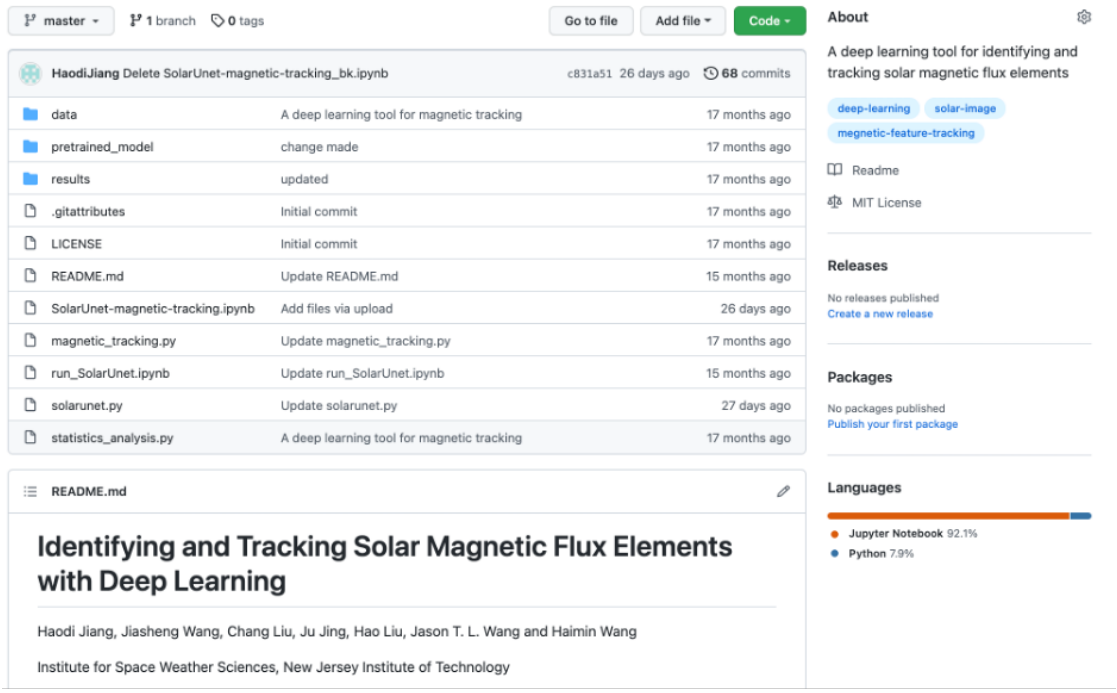

Figure 6.1 Illustration of the Github repository page of SolarUnet.

Figure 6.8 shows Github repository of the FibrilNet based on Jiang et al. [74].

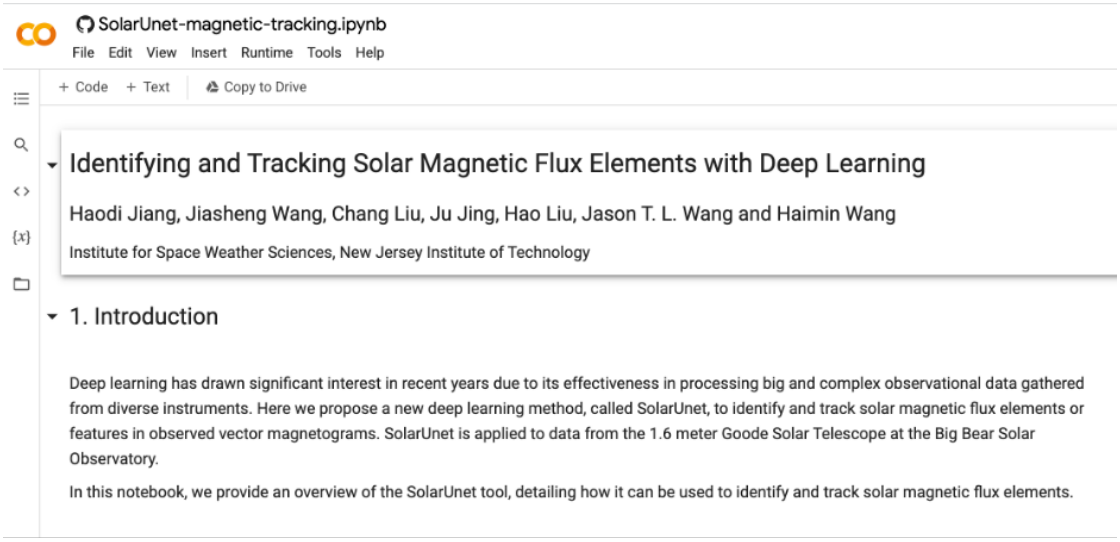

Figure 6.2 Illustration of the Google Colab page of SolarUnet: the introduction.

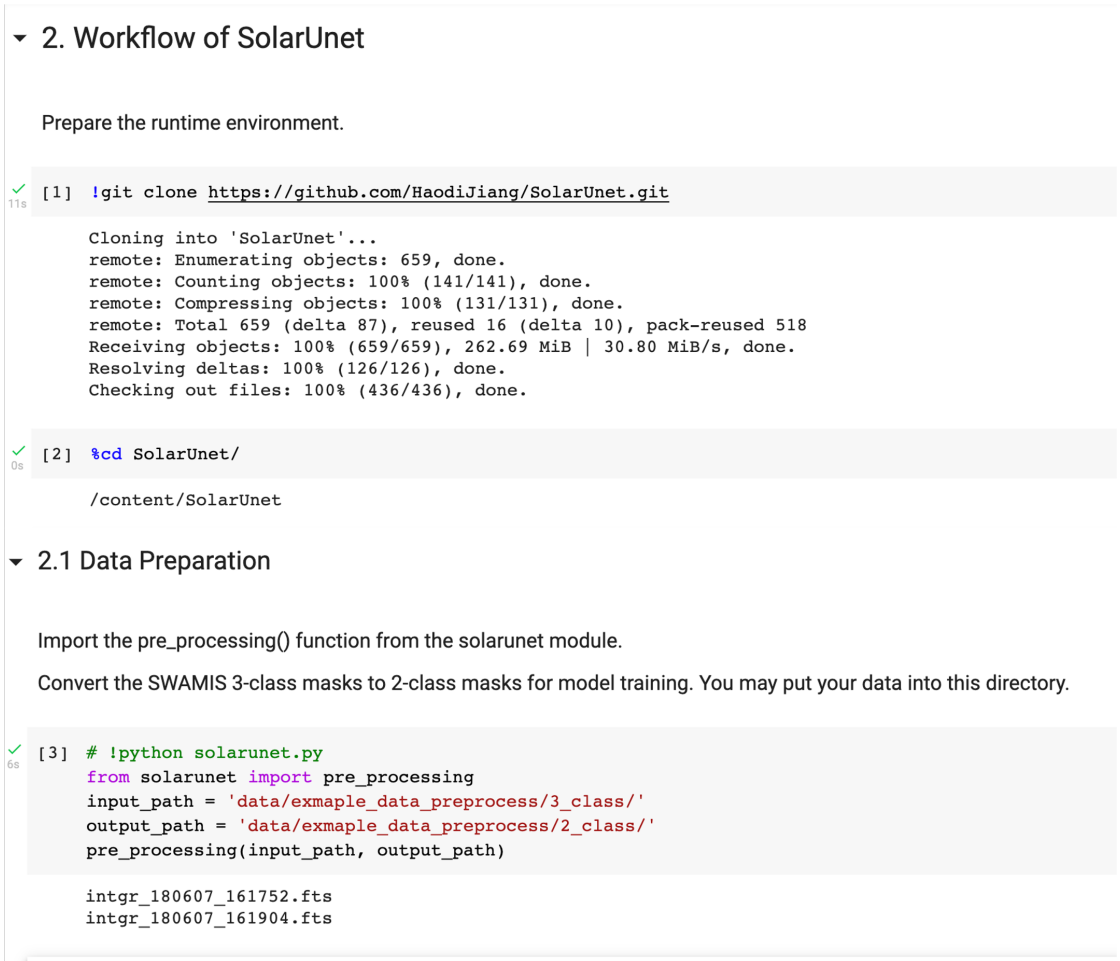

Figure 6.3 Illustration of the Google Colab page of SolarUnet: runtime environment and data preparation.

2.2 Model Training and Predicting

You may train the model with your own data or directly use the pretrained model for prediction and feature tracking. Here, we show how to train the model using the data prepared in Section 2.1.

2.2.1 Training with Default Data and Predicting

Import the model_training(), model_predicting() functions from the solarunet module.

The model is trained with the data prepared in Section 2.1 and tested on the given magnetorgrams. Please make sure your input data is at the given directory or you may create your own directory and modify the path. The predicted results will be saved in the given path.

```
from solarunet import model_training, model_predicting
input_path ='data/magnetic/'
output_path ='results/predicted_mask/'
model_training(input_path)
model_predicting(input_path, output_path)
```

```
Found 196 images belonging to 1 classes.
Found 196 images belonging to 1 classes.
   57/10000 [..............................] - ETA: 1:48:59 - loss: 0.2576 - accuracy: 0.8894
```

2.2.2 Predicting with the Pretrained Model

Import the model_predicting() function from the solarunet module.

Predict the binary masks of the given magnetograms by using the pretrained model. We set the 3rd argument of model_predicting() as True. The predicted results will be saved in the given path.

```
[5] from solarunet import model_predicting
    input_path ='data/magnetic/'
    output_path ='results/predicted_mask/'
    model_predicting(input_path, output_path, pretrain=True)
```

```
3/3 [==============================] - 1s 265ms/step
Prediction on the given data done
```

Figure 6.4 Illustration of the Google Colab page of SolarUnet: model training and predicting.

▾ 2.3 Postprocessing Data

Import the post_processing() and plot_mask() functions from the solarunet module.

Convert the predicted binary masks to 3-class masks and use the plot_mask() function to draw the SolarUnet masks of the testing magnetorgrams.

```
[6] from solarunet import post_processing
    from solarunet import plot_mask
    %matplotlib inline
    post_processing()
    plot_mask()
```

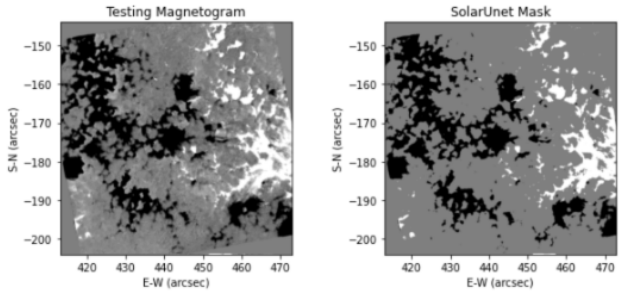

Figure 6.5 Illustration of the Google Colab page of SolarUnet: postprocessing data

▾ 2.4 Magnetic Tracking

Import the magnetic_tracking() function from the magnetic_tracking module.

Magnetic tracking algorithms are performed on the three consecutive testing magnetograms. An option of saving feature lifetime was provided by the 3rd argument of the magnetic_tracking() function. The tracking results will be saved at the given path.

```
[7] from magnetic_tracking import magnetic_tracking
    input_path = 'results/processed_data_for_tracking/'
    output_path = 'results/tracking_results/'
    magnetic_tracking(input_path, output_path)
    # lifetime_path = 'data/statistics_analysis/lifetime'
    # magnetic_tracking(input_path, output_path, lifetime_path)
```

```
============magnetic tracking start============
-----------------process frame 1------------------
-----------------process frame 2------------------
-----------------process frame 3------------------
-----------------Done------------------
```

```
[8] from solarunet import plot_tracking_results
    %matplotlib inline
    plot_tracking_results()
```

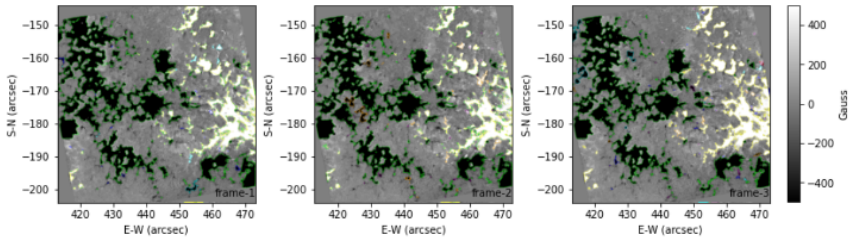

Figure 6.6 Illustration of the Google Colab page of SolarUnet: magnetic tracking.

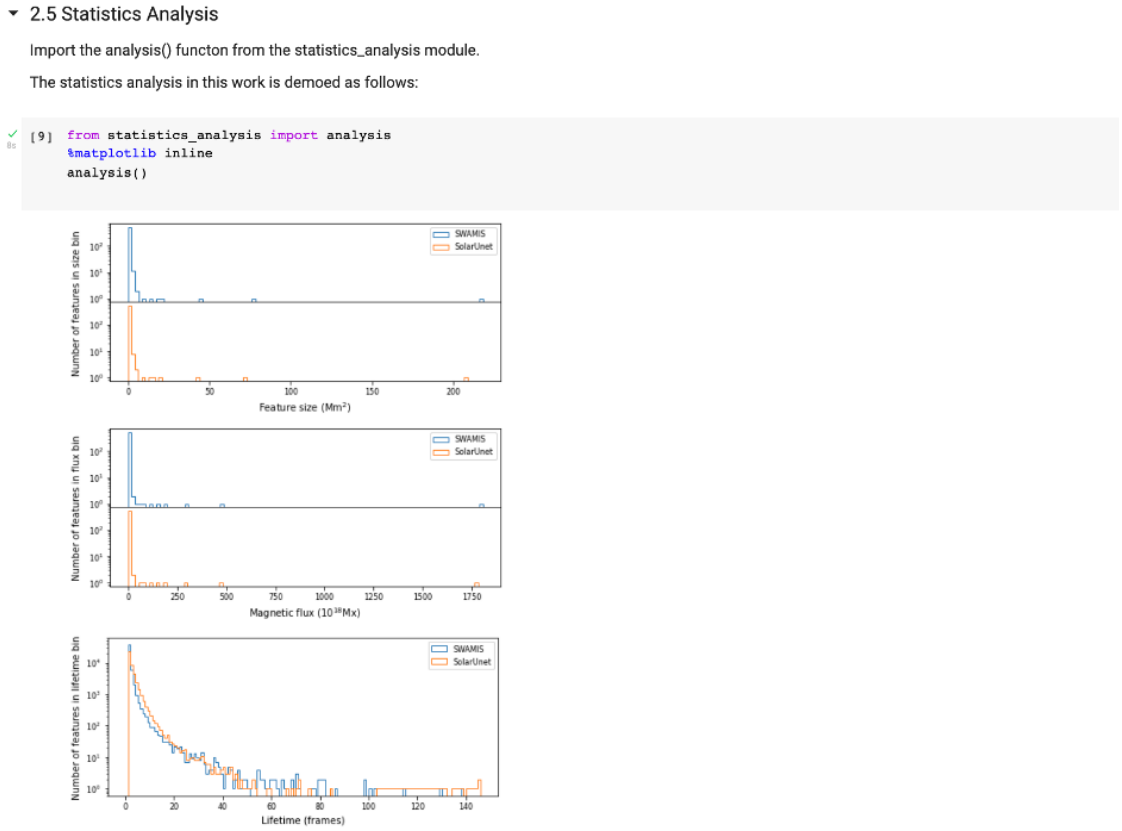

Figure 6.7 Illustration of the Google Colab page of SolarUnet: statistics analysis.

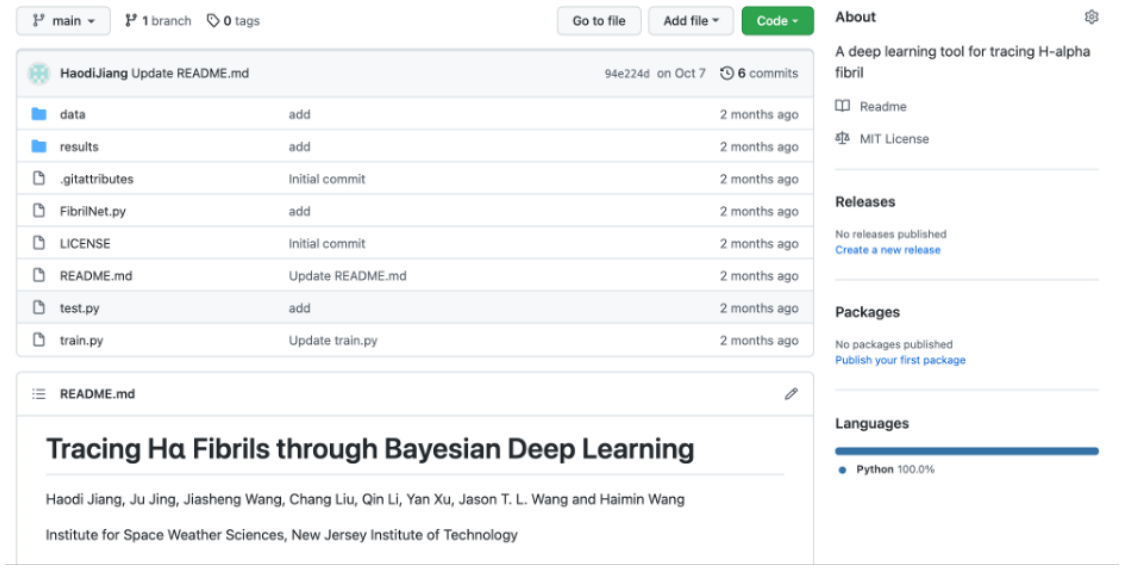

Figure 6.8 Illustration of the Github repository page of FibrilNet.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

This dissertation addresses four interrelated problems in solar physics: magnetic flux tracking, fibril tracing, Stokes inversion and vector magnetogram generation. First, this dissertation presents an encode-decoder deep learning method, SolarUnet to identify and track solar magnetic flux elements in observed vector magnetograms. Second, this dissertation presents presents a Bayesian convolutional neural network for probabilistic image segmentation with uncertainty quantification, to tracing chromospheric fibrils in Hα images of solar observations. Third, this dissertation presents a stacked deep neural network (SDNN) for inferring line-of-sight (LOS) velocities and Doppler widths from Stokes profiles. Fourth, this dissertation presents a generative adversarial network, named MagNet, for generating vector components to create synthetic vector magnetograms of solar active regions. Substantial experimental results have validated and demonstrated of good performance of these methods. Finally, this dissertation presents the implementations of some of the deep learning-based methods using Jupyter notebooks and Google Colab with GitHub.

This dissertation develops new machine learning and computer vision based tools to help understand solar activities and predict violent solar eruptions. In the future work, the Bayesian convolutional neural network with Monte Carlo dropout sampling will be investigated and extended to more solar physics related tasks, such as magnetic tracking, Stokes inversion and vector magnetograms generation. A operational Stoke inversion model will be developed based on the all types of ARs, where those ARs will be collected based on a classification algorithm, e.g. Mount Wilson magnetic classifications. The image super-resolution techniques based deep learning method will be studied and used to improve the image quality of Hα images

and MDI LOS magnetograms, as a consequence, the performance of MagNet could be improved.

REFERENCES

[1] Y. Abduallah, J. T. L. Wang, Y. Nie, C. Liu, and H. Wang. DeepSun: Machine-Learning-as-a-Service for Solar Flare Prediction. *Research in Astronomy and Astrophysics*, 21:160, 2021.

[2] C. C. Aggarwal. *Neural Networks and Deep Learning*. Springer, Cham, NY, 2018.

[3] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. *Journal of Machine Learning Research*, 1:113141, September 2001.

[4] S. K. Antiochos, C. R. DeVore, and J. A. Klimchuk. A Model for Solar Coronal Mass Ejections. *The Astrophysical Journal*, 510:485–493, January 1999.

[5] M. J. Aschwanden, K. Reardon, and D. B. Jess. Tracing the Chromospheric and Coronal Magnetic Field with AIA, IRIS, IBIS, and ROSA Data. *The Astrophysical Journal*, 826(1):61, July 2016.

[6] M. J. Aschwanden, Y. Xu, and J. Jing. Global Energetics of Solar Flares. I. Magnetic Energies. *The Astrophysical Journal*, 797(1):50, December 2014.

[7] A. Asensio Ramos, J. de la Cruz Rodríguez, M. J. Martínez González, and H. Socas-Navarro. Inference of the Chromospheric Magnetic Field Orientation in the Ca II 8542 Å Line Fibrils. *Astronomy and Astrophysics*, 599:A133, March 2017.

[8] A. Asensio Ramos and C. J. Díaz Baso. Stokes Inversion Based on Convolutional Neural Networks. *Astronomy and Astrophysics*, 626:A102, 2019.

[9] A. Asensio Ramos, J. Trujillo Bueno, and E. Landi Degl'Innocenti. Advanced Forward Modeling and Inversion of Stokes Profiles Resulting from the Joint Action of the Hanle and Zeeman Effects. *The Astrophysical Journal*, 683(1):542–565, August 2008.

[10] Astropy Collaboration, T. P. Robitaille, E. J. Tollerud, P. Greenfield, M. Droettboom, E. Bray, T. Aldcroft, M. Davis, A. Ginsburg, A. M. Price-Whelan, W. E. Kerzendorf, A. Conley, N. Crighton, K. Barbary, D. Muna, H. Ferguson, F. Grollier, M. M. Parikh, P. H. Nair, H. M. Unther, C. Deil, J. Woillez, S. Conseil, R. Kramer, J. E. H. Turner, L. Singer, R. Fox, B. A. Weaver, V. Zabalza, Z. I. Edwards, K. Azalee Bostroem, D. J. Burke, A. R. Casey, S. M. Crawford, N. Dencheva, J. Ely, T. Jenness, K. Labrie, P. L. Lim, F. Pierfederici, A. Pontzen, A. Ptak, B. Refsdal, M. Servillat, and O. Streicher. Astropy: A Community Python Package for Astronomy. *Astronomy and Astrophysics*, 558:A33, October 2013.

[11] L. H. Auer, J. N. Heasley, and L. L. House. The Determination of Vector Magnetic Fields From Stokes Profiles. *Solar Physics*, 55(1):47–61, November 1977.

[12] V. Badrinarayanan, A. Kendall, and R. Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.

[13] X. Bai, H. Liu, Y. Deng, J. Jiang, J. Guo, Y. Bi, T. Feng, Z. Jin, W. Cao, J. Su, and K. Ji. A Deep Learning Method to Estimate Magnetic Fields in Solar Active Regions from Photospheric Continuum Images. *Astronomy and Astrophysics*, 652:A143, August 2021.

[14] L. Balles and P. Hennig. Dissecting Adam: The Sign, Magnitude and Variance of Stochastic Gradients. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 413–422. PMLR, 2018.

[15] V. T. Bickel, B. Moseley, I. Lopez-Francos, and M. Shirley. Peering into Lunar Permanently Shadowed Regions with Deep Learning. *Nature Communications*, 12(1):5607, 2021.

[16] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[17] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[18] M. G. Bobra and S. Couvidat. Solar Flare Prediction Using SDO/HMI Vector Magnetic Field Data with a Machine-learning Algorithm. *The Astrophysical Journal*, 798(2):135, January 2015.

[19] M. G. Bobra, X. Sun, J. T. Hoeksema, M. Turmon, Y. Liu, K. Hayashi, G. Barnes, and K. D. Leka. The Helioseismic and Magnetic Imager (HMI) Vector Magnetic Field Pipeline: SHARPs - Space-weather HMI Active Region Patches. *Solar Physics*, 289(9):3549–3578, September 2014.

[20] M. G. Bobra, A. A. van Ballegooijen, and E. E. DeLuca. Modeling Nonpotential Magnetic Fields in Solar Active Regions. *The Astrophysical Journal*, 672(2):1209–1220, January 2008.

[21] V. Bommier. Milne-Eddington Inversion for Unresolved Magnetic Structures in the Quiet Sun photosphere. *Journal of Geophysical Research (Space Physics)*, 121(6):5025–5040, June 2016.

[22] J. M. Borrero, S. Tomczyk, M. Kubo, H. Socas-Navarro, J. Schou, S. Couvidat, and R. Bogart. VFISV: Very Fast Inversion of the Stokes Vector for the Helioseismic and Magnetic Imager. *Solar Physics*, 273(1):267–293, October 2011.

[23] J. M. Borrero, S. Tomczyk, A. Norton, T. Darnell, J. Schou, P. Scherrer, R. Bush, and Y. Liu. Magnetic Field Vector Retrieval With the Helioseismic and Magnetic Imager. *Solar Physics*, 240(1):177–196, January 2007.

[24] W. Cao, P. R. , K. Ahn, N. Gorceix, W. Schmidt, and H. Lin. NIRIS: The Second Generation Near-Infrared Imaging Spectro-polarimeter for the 1.6 Meter New Solar Telescope. In T. R. Rimmele, A. Tritschler, F. Wöger, M. Collados Vera, H. Socas-Navarro, R. Schlichenmaier, M. Carlsson, T. Berger, A. Cadavid, P. R. Gilbert, P. R. Goode, and M. Knölker, editors, *Second (Advanced Solar Technology Telescope) ATST-EAST Meeting: Magnetic Fields from the Photosphere to the Corona.*, volume 463 of *Astronomical Society of the Pacific Conference Series*, page 291, December 2012.

[25] W. Cao, N. Gorceix, R. Coulter, K. Ahn, T. R. Rimmele, and P. R. Goode. Scientific instrumentation for the 1.6 m New Solar Telescope in Big Bear. *Astronomische Nachrichten*, 331:636, June 2010.

[26] T. A. Carroll and M. Kopf. Zeeman-tomography of the Solar Photosphere. Three-dimensional Surface Structures Retrieved from Hinode Observations. *Astronomy and Astrophysics*, 481(1):L37–L40, April 2008.

[27] X. Chen, N. Deng, D. A. Lamb, J. Jing, C. Liu, R. Liu, S.-H. Park, and H. Wang. Development of Technique to Detect and Classify Small-scale Magnetic Flux Cancellation and Rapid Blue-shifted Excursions. *Research in Astronomy and Astrophysics*, 15(7):1012–1026, June 2015.

[28] Y. Chen, W. B. Manchester, A. O. Hero, G. Toth, B. DuFumier, T. Zhou, X. Wang, H. Zhu, Z. Sun, and T. I. Gombosi. Identifying Solar Flare Precursors Using Time Series of SDO/HMI Images and SHARP Parameters. *Space Weather*, 17(10):1404–1426, October 2019.

[29] C. E. DeForest, H. J. Hagenaar, D. A. Lamb, C. E. Parnell, and B. T. Welsch. Solar Magnetic Tracking. I. Software Comparison and Recommended Practices. *The Astrophysical Journal*, 666(1):576–587, September 2007.

[30] Y. Deng, C. C. Loy, and X. Tang. Aesthetic-Driven Image Enhancement by Adversarial Learning. In *Proceedings of the ACM International Conference on Multimedia*, pages 870–878, 2018.

[31] C. Denker, A. Johannesson, W. Marquette, P. R. Goode, H. Wang, and H. Zirin. Synoptic Hα Full-Disk Observations of the Sun from Big Bear Solar Observatory - I. Instrumentation, Image Processing, Data Products, and First Results. *Solar Physics*, 184(1):87–102, January 1999.

[32] J. S. Denker and Y. LeCun. Transforming Neural-Net Output Levels to Probability Distributions. In *Proceedings of the 3rd International Conference on Neural Information Processing Systems*, NIPS'90, page 853859, San Francisco, CA, USA, 1990. Morgan Kaufmann Publishers Inc.

[33] T. W. Epps and K. J. Singleton. An Omnibus Test for the Two-sample Problem Using the Empirical Characteristic Function, journal = Journal of Statistical Computation and Simulation. 26(3-4):177–203, 1986.

[34] T. Falk, D. Mai, R. Bensch, Ö. Çiçek, A. Abdulkadir, Y. Marrakchi, A. Böhm, J. Deubner, Z. Jäckel, K. Seiwald, A. Dovzhenko, O. Tietz, C. Dal Bosco, S. Walsh, D. Saltukoglu, T. L. Tay, M. Prinz, K. Palme, M. Simons, I. Diester, T. Brox, and O. Ronneberger. U-Net: Deep Learning for Cell Counting, Detection, and Morphometry. *Nature Methods*, 16(1):67–70, 2019.

[35] T. Falk, D. Mai, R. Bensch, . iek, A. Abdulkadir, Y. Marrakchi, A. Bhm, J. Deubner, Z. Jaeckel, K. Seiwald, O. Dovzhenko, O. Tietz, C. Dal Bosco, S. Walsh, D. Saltukoglu, T. Leng Tay, M. Prinz, K. Palme, M. Simons, and O. Ronneberger. U-Net: Deep Learning for Cell Counting, Detection, and Morphometry. *Nature Methods*, 16(1):67–70, 2019.

[36] G. Fleishman, I. Mysh'yakov, A. Stupishin, M. Loukitcheva, and S. Anfinogentov. Force-free Field Reconstructions Enhanced by Chromospheric Magnetic Field Data. *The Astrophysical Journal*, 870(2):101, January 2019.

[37] T. G. Forbes. A Review on the Genesis of Coronal Mass Ejections. *Journal of Geophysical Research*, 105(A10):23153–23166, Oct 2000.

[38] A. Fossum and M. Carlsson. Determination of the Acoustic Wave Flux in the Lower Solar Chromosphere. *The Astrophysical Journal*, 646(1):579–592, July 2006.

[39] P. Foukal. Hα Fine Structure and the Chromospheric Field. *Solar Physics*, 20(2):298–309, November 1971.

[40] P. Foukal. Morphological Relationships in the Chromospheric Hα Fine Structure. *Solar Physics*, 19(1):59–71, August 1971.

[41] S. L. Freeland and B. N. Handy. Data Analysis with the SolarSoft System. *Solar Physics*, 182:497–500, 1998.

[42] C. Frutiger, S. K. Solanki, M. Fligge, and J. H. M. J. Bruls. Properties of the Solar Granulation Obtained from the Inversion of Low Spatial Resolution Spectra. *Astronomy and Astrophysics*, 358:1109–1121, June 2000.

[43] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu. Dual Attention Network for Scene Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[44] R. Gafeira, A. Lagg, S. K. Solanki, S. Jafarzadeh, M. van Noort, P. Barthol, J. Blanco Rodríguez, J. C. del Toro Iniesta, A. Gandorfer, L. Gizon, J. Hirzberger, M. Knölker, D. Orozco Suárez, T. L. Riethmüller, and W. Schmidt. Morphological Properties of Slender Ca II H Fibrils Observed by Sunrise II. *The Astrophysical Journals*, 229:6, March 2017.

[45] Y. Gal and Z. Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In M. Balcan and K. Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org, 2016.

[46] F. Galton. Regression Towards Mediocrity in Hereditary Stature. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 15:246–263, 1886.

[47] R. Galvez, D. F. Fouhey, M. Jin, A. Szenicer, A. Muñoz-Jaramillo, M. C. Cheung, P. J. Wright, M. G. Bobra, Y. Liu, J. Mason, et al. A Machine-learning Data Set Prepared from the NASA Solar Dynamics Observatory Mission. *The Astrophysical Journal Supplement Series*, 242:7, 2019.

[48] D. Gao, E. Morini, M. Salani, A. J. Krauson, A. Chekuri, N. Sharma, A. Ragavendran, S. Erdin, E. M. Logan, W. Li, A. Dakka, J. Narasimhan, X. Zhao, N. Naryshkin, C. R. Trotta, K. A. Effenberger, M. G. Woll, V. Gabbeta, G. Karp, Y. Yu, G. Johnson, W. D. Paquette, G. R. Cutting, M. E. Talkowski, and S. A. Slaugenhaupt. A Deep Learning Approach to Identify Gene Targets of a Therapeutic for Human Splicing Disorders. *Nature Communications*, 12(1):3332, 2021.

[49] Y. Gao and Y. Cui. Deep Transfer Learning for Reducing Health Care Disparities Arising from Biomedical Data Inequality. *Nature Communications*, 11(1):5131, 2020.

[50] J. Gehring, Y. Miao, F. Metze, and A. Waibel. Extracting Deep Bottleneck Features Using Stacked Auto-encoders. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*, pages 3377–3381, 2013.

[51] F. Giannattasio, F. Berrilli, G. Consolini, D. Del Moro, M. Gošić, and L. Bellot Rubio. Occurrence and Persistence of Magnetic Elements in the Quiet Sun. *Astronomy and Astrophysics*, 611:A56, March 2018.

[52] J. D. Gibbons and S. Chakraborti. Nonparametric statistical inference. In M. Lovric, editor, *International Encyclopedia of Statistical Science*, pages 977–979. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[53] S. J. Goerg and J. Kaiser. Nonparametric Testing of Distributions–the Epps-Singleton Two-Sample Test using the Empirical Characteristic Function. *The Stata Journal*, 9(3):454–465, 2009.

[54] P. R. Goode and W. Cao. The 1.6 m off-axis New Solar Telescope (NST) in Big Bear. In L. M. Stepp, R. Gilmozzi, and H. J. Hall, editors, *Ground-based and Airborne Telescopes IV*, volume 8444, pages 1–8. International Society for Optics and Photonics, SPIE, 2012.

[55] P. R. Goode and W. Cao. The 1.6 m off-axis New Solar Telescope (NST) in Big Bear. In L. M. Stepp, R. Gilmozzi, and H. J. Hall, editors, *Ground-based and Airborne Telescopes IV*, volume 8444, pages 1–8. International Society for Optics and Photonics, SPIE, 2012.

[56] P. R. Goode, V. Yurchyshyn, W. Cao, V. Abramenko, A. Andic, K. Ahn, and J. Chae. Highest Resolution Observations of the Quietest Sun. *The Astrophysical Journal Letters*, 714:L31–L35, May 2010.

[57] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, 2016. `http://www.deeplearningbook.org`.

[58] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, 2020.

[59] A. Graves. Practical Variational Inference for Neural Networks. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pages 2348–2356, 2011.

[60] M. A. Gruet, M. Chandorkar, A. Sicard, and E. Camporeale. Multiple-Hour-Ahead Forecast of the Dst Index Using a Combination of Long Short-Term Memory Neural Network and Gaussian Process. *Space Weather*, 16(11):1882–1896, November 2018.

[61] M. R. Gupta, S. Bengio, and J. Weston. Training Highly Multiclass Classifiers. *Journal of Machine Learning Research*, 15(1):14611492, January 2014.

[62] H. J. Hagenaar, C. J. Schrijver, and A. M. Title. The Properties of Small Magnetic Regions on the Solar Surface and the Implications for the Solar Dynamo(s). *The Astrophysical Journal*, 584(2):1107–1119, February 2003.

[63] J. W. Harvey, F. Hill, R. P. Hubbard, J. R. Kennedy, J. W. Leibacher, J. A. Pintar, P. A. Gilman, R. W. Noyes, A. M. Title, J. Toomre, R. K. Ulrich, A. Bhatnagar, J. A. Kennewell, W. Marquette, J. Patron, O. Saa, and E. Yasukawa. The Global Oscillation Network Group (GONG) Project. *Science*, 272(5266):1284–1286, May 1996.

[64] K. He, X. Zhang, S. Ren, and J. Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[65] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[66] L. He, Y. Chao, K. Suzuki, and K. Wu. Fast Connected-component Labeling. *Pattern Recogn.*, 42(9):1977–1987, September 2009.

[67] P. Heinzel and B. Schmieder. Chromospheric Fine Structure: Black & White Mottles. *Astronomy and Astrophysics*, 282(3):939–954, February 1994.

[68] Z. Hu, T. Turki, and J. T. L. Wang. Generative Adversarial Networks for Stochastic Video Prediction With Action Control. *IEEE Access*, 8:63336–63348, 2020.

[69] M. Huertas-Company, J. R. Primack, A. Dekel, D. C. Koo, S. Lapiner, D. Ceverino, R. C. Simons, G. F. Snyder, M. Bernardi, Z. Chen, H. Domínguez-Sánchez, C. T. Lee, B. Margalef-Bentabol, and D. Tuccillo. Deep Learning Identifies High-z Galaxies in a Central Blue Nugget Phase in a Characteristic Mass Range. *The Astrophysical Journal*, 858(2):114, May 2018.

[70] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In F. R. Bach and D. M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org, 2015.

[71] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image Translation with Conditional Adversarial Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017.

[72] S. A. Jaeggli and A. A. Norton. The Magnetic Classification of Solar Active regions1992–2015. *The Astrophysical Journal*, 820(1):L11, March 2016.

[73] S. Jafarzadeh, S. K. Solanki, R. Gafeira, M. van Noort, P. Barthol, J. Blanco Rodríguez, J. C. del Toro Iniesta, A. dorfer, L. Gizon, J. Hirzberger, M. Knölker, D. Orozco Suárez, T. L. Riethmüller, and W. Schmidt. Transverse Oscillations in Slender Ca II H Fibrils Observed with Sunrise/SuFI. *The Astrophysical Journals*, 229:9, March 2017.

[74] H. Jiang, J. Jing, J. Wang, C. Liu, Q. Li, Y. Xu, J. T. L. Wang, and H. Wang. Tracing Hα fibrils through Bayesian deep learning. *The Astrophysical Journal Supplement Series*, 256(1):20, September 2021.

[75] H. Jiang, J. Wang, C. Liu, J. Jing, H. Liu, J. T. L. Wang, and H. Wang. Identifying and Tracking Solar Magnetic Flux Elements with Deep Learning. *The Astrophysical Journal Supplement Series*, 250(1):5, September 2020.

[76] J. Jing, Q. Li, C. Liu, J. Lee, Y. Xu, W. Cao, and H. Wang. High-resolution Observations of Dynamics of Superpenumbral H-alpha Fibrils. *The Astrophysical Journal*, 880(2):143, August 2019.

[77] J. Jing, Y. Yuan, K. Reardon, T. Wiegelmann, Y. Xu, and H. Wang. Nonpotentiality of Chromospheric Fibrils in NOAA Active Regions 11092 and 9661. *The Astrophysical Journal*, 739(2):67, September 2011.

[78] J. T. Karpen, S. K. Antiochos, and C. R. DeVore. The Mechanisms for the Onset and Explosive Eruption of Coronal Mass Ejections and Eruptive Flares. *The Astrophysical Journal*, 760(1):81, November 2012.

[79] A. Kendall and Y. Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 5580–5590, Red Hook, NY, USA, 2017. Curran Associates Inc.

[80] P. H. Keys, O. Steiner, and G. Vigeesh. On the Effect of Oscillatory Phenomena on Stokes Inversion Results. *Philosophical Transactions of the Royal Society of London Series A*, 379(2190):20200182, February 2021.

[81] T. Kim, E. Park, H. Lee, Y.-J. Moon, S.-H. Bae, D. Lim, S. Jang, L. Kim, I.-H. Cho, M. Choi, and K.-S. Cho. Solar Farside Magnetograms from Deep Learning Analysis of STEREO/EUVI Data. *Nature Astronomy*, 3(5):397–400, 2019.

[82] S. Kiranyaz, T. Ince, O. Abdeljaber, O. Avci, and M. Gabbouj. 1-D Convolutional Neural Networks for Signal Processing Applications. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8360–8364, 2019.

[83] W. Kirch, editor. *Pearson's Correlation Coefficient*, pages 1090–1091. Springer Netherlands, Dordrecht, 2008.

[84] B. Kliem, Y. N. Su, A. A. van Ballegooijen, and E. E. DeLuca. Magnetohydrodynamic Modeling of the Solar Eruption on 2010 April 8. *The Astrophysical Journal*, 779:129, December 2013.

[85] B. Kliem and T. Török. Torus Instability. *Physical Review Letters*, 96(25):255002, June 2006.

[86] Y. Kwon, J.-H. Won, B. J. Kim, and M. C. Paik. Uncertainty Quantification Using Bayesian neural networks in Classification: Application to Biomedical Image Segmentation. *Computational Statistics and Data Analysis*, 142:106816, 2020.

[87] A. Lagg, J. Woch, N. Krupp, and S. K. Solanki. Retrieval of the Full Magnetic Vector with the He I Multiplet at 1083 nm. Maps of an Emerging Flux Region. *Astronomy and Astrophysics*, 414:1109–1120, February 2004.

[88] D. A. Lamb, C. E. DeForest, H. J. Hagenaar, C. E. Parnell, and B. T. Welsch. Solar Magnetic Tracking. III. Apparent Unipolar Flux Emergence in High-resolution Observations. *The Astrophysical Journal*, 720(2):1405–1416, August 2010.

[89] D. A. Lamb, C. E. DeForest, H. J., C. E. Parnell, and B. T. Welsch. Solar Magnetic Tracking. II. The Apparent Unipolar Origin of Quiet-Sun Flux. *The Astrophysical Journal*, 674(1):520–529, February 2008.

[90] D. A. Lamb, T. A. Howard, C. E. DeForest, C. E. Parnell, and B. T. Welsch. Solar Magnetic Tracking. IV. The Death of Magnetic Features. *The Astrophysical Journal*, 774(2):127, August 2013.

[91] E. Landi Degl'Innocenti. Polarization in Spectral Lines. *Solar Physics*, 85(1):3–31, May 1983.

[92] E. Landi Degl'Innocenti and M. Landolfi. *Polarization in Spectral Lines*, volume 307. Kluwer Academic Publishers, Dordrecht, 2004.

[93] Ø. Langangen, B. De Pontieu, M. Carlsson, V. H. Hansteen, G. Cauzzi, and K. Reardon. Search for High Velocities in the Disk Counterpart of Type II Spicules. *The Astrophysical Journal Letters*, 679(2):L167, June 2008.

[94] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. *Nature*, 521:436–444, May 2015.

[95] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 105–114, 2017.

[96] J. Leenaarts, M. Carlsson, and L. Rouppe van der Voort. On Fibrils and Field Lines: the Nature of H-alpha Fibrils in the Solar Chromosphere. *The Astrophysical Journal*, 802(2):136, April 2015.

[97] K. D. Leka, G. Barnes, and A. Crouch. An Automated Ambiguity-Resolution Code for Hinode/SP Vector Magnetic Field Data. In B. Lites, M. Cheung, T. Magara, J. Mariska, and K. Reeves, editors, *The Second Hinode Science Meeting: Beyond Discovery-Toward Understanding*, volume 415 of *Astronomical Society of the Pacific Conference Series*, page 365. Astronomical Society of the Pacific, December 2009.

[98] H. W. Leung and J. Bovy. Deep Learning of Multi-element Abundances from High-resolution Spectroscopic Data. *Monthly Notices of the Royal Astronomical Society*, 483(3):3255–3277, 11 2018.

[99] M. Lieu, L. Conversi, B. Altieri, and B. Carry. Detecting Solar System Objects with Convolutional Neural Networks. *Monthly Notices of the Royal Astronomical Society*, 485(4):5831–5842, 03 2019.

[100] B. Lites, R. Casini, J. Garcia, and H. Socas-Navarro. A Suite of Community Tools for Spectro-polarimetric Analysis. *Memorie della Societa Astronomica Italiana*, 78:148, January 2007.

[101] C. Liu, W. Cao, J. Chae, K. Ahn, D. P. Choudhary, J. Lee, R. Liu, N. Deng, J. Wang, and H. Wang. Evolution of Photospheric Vector Magnetic Field Associated

with Moving Flare Ribbons as Seen by GST. *The Astrophysical Journal*, 869(1):21, December 2018.

[102] C. Liu, N. Deng, J. T. L. Wang, and H. Wang. Predicting Solar Flares Using SDO/HMI Vector Magnetic Data Products and the Random Forest Algorithm. *The Astrophysical Journal*, 843:104, July 2017.

[103] H. Liu, C. Liu, J. T. L. Wang, and H. Wang. Predicting Solar Flares Using a Long Short-term Memory Network. *The Astrophysical Journal*, 877(2):121, Jun 2019.

[104] H. Liu, C. Liu, J. T. L. Wang, and H. Wang. Predicting Coronal Mass Ejections Using SDO/HMI Vector Magnetic Data Products and Recurrent Neural Networks. *The Astrophysical Journal*, 890(1):12, February 2020.

[105] H. Liu, Y. Xu, J. Wang, J. Jing, C. Liu, J. T. L. Wang, and H. Wang. Inferring Vector Magnetic Fields from Stokes Profiles of GST/NIRIS Using a Convolutional Neural Network. *The Astrophysical Journal*, 894(1):70, May 2020.

[106] J. Liu, Y. Wang, X. Huang, M. B. Korsos, Y. Jiang, Y. Wang, and R. Erdelyi. Reliability of AI-generated Magnetograms from Only EUV Images. *Nature Astronomy*, 5:108110, 2021.

[107] Y. Liu, J. T. Hoeksema, P. H. Scherrer, J. Schou, S. Couvidat, R. I. Bush, T. L. Duvall, K. Hayashi, X. Sun, and X. Zhao. Comparison of Line-of-Sight Magnetograms Taken by the Solar Dynamics Observatory/Helioseismic and Magnetic Imager and Solar and Heliospheric Observatory/Michelson Doppler Imager. *Solar Physics*, 279(1):295–316, July 2012.

[108] R. E. Loughhead. High-Resolution Photography of the Solar Chromosphere. *Solar Physics*, 5(4):489–497, December 1968.

[109] S. F. Martin. Conditions for the Formation and Maintenance of Filaments (Invited Review). *Solar Physics*, 182(1):107–137, September 1998.

[110] M. L. McHugh. Interrater Reliability: the Kappa Statistic. *Biochemia Medica*, 22:276–282, 2012.

[111] T. R. Metcalf, K. D. Leka, and D. L. Mickey. Magnetic Free Energy in NOAA Active Region 10486 on 2003 October 29. *The Astrophysical Journal Letters*, 623(1):L53–L56, April 2005.

[112] D. Moore, W. Notz, and M. Fligner. *The Basic Practice of Statistics*. W.H. Freeman and Company, 2013.

[113] K. Mooroogen, R. J. Morton, and V. Henriques. Dynamics of internetwork chromospheric fibrils: Basic properties and magnetohydrodynamic kink waves. *Astronomy and Astrophysics*, 607:A46, November 2017.

[114] F. Moreno-Insertis, J. Martinez-Sykora, V. H. Hansteen, and D. Muñoz. Small-scale Magnetic Flux Emergence in the Quiet Sun. *The Astrophysical Journal Letters*, 859:L26, June 2018.

[115] D. Orozco Suárez, L. R. Bellot Rubio, J. C. del Toro Iniesta, S. Tsuneta, B. W. Lites, K. Ichimoto, Y. Katsukawa, S. Nagata, T. Shimizu, R. A. Shine, Y. Suematsu, T. D. Tarbell, and A. M. Title. Quiet-Sun Internetwork Magnetic Fields from the Inversion of Hinode Measurements. *The Astrophysical Journal Letters*, 670(1):L61–L64, November 2007.

[116] A. Ortiz, L. R. B. Rubio, V. H. Hansteen, J. de la Cruz Rodríguez, and L. R. van der Voort. Emergence of Granular-sized Magnetic Bubbles through the Solar Atmosphere. I. Spectropolarimetric Observations and Simulations. *The Astrophysical Journal*, 781(2):126, January 2014.

[117] E. Ostertagov. Modelling using Polynomial Regression. *Procedia Engineering*, 48:500 – 506, 2012. Modelling of Mechanical and Mechatronics Systems.

[118] W. Otruba. High Cadence Digital Full Disk Hα Patrol Device at Kanzelhöhe. In B. Schmieder, A. Hofmann, and J. Staude, editors, *Third Advances in Solar Physics Euroconference: Magnetic Fields and Oscillations*, volume 184 of *Astronomical Society of the Pacific Conference Series*, pages 314–318, September 1999.

[119] W. Otruba, H. Freislich, and A. Hanslmeier. Kanzelhöhe Photosphere Telescope (KPT). *Central European Astrophysical Bulletin*, 32:1–8, January 2008.

[120] E. N. Parker. Exploring the Innermost Solar Atmosphere. *Nature Astronomy*, 4:19–20, January 2020.

[121] C. Parnell. Nature of the Magnetic Carpet I. Distribution of Magnetic Fluxes. *Monthly Notices of the Royal Astronomical Society*, 335:389–398, 09 2002.

[122] K. Pearson. Notes on Regression and Inheritance in the Case of Two Parents. *Proceedings of the Royal Society of London*, 58(347-352):240–242, January 1895.

[123] W. D. Pesnell, B. J. Thompson, and P. C. Chamberlin. The Solar Dynamics Observatory (SDO). *Solar Physics*, 275(1-2):3–15, January 2012.

[124] S. B. Pikel'ner. Nature of the Fine Structure of the Middle Chromosphere. *Solar Physics*, 20(2):286–294, November 1971.

[125] J. E. Plowman and T. E. Berger. Calibrating GONG Magnetograms with End-to-end Instrument Simulation I: Background, the GONG Instrument, and End-to-end Simulation. *Solar Physics*, 295(10):143, 2020.

[126] W. M. Rand. Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[127] D. Rees, Y. Guo, A. L. Ariste, and J. Graham. Real Time Stokes Inversion Using Multiple Support Vector Regression. In M. G. Negoita, R. J. Howlett, and L. C. Jain, editors, *Knowledge-Based Intelligent Information and Engineering Systems, 8th International Conference, KES 2004, Wellington, New Zealand, September 20-25, 2004. Proceedings. Part III*, volume 3215 of *Lecture Notes in Computer Science*, pages 388–394. Springer, 2004.

[128] L. Rouppe van der Voort, J. Leenaarts, B. de Pontieu, M. Carlsson, and G. Vissers. On-disk Counterparts of Type II Spicules in the Ca II 854.2 nm and Hα Lines. *The Astrophysical Journal*, 705(1):272–284, November 2009.

[129] B. Ruiz Cobo and J. C. del Toro Iniesta. Inversion of Stokes Profiles. *The Astrophysical Journal*, 398:375, October 1992.

[130] T. Samanta, H. Tian, V. Yurchyshyn, H. Peter, W. Cao, A. Sterling, R. Erdélyi, K. Ahn, S. Feng, D. Utz, D. Banerjee, and Y. Chen. Generation of Solar Spicules and Subsequent Atmospheric Heating. *Science*, 366(6467):890–894, November 2019.

[131] C. Sammut and G. I. Webb, editors. *Mean Absolute Error*, pages 652–652. Springer US, Boston, MA, 2010.

[132] T. Schad. Automated Spatiotemporal Analysis of Fibrils and Coronal Rain Using the Rolling Hough Transform. *Solar Physics*, 292(9):132, September 2017.

[133] P. H. Scherrer, R. S. Bogart, R. I. Bush, J. T. Hoeksema, A. G. Kosovichev, J. Schou, W. Rosenberg, L. Springer, T. D. Tarbell, A. Title, C. J. Wolfson, I. Zayer, and MDI Engineering Team. The solar oscillations investigation - Michelson Doppler Imager. *Solar Physics*, 162(1-2):129–188, December 1995.

[134] P. H. Scherrer, J. Schou, R. I. Bush, A. G. Kosovichev, R. S. Bogart, J. T. Hoeksema, Y. Liu, T. L. Duvall, J. Zhao, A. M. Title, C. J. Schrijver, T. D. Tarbell, and S. Tomczyk. The Helioseismic and Magnetic Imager (HMI) investigation for the Solar Dynamics Observatory (SDO). *Solar Physics*, 275(1-2):207–227, January 2012.

[135] C. J. Schrijver and M. L. De Rosa. Photospheric and Heliospheric Magnetic Fields. *Solar Physics*, 212:165–200, January 2003.

[136] C. J. Schrijver, M. L. De Rosa, A. M. Title, and T. R. Metcalf. The Nonpotentiality of Active-Region Coronae and the Dynamics of the Photospheric Magnetic Field. *The Astrophysical Journal*, 628(1):501–513, July 2005.

[137] C. J. Schrijver, M. L. DeRosa, T. Metcalf, G. Barnes, B. Lites, T. Tarbell, J. McTiernan, G. Valori, T. Wiegelmann, M. S. Wheatland, T. Amari, G. Aulanier, P. Démoulin, M. Fuhrmann, K. Kusano, S. Régnier, and J. K. Thalmann. Nonlinear Force-free Field Modeling of a Solar Active Region around the Time of a Major Flare and Coronal Mass Ejection. *The Astrophysical Journal*, 675(2):1637–1644, March 2008.

[138] A. Sen and M. Srivastava. *Regression Analysis*. Springer-Verlag New York, 1990.

[139] E. Shelhamer, J. Long, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(04):640–651, April 2017.

[140] S. Shumko, N. Gorceix, S. Choi, A. Kellerer, W. Cao, P. R. Goode, V. Abramenko, K. Richards, T. R. Rimmele, and J. Marino. AO-308: the High-order Adaptive Optics System at Big Bear Solar Observatory. In E. Marchetti, L. M. Close, and J.-P. Vran, editors, *Adaptive Optics Systems IV*, volume 9148 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 914835, August 2014.

[141] H. Socas-Navarro, J. de la Cruz Rodríguez, A. Asensio Ramos, J. Trujillo Bueno, and B. Ruiz Cobo. An Open-source, Massively Parallel Code for Non-LTE Synthesis and Inversion of Spectral Lines and Zeeman-induced Stokes Profiles. *Astronomy and Astrophysics*, 577:A7, May 2015.

[142] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(1):19291958, January 2014.

[143] X. Sun, J. T. Hoeksema, Y. Liu, T. Wiegelmann, K. Hayashi, Q. Chen, and J. Thalmann. Evolution of Magnetic Field and Energy in a Major Eruptive Active Region Based on SDO/HMI Observation. *The Astrophysical Journal*, 748:77, April 2012.

[144] R. Taujale, Z. Zhou, W. Yeung, K. W. Moremen, S. Li, and N. Kannan. Mapping the Glycosyltransferase Fold Landscape Using Interpretable Deep Learning. *Nature Communications*, 12(1):5656, 2021.

[145] F. Teng and Y.-Y. Deng. Some Tests and Improvements to the VFISV: Very Fast Inversion of the Stokes Vector for the Helioseismic and Magnetic Imager. *Research in Astronomy and Astrophysics*, 14(11):1469–1480, November 2014.

[146] T. Török and B. Kliem. Confined and Ejective Eruptions of Kink-unstable Flux Ropes. *The Astrophysical Journal Letters*, 630:L97–L100, September 2005.

[147] S. E. Umbaugh. *Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIPtools, Second Edition*. CRC Press, Inc., Boca Raton, FL, USA, 2nd edition, 2010.

[148] R. Unnikrishnan, C. Pantofaru, and M. Hebert. A Measure for Objective Evaluation of Image Segmentation Algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2005, San Diego, CA, USA, 21-23 September, 2005*, page 34. IEEE Computer Society, 2005.

[149] J. Varsik, C. Plymate, P. Goode, A. Kosovichev, W. Cao, R. Coulter, K. Ahn, N. Gorceix, and S. Shumko. Control and operation of the 1.6 m New Solar Telescope in Big Bear. In *Ground-based and Airborne Instrumentation for Astronomy V*, volume 9147 of *Proceedings of the SPIE*, page 91475D, August 2014.

[150] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. . . Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[151] H. Wang, W. Cao, C. Liu, Y. Xu, R. Liu, Z. Zeng, J. Chae, and H. Ji. Witnessing Magnetic Twist with High-resolution Observation from the 1.6-m New Solar Telescope. *Nature Communications*, 6:7008, April 2015.

[152] H. Wang, V. Yurchyshyn, C. Liu, K. Ahn, S. Toriumi, and W. Cao. Strong Transverse Photosphere Magnetic Fields and Twist in Light Bridge Dividing Delta Sunspot of Active Region 12673. *Research Notes of the AAS*, 2(1):8, January 2018.

[153] J. Wang, W. Li, C. Denker, C. Lee, H. Wang, P. R. Goode, A. McAllister, and S. F. Martin. Minifilament Eruption on the Quiet Sun. I. Observations at H-alpha Central Line. *The Astrophysical Journal*, 530(2):1071–1084, February 2000.

[154] J. Wang, C. Liu, N. Deng, and H. Wang. Evolution of Photospheric Flow and Magnetic Fields Associated with the 2015 June 22 M6.5 Flare. *The Astrophysical Journal*, 853(2):143, February 2018.

[155] X. Wang, Y. Chen, G. Toth, W. B. Manchester, T. I. Gombosi, A. O. Hero, Z. Jiao, H. Sun, M. Jin, and Y. Liu. Predicting Solar Flares with Machine Learning: Investigating Solar Cycle Dependence. *The Astrophysical Journal*, 895(1):3, May 2020.

[156] M. S. Wheatland, P. A. Sturrock, and G. Roumeliotis. An Optimization Approach to Reconstructing Force-free Fields. *The Astrophysical Journal*, 540:1150–1155, September 2000.

[157] T. Wiegelmann. Optimization Code with Weighting Function for the Reconstruction of Coronal Magnetic Fields. *Solar Physics*, 219:87–108, January 2004.

[158] T. Wiegelmann, J. K. Thalmann, C. J. Schrijver, M. L. De Rosa, and T. R. Metcalf. Can We Improve the Preprocessing of Photospheric Vector Magnetograms by the Inclusion of Chromospheric Observations? *Solar Physics*, 247(2):249–267, February 2008.

[159] T. Wiegelmann, J. K. Thalmann, C. J. Schrijver, M. L. De Rosa, and T. R. Metcalf. Can We Improve the Preprocessing of Photospheric Vector Magnetograms by the Inclusion of Chromospheric Observations? *Solar Physics*, 247:249–267, February 2008.

[160] M. Wilhelm, D. P. Zolg, M. Graber, S. Gessulat, T. Schmidt, K. Schnatbaum, C. Schwencke-Westphal, P. Seifert, N. de Andrade Krätzig, J. Zerweck, T. Knaute, E. Bräunlein, P. Samaras, L. Lautenbacher, S. Klaeger, H. Wenschuh, R. Rad, B. Delanghe, A. Huhmer, S. A. Carr, K. R. Clauser, A. M. Krackhardt, U. Reimer, and B. Kuster. Deep Learning Boosts Sensitivity of Mass Spectrometry-based Immunopeptidomics. *Nature Communications*, 12(1):3346, 2021.

[161] C. J. Willmott and K. Matsuura. Advantages of the Mean Absolute Error (MAE) Over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research*, 30(1):79–82, 2005.

[162] F. Wöger, O. von der Lühe, and K. Reardon. Speckle Interferometry With Adaptive Optics Corrected Solar Data. *Astronomy and Astrophysics*, 488:375–381, September 2008.

[163] J. F. Wu and S. Boada. Using Convolutional Neural Networks to Predict Galaxy Metallicity from Three-colour Images. *Monthly Notices of the Royal Astronomical Society*, 484(4):4683–4694, 02 2019.

[164] Y. Xiao and W. Y. Wang. Quantifying Uncertainties in Natural Language Processing Tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):7322–7329, Jul. 2019.

[165] S. B. Xu, S. Y. Huang, Z. G. Yuan, X. H. Deng, and K. Jiang. Prediction of the Dst Index with Bagging Ensemble-learning Algorithm. *The Astrophysical Journal Supplement Series*, 248(1):14, May 2020.

[166] Y. Xu, W. Cao, K. Ahn, J. Jing, C. Liu, J. Chae, N. Huang, N. Deng, D. E. Gary, and H. Wang. Transient Rotation of Photospheric Vector Magnetic Fields Associated with a Solar Flare. *Nature Communications*, 9:46, January 2018.

[167] Y. Xu, W. Cao, M. Ding, L. Kleint, J. Su, C. Liu, H. Ji, J. Chae, J. Jing, K. Cho, K. Cho, D. Gary, and H. Wang. Ultra-narrow Negative Flare Front Observed in Helium-10830 Å Using the 1.6 m New Solar Telescope. *The Astrophysical Journal*, 819(2):89, March 2016.

[168] H. Zhao, J. Jia, and V. Koltun. Exploring Self-attention for Image Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085, 2020.

[169] C. Zwaan. Elements and Patterns in the Solar Magnetic Field. *Annual Review of Astronomy and Astrophysics*, 25:83–111, January 1987.