

PROGRESS REPORT: NEURAL NETWORKS FOR COMPUTED TOMOGRAPHY IMAGING SPECTROSCOPY OF THE SOLAR ATMOSPHERE

Roy Smart
roy.smart@montana.edu
Montana State University, Department of Physics
Bozeman, MT 59717, USA

March 9, 2018

1 Introduction

The goal outlined in my NESSF17 proposal was to classify explosive events (EEs) in the solar transition region (TR) according to their 2D spatial structure. I proposed to accomplish this goal using two instruments optimized towards observing explosive events, the *Multi-order Solar EUV Spectrograph* (MOSES), and the *EUV Snapshot Imaging Spectrograph* (ESIS). These instruments are members of a class of spectrographs known as snapshot imaging spectrographs, which can spectrally-resolve a 2D field-of-view (FOV) in a single snapshot, as opposed to imaging spectrographs, such as *Interface Region Imaging Spectrograph* (IRIS), which can only spectrally-resolve a 2D FOV through rastering. Snapshot imaging spectrographs are needed to adequately characterize the spatial structure of EEs, since the EEs evolve faster than the rastering timescale of IRIS.

MOSES and ESIS are a type of snapshot imaging spectrograph known as a computed tomography imaging spectroscopy (CTIS), which takes a snapshot of a 2D FOV by simply removing the slit used by imaging spectrographs. Without the slit, space and spectrum become confused, and so images are taken at several diffraction orders/angles to provide more information about the scene. These images can be combined to form a spectrally-resolved 2D scene using computed tomography (CT) inversion algorithms, hence the name CTIS. Spectrally-resolving MOSES and ESIS data is an ill-posed inversion problem because they only image in a limited number (3-6) of diffraction orders/angles.

A brute-force approach to addressing this ill-posed inversion problem would be to use a forward model of a MOSES/ESIS and an a priori model of solar TR spectra to create a dictionary of spectral structures and their corresponding signature on the CTIS detectors. This would allow observations to be matched to entries in this dictionary to recover a spectrally-resolved scene. I am developing a convolutional neural network (CNN)-based procedure, that uses IRIS *iv* 1403 Å observations as a model of the solar TR, as an approximation to the above brute-force approach, which will spectrally-resolve the observations gathered by MOSES and ESIS.

I have completed a simple realization of a CTIS inversion neural network (CINN) for the MOSES instrument, the Doppler inversion network (DIN), which reconstructs the Doppler-shift of the spectral line profile. We found that our implementation of a DIN was more accurate at large Doppler velocities than existing CT algorithms.

2 Progress Report

While the development of inversion routines is the most discussed milestone of this investigation, assembling the MOSES and ESIS instruments for flight in 2019 is an important goal of my research group. My most significant objective in the instrument assembly is the optical alignment and focus of both ESIS and MOSES.

This effort has provided much needed experience using the instruments in this study, which will allow us to be more effective in the development of our inversion routines.

Parallel to the advancement of MOSES and ESIS, I was also involved in the design phase of the FURST mission, which was just accepted in this latest LCAS proposal round. The design process of FURST gave me useful experience using optical modeling software, which will allow for the implementation of more sophisticated instrument models in our inversion routine.

2.1 CTIS Inversion Neural Networks

Since the ultimate goal of solar spectroscopy is the determination of plasma parameters such as bulk velocity, temperature, density, etc., we don't necessarily need to recover the spectrum from MOSES observations, just the derived physical quantities that we're interested in. Finding a few physical quantities is easier than reconstructing the entire spectrum since they contain less information, easing the ill-posedness of the inversion problem. Contrary to the plan outlined in the NESSF17 proposal, we decided to first construct a neural network that would derive the bulk velocity of TR plasma by finding the central-tendency (doppler shift) of an input spectrum from a MOSES observation.

This branch from the proposal was temporarily taken since we wanted to be sure that we could solve this simple problem before putting too much effort into a full spectrum inversion. From the development of our DIN, we were able to see the necessity of a high-performance despiking algorithm for the input dataset. We have nearly completed a pixel-identification module of our despiking algorithm, and are planning a simple implementation of a pixel replacement module.

2.1.1 Doppler Inversion Network

Our DIN is a sophisticated 21x21 kernel that is convolved with MOSES observations to yield the Doppler shift at each pixel. The convolutional kernel is learned through training on MOSES observations where the Doppler shift at each pixel is known a priori. This a priori dataset is formed by using the IRIS Si IV 1403 Å spectral line as model for the He II 304 Å line observed by MOSES. The Doppler shift is taken to be the mean along the spectral dimension of the IRIS observations in a -150 to 150 km/s window centered around the Si IV 1403 Å spectral line.

During the development of our DIN, we tested two neural networks to understand how increasing the complexity of the network affects the quality of the results. The first network was dubbed the "test" network, and contained relatively few free parameters, while the "Final" network had many more free parameters, to increase the predictive power of the network. In Table 1, we have provided both parameters and results of both networks. Here, we can see that both networks were able to achieve an RMS velocity error better than the theoretical resolution of MOSES (29 km/s).

	Test	Final
Free Parameters	1.2k	836k
Training Images	4.5k	4.5k
Training time (min)	3	15
RMS error (km/s)	12.7	10.3
Pearson's r	0.500	0.701

Table 1: Description of the characteristics and results of two neural networks tested for this progress report. More comprehensive results from the "Final" network are presented in later figures.

In Figure 1 we have provided a few validation examples of our method, where we have plotted the reconstructed Doppler velocity along with the "true" velocity for comparison. We chose to show these validation examples since they demonstrated reasonably-high Doppler velocities, characteristic of some types of EEs. We can see that in areas of high signal the network does reasonably well at reproducing the true Doppler shift, while in areas of low signal the Doppler shift is underestimated. We anticipate that the despiking procedure outlined in Section 2.1.2 will improve results in these low-signal areas, since spikes are more effective at changing the estimated Doppler shift in these areas.

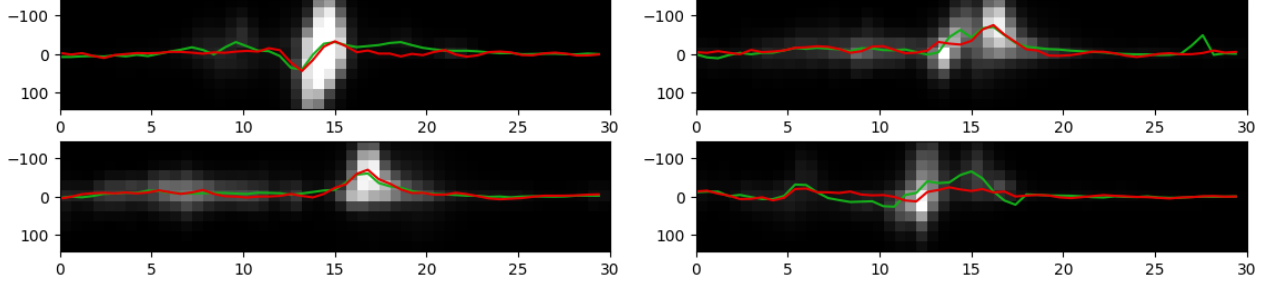


Figure 1: Examples of Doppler inversions. Each image is an IRIS Si IV 1403 Å spectra rebinned into MOSES resolution. The vertical axis is in km/s and the horizontal axis is in arcsec. The true line center is plotted in orange, and the reconstructed line center is plotted in blue.

In Figure 2b we constructed a histogram of the true velocity vs. the reconstructed velocity for every pixel in our validation dataset to demonstrate the statistical effectiveness of our method. If the DIN reconstruction were perfect, the result of this plot would be a straight line with slope unity (plotted in blue). Immediately, we can see that the reconstruction is not flawless. There is an obvious spread about the line of perfect reconstruction, biased towards slower speeds.

For comparison with Figure 2b we have provided an analogous image (Figure 2a) prepared using the smooth multiplicative algebraic reconstruction technique (SMART), the de-facto standard algorithm for inverting MOSES observations. This image is a scatterplot constructed from a single SUMER O III 703.87 Å raster. Since Figure 2a is only constructed from a single raster, it contains much less samples than Figure 2b, and therefore the two should not be compared directly. However, SMART’s underestimation of velocity is a well-studied phenomenon, so we know that Figure 2a reasonably demonstrates SMART’s capabilities.

With the above considerations we can see that according to Figure 2 the DIN is much less likely to underestimate the velocity than this implementation of SMART. However these results are still preliminary and obviously need to be verified by applying SMART to the same IRIS dataset used to validate the DIN.

A quantitative test that we performed on Figure 2b was the calculation of Pearson’s r , a test of linearity. The result, enumerated in Table 1 as $r = 0.701$, we interpret as modest performance, likely polluted by spikes in the input dataset.

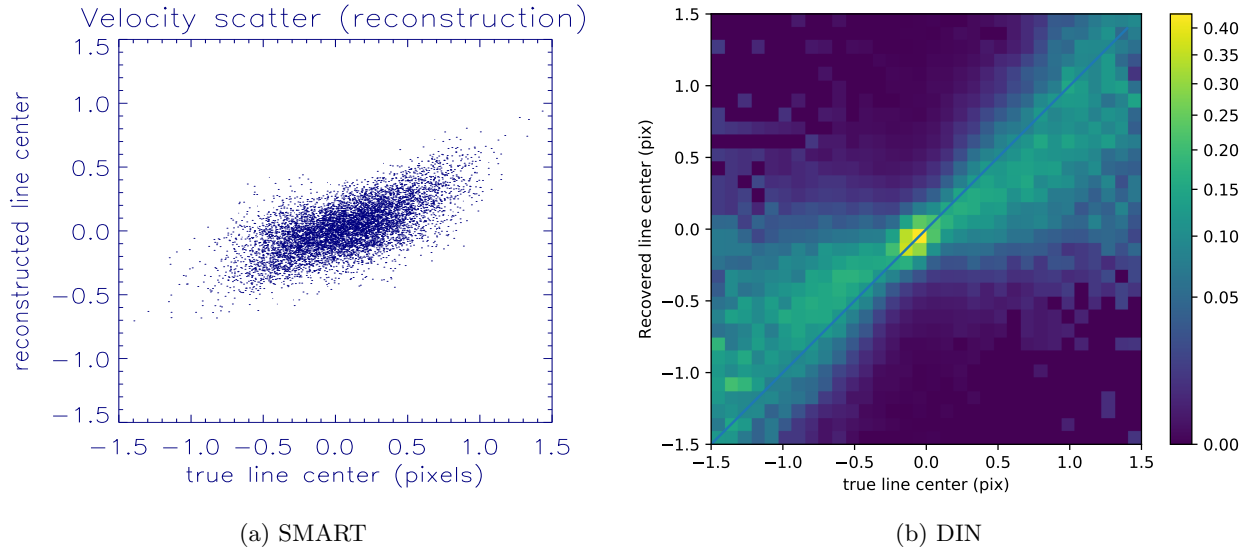


Figure 2: Reconstructed vs. true Doppler velocity using both the standard SMART algorithm (applied to SUMER O III 703.87 Å raster; courtesy of Charles Kankelborg), and the CINN algorithm (applied to IRIS Si IV 1403 Å.)

2.1.2 Despiking Training Data

Spikes in IRIS observations are a stochastic process, primarily due to ionospheric particles impacting the CCD detectors. Removal of spikes in our training dataset is important since the network will become distracted trying to reconstruct the spikes, a futile undertaking. This was not considered to be a serious problem during the proposal phase, as there have been several despiking routines written for IRIS. However we failed to appreciate how difficult it would be to apply these routines in an automated fashion to a large section of the IRIS Si IV 1403 Å dataset.

There are a multitude of despiking routines available on the IDL SolarSoft libraries, such as `nospike.pro`, `array_despike.pro`, `iris_prep_despike.pro`. Unfortunately they all identify spikes using the same method: convolution of some kernel with an image to estimate a local mean and standard deviation, and a hard threshold to exclude pixels some number of standard deviations above the mean. This method is often too aggressive in areas of high signal intensity and not aggressive enough in areas of low signal intensity. In Figure 3, we can see an example of this behavior: the explosive event in the bottom row has become eroded from `iris_prep_despike.pro`, while many of the spikes in the top row are only partially identified. Considering this behavior, to use these procedures we found that we would have to manually tune them for each observation, which would become prohibitive for a training dataset composed of many IRIS observations.

Instead of estimating the mean and standard deviation, our method uses a median-percentile based approach, where pixels are marked as spikes if their value is larger than 99% (for example) of all other pixels with the same median. This is accomplished by calculating a local median for every pixel in the observation, and then constructing a histogram of median vs. pixel value for each observation. From this histogram, we can then determine the bad pixel threshold as a function of the local median. Finally, this procedure was performed independently along each axis (wavelength, space, time) and we required that a pixel must be above the threshold for all three axes to be marked as a spike. A visualization of this procedure is presented in Figure 4, where we can see the threshold used to perform the despiking seen in Figure 3. We have developed a GPU implementation of this despiking procedure, resulting in code that is at least 10x faster than current methods while being much more discriminatory.

2.2 FURST Design

3 Proposal for Renewal

3.1 MOSES Inversion GAN

3.2 Explosive Event Classification

4 Conclusion

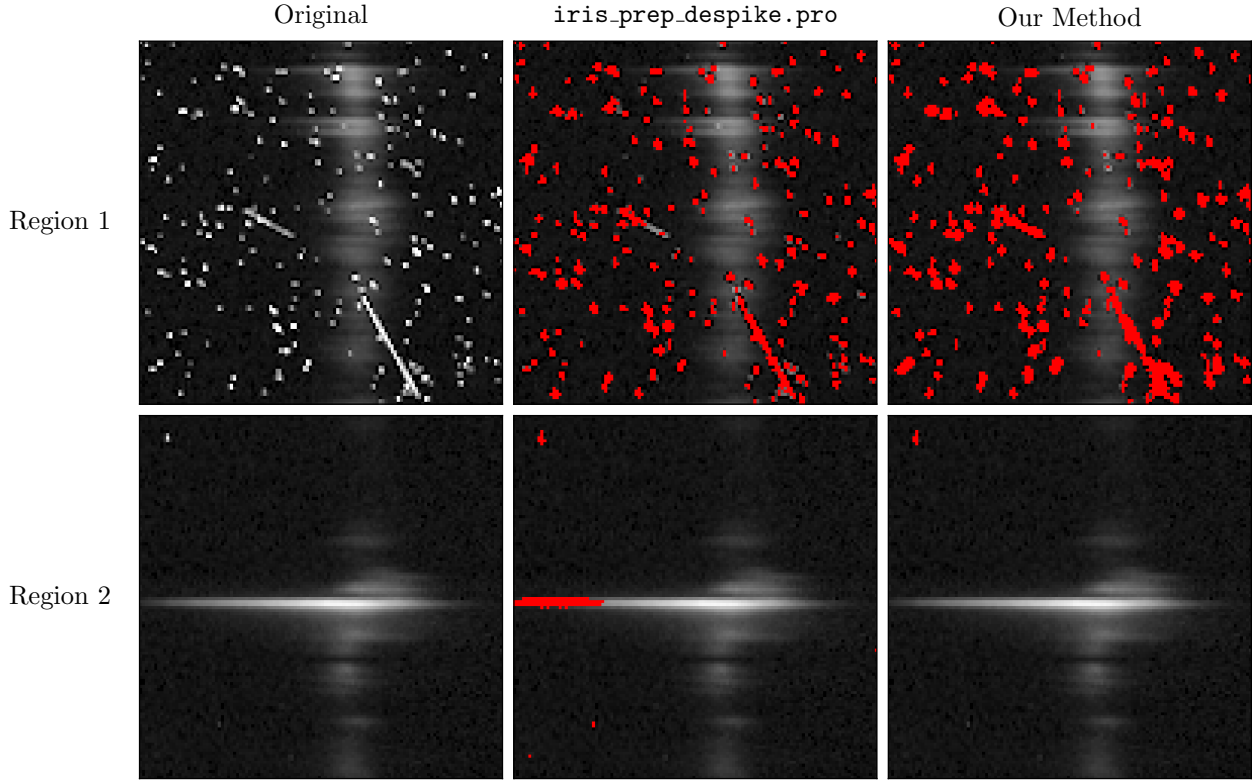


Figure 3: An example of both the standard despiking algorithm and our spike identification algorithm applied to an IRIS Si IV 1403 Å observation gathered at 07:24:26 on 06-16-2015. Pixels containing spikes are marked red by each algorithm. We present cutouts from frames 24 and 50 to demonstrate how susceptible each algorithm is to false spike identifications. The frame on the top row was taken inside the SAA, and provides examples of many types of spikes intended to show the false negatives identified by each algorithm. The frame on the bottom row is an example of an EE, and serves as an example of the false positives identified by each algorithm.

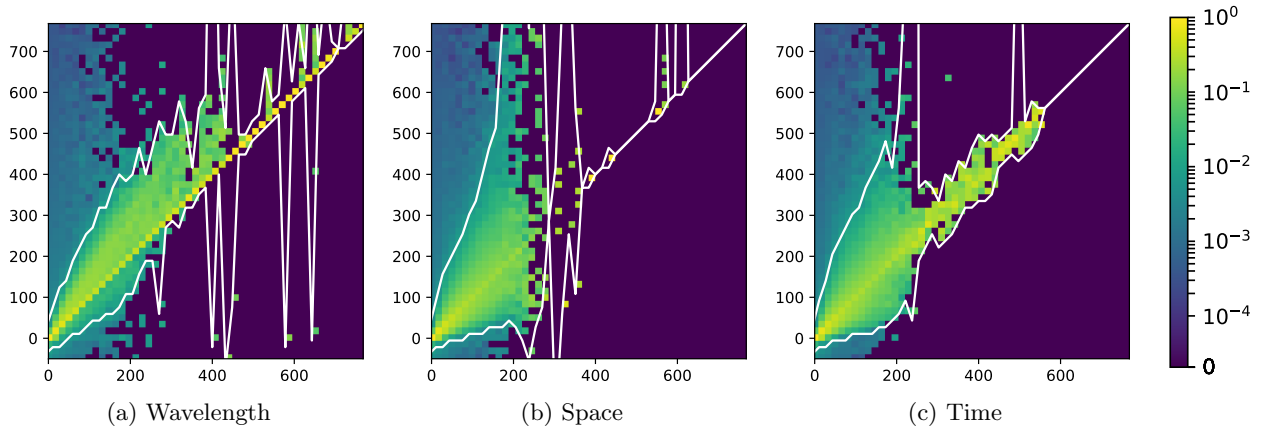


Figure 4: Column-normalized histogram along each axis for the observation in Figure 3. Each column has been divided by its total to understand the distribution of values about a particular median. The 1% and 99% thresholds have been plotted in white. Horizontal and vertical axes are in DN.

ID		Task Mode	Task Name	Duration	Start	Finish																												
1			NESSF18 Renewal Due Date	0 days	Thu 3/15/18	Thu 3/15/18																												
2			NESSF19 Renewal Due Date	0 days	Fri 3/15/19	Fri 3/15/19																												
3			NESSF17 End Date	0 days	Sat 9/1/18	Sat 9/1/18																												
4			NESSF18 End Date	0 days	Sun 9/1/19	Sun 9/1/19																												
5			MOSES Doppler Inversion CNN	170 days	Thu 3/15/18	Wed 11/7/18																												
6			Development	90 days	Thu 3/15/18	Wed 7/18/18																												
7			Complete Despiking Procedures	1 mon	Thu 3/15/18	Wed 4/11/18																												
8			IRIS Level 2 Despiking	2 wks	Thu 4/12/18	Wed 4/25/18																												
9			DIN Development/Testing	1 mon	Thu 4/26/18	Wed 5/23/18																												
10			MOSES Data Preparation	1 mon	Thu 5/24/18	Wed 6/20/18																												
11			MOSES Doppler Inversion	2 wks	Thu 6/21/18	Wed 7/4/18																												
12			Validation Against SMART	2 wks	Thu 7/5/18	Wed 7/18/18																												
13			Publication	80 days	Thu 7/19/18	Wed 11/7/18																												
14			Writing	2 mons	Thu 7/19/18	Wed 9/12/18																												
15			Peer Review	1 mon	Thu 9/13/18	Wed 10/10/18																												
16			Response to Review	1 mon	Thu 10/11/18	Wed 11/7/18																												
17			MOSES Inversion GAN	140 days	Thu 11/8/18	Wed 5/22/19																												
18			Development	60 days	Thu 11/8/18	Wed 1/30/19																												
19			GAN Research	2 wks	Thu 11/8/18	Wed 11/21/18																												
20			MIG Development/Validation	2 mons	Thu 11/22/18	Wed 1/16/19																												
21			MOSES Spectral Inversion	2 wks	Thu 1/17/19	Wed 1/30/19																												
22			Publication	80 days	Thu 1/31/19	Wed 5/22/19																												
23			Writing	2 mons	Thu 1/31/19	Wed 3/27/19																												
24			Peer Review	1 mon	Thu 3/28/19	Wed 4/24/19																												
25			Response to Review	1 mon	Thu 4/25/19	Wed 5/22/19																												
26			EE Classifier	140 days	Thu 5/23/19	Wed 12/4/19																												
27			Development	60 days	Thu 5/23/19	Wed 8/14/19																												
28			Classifier Research	2 wks	Thu 5/23/19	Wed 6/5/19																												
29			EEC Development	2 mons	Thu 6/6/19	Wed 7/31/19																												
30			Compare Results to Literature	2 wks	Thu 8/1/19	Wed 8/14/19																												
31			Publication	80 days	Thu 8/15/19	Wed 12/4/19																												
32			Writing	2 mons	Thu 8/15/19	Wed 10/9/19																												
33			Peer Review	1 mon	Thu 10/10/19	Wed 11/6/19																												
34			Response to Review	1 mon	Thu 11/7/19	Wed 12/4/19																												