# MOSES Data Inversion With Convolutional Neural Networks

Roy Smart

Montana State University
roytsmart@gmail.com

## I. Introduction

The Multi-Order Solar EUV Spectrograph (MOSES) is a unique instrument that captures solar images in three diffraction orders: $m = 0, +1$, and $-1$. Unlike most spectrographs, MOSES is not equipped with a slit to restrict the field of view in the direction of dispersion. This advantage of this configuration is that spectral and spatial information is simultaneously viewed by the instrument, allowing interesting solar features to be more easily identified. However this lack of spatial restriction by a slit means that spectral and spatial information are convolved together to form the images captured by MOSES. These images are known as "overlappograms."
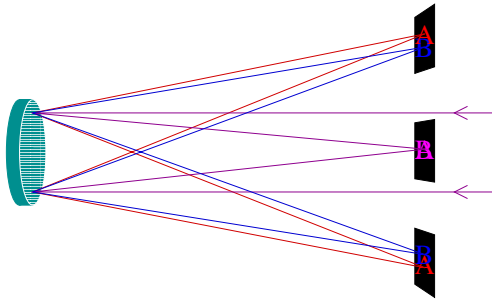


**Figure 1:** *Layout of the MOSES instrument demonstrating how different wavelengths are overlapped onto each detector.*

The main objective of the MOSES instrument is to determine Doppler shifts of the structures observed in the solar transition region. To find this quantity, we must perform what we call an "inversion." This is best described as taking the 2-dimensional spectral and spatial information from each of the three detectors and constructing a spectral "cube" in three di-

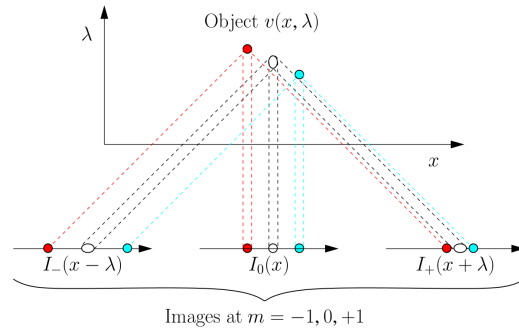mensions, with two spatial dimensions and one spectral dimension.



**Figure 2:** *Diagram demonstrating how compact objects in the data cube are convolved to form the images at each detector [1].*

The main challenge with inverting the MOSES data is that it is an obviously ill-posed problem, i.e. the spectral cube contains more information than is provided by the instrument. This is mathematically described using the expression

$$I_m(x', y') = \int_B v(x' - m\lambda, y', \lambda) d\lambda \qquad (1)$$

Where $I(x', y')$ is the intensity measured by the instrument $v(x', y', \lambda)$ is an object located in the spectral cube, $m$ is the spectral order, $(x', y')$ are the detector coordinates, and the domain $B$ is the passband of the instrument.

Since Equation (1) doesn't have a unique solution, there are many possible cubes that could produce the same images captured by MOSES. The task then, is to try and use what we know about physics to try and trim down the large number of potential data cubes in an attempt to learn something about the solar transition region.

## II. Motivation

Many researchers over the history of the MOSES research project have developed computational tools to solve the inversion problem. These methods are discussed in detail in the literature and include Smoothed Multiplicative Algebraic Reconstruction Technique (SMART)[2], Fourier backprojection and pixon inversion [3].

These methods have their advantages and disadvantages, but in general they have been successful in revealing the interior structure of the transition region [1]. However there is still room for progress to be made towards full inversions of the entire MOSES data stack. These algorithms have a tendency to produce what is known as "plaid." This is where bright objects on the spectral cube tend to be smeared across the direction of the spectral projections, producing decidedly unphysical inversions.
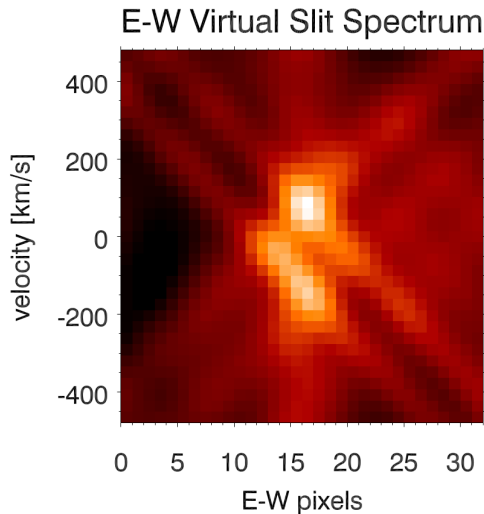


**Figure 3:** *An example of plaid. Notice how the bright pixels in the center are smeared diagonally across the image. This is an example of an unphysical inversion. [4]*

Determining an inversion method that is resistant to plaid would be very beneficial to Kankelborg's research group, as inversions would contain more viable scientific data.

## III. Proposed Inversion Method

### I. Machine Learning

Thus far, all of the attempts to invert the MOSES data have relied on traditional, human-designed algorithms to carry out the computation. However, in the past few years, great strides have been made in machine learning, where a computer program is presented with a problem and its solution and then "trained" to solve the problem using a large number of problem/solution examples.

I propose that a method based off of machine learning might allow some progress to be made towards the goal of plaid-resistant MOSES inversions. This is made possible by the fact that the learning algorithm could be penalized for producing plaid solutions to the inversion problem and rewarded for producing physical solutions.

While all this machine learning sounds fantastic, there is still a great deal of challenge to the problem. First, the structure of the learning algorithm is very important, and unlike traditional algorithms, the right structure can only be found by trial and error. Furthermore, the learning process often takes a large number of examples to converge on the solution. Solving these problems will be discussed in the following pages.

### II. Artificial Neural Networks

Artificial neural networks (ANNs) are machine learning algorithms that are modeled off of the structure of the human brain. ANNs can be described as a system of "neurons" that can communicate with each other. The behavior of each neuron is dictated by an activation function. The activation function maps neuron inputs to outputs, e.g. determines when the neuron is *active*. Active neurons propagate information through the network while dormant neurons restrict it.

Communication is facilitated through a series of connections between the neuron, where each connection may have a different weight representing the amount of influence one neu-

ron may have on another. The weights are tuned based off of experience and taken with the activation function, allow the neural network to learn.

The neurons are usually organized into layers, where each layer can be interpreted as a different computational step used to solve the problem. The first layer is known as the input layer, and the number of input neurons corresponds to the amount of information supplied to the network. The output layer is the last layer, and the number of output neurons equals the amount of information to be expected in the solution of the problem being solved. In between the input and output layers, there is a number of hidden layers used to compute the solution. The amount of hidden layers and the number of neurons per hidden layer depends on the complexity of the problem being solved.
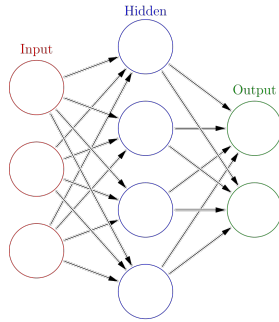


**Figure 4:** *An example of a simple neural network.*

Using this simple idea, many different species of neural networks may be constructed. The choices of network depth, number of connections, and activation function are examples of what are called *hyperparameters*, and determine the behavior of the network.

## III.   Convolutional Neural Networks

Vanilla ANNs usually connect each neuron in one layer to every other neuron in the preceding layer. This configuration is sufficient when the number of input neurons is small. However, when the number of input neurons grows as

large as an image, for example, the huge number of free parameters quickly leads to overfitting. Furthermore, the Vanilla ANN treats pixels with large spatial separation equally, which is often undesirable for image processing.

Convolutional Neural Networks (CNNs) solve this problem by connecting only a small number of the input neurons (known as the receptive field) to the layers below, known as convolutional layers.
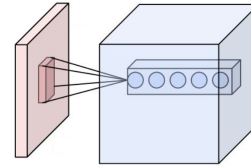


**Figure 5:** *The red box represents the input neurons, with the receptive field connected to the next five layers.*

The receptive fields are tiled such that they overlap one another. Overlapping receptive fields, taken with parameter sharing with the next layer in the network provides translational invariance and amounts to a convolution operation. There are many other layers that go into a CNN that will not be discussed here, but they include: pooling, RELU, and fully connected layers. These layers serve to increase the computational power and decrease the training time of the network.

CNNs have been used successfully in many fields, most notably in image recognition [5]. We think that this type of neural network will be best suited to MOSES data inversion because spectral features in MOSESChoosing images are separated spatially, and CNNs are designed to detect spatial features in images.

## IV.   IMPLEMENTATION

## I.   Selecting Hyperparameters

## II.   Training Dataset

## III.   Software

## References

[1] Charles C. Kankelborg J. Lewis Fox and Roger J. Thomas. A transition region explosive event observed in He II with the moses sounding rocket. *The Astrophysical Journal*, 719:1132–1143, August 2010. `http://solar.physics.montana.edu/MOSES/papers/2010/FoxKankelborgThomas2010.pdf`.

[2] T. Rust, L. Fox, C. Kankelborg, H. Courrier, and J. Plovanic. MOSES Inversions using Multiresolution SMART. 224:414.06, June 2014.

[3] J. L. Fox, C. C. Kankelborg, and T. R. Metcalf. Data inversion for the Multi-Order Solar Extreme-Ultraviolet Spectrograph. 5157:124–132, November 2003.

[4] T. Rust and C Kankelborg. Imaging spectroscopy with moses sounding rocket data. *SPD*, 2015.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. `http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf`.