Processing Solar Images to forecast Coronal Mass Ejections using Artificial Intelligence

Savvas Raptis[1,2], Jorge Amaya[2], Adam Shamash[2], Gilles Depypere[2], Giovanni Lapenta[2]

[1]KTH Royal Institute of Technology, Department of Space and Plasma Physics
[2]KU Leuven, Centre for mathematical Plasma Astrophysics

Introduction

Energetic activity on the Sun influences space missions, satellites, airplanes, electrical/electronic devices and grid networks. The origin of these disastrous effects is the phenomenon of Coronal Mass Ejection (CME).

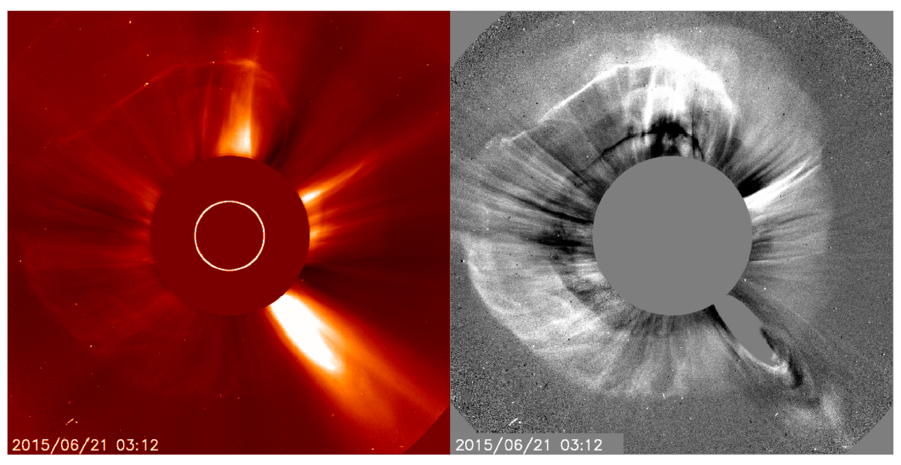


Figure 1: Images of the Sun showing a halo CME on 20/6/2015. Figure Courtesy: ESA & NASA SOHO.

CME is as an extensive structure of magnetized plasma which propagates away from the Sun into space, driven by magnetic forces. A specific subcategory of CMEs are the halo CMEs that propagate in the direction of the Earth.

Method – Neural Networks

Neural Network (NN) is specialized machine learning algorithm "trained" to perform a specific task. The training is being done by introducing numerous data several times to the NN and by then optimizing the NN's parameters according to these examples ("back propagation"). The basic idea behind neural networks is that after parametrizing a network to classify known data, the network can be used for prediction, given new unknown information.

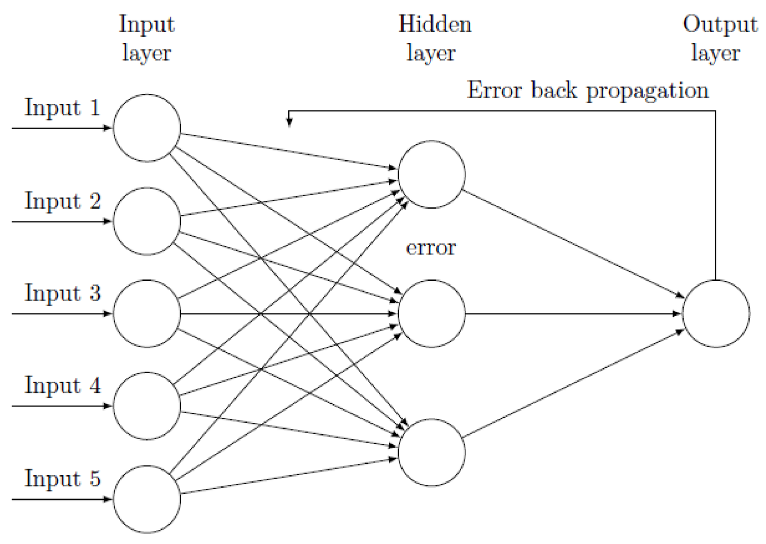


Figure 2: Neural Network (NN) and visualization of back propagation.

Convolution Neural Network (CNN) is an advanced neural network that works ideally when dealing with images. It consists of different layers and parameters that try to obtain features originating from the input images in order to perform a specific task.

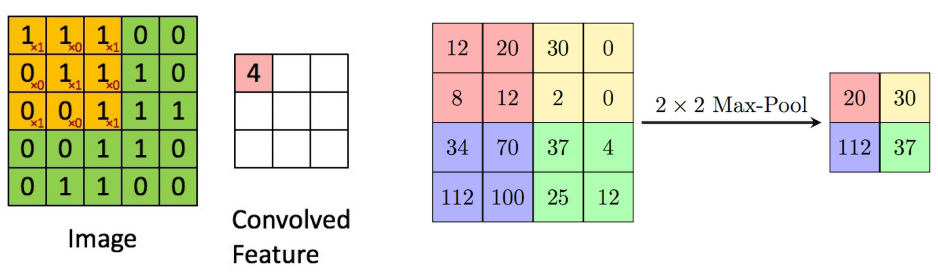


Figure 3: [Left]: Visualization of Convolution layer. [Right] Visualization of Max pooling/Subsampling layer. Figure courtesy: Cambridge Spark Ltd

In our case, the task is the prediction of CMEs and their classification as halo and non-halo.

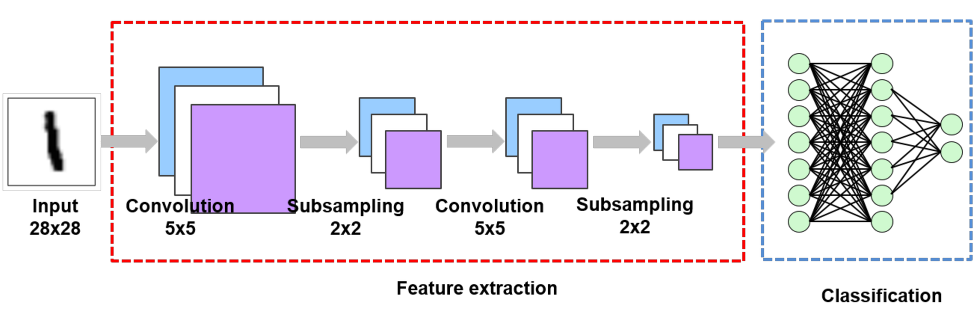


Figure 4: Visualization of CNN used to classify the number "1". Figure Courtesy: Suhyun Kim, iSystems Design Lab.

Machine Learning Project

Layer	Details & Operations	Output shape
Input	-	[512,512,13]
Convolution	Convolution [14] & 3x3 Kernel	[510,510,14]
Convolution	Convolution [16] & 3x3 Kernel	[508,508,16]
Convolution	Convolution [18] & 3x3 Kernel	[506,506,18]
Max Pooling	Max Pooling with 2x2 Kernel	[253,253,18]
Dropout	20 % Dropout	[253,253,18]
Convolution	Convolution [20] & 3x3 Kernel	[251,251,20]
Convolution	Convolution [28] & 3x3 Kernel	[249,249,28]
Convolution	Convolution [36] & 3x3 Kernel	[247,247,36]
Max Pooling	Max Pooling with 2x2 Kernel	[247,247,36]
Dropout	20 % Dropout	[123,123,36]
Convolution	Convolution [40] & 3x3 Kernel	[121,121,40]
Convolution	Convolution [56] & 3x3 Kernel	[119,119,56]
Convolution	Convolution [72] & 3x3 Kernel	[117,117,72]
Max Pooling	Max Pooling with 2x2 Kernel	[58,58,72]
Dropout	40 % Dropout	[253,253,18]
Convolution	Convolution [80] & 3x3 Kernel	[56,56,80]
Convolution	Convolution [112] & 3x3 Kernel	[54,54,112]
Convolution	Convolution [144] & 3x3 Kernel	[52,52,144]
Max Pooling	Max Pooling with 2x2 Kernel	[26,26,144]
Flatten	Flattening of the input	97344
Fully Connected	400 Neuron - Dense layer	400
Fully Connected	200 Neuron - Dense layer	200
Fully Connected	2 Neuron - Dense layer	2
Output	Classifier, 0.5 Threshold Sigmoid	2

Table 1: Basic architecture of the CNN model used for the prediction of CMEs and its classification between halo and non-halo.

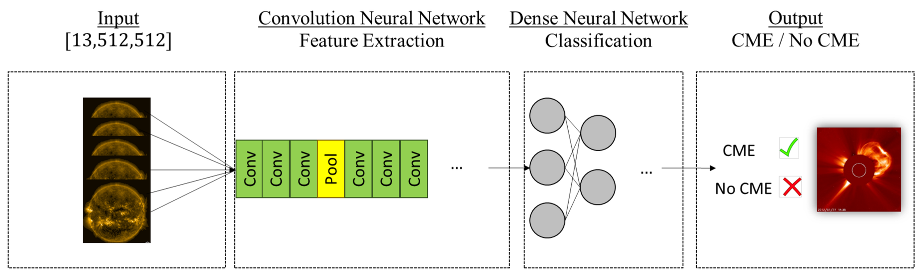


Figure 5: Diagram of the CNN implementation done for the machine learning project.

For the year 2014 and by using SDO data of 512x512 resolution and $\lambda = 171$ [Å], we obtained:

- An accuracy of **76.6%** for the prediction of the CMEs as described in CACTUS catalog.
- An **83.5%** classification between halo and non-halo CMEs as shown in LASCO catalog.

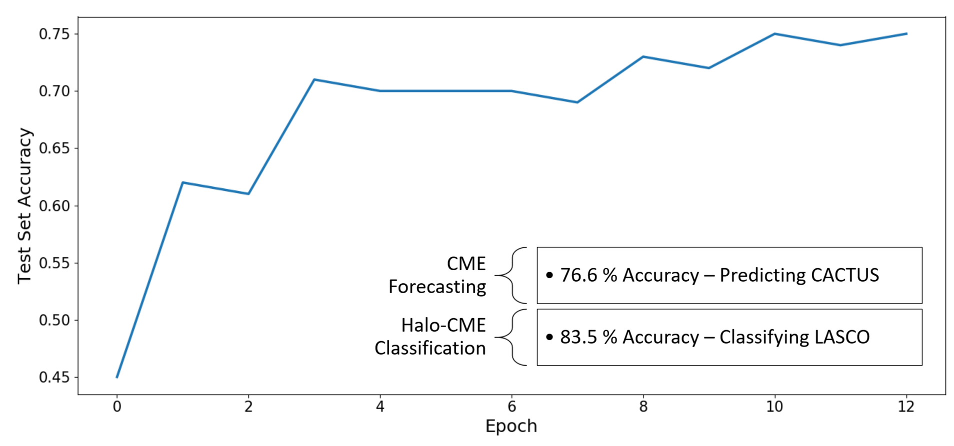


Figure 6: Accuracy of test set versus epoch. Accuracy is given as a percentage, $y \in [0.1]$

For the download and process of data along with the creation of the Pre-processing tool, we used the Sunpy (http://sunpy.org) library of Python language. For the implementation of CNNs we used as backend the TensorFlow library (https://www.tensorflow.org) and the Keras API (https://keras.io).

For the training of the CNNs and for the image processing, we used the Flemish Supercomputer Centre (VSC).

Pre-Processing Tool

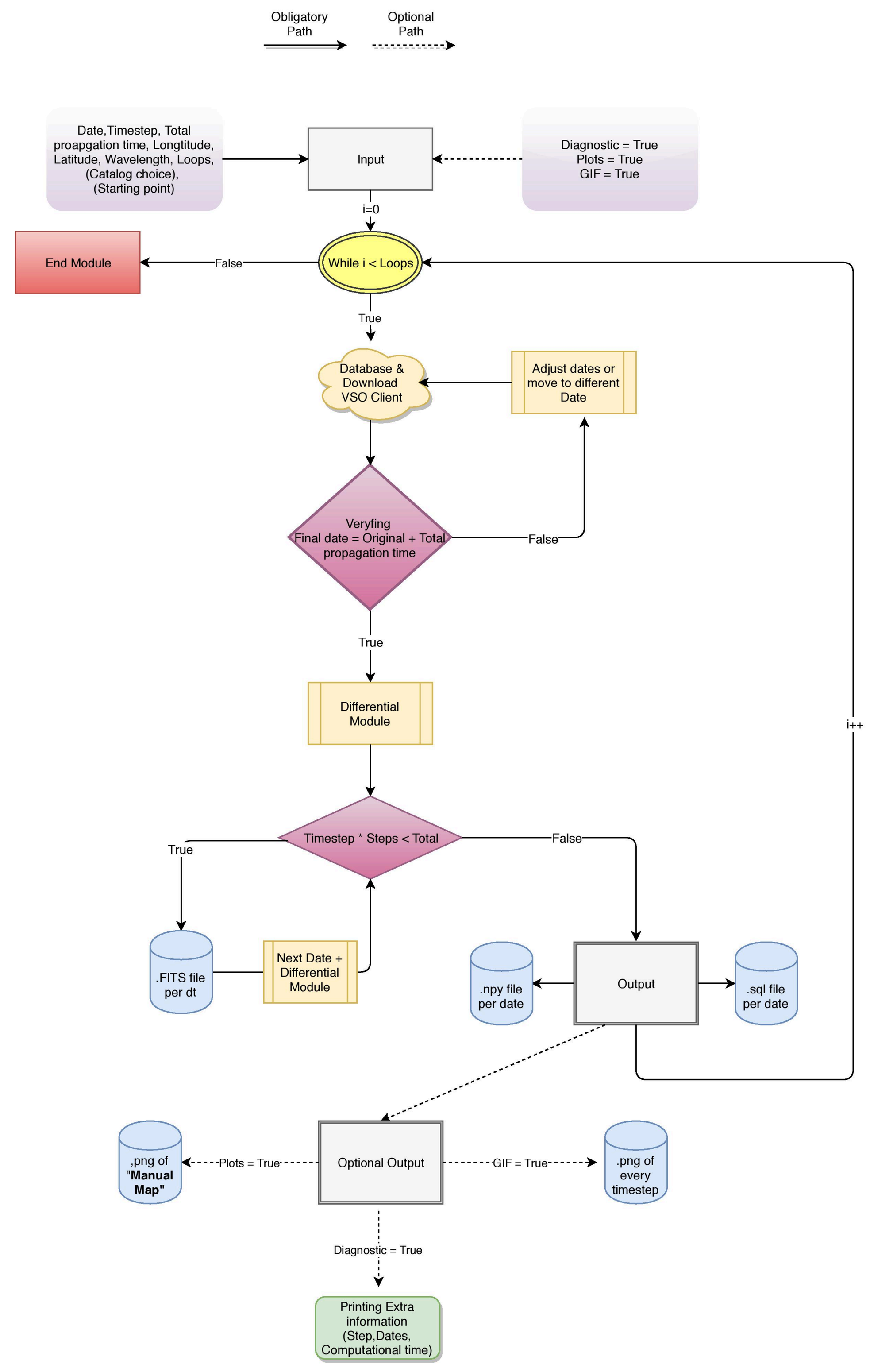


Figure 7: A diagram showing the pre-processing tool algorithm.

The pre-processing tool downloads, processes and organizes images of the Sun derived from the VSO client. It consists of an automatic procedure that uses the Sun's differential rotation to derive new features (**History maps**) that may be used in Solar data analysis and machine learning applications.

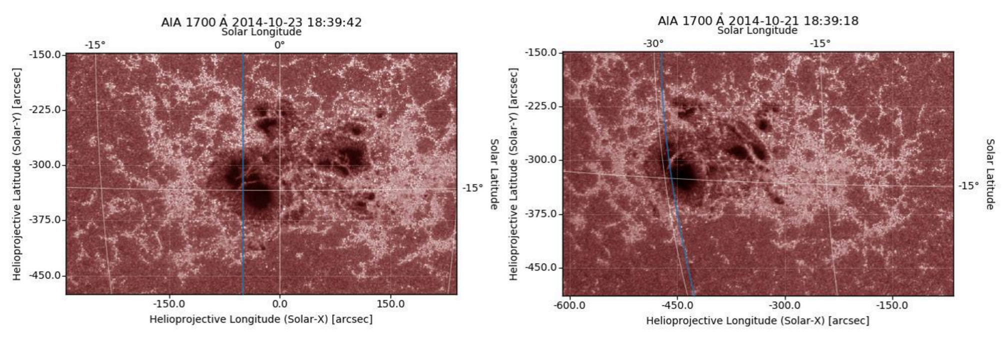


Figure 8: Visualization of a Sunspot as shown in a time-lapse of 2 days. A longitude line at $-2°$ is shown which follows the differential rotation of the Sun back in time.

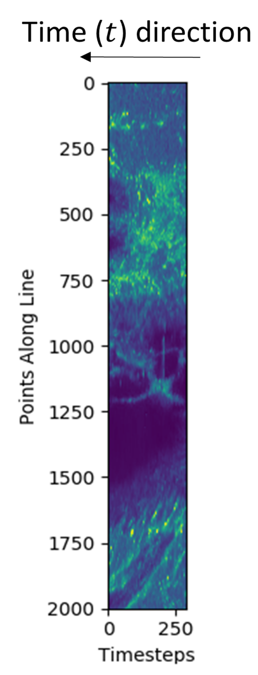


Figure 9: **A History Map (HM)** of a sunspot showing its time evolution. Total propagation time is 2 days and timestep is $dt = 10\ [min]$.

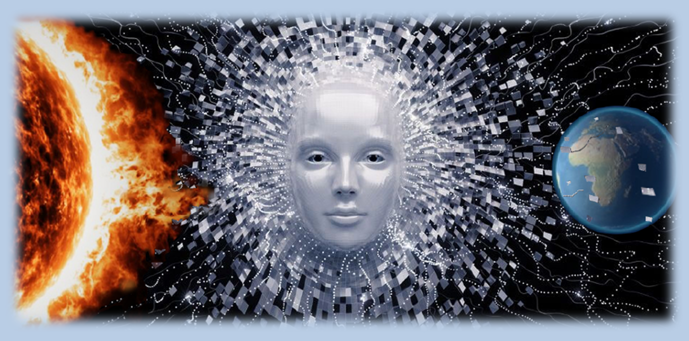
Conclusion & Discussion

- A promising result was obtained for the prediction and the classification of CMEs as described by CACTUS and LASCO catalogs using Artificial Intelligence techniques. In specific, **76.6%** prediction and **83.5%** classification between halo and non-halo CME.
- A new tool was developed that may be a valuable asset in analyzing and predicting solar related phenomena (CMEs, Sunspots, Solar Flares etc.)

Contact / More information

Savvas Raptis
KTH Royal Institute of Technology
savvra@kth.se

Vlaams Supercomputer Centrum



This work is part of the AIDA H2020 (www.aida-space.eu) project funded by the European Commission (EC).