



Identifying Magnetic Reconnection in 2D Hybrid Vlasov Maxwell Simulations with Convolutional Neural Networks

A. Hu¹ , M. Sisti^{2,3} , F. Finelli² , F. Califano² , J. Dargent² , M. Faganello³ , E. Camporeale^{1,4,5} , and J. Teunissen¹

¹ Centrum Wiskunde & Informatica, Amsterdam, The Netherlands; andong@cwi.nl

² Dipartimento di Fisica, Università di Pisa, Italy

³ Aix-Marseille University, CNRS, PIIM UMR 7345, Marseille, France

⁴ CIRES, University of Colorado, Boulder, CO, USA

⁵ NOAA Space Weather Prediction Center, Boulder, CO, USA

Received 2020 June 5; revised 2020 July 9; accepted 2020 July 10; published 2020 September 3

Abstract

Magnetic reconnection is a fundamental process that quickly releases magnetic energy stored in a plasma. Identifying from simulation outputs where reconnection is taking place is nontrivial and, in general, has to be performed by human experts. Hence, it would be valuable if such an identification process could be automated. Here, we demonstrate that a machine-learning algorithm can help to identify reconnection in 2D simulations of collisionless plasma turbulence. Using a Hybrid Vlasov Maxwell model, a data set containing over 2000 potential reconnection events was generated and subsequently labeled by human experts. We test and compare two machine-learning approaches with different configurations on this data set. The best results are obtained with a convolutional neural network combined with an “image cropping” step that zooms in on potential reconnection sites. With this method, more than 70% of reconnection events can be identified correctly. The importance of different physical variables is evaluated by studying how they affect the accuracy of predictions. Finally, we also discuss various possible causes for wrong predictions from the proposed model.

Unified Astronomy Thesaurus concepts: [Convolutional neural networks \(1938\)](#); [Solar magnetic reconnection \(1504\)](#)

1. Introduction

Magnetic reconnection is a fundamental process in space and laboratory plasmas in which magnetic energy is converted into kinetic energy, released in the form of accelerated particles, flows, and heating (Cassak & Shay 2007). Although the process itself is highly localized, it eventually leads to a global change of the magnetic field topology.

Reconnection typically occurs in the presence of thin, elongated current sheets (CSs), which can locally become unstable but eventually rearrange the connectivity of magnetic field lines on a global scale. This process is forbidden at large fluid scales (with respect to kinetic and/or diffusive scales) where ideal magnetohydrodynamic (MHD) holds. At such scales, the initial connectivity of field lines is preserved, as field lines are “frozen-in” in the fluid motion of the plasma (and vice versa). Local violation of ideal MHD laws leads to the onset of reconnection (Furth et al. 1963; Coppi et al. 1979; White 1980).

In space, magnetic reconnection is today recognized as the energetic driver of several important energetic processes as, for instance, solar flares and coronal mass ejections (Priest 1982). It also occurs routinely at the dayside boundary between the solar wind and the Earth’s magnetosphere, as well as in the magnetotail. As a consequence, accelerated particles are injected into the magnetosphere, in some cases down to the Earth polar regions (Dungey 1961; Priest & Forbes 2001). Magnetic reconnection is therefore behind many of the risks associated with space weather, including electronic damage to satellites, endangering astronauts, disturbing Global Navigation Satellite System signals and even impacting power grids (Cassak 2016). Magnetic reconnection also occurs in conditions where CSs are naturally created by the presence of global large-scale unsteady flows, as for dayside or tail reconnection. Furthermore, CSs can be created by the development of

MHD-scale vortices driven by the Kelvin–Helmholtz instability along the magnetospheric flanks (Faganello & Califano 2017) or by the nonlinear dynamics of magnetic field fluctuations, such as small scale vortex motion in solar wind turbulence (Retinò et al. 2007; Haynes et al. 2014; Phan et al. 2018).

Magnetic reconnection plays a key role in the context of plasma turbulence, a phenomenon today routinely observed by satellites in the solar wind and in the Earth’s magnetosphere. With respect to a turbulent fluid where energy is transferred by wave–wave interactions, reconnection has been recognized to represent an alternative path for energy transfer in plasmas (Karimabadi et al. 2013; Cerri & Califano 2017; Camporeale et al. 2018a). First, the local formation of CSs efficiently transfers energy from the large MHD scales to the small ion-kinetic scale (which governs the thickness of CSs). Second, CS disruption allows to inject energy directly on the sub-ion scales involving also the electrons strictly coupled to the magnetic small scale dynamics. These dynamics are generic and visible in almost all simulations of plasma turbulence.

In 2D, the CS structure is characterized by the presence of a thin in-plane magnetic field inversion region associated with an out-of-plane directed current. It corresponds to a magnetic field that points in the opposite direction after crossing the so-called neutral line where the in-plane field goes to zero. In the presence of a nearly constant out-of-plane magnetic field dubbed “guide field,” the plasma is in the so-called “guide field regime.” In this regime, the nonlinear dynamics is dominated by in-plane interactions (quasi-2D regime). This is the regime adopted in the present paper. We underline that in 2D reconnection can more easily be identified by human experts than in 3D. Another advantage is that simulations are computationally much cheaper in 2D. The use of 2D simulations therefore allows us, as a first

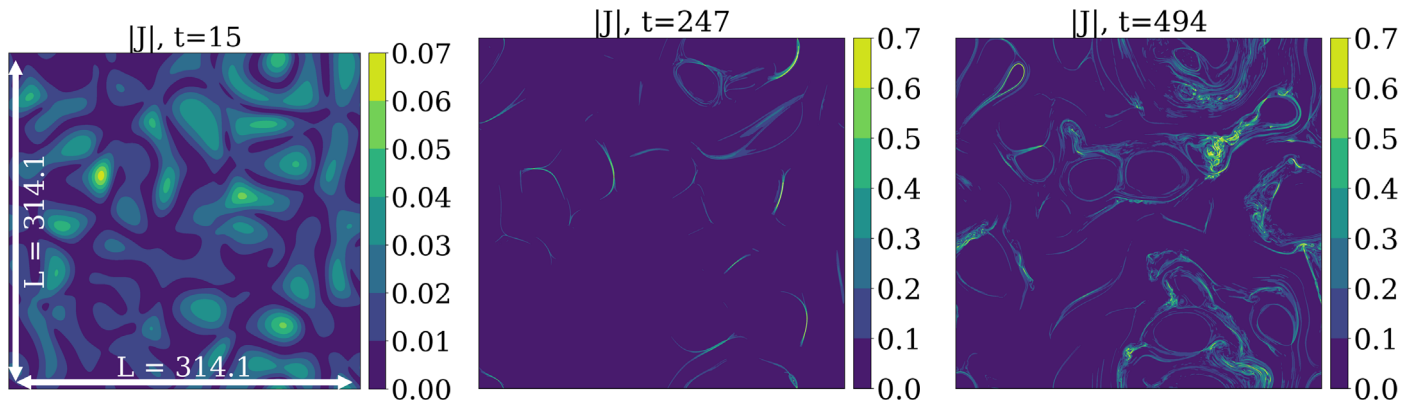


Figure 1. Evolution of the current density $|J|$ in Sim 1; see Table 1. Time is indicated in units of the inverse ion cyclotron frequency Ω_{ci}^{-1} . At $t = 15$, the beginning of the simulation, the initial perturbations are visible. At $t = 247$, which corresponds to roughly one eddy turnover time ($t_e \sim 260$), current sheets have formed. They show up as thin and elongated peaks in the current density. At $t = 494$, turbulence is fully developed, and current sheets are broken up by the dynamics at small scales.

step, to carefully assess the viability of the proposed machine-learning approach.

An example of a 2D kinetic Vlasov simulation of a turbulent magnetized plasma is shown in Figure 1. In this simulation (presented in detail in Section 2.1), turbulence is generated by initial large-amplitude magnetic perturbations of wavelengths of the order of the domain size. After approximately one eddy turnover time for the largest wavelength perturbations (at $t = 247 \Omega_{ci}^{-1}$, where Ω_{ci} is the ion cyclotron frequency; see Section 2.1), we see that after an initial transient thin CSs (with respect to the domain size) form. After another eddy turnover time, at $t = 494 \Omega_{ci}^{-1}$ the plasma is in a fully turbulent regime where the CSs interact, merge, or disrupt. It becomes harder to distinguish individual CSs, but reconnection continues to occur.

An increasing amount of data for the study of magnetic reconnection is continuously produced by simulations and by satellite measurements. It is therefore important to find ways to reliably and efficiently locate reconnection events in these data. Recognizing magnetic reconnection is relatively straightforward in idealized configurations that are prepared ad hoc, such as a symmetric isolated CS, usually modeled in simulations as a 1D equilibrium (the so-called Harris sheet; see, e.g., Camporeale & Lapenta 2005). However, in more realistic dynamical configurations, detecting reconnection is much less trivial as, for instance, in the context of large-scale vortex dynamics (Daughton et al. 2014; Borgogno et al. 2015; Sisti et al. 2019) or in turbulent simulations (Servidio et al. 2009; Zhdankin et al. 2013). As of today, there is no single optimal way to automatically identify magnetic reconnection in simulations of “nonidealized” plasma configurations. With observational data, this task is even more difficult, as most signatures of reconnection, such as large-amplitude field variations or particle accelerations, can be caused by other phenomena (e.g., large-scale vortices, plasma turbulence, shocks). Up to now, reconnection events have been identified by human experts, which is time consuming and can lead to subjective results. Therefore, a method that can automatically and reliably identify reconnection events would be valuable.

Machine-learning techniques have recently been used to identify regions with reconnection based on signatures in the particle velocity distribution function (Dupuis et al. 2020). In this work, we instead focus on the electromagnetic, density, and velocity fields to detect magnetic structures where reconnection is occurring using supervised machine-learning techniques. More

specifically, we have developed a method capable of identifying reconnection in 2D simulations of plasma turbulence performed with a Hybrid Vlasov Maxwell (2D-HVM) numerical code (Valentini et al. 2007, 2014). The simulations are described in Section 2.1. Compared with a more demanding fully kinetic description, a proper description of the electron reconnection physics is lacking here. Nevertheless, the use of a hybrid model has several advantages. A hybrid model is able to simulate a physical domain much larger than the ion-kinetic scale, which is a typical scale for the thickness of CSs. A simulation can therefore contain many CSs, while still resolving ion-kinetic effects for a proper descriptions of turbulence at the ion/CS scale. There are also several reasons for using 2D simulations. As mentioned above, reconnection can clearly be identified in 2D, whereas in 3D reconnection is not clearly defined, and is much harder to identify; computational costs are much lower in 2D. Also, although 2D plasma turbulence is geometrically simpler than its 3D counterpart, it still creates a large number of magnetic field geometries where reconnection can occur in “nonidealized” configurations. Furthermore, magnetic reconnection can be identified more reliably by human experts in 2D data than in 1D (time series) observational data. Simulations provide more information as compared with a satellite’s 1D time series, as they contain a full 2D description of the plasma dynamics and fields shaping around a reconnection zone. Using simulations can be seen as a first step in the automatic identification of reconnection. Finally, additional data can readily be generated in the future for refining the method. The long-term objective is to apply machine-learning methods to time series data, such as those provided by a virtual satellite technique, and real satellite measurements.

In this paper we use supervised machine-learning methods, which means that a training data set consisting of input–output pairs is required. After the training phase, the model can predict outputs for other unseen inputs. The inputs here consist of simulation data in a 200^2 pixels neighborhood around a potential reconnection zone, and the output is a binary value indicating whether reconnection is taking place. It should be noted that a 200^2 pixel area is able to capture most of the important information around the reconnection site and is convenient for human experts for labeling the samples. A challenge in this binary classification problem is that the input is relatively high dimensional. Several machine-learning approaches are considered. In the simplest one, we extract only a few statistics from the input data, and then apply a

Table 1
Description of the Two 2D-HVM Simulations Used in This Paper

Name	Description	Grid Size	dl/d_i	N_{samples}	% Reconnection	Time Range (Ω_{ci}^{-1})
Sim 1	All data	3072 ²	0.1	2069	42%	[0, 370]
	Training set			1205	34.7%	[0, 260], [340, 370]
	Validation set			437	56%	[280, 320]
Sim 2	Test set	2048 ²	0.15	124	56.5%	[205, 233]

Note. Both simulations are performed on a square $L \times L$ domain with $L = 50 \times 2\pi$. The resolution of Sim 1 is higher, as indicated by its smaller grid spacing dl . N_{samples} is the number of labeled samples; in the training, validation, and test data sets only nonambiguous samples are included. The computational costs of Sim 1 and Sim 2 were about 5 M and 2 M core hours, respectively.

so-called decision tree classifier. The other classifiers that we use are based on neural networks.

Neural networks (NNs) are widely used for machine learning. Nonlinear NNs contain few assumptions on the “basis functions” used to map inputs to outputs, in contrast to, e.g., a linear model. In principle, NNs of sufficient complexity can approximate any continuous function mapping inputs to outputs. Their flexibility makes NNs a powerful tool for space physics modeling, and they have been applied in various contexts, e.g., forecasting geomagnetic indices (Wu & Lundstedt 1996, 1997; Wing et al. 2005; Bala et al. 2009; Wintoft et al. 2017; Camporeale et al. 2018b; Gruet et al. 2018) and modeling for the upper atmosphere (Hoque & Jakowski 2011; Hu & Zhang 2018; Hu et al. 2019, 2020). The use of NNs is attractive here because the precise relationship between magnetic reconnection and selected physical parameters is not entirely known.

Here, we use convolutional neural networks (CNNs), which are particularly well suited for dealing with images. A CNN employs a type of filtering that can extract spatial features at a given characteristic scale, while retaining spatial transformation invariance. The repeated application of these filters can process the input image on a number of different scales and at different levels of feature abstraction. Many of the recent breakthroughs in image recognition have been achieved with CNNs. In a space context, CNNs have been used, for example, for space object detection (Linares & Furfaro 2016) and solar flare prediction (Huang et al. 2018; Park et al. 2019).

The paper is organized as follows. In Section 2.1, the 2D-HVM model and the simulation data are introduced. The generation of the reconnection data set is described in Section 2.2, and the proposed machine-learning models and an image-cropping method are introduced in Section 2.3. The accuracy of the machine-learning models is assessed in Section 3, by comparing the performance of the different methods on unseen data. The contribution of individual physical variables is analyzed in Section 4, where we also study an optimal window size for the image-cropping method. The importance of each variable is also investigated in this section. Several illustrative examples of misclassifications are also discussed in Section 4. Finally, our results are summarized in Section 5 where we also discuss their relevance for the automatic classification of real observational data as e.g., those from the Magnetospheric Multiscale Mission (MMS).

2. Methods and Data

2.1. Simulations

Data are provided by means of high-resolution 2D-HVM simulations of turbulence. In this model, ions are fully kinetic

and electrons are modeled as a neutralizing fluid with mass through a generalized Ohm’s law (Valentini et al. 2007; Perrone et al. 2012). Quasi-neutrality, $n_i \simeq n_e \simeq n$, is assumed. Then, the system of equations is given by the Vlasov equation for the ion distribution function $f_i = f_i(\mathbf{x}, \mathbf{v}, t)$

$$\frac{\partial f_i}{\partial t} + \mathbf{v} \cdot \nabla f_i + (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_i}{\partial \mathbf{v}} = 0, \quad (1)$$

where \mathbf{E} and \mathbf{B} are the electric and magnetic field. The generalized Ohm’s equation for the electron response reads

$$\begin{aligned} \mathbf{E} - d_e^2 \nabla^2 \mathbf{E} = & -(\mathbf{u} \times \mathbf{B}) + \frac{1}{n} (\mathbf{J} \times \mathbf{B}) - \frac{1}{n} \nabla P_e \\ & + \frac{d_e^2}{n} \nabla \cdot [\mathbf{u} \mathbf{J} + \mathbf{J} \mathbf{u}] - \frac{1}{n} d_e^2 \nabla \cdot \left(\frac{\Gamma \Gamma}{n} \right). \end{aligned} \quad (2)$$

Furthermore, the Faraday and Ampere equations are given by

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}; \quad \nabla \times \mathbf{B} = \mathbf{J}, \quad (3)$$

where the displacement current has been neglected (low-frequency regime). The ion density n and the ion fluid velocity \mathbf{u} are obtained by taking the zeroth and first-order velocity moment of f_i , respectively. All equations are normalized to the ion mass m_i , the initial ion cyclotron frequency $\Omega_{ci} = eB_0/m_i c$, where B_0 is the magnitude of the initial guide field along the z -direction, and the Alfvén velocity v_A (or, equivalently, to the ion skin depth $d_i = v_A \Omega_{ci}^{-1}$). As a result, the electron skin depth is given by $d_e = \sqrt{m_e/m_i}$, where m_e is the electron mass. We assume an isothermal equation of state for the electron pressure, $P_e = nT_{0e}$. The set of Equations (1)–(3) is solved in a 2D-3V phase space using an Eulerian algorithm (Mangeney et al. 2002) that combines the so-called splitting scheme with the current advanced method (Valentini et al. 2007).

We use data from two simulations listed in Table 1. In both simulations we take $B_z(t=0) = B_0 = 1$ and $B_x(t=0) = B_y(t=0) = 0$, where the z -direction is perpendicular to the simulation plane. Random isotropic magnetic field perturbations are added to the initial equilibrium configuration to initiate turbulence. The plasma response self-consistently generates velocity and density fluctuations. The initial perturbations have wavenumber magnitudes $k \in [0.02, 0.12]$ and a root mean squared value of the magnetic fluctuations equal to $\delta B_{\text{rms}} \simeq 0.3$ in both simulations.

The initial distribution functions are Maxwellian with a uniform initial temperature $T_{0i} = T_{0e}$ such that the ion beta parameter $\beta_i \doteq 2nT_{0i}/B_0^2$ is equal to one. The velocity space is sampled by 51^3 uniformly distributed grid points spanning

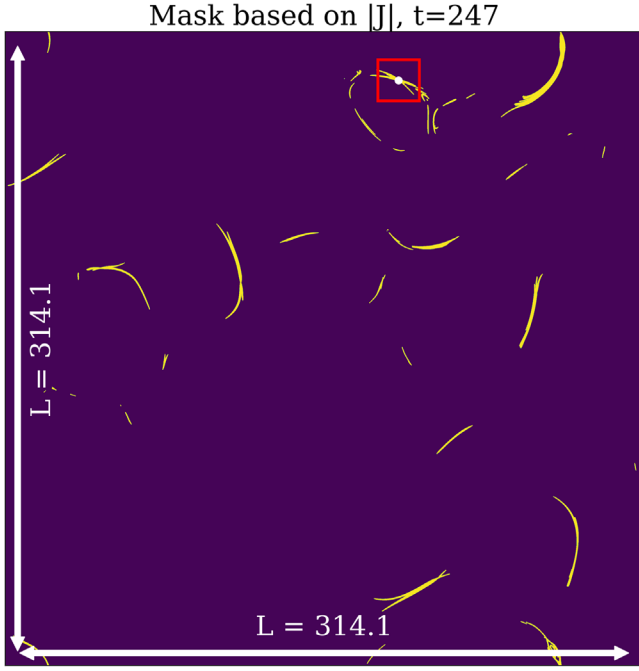


Figure 2. Example showing how potential reconnection sites in the simulation output are selected. First, a mask is constructed based on a threshold for the current density (shown in yellow). A list of current sheets is extracted, which are centered on their local peaks (e.g., the red square centered on the white dot). Note that there are typically multiple reconnection events that occur simultaneously

$[-5v_{th,i}, 5v_{th,i}]$ in each direction, where $v_{th,i} = \sqrt{\beta_i/2}$ is the initial ion thermal velocity. We set the reduced mass ratio to $m_i/m_e = 100$ so that d_i and d_e are separated by one decade.

2.2. Extraction and Labeling of Images from Simulations

Magnetic reconnection usually takes place in or nearby a maximum of the current density. The current density can therefore be used to extract potential reconnection sites from the simulation data. For each time at which simulation output was saved, all regions are marked where the magnitude $J = |\mathbf{J}|$ of the current density exceeds a given threshold. The threshold is defined by $J_{th} = \max(\sqrt{\langle J^2 \rangle} + 3\sigma(J^2))$, where $\sigma(J^2) = \sqrt{\langle J^4 \rangle - (\langle J^2 \rangle)^2}$ (Zhdankin et al. 2013), where the brackets denote the average over the whole physical domain. The center of each reconnection site is determined by each local maximum of the current density. Around a local maximum, one finds a long and narrow region where the current density exceeds the threshold. For such a region, we first determine the maximal distance d_{max} to the local maximum. Then a square of size $(2d_{max})^2$ is extracted, centered on the local maximum.

An example of a potential reconnection site is highlighted by a red square box in Figure 2. In some cases, multiple potential reconnection sites can be present in the same region. All of these regions are extracted, and together they constitute the samples of the data set. All samples are rescaled to 200^2 pixels to facilitate their use in machine-learning methods. An example of the extracted data is shown in Figure 3. The data extracted from the simulation outputs used in the machine-learning

models contain the seven physical variables that are listed below.

1. Current magnitude $J = |\mathbf{J}|$.
2. Electron fluid velocity along z -direction $V_{e,z} = u_z - J_z/ne$.
3. In-plane electron fluid velocity $V_{plane} = \sqrt{V_{e,x}^2 + V_{e,y}^2}$, where $V_{e,x/y} = u_{x/y} - J_{x/y}/(ne)$.
4. In-plane magnetic field $B_{plane} = \sqrt{B_x^2 + B_y^2}$.
5. Magnetic field fluctuation along the z -direction $\delta B_z = B_z - \langle B_z \rangle$.
6. Flux function Ψ , in 2D, related to the magnetic field through $\mathbf{B}_{plane} = \nabla \Psi \times \mathbf{z}$.
7. Electron decoupling defined by $E_{dec,e} = |(\mathbf{E} + \mathbf{V}_e \times \mathbf{B})_z|$, which corresponds to the z -component of the electric field in the electron rest frame. In 2D, it is nonzero only when the magnetic field dynamics and the electron motion are decoupled, which is a necessary condition for reconnection to occur.

We remark that the ion decoupling term $E_{dec,i} = |(\mathbf{E} + \mathbf{u} \times \mathbf{B})_z|$ is included in the pictures for classification by a human expert, but it is not used as a variable for the machine-learning models. There are several reasons for this. First, our experts use $E_{dec,i}$ only as an auxiliary variable to $E_{dec,e}$. Because $E_{dec,i}$ is more sensitive to large-scale structures than $E_{dec,e}$, it is heavily influenced by the environment around the candidate site, making it less useful. Furthermore, the ML method will only benefit from additional variables if enough training samples are available. We “only” have around 3000 labeled samples, and leaving out some rather uninformative variables can be beneficial.

As listed in Table 1, two data sets have been generated. Sim 1 is the main one including 2069 samples. Sim 1 is divided into two subsets: a training set and a validation set. The training set is used to train the model and the validation set is used to find the optimal parameters for the trained model. A time interval of $20 \Omega_{ci}^{-1}$ is used between these two data sets to keep them independent, as shown in Figure 4. This delay is important to prevent information from leaking from the training to the validation set, as it takes some time for the morphology of reconnection sites to change. Sim 2 is used as the test set to assess the accuracy of the developed models as well as their performance on simulation data with a different resolution than the one used for training.

Once a sufficient number of 200^2 images have been extracted from the simulations, they need to be labeled by human experts. We make use of an automated workflow on zooniverse.org, which is a platform aimed at involving the general public in the labeling of scientific data sets. The project can be accessed via <http://aida-space.eu/reconnection>, together with a tutorial on how to identify reconnection sites. The project is public, so any expert can help with the labeling.

It is worth noticing that magnetic reconnection in “non-idealized” configurations as those created by turbulence can be difficult to identify, even for experts. Therefore, the possible labels assigned to a picture are: 1 for reconnection, 0 for no reconnection, and 0.5 for ambiguous cases. Every picture is labeled by up to three human experts, after which the labels are averaged, so that in the end each case has a single label. The distribution of labels in Sim 1 is shown in Figure 5. Labels

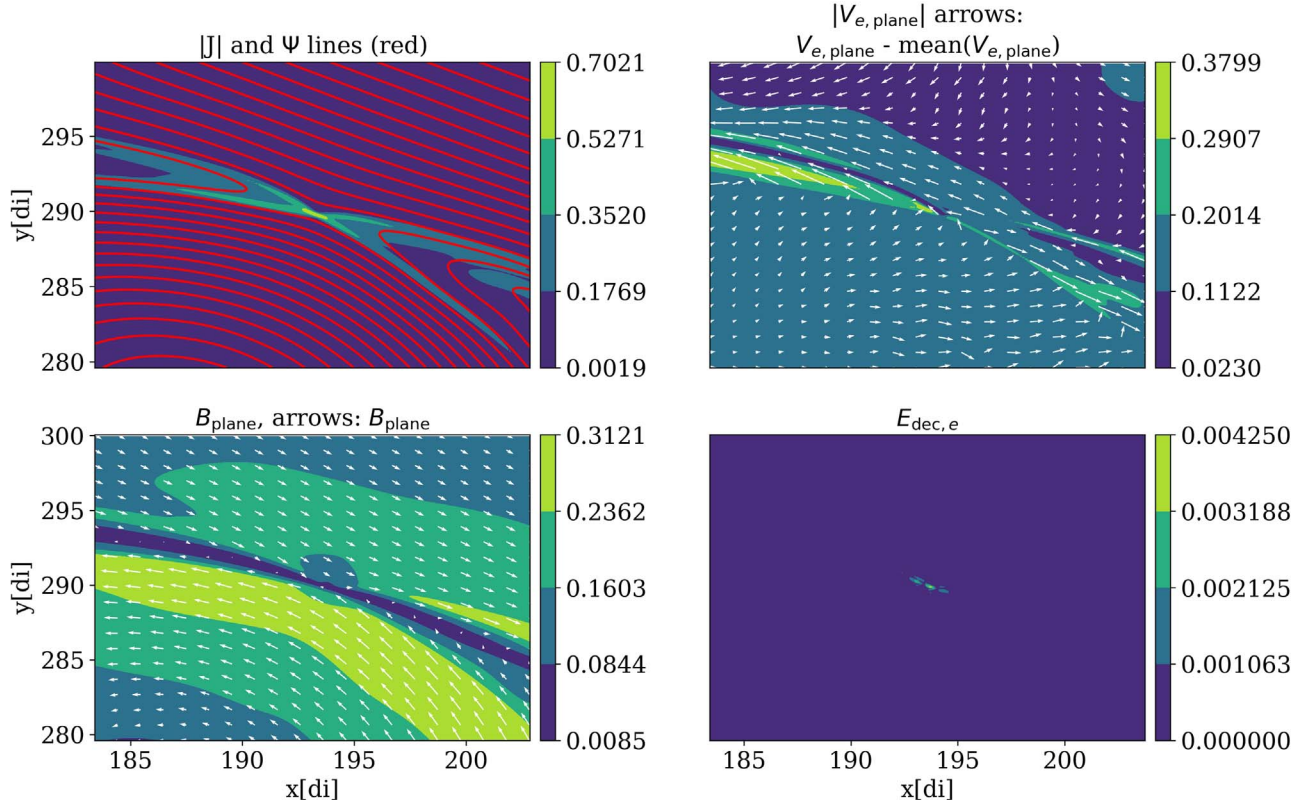


Figure 3. Example of the pictures used for the classification of magnetic reconnection by human experts. The selected variables $|J|$, Ψ , $V_{e, \text{plane}}$, B_{plane} , and $E_{\text{dec}, e}$ that can be indicators for reconnection are shown. Note that arrows corresponding to $V_{e, \text{plane}}$ and B_{plane} help human classification, but only the magnitudes of these vectors is used in the machine-learning models. This example corresponds to the red box drawn in Figure 2.

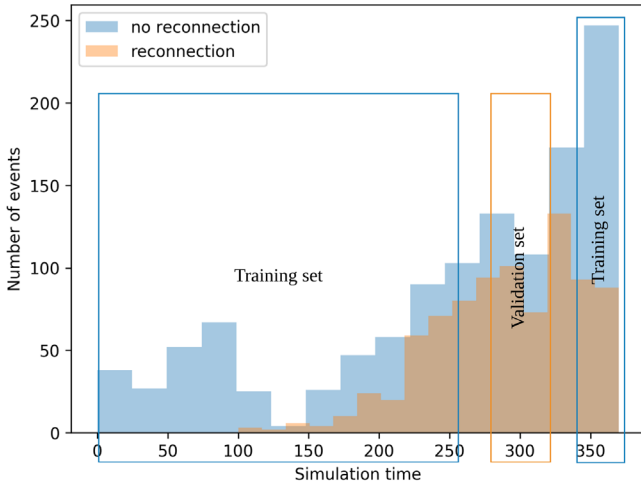


Figure 4. Distribution of the samples and their labels of Sim 1. The blue rectangles indicate the time range for the training set, and the yellow rectangle the time range for the validation set. A gap of $\delta t = 20$ is used between the training and validation set to keep them independent.

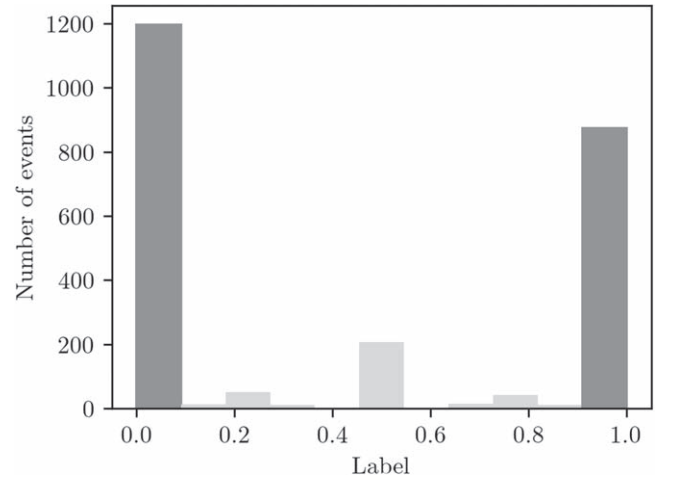


Figure 5. Distribution of labels in Sim 1; see Table 1. Here, 1 indicates reconnection, 0 indicates no reconnection, and values in between are ambiguous cases. Note that the data set is fairly balanced. In this study, only nonambiguous events (indicated by the darker shade) are used to train and evaluate the machine-learning models.

unequal to zero or one are considered to be “ambiguous.” There are around 1200 negative cases, 900 positive cases, and 300 ambiguous cases. To convert these labels to binary values, all ambiguous cases with a label unequal to 0 or 1 were dropped. However, this does not mean all included cases are completely unambiguous. At the time of writing, about 59% of cases was labeled by one human expert, 32% by two experts, and 9% by

three experts. The corresponding fractions of ambiguous cases were 8%, 25%, and 30%, showing that the number of ambiguous cases goes up significantly when cases are inspected by multiple experts. From these numbers, we can estimate that roughly 14% of cases in our filtered data set would still be marked ambiguous, if all cases had been labeled by three experts. For the smaller Sim 2 data set, all cases were

labeled by three experts, so the above estimate does not apply. In future research, it would be interesting to include ambiguous events in the data set. One could then investigate whether such events can be identified by a machine-learning approach, and it would also allow for a direct comparison between the performance of a machine-learning model and that of a human expert.

2.3. Machine-learning Approaches

We consider two machine-learning models. CNN-X takes a region of size X^2 as input. This subset is determined by a physics-based image-cropping approach. The whole image is used when $X = 200$. A decision tree classifier is used for comparison. These models are described in more detail below.

2.3.1. CNN-X

As already mentioned, CNNs are widely used for image processing because their convolutional layers can extract essential information from images in a generic way. In this study, a standard CNN is used and implemented using PyTorch (Paszke et al. 2019). Each input channel is convolved with its own set of filters. The designed architecture of a CNN-X model is shown in Figure 6.

Because reconnection occurs when the dynamics of the magnetic field decouples from the electron fluid motion, there are two main signatures to consider: a peak in the current density $|\mathbf{J}|$, and a peak in the electron decoupling $E_{\text{dec},e}$ term (see Section 2.2). Based on these signatures, a heuristic method has been constructed to extract an area of X^2 pixels from the original 200^2 input. In this study, 32 was found to be the optimal value for X , as discussed in Section 4. A flowchart describing this CNN-32 approach is shown in Figure 7. It consists of the following steps:

1. For each image, find the closest pair of local maxima in $|\mathbf{J}|$ and $E_{\text{dec},e}$. This is done by first constructing a list of local maxima for each variable, consisting of all maxima that lie at least 10 pixels apart. Only local maxima whose amplitude is at least 70% of the global maximum of the image are considered.
2. If the closest pair is within a 20 pixel distance, extract a square region of 32×32 pixels centered on the middle of the two peaks. If not, return a 32^2 region centered on the maximum of $|\mathbf{J}|$ in the image. All seven physical variables are extracted.

The resulting data set contains 7×32^2 values (seven is the number of physical variables) per image, so it is about 40 times smaller than the original data set with 7×200^2 values per image. However, the 200^2 images are still used for human labeling because they contain more information and are easier for human experts to label than the 32^2 images.

2.3.2. Decision Tree Classifier

This classifier takes as input only the min, max, and mean of each variable inside the 32^2 area as illustrated in Figure 7. Its input therefore consists of 21 variables (seven physical variables times three). The decision tree classifier is implemented using the *scikit-learn* library (Pedregosa et al. 2011), and its optimal depth is found with a grid search method.

2.3.3. Optimal Window Size

As introduced in Section 2.3.1, a physics-based image-cropping method is used in the CNN-X model and decision tree models to extract the most important X^2 pixel region out of the 200^2 pixel input. When the “correct” region (i.e., the one containing the potential reconnection zone) is extracted, this approach increases the signal-to-noise ratio (S/N). A further benefit is that it reduces the computational cost of the CNNs, as the dimension of their input is reduced. However, the reconnection site marked by our human experts is not always captured inside the extracted region. Figure 8 shows the percentage of reconnection sites captured versus the window size of the extracted region. An “elbow” can be seen around a size of about 20 pixels, at which size more than 70% of reconnection sites can be captured. There are several reasons why not all reconnection sites are captured. For example, more than one reconnection site can be present in one sample, or it could be that no local maximum in $E_{\text{dec},e}$ can be found.

The accuracy of the models for different window sizes can be assessed by looking at the number of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN), where positives refer to reconnection events. To compare different models, these numbers are converted to a single score. We consider two of such scores: Matthews’ Correlation Coefficient (MCC) and the True Skill Statistic (TSS), which are defined as

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \times (\text{FN} + \text{TN}) \times (\text{FP} + \text{TN}) \times (\text{TP} + \text{FN})}} \quad (4)$$

$$\text{TSS} = \text{TPR} + \text{TNR} - 1, \text{ where} \quad (5)$$

$$\text{TPR} = \frac{\text{TP}}{\text{FN} + \text{TP}} \quad (6)$$

$$\text{TNR} = \frac{\text{TN}}{\text{FP} + \text{TN}}. \quad (7)$$

To be able to classify a potential reconnection site, it can be beneficial to have information available in a neighborhood around the site, so there are two competing factors: a smaller window size can increase the S/N, whereas more reconnection sites and more of their surroundings can be captured with a larger window size. To determine the optimal window size, we compare the performance of the CNN-16, CNN-32, CNN-64, CNN-128, and CNN-200 models on the validation set described in Table 1. The TSS, MCC, and confusion matrix of these models are shown in Table 2. The listed numbers are average values obtained by training the model 10 times with the same configuration but different initial random values for the CNN weights. Table 2 shows that the optimal window size is 32 for this application. The “image-cropping” method described in Section 2.3 can therefore improve both the accuracy and efficiency of the CNN-X model. With a larger window size, the performance degrades because there is more noise in the input, such as other potential reconnection sites or complex structures due to turbulence. This noise makes it more difficult to train a CNN that performs well on the test set. We remark that with a larger data set a larger window

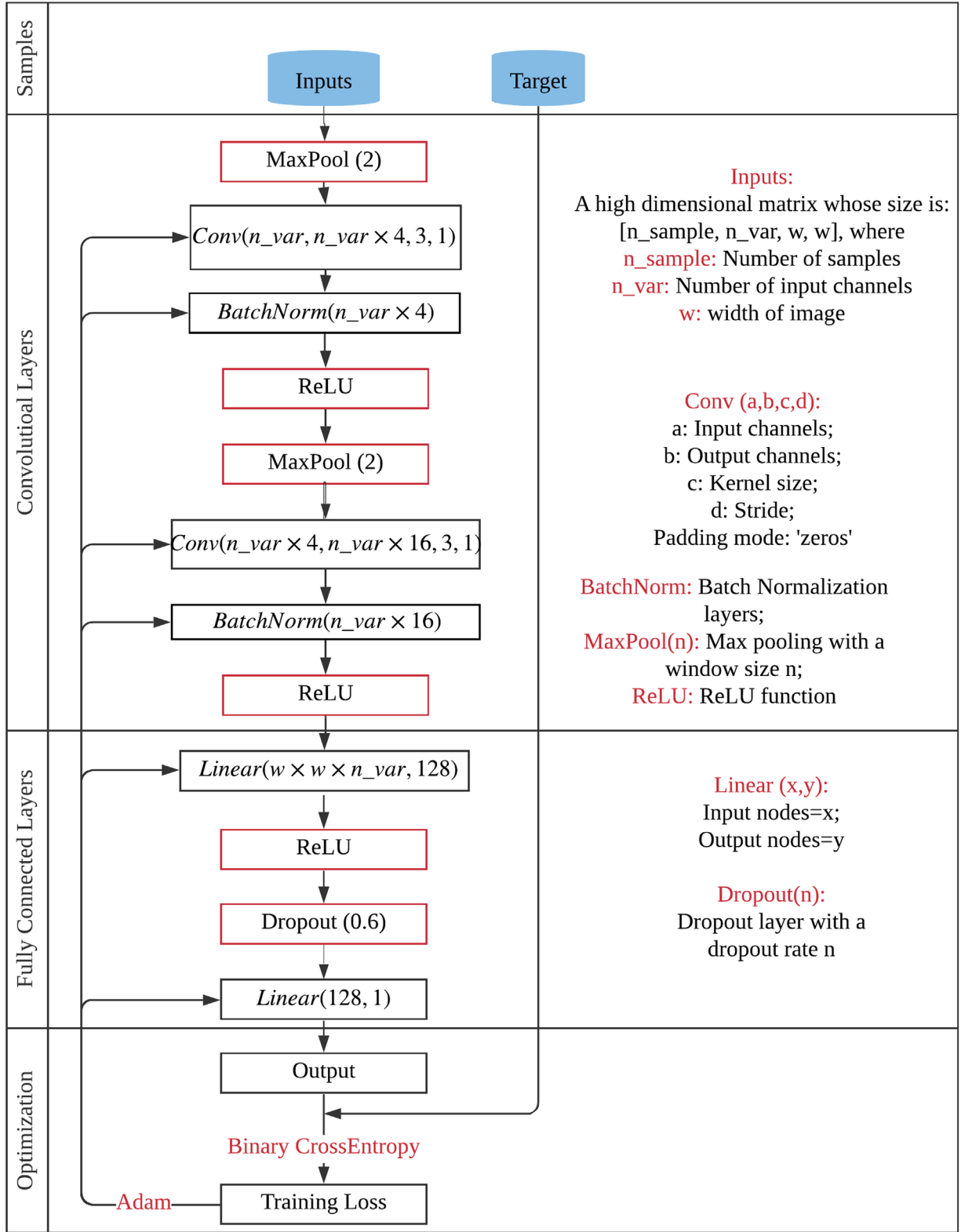


Figure 6. Architecture of CNN-X models.

size could work better because noise would be less of a concern.

With a window size of 32, the fraction of cases that is misclassified is about 26%. For reference, we expect that about 14% of cases in the data set would be marked ambiguous if they all had been labeled by three human experts; see Section 2.2.

3. Results

In this section, we evaluate the accuracy of the two machine-learning approaches introduced in Section 2.3, namely the decision tree model and the CNN-32 model, which was found to be optimal in Section 2.3.3. The training set introduced in

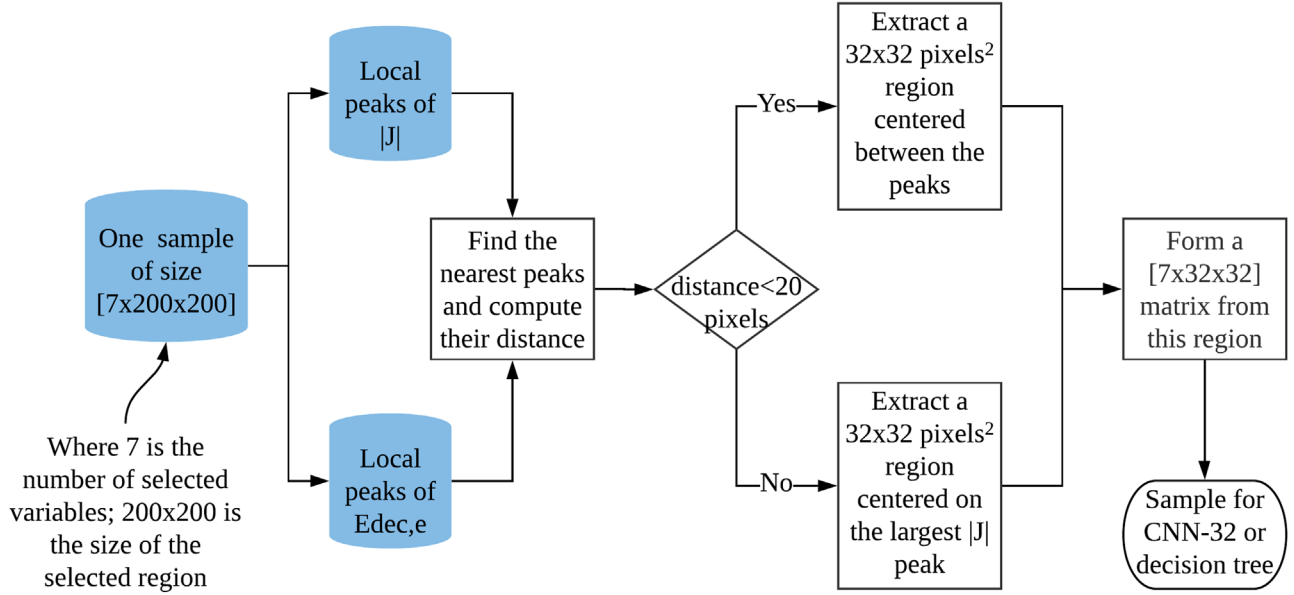


Figure 7. Flowchart of the image-cropping approach, which extracts an 32^2 pixel area from the 200^2 pixel input.

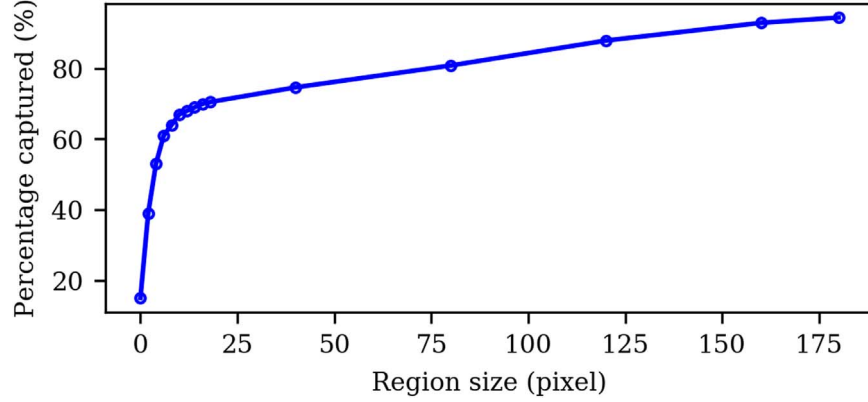


Figure 8. Percentage of reconnection sites captured vs. window size for the image-cropping approach described in Section 2.3.1.

Table 2

Accuracy of the CNN-X Models with Different Window Sizes X, Evaluated on the Out-of-sample Sim 1 Data

Window Size	TSS	MCC	TP	FP	TN	FN
16	0.29	0.32	85	12	176	158
32	0.56	0.55	170	28	161	70
64	0.42	0.41	138	29	159	103
128	0.43	0.44	133	23	166	108
200	0.39	0.44	154	48	141	87

Note. The results shown are averages over 10 trained models with different initial random coefficients. The best results are marked in bold.

Table 1 is used to train these models. In Section 3.1, the test set, which is Sim 2, is used to assess the accuracy of both models. In Section 3.2, the importance of the physical variables is investigated by comparing the accuracy of the CNN-32 model on the test set with different inputs.

3.1. Model Accuracy

The scores of the CNN-32 model (which is the optimal CNN-X model) and the decision tree model are shown in

Table 3

Accuracy of the Machine-learning Models Evaluated on Test Data Set; See Table 1

Model	TSS	MCC	TP	FP	TN	FN
CNN-32	0.50	0.51	48	10	43	22
Decision tree	0.28	0.30	55	27	26	15

Table 3. Both models are evaluated on the test set described in Table 1. As before, the score of the CNN-32 model is averaged over 10 model instances. The CNN-32 model significantly outperforms the simple decision tree classifier. It has a true positive rate (TPR or sensitivity; details in Equation (6)) of 0.82 and a true negative rate (TNR or specificity; details in Equation (6)) of 0.66. For the decision tree, these rates are 0.67 and 0.63, respectively.

The performance of the CNN-32 model on the test set is almost as good as on the out-of-sample validation data (Table 2). This indicates that the model can be applied to independent simulations performed with the same numerical code, provided that the grid resolution is not too different. To apply the model to a simulation performed at a very different

Table 4

Performance of CNN-32 Models when Some of the Variables Are Randomly Shuffled within the Test Set, Meaning They Are No Longer Informative of Reconnection

MCC								
No.	$ J $	B_{plane}	$V_{e,z}$	$E_{\text{dec},e}$	δB_z	Ψ	$V_{e,\text{plane}}$	Improvement
1	0.33	0.051	0.16	0.024	0.001	−0.044	0.015	0.33
2	✓	0.43	0.39	0.32	0.34	0.33	0.35	0.098
3	✓	✓	0.49	0.40	0.42	0.39	0.42	0.062
4	✓	✓	✓	0.50	0.48	0.40	0.43	0.007
5	✓	✓	✓	✓	0.50	0.49	0.48	0.004
6	✓	✓	✓	✓	✓	0.50	0.49	0.003
7	✓	✓	✓	✓	✓	✓	0.50	0.0

Note. The numbers in each row indicate how much the MCC score improves when the corresponding variables are unshuffled. The most important variable per row is in bold. These variables are included unshuffled in the rows below. Improvements in the MCC score are shown in the right-most column. The reported numbers are averages over 10 CNN-32 model instances.

resolution, some scaling of the input would probably be required, so that a 32^2 pixel window would capture a region of similar physical size. In future work, it would also be interesting to investigate the performance of the CNN-32 model on 2D reconnection simulations performed with different numerical codes.

3.2. Importance of Physical Variables to Classifier Accuracy

One of the objectives of this study is to investigate which physical variables are important for the classification of magnetic reconnection. We do this in two ways. The first approach is to shuffle the variables in the test set, and then determine a sequential order for their importance as follows:

1. Train the CNN-32 model 10 times to have 10 different CNN-32 models.
2. Shuffle each physical variable in the test set, by randomly permuting the corresponding 32^2 pixel data over the samples.
3. Put the original values back in the test set for one variable at a time to check the influence of each variable on the classification.
4. Find the variable which has the largest influence (largest mean MCC score in 10 models) and “freeze” it, meaning that it will not be shuffled anymore. Then go back to step 3 to test other variables, until no variables are left.

The results are shown in Table 4. They reveal that, statistically, $|J|$ contributes most in this classification model. B_{plane} and $V_{e,z}$ are slightly less significant. The other variables $E_{\text{dec},e}$, δB_z , Ψ and $V_{e,\text{plane}}$ are not very significant, as the MCC score improves by less than 1% when one of them is added.

Physically, these results seem reasonable because (1) $|J|$, B_{plane} and $V_{e,z}$ are all directly related to the reconnection process, and they are highly correlated; (2) $E_{\text{dec},e}$ has already been used for the image-cropping method (as introduced in Section 2.3.1); and (3) $V_{e,\text{plane}}$ variations are a consequence of reconnection, but they can also be caused by turbulence. Overall, Table 4 indicates that the first three variables, $|J|$, $V_{e,z}$, B_{plane} , might be enough to develop a reconnection classification model as good as the one based on all seven variables.

The above results say something about the importance of variables in a model that was trained on all variables. A slightly different question is which variables are the most important when a model with fewer inputs is used. To investigate this, we have trained CNN-32 models using one, two, or three physical variables as input. All possible combinations of variables were

Table 5

MCC Scores of CNN-32 Models that Only Take the Listed Variables as Input, Evaluated on the Test Set

One Variable	MCC	Two Variables	MCC	Three Variables	MCC
$ J $	0.44	$ J , B_{\text{plane}}$	0.51	$ J , V_{e,z}, B_{\text{plane}}$	0.56
$V_{e,z}$	0.39	$ J , V_{e,z}$	0.49	$ J , B_{\text{plane}}, E_{\text{dec},e}$	0.55
$V_{e,\text{plane}}$	0.13	$ J , E_{\text{dec},e}$	0.44	$ J , V_{e,z}, \Psi$	0.50
B_{plane}	0.11	$ J , \Psi$	0.42	$ J , \Psi, E_{\text{dec},e}$	0.50
δB_z	0.09	$ J , V_{e,\text{plane}}$	0.40	$ J , V_{e,z}, E_{\text{dec},e}$	0.48

Note. The best five combinations of variables are shown for one, two, or three physical input variables. The results are averages over 10 model instances.

tested. Table 5 shows the MCC scores of the top five combinations. Again, $|J|$ is the most important variable followed by B_{plane} and $V_{e,z}$, in agreement with the results from Table 4.

4. Discussion

4.1. Analysis of Wrong Predictions

In this section, we investigate why the CNN-32 model in some cases makes a wrong prediction. Three examples of FN and three examples of FP are shown in Figure 9. Here, a FP refers to a case labeled 0 (no reconnection) for which the model predicts a label 1 (reconnection), and a FN is a case labeled 1 for which the predicted label is 0. Each panel is a “candidate” magnetic reconnection event similar to Figure 3, where panels (a)–(c) display FN cases, and panels (d)–(f) display FP cases.

Each case illustrates a different type of misclassification. In Figure 9(a), a so-called plasmoid is present with a relatively complex structure around it. However, the reason the labeled reconnection point was not recognized is probably that it does not coincide with a significant peak in the current density. The reconnection site in Figure 9(b) is at the edge of the image. Such sites are probably harder to classify because only “half” of the reconnection zone is visible. Case (c) is ambiguous, even for human experts. For the FP, case (d) is distorted by plasma turbulence and, as a consequence, has a complex morphology. Finally, cases (e) and (f) are actually correctly predicted but have a wrong label. These cases correspond to the same physical region at different times. It is clear that reconnection proceeds at the lower edge of the magnetic flux rope in the center of case (e), where an X-point is clearly visible. Indeed, the difference between the Ψ value at the X-point and its value at the O-point (the center of the flux rope) increases in time

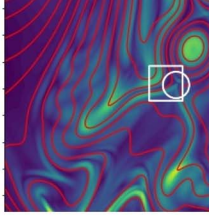
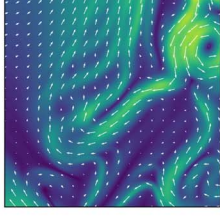
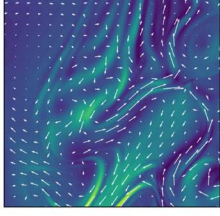
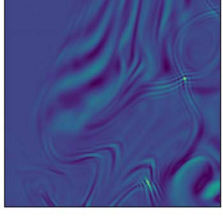
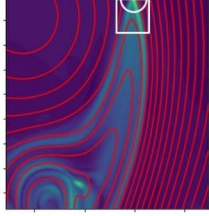
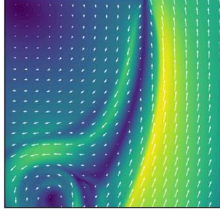
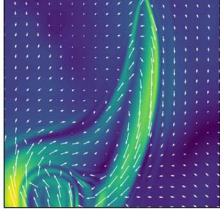

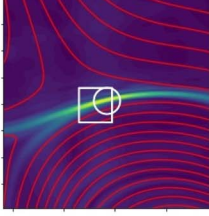
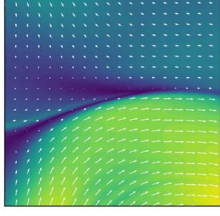
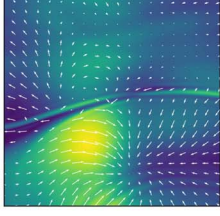
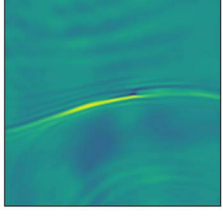
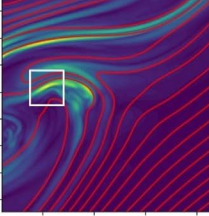
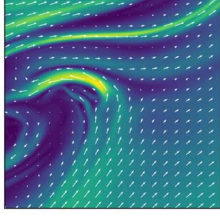
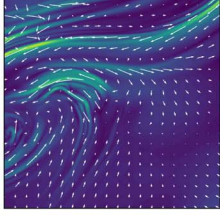

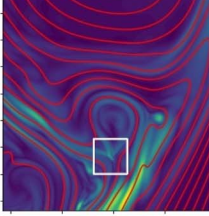
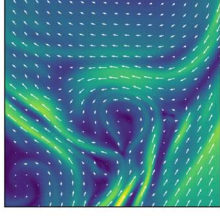
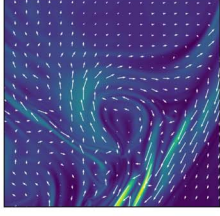
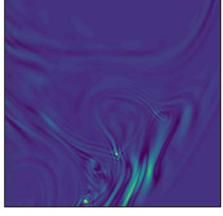
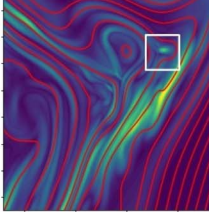
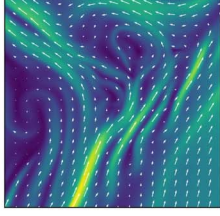
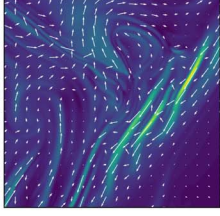
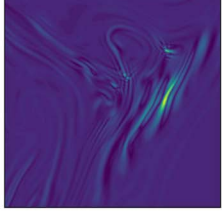
	Reason	$ \vec{J} $	B_{plane}	$V_{e,plane}$	$E_{dec,e}$
a) Epoch: 330 Case: 49 Label: 1 Predicted: 0	contaminated by plasma turbulence				
b) Epoch: 330 Case: 89 Label: 1 Predicted: 0	Reconnection site is at the edge				
c) Epoch: 335 Case: 12 Label: 1 Predicted: 0	Ambiguous case				
d) Epoch: 330 Case: 51 Label: 0 Predicted: 1	contaminated by plasma turbulence				
e) Epoch: 345 Case: 97 Label: 0 Predicted: 1	misabeled				
f) Epoch: 355 Case: 100 Label: 0 Predicted: 1	site is not in the cropped region				

Figure 9. Examples of false negatives (rows a)–c)) and false positives (rows d)–f)). Reconnection sites labeled by human experts are indicated by white circles. The region extracted by the image-cropping method is shown by white squares. Possible reasons for the false predictions are indicated. The images show the same variables as in Figure 3, in a 200^2 pixel window.

passing from (e) to (f). The fact that the main current peak is away from the X-point misled humans during classification, but the machine-learning model correctly catches reconnection going on. However, in case f), the cropping method in this particular case selects a wrong region. This is due to the presence of a high peak in the current density and in E_{dec} at some distance from the true reconnecting site.

These cases show that the CNN-32 model is in some cases able to find reconnection sites that are initially missed by human experts.

5. Summary and Outlook

The first extensive labeled data set for magnetic reconnection in 2D-HVM simulations has been constructed with currently over 2000 samples labeled by human experts. We have developed a

classifier able to automatically identify magnetic reconnection in such simulations using convolutional neural networks (CNNs). An important part of this classifier is a physics-based image-cropping method that zooms in on potential reconnection sites. The overall model is called “CNN-X,” where X indicates the size of the cropped window. Our results show that

1. The image cropping method can improve the accuracy of the CNN models by increasing the signal-to-noise ratio. The optimal window size around potential reconnection sites was found to be 32^2 pixels. The corresponding CNN-32 model had a true positive rate (sensitivity) of 89% and a true negative rate (specificity) of 70% when evaluated on an out-of-sample validation set.
2. The CNN-32 model was also evaluated on a fully independent test set that was constructed from a simulation with lower resolution. The model then had a true positive rate of 82% and a true negative rate of 66%. This indicates the developed CNN-32 model is generic and can be applied to other simulations. Furthermore, in some cases, the CNN-32 model was able to find reconnection sites that were initially missed by a human expert.
3. We have investigated the importance of different physical variables for the prediction of reconnection. Three variables were found to be the most important reconnection markers: the current density $|J|$, the out-of-plane electron velocity $V_{e,z}$ and the in-plane magnetic field B_{plane} .

This study is a first step in adopting machine learning for the automatic identification of magnetic reconnection. We think that with more labeled data from different types of simulations the model’s accuracy would improve. This would then open up the possibility of using machine learning to detect reconnection in other types of data, such as artificial and real satellite measurements. In particular, the long-term goal of this study is to use the the classifications obtained using the model developed here to analyze time series data created using a virtual satellite technique. These time series could then be collected in a large labeled database, which would help to detect reconnection in time series recorded by real satellites.

This project has received funding from the European Union’s Horizon 2020 research and innovation program under grant agreement No. 776262 (AIDA, www.aida-space.eu) Numerical simulations have been performed on Marconi at CINECA (Italy) under the ISCRA initiative. F.C. thank Dr. M. Guarrasi (CINECA, Italy) for his essential contribution on code implementation on Marconi.

This publication uses data generated via the Zooniverse.org platform, development of which is funded by generous support, including a Global Impact Award from Google, and by a grant from the Alfred P. Sloan Foundation. The simulation data set (UNIP1_TURB_2D, UNIP1_TURB_2D_2048) is available at Cineca on the AIDA-DB. To access the meta-information and the link to the raw data, see the tutorial at <http://aida-space.eu/AIDAdB-iRODS>.

For their help in labeling the data set, we thank Sid Fadanelli, Giuseppe Arrò Giulia Cozzani, Silvio Sergio Cerri, Francesco Pucci, Francesco Pegoraro, Alessandro Retinò Oreste Pezzi, Jörg Büchner, Amir Chatraee, and Neeraj Jain.

The current version of the labeled data set and the code for the machine-learning models are available on Zenodo ([10.5281/zenodo.3907309](https://zenodo.org/record/3907309) and [10.5281/zenodo.3935887](https://zenodo.org/record/3935887), respectively).

Please check the AIDA website at <http://aida-space.eu/reconnection> for further updates.

ORCID iDs

A. Hu  <https://orcid.org/0000-0002-6929-2158>
M. Sisti  <https://orcid.org/0000-0003-3331-0606>
F. Finelli  <https://orcid.org/0000-0002-9160-4840>
F. Califano  <https://orcid.org/0000-0002-9626-4371>
J. Dargent  <https://orcid.org/0000-0002-7131-3587>
M. Faganello  <https://orcid.org/0000-0002-5232-0108>
E. Camporeale  <https://orcid.org/0000-0002-7862-6383>
J. Teunissen  <https://orcid.org/0000-0003-0811-5091>

References

- Bala, R., Reiff, P., & Landivar, J. 2009, *SpWea*, **7**, 04003
Borgogno, D., Califano, F., Faganello, M., & Pegoraro, F. 2015, *PhPI*, **22**, 032301
Camporeale, E., & Lapenta, G. 2005, *JGRA*, **110**, A07206
Camporeale, E., Sorriso-Valvo, L., Califano, F., & Retinò, A. 2018a, *PhRvL*, **120**, 125101
Camporeale, E., Wing, S., & Johnson, J. 2018b, *Machine Learning Techniques for Space Weather* (Amsterdam: Elsevier)
Cassak, P., & Shay, M. 2007, *PhPI*, **14**, 102114
Cassak, P. A. 2016, *SpWea*, **14**, 186
Cerri, S. S., & Califano, F. 2017, *NJPh*, **19**, 025007
Coppi, B., Mark, J. W.-K., Sugiyama, L., & Bertin, G. 1979, *AnPhy*, **119**, 370
Daughton, W., Nakamura, T. K. M., Karimabadi, H., Roytershteyn, V., & Loring, B. 2014, *PhPI*, **21**, 052307
Dungey, J. W. 1961, *PhRvL*, **6**, 47
Dupuis, R., Goldman, M. V., Newman, D. L., Amaya, J., & Lapenta, G. 2020, *ApJ*, **889**, 22
Faganello, M., & Califano, F. 2017, *JPIPh*, **83**, 535830601
Furth, H. P., Killeen, J., & Rosenbluth, M. N. 1963, *PhFI*, **6**, 459
Gruet, M. A., Chandorkar, M., Sicard, A., & Camporeale, E. 2018, *SpWea*, **16**, 1882
Haynes, C. T., Burgess, D., & Camporeale, E. 2014, *ApJ*, **783**, 38
Hoque, M. M., & Jakowski, N. 2011, *RaSc*, **46**, RS6015
Hu, A., Carter, B., Currie, J., et al. 2020, *JGRA*, **125**, e27263
Hu, A., Carter, B., Julie, C., et al. 2019, *JGRA*, **124**, 4926
Hu, A., & Zhang, K. 2018, *RemS*, **10**, 1658
Huang, X., Wang, H., Xu, L., et al. 2018, *ApJ*, **856**, 7
Karimabadi, H., Roytershteyn, V., Wan, M., et al. 2013, *PhPI*, **20**, 012303
Linares, R., & Furfaro, R. 2016, in 2016 19th Int. Conf. Information Fusion (FUSION) (Piscataway, NJ: IEEE), 1140
Mangeney, A., Califano, F., Cavazzoni, C., & Travnicek, P. 2002, *JCoPh*, **179**, 495
Park, E., Moon, Y.-J., Lee, J.-Y., et al. 2019, *ApJL*, **884**, L23
Paszke, A., Gross, S., Massa, F., et al. 2019, *Advances in Neural Information Processing Systems 32 (NIPS 2019)*, <https://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library>
Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, *Journal of Machine Learning Research*, **12**, 2825
Perrone, D., Valentini, F., Servidio, S., Dalena, S., & Veltri, P. 2012, *ApJ*, **762**, 99
Phan, T. D., Eastwood, J. P., Shay, M. A., et al. 2018, *Natur*, **557**, 202
Priest, E. R. 1982, *Solar Magnetohydrodynamics* (Dordrecht: Reidel)
Priest, E. R., & Forbes, T. G. 2001, *JPIPh*, **66**, 363
Retinò, A., Sundkvist, D., Vaivads, A., et al. 2007, *NatPh*, **3**, 235
Servidio, S., Matthaeus, W., Shay, M., Cassak, P., & Dmitruk, P. 2009, *PhRvL*, **102**, 115003
Sisti, M., Faganello, M., Califano, F., & Lavraud, B. 2019, *GeoRL*, **46**, 11597
Valentini, F., Califano, F., Hellinger, P., & Mangeney, A. 2007, *JCoPh*, **225**, 753
Valentini, F., Servidio, S., Perrone, D., et al. 2014, *PhPI*, **21**, 082307
White, R. 1980, *Resistive Instabilities and Field Line Reconnection*, Tech rep. PPL-1655, (Princeton Univ. Plasma Physics Lab), doi:[10.2172/5423144](https://doi.org/10.2172/5423144)
Wing, S., Johnson, J., Jen, J., et al. 2005, *JGRA*, **110**, A04203
Wintoft, P., Wik, M., Matzka, J., & Shprits, Y. 2017, *JSWSC*, **7**, A29
Wu, J.-G., & Lundstedt, H. 1996, *GeoRL*, **23**, 319
Wu, J.-G., & Lundstedt, H. 1997, *JGR*, **102**, 14255
Zhdankin, V., Uzdensky, D. A., Perez, J. C., & Boldyrev, S. 2013, *ApJ*, **771**, 124