The Astrophysical Journal Supplement Series, 271:29 (11pp), 2024 March https://doi.org/10.3847/1538-4365/ad1dd4

Image Synthesis for Solar Flare Prediction

Elad Amar and Ohad Ben-Shahar[1]
Ben Gurion University of the Negev, Be'er Sheva, Israel
Received 2023 September 8; revised 2024 January 9; accepted 2024 January 10; published 2024 March 1

Abstract

Solar flare prediction is a topic of interest to many researchers owing to the potential of solar flares to affect various technological systems, both terrestrial and in orbit. In recent years, the forecasting task has become progressively more reliant on data-driven computations and machine-learning algorithms. Although these efforts have improved solar flare predictions, they still falter in doing so for large solar flares, in particular under operational conditions, since large-flare data are very scarce and labeled data are heavily imbalanced. In this work, we seek to address this fundamental issue and present a scheme for generating synthetic magnetograms to reduce the imbalance in the data. Our method consists of (1) synthetic oversampling of line-of-sight magnetograms using Gaussian mixture model representation, followed by (2) a global optimization technique to ensure consistency of both physical features and flare precursors, and (3) the mapping of the generated representations to realistic magnetogram images using deep generative models. We show that these synthetically generated data indeed improve the capacity of solar flare prediction models and that, when tested on such a state-of-the-art model, it significantly enhances its forecasting performance, achieving an F1-score as high as 0.43 ± 0.08 and a true skill statistic of 0.64 ± 0.10 for X-class flares in the 24 hr operational solar flare data split.

Unified Astronomy Thesaurus concepts: Space weather (2037); Solar active regions (1974); Neural networks (1933); Gaussian mixture model (1937); Solar flares (1496)

1. Introduction

Solar flares are sudden, rapid, and intense eruptions of electromagnetic radiation emerging from the Sun's atmosphere. They are caused by complex patterns of reorganization of the Sun's magnetic field in temporary regions in its atmosphere (Hesse & Cassak 2020). These regions, known as active regions (ARs), may consist of several sunspots—subregions where the magnetic field wells up to the surface of the Sun, inhibits the convection of hot plasma from its interior, and thus induces cooler regions that appear darker than their surroundings. Sunspots usually group in pairs, one for each magnetic pole of the magnetic field that penetrates the Sun's surface.

Solar flares are classified on a logarithmic scale into five progressive classes labeled A, B, C, M, and X according to the peak emission of soft X-ray radiation they emit. While the $\leqslant$M-class flares have minor or inconspicuous effects on Earth,[2] the energy emitted during X-class flares might disturb Earth's outer atmosphere and can interrupt power grids, communication, and navigation infrastructures (Camporeale et al. 2018), both in orbit and terrestrial. While forecasting efforts of these space weather events are thus essential, X-class flares are considered rare (Georgoulis et al. 2021), and in the last decades only a few hundred have been recorded, as opposed to the tens of thousands of weaker flares. But regardless of their strength, solar flare prediction can be considered a binary classification task based on AR data during a predefined time interval. If a flare of a specified class or stronger indeed bursts during the time interval of interest, the AR data are labeled "positive" for that class, and "negative" otherwise.

With no clear and precise physical model of solar flares yet available, one way to address their prediction is driven by data. And data are in no shortage, especially since the 2010 launch of NASA's Solar Dynamic Observatory (SDO). The satellite produces a variety of data, including images, dopplergrams, magnetograms, and spectra (Pesnell et al. 2012). One of the data sources on board the SDO is the Helioseismic and Magnetic Imager (HMI) instrument, which continuously maps the photospheric vector magnetic field, including the line-of-sight (LOS) component. In addition, the HMI team also produces the Space-weather HMI Active Region Patches (SHARPs; Bobra et al. 2014), which consist of patches of magnetic concentrations on the Sun's surface. Each HMI Active Region Patch (HARP) is localized and tracked automatically during the Sun rotation, providing the type of extended temporal data useful for flare prediction. Two more instruments aboard the SDO are the Atmospheric Imaging Assembly (AIA) and the Extreme ultraviolet Variability Experiment (EVE). The AIA takes images of the Sun in multiple wavelengths, while EVE measures the solar extreme-ultraviolet irradiance. Taken in combination, SDO's instruments produce around 1.5 TB of data every day, and with so much data it begs to consider empirical forecasting of solar flares using machine-learning algorithms. Recent years have indeed experienced rapid growth in this direction (Lugaz et al. 2021).

Several attempts to forecast solar flares using handcrafted features were established throughout the years, depending mainly on physics-based features. Early work focused on identifying parameters that are associated with (mainly large) solar flares by applying discriminant analysis (Leka & Barnes 2003a, 2003b, 2007) on data from the Imaging Vector Magnetograph (Mickey et al. 1996). Georgoulis & Rust (2007) defined the effective connected magnetic field as a significant

[1] Corresponding author.
[2] https://svs.gsfc.nasa.gov/10109/

flare precursor, and Schrijver (2007) pointed out the gradient next to polarity inversion lines (PILs). Both used data produced by the Michelson Doppler Imager (Scherrer et al. 1995) on board the Solar and Heliospheric Observatory. However, it was argued that conclusions may be sensitive to the data collection and the metrics employed (Barnes & Leka 2008). Indeed, after the launch of the SDO, most subsequent research employed physical parameters derived from its data, with various machine-learning tools and skill scores for the analysis.

Employing this general trend, Bobra & Couvidat (2015) used support vector machine (SVM) with features calculated from the vector magnetogram maps. Liu et al. (2017) introduced the random forest algorithm to the problem. Both methods have shown that some of the magnetic parameters are redundant and do not contribute to flare forecasting. Nishizuka et al. (2017) added more solar features from multiple data sources and showed that *k*-nearest neighbors outperforms both SVM and extremely randomized trees. With the emergence of deep neural networks (DNNs) the task was addressed by various models. Nishizuka et al. (2018) extracted parameters from SDO/AIA in addition to those from prior research as input to a neural network. Unlike the preceding algorithms, the neural network achieved good results on forecasting $\geqslant$M-class flares also in operational data splits (Leka et al. 2019), where the training and testing are split into dichotomous chronological intervals. But due to an insufficient number of samples, the authors abstained from forecasting X-class flares. Another DNN model is the long short-term memory (LSTM), which is widely used in language processing owing to its ability to remember and process dependencies through time. This property makes LSTMs useful for solar flare prediction task too (Chen et al. 2019; Liu et al. 2019; Jiao et al. 2020), facilitating the exploition of the changes in time of the parameters describing the ARs. For the present, however, the triggering mechanism of solar flares is still unknown, making the feature selection challenging.

In recent years the rise of machine-learning algorithms based on automatically extracted features did not skip the space weather community, with notable deployment of methods such as convolutional neural networks (CNNs). For example, Huang et al. (2018) used LOS magnetograms with a two-layered network architecture, and Tang et al. (2021) proposed a fusion model composed of CNN and LSTM independently. And yet, as mentioned above, large solar flares are considered rare events, resulting in extremely imbalanced data sets, and while these models perform well in nonoperational data splits, they are biased toward the majority class, obtaining low precision and high recall scores. In general, imbalanced data sets can pose challenges for deep-learning models because they can lead to biased or skewed results (Krawczyk 2016). To tackle the issue, several methods have been proposed, such as the use of data augmentations for the minority class or undersampling the majority class. For example, Deshmukh et al. (2022) suggested adding extremely randomized trees after the CNN and combining both visual properties and physics-based features to reduce false positives, stating that augmenting the minority class did not improve the predictive skill. Zheng et al. (2019, 2021) used both of these schemes with an ensemble of independent binary CNNs to improve prediction performance. While such methods indeed improve results, image augmentation is still limited to the measured data and cannot ensure robustness in operational data splits as mentioned by the authors.

While the class imbalance problem confounds machine-learning algorithms (Abd Elrahman & Abraham 2013; Branco et al. 2016), a different approach to address the issue is by generating new data samples from the minority class, for example, using the synthetic minority oversampling technique (SMOTE; Chawla et al. 2002) and its extensions (Fernández et al. 2018). In order to address the class imbalance problem for solar flare prediction while exploiting the advantages of visual properties of ARs as flare precursors, in this work we propose using SMOTE for generating synthetic positive samples. Subsequently, the training data set, now composed of genuine and synthetic data at a desired and controllable balance, is utilized to enhance solar flare prediction through machine-learning algorithms.

The general idea of generating synthetic images is not new and has been improving dramatically over the years, including for tackling class imbalance in various domains such as medical applications (Frid-Adar et al. 2018; Iqbal & Ali 2018; Nie et al. 2018), classification tasks (Ali-Gombe & Elyan 2019; Lee & Park 2021; Zhu & Pan 2022), and weather forecasting (Xu et al. 2019; Li et al. 2021). So far, previous work on space weather image synthesis has focused on converting SDO/AIA imagery to SDO/HMI imagery (Dani et al. 2022; Sun et al. 2022), or vice versa (Dash et al. 2022), and mapping SDO/AIA images to corresponding images with different extreme-ultraviolet channels (Salvatelli et al. 2022). Such image-to-image translation is a class of computer vision tasks that typically involves learning a mapping function between the two domains, such that the output image preserves the content and key features of the input image.

Two well-known architectures for generating synthetic images are generative adversarial networks (GANs; Goodfellow et al. 2014) and denoising diffusion probabilistic models, or diffusion models in short (Ho et al. 2020; Nichol & Dhariwal 2021). GANs are composed of two CNNs, a generator and a discriminator, that play a two-player minimax game. The generator network learns to create samples resembling the training data, while the discriminator network is trained to differentiate between real training samples and the fake ones generated by the generator. The two networks are trained together until the generator produces samples indistinguishable from the discriminator. A conditional GAN (cGAN) is a modification of GAN trained to generate synthetic data that are conditioned on a set of input variables or input images (Mirza & Osindero 2014; Isola et al. 2017).

Next to GANs and cGANS, diffusion models are a class of likelihood-based models that are trained to generate images or signals. These models work by sequentially adding noise to the data and then learning the opposite mapping function between the noisy input and a denoised version of it. Recently diffusion models have outperformed GANs (Dhariwal & Nichol 2021) and have also been introduced for image-to-image translation tasks (Saharia et al. 2022).

2. Data

We use LOS magnetograms from HMI taken between the years 2011 and 2017 inclusive. The data set consists of 67,180 images with resolutions ranging from 480×370 to 2356×785. In order to synthesize images correlated visually to large flares, for the image-to-image translation task we considered only magnetograms of ARs that produced the rare X-class flares, with a 12 minute cadence. For the prediction

Table 1
Data Split into Training, Validation, and Test Sets, Guided by Operational Setting

Data Split	Year	# Negatives	# Positives
Train	2011–2013	33,462	140
Validation	2014	12,915	111
Test	2015–2017	20,514	38

task, the data set consists of C-, M-, and X-class flares. Due to the minor visual changes in successive images of AR, the images were taken with 96 minute cadence. This helps remove redundant images and train the model much faster. Furthermore, ARs on the limb, i.e., those extending beyond $\pm 70^{\circ}$ from the central meridian, were removed from the data set owing to distortion in the images. Following Leka et al. (2019), we adopted operational data splitting, where the train, validation, and test sets are split into dichotomous chronological intervals. As opposed to other data splitting approaches, such as randomly splitting all the magnetograms or placing magnetograms belonging to an AR in one specific set, this chronological data split mimics reality and allows a better assessment of the prediction model in case it will be used for future forecasting. For the evaluation, we used a 24 hr preflare time interval of X-class flares. The data set is split into training, validation, and test sets as displayed in Table 1.

3. Methods

We propose a method for generating synthetic positive data samples to facilitate better solar flare prediction using machine-learning algorithms despite the inherent imbalance in solar flare data sets. To be effective, these novel synthetic data need to be perceptually realistic (namely to perceptually resemble plausible SDO/HMI data) while also reflecting real ARs' physical features. At the same time, they need to have a different sunspot structure than real data in order to add diversity and enable forecasting on unseen data.

In general, evaluating the perceptual quality of generated images can be done through subjective and objective methods. Since subjective assessments involve human opinion-based tests, objective quality assessment is generally preferred, where one employs distance metrics between real and generated image distributions (Blau & Michaeli 2018). Indeed, in this work, we suggest using an objective assessment for perceptual quality by minimizing the disparity between the actual and predicted probability distributions for real and generated images. Such minimization can facilitate samples having realistic visual properties despite never really being observed in SDO/HMI magnetograms. The bulk of our contribution is thus formalizing and implementing such a synthesis method and showing that it indeed improves the prediction and classification capacity of learning schemes on real data.

Formally, let p_X be the true distribution of flaring magnetograms, $p_{\hat{X}}$ a sought-after distribution of synthesized magnetogram images, and $P(p_X, p_{\hat{X}})$ a nonnegative divergence measure between them. Given $x \in p_X$ and $\hat{x} \in p_{\hat{X}}$, let $F(x, \hat{x})$ be an estimator of similarity between the physical properties of x and $\hat{x}$, while $D(x, \hat{x})$ is a measure of their visual dissimilarity. Given $x \in p_X$, our goal is to find a distribution $p_{\hat{X}}$ that solves the following optimization problem:

$$\begin{aligned} &\arg\min_{p_{\hat{X}}} \quad \mathbb{E}_{\hat{x}\sim p_{\hat{X}}}[F(x, \hat{x})] + \lambda \cdot P(p_X, p_{\hat{X}}) \\ &\text{s.t.} \quad \min_{\hat{x}\sim p_{\hat{X}}} D(x, \hat{x}) \geqslant d, \end{aligned} \tag{1}$$

where $d \in \mathbb{R}$ is a dissimilarity tolerance and λ is a weight parameter. A solution for the optimization problem in Equation (1) provides a distribution of images that closely resembles the original data distribution with physical features similar to a real magnetogram. At the same time, it ensures that samples drawn from the generated distribution are not too similar visually to existing flaring magnetograms (a condition we will test for explicitly before adopting generated samples). Given that p_X is unknown, we consider the positively labeled magnetograms as samples drawn from it. The following subsections present the different building blocks of our proposed method.

3.1. GMM Representation of Magnetograms

Processing the magnetograms directly can be computationally expensive owing to their size and resolution. Each magnetogram has a different aspect ratio and different size depending on the AR evolution and its location on the solar disk, making comparison of magnetograms more difficult. Hence, we seek a representation of magnetograms that not only is as compact as possible but also maintains sunspots' visual information in a fashion independent of resolution and other image parameters. One natural choice is using Gaussian mixture models (GMMs; Reynolds 2009), a type of probabilistic model that represents a data distribution as a mixture of multiple normal distributions. The mixture estimation for $\boldsymbol{x} \in \mathbb{R}^d$ is given by

$$f(\boldsymbol{x}) = \sum_{i=1}^{K} w_i N(\boldsymbol{x};\, \mu_i,\, \Sigma_i)), \tag{2}$$

where K is the number of components, w_i are the mixture weights and satisfy $\sum_{i=1}^{K} w_i = 1$, and μ_i and Σ_i are the mean and covariance matrix of each Gaussian component, respectively. Each component of the mixture is the multivariate normal distribution:

$$\mathcal{N}(\boldsymbol{x};\, \mu_i,\, \Sigma_i) = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu_i})^{\mathrm{T}}\boldsymbol{\Sigma_i}^{-1}(\boldsymbol{x} - \boldsymbol{\mu_i})\right)}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma_i}|}}. \tag{3}$$

Given a gray-scale SHARP LOS magnetogram with N sunspots, we consider each sunspot as a set of image points (pixels) that we seek to model as a GMM. Toward this end, we first segment out the different sunspots, each being a connected component of consistent polarity in the AR. This computation begins with the binarization of the image, followed by edge detection to find the contour of each sunspot. Morphological closing (Serra 1986) is used to smooth noise and areas with weak polarization (aka quiet Sun) and to close holes inside the sunspot's boundary. Next, flood fill (Smith 1979) is used in order to detect and segment all sunspots in the AR. The set of points (pixels) of each sunspot is then fitted with a K-GMM using an expectation-maximization process (Moon 1996), and the result is a representation of the sunspot as a set of GMM parameters. Clearly, the more GMM components are allowed (i.e., the larger the value of K), the finer-grained the fitting result, but also the more complex and higher dimensional its representation. The visual approximation of

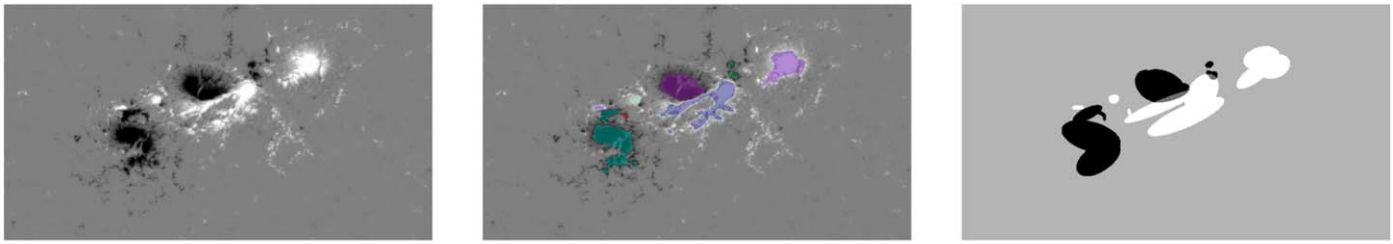


Figure 1. Steps in the magnetogram's GMM representation process. (a) The initial image, HARP 377 on 2011 February 15 at 01:36:00 TAI. (b) Segmentation of the sunspots. (c) Result of fitting a GMM to the segments in the segmentation result and plotting the corresponding confidence ellipses. Here $N = 9$ and $K = 3$.

the sunspot from the GMM model constitutes the direct plotting of each component of the GMM as a confidence ellipse, where its mean determines the center of the distribution and its covariance matrix sets its orientation and size. Since each magnetogram has a different size, we normalize the mean vectors by the HARP size. This way, the GMM representation maintains the relative distance between the sunspots in the AR and ignores their absolute location in the magnetogram. The entire AR is thus fitted with a collection of N K-GMMs based on the number of sunspots that were detected. Figure 1 exemplifies this process for one selected AR, and it also demonstrates how qualitatively the minimal number of components for the GMM that provides good fits was found empirically to be $K = 3$. This choice was set for all our implementations.

The GMM abstraction and representation just described are carried out for positive and negative sunspots separately. It preserves sunspots' position, size, and orientation, as well as their relative distance from one another, while discarding noise, minute magnetic fluctuations, and quiet-Sun information. As a result, it produces a relatively low-dimensional vector on which it may be easier to apply additional processing. In particular, it enables choosing the dissimilarity function to be the L_2 distance between the log-probability density functions f_x and $f_{\hat{x}}$ of two GMMs (Cui & Datcu 2015):

$$D(x, \hat{x}) = \sqrt{\sum_{i=1}^{n} (\log f_x(u_i) - \log f_{\hat{x}}(u_i))^2}\,. \tag{4}$$

By incorporating visual dissimilarity as a constraint, the optimization procedure can seek a broader range of visual differences, allowing for more flexibility and nuances in the generated output. The dissimilarity tolerance d itself was set by calculating the dissimilarity of two real random X-class flaring magnetograms from two distinct ARs.

3.2. AR Synthesis for Minority Class

Although deep-learning algorithms have shown immense potential in data generation tasks, one of the notable limitations is their reliance on the availability of vast and diverse data sets (Noguchi & Harada 2019). To address the class imbalance problem, an alternative approach is SMOTE. This method is a preprocessing stage that generates synthetic samples of the minority class rather than augmenting existing samples in the minority class. The first step of the algorithm starts by choosing a random sample of the minority class and finding its k-nearest neighbors. Out of the k-nearest neighbors the algorithm chooses randomly a subset of neighbors; the subset size is determined based on the amount of oversampling needed. Then, the new samples are generated randomly on the segments between the sample and each of the chosen nearest neighbors.

To use SMOTE for the imbalanced solar flare data set, we first split the set of magnetograms $\mathcal{M}$ into flaring magnetograms $\mathcal{M}_f$ and nonflaring magnetograms $\mathcal{M}_{nf} = \mathcal{M} \backslash \mathcal{M}_f$. The next step includes transforming the magnetograms to their GMM representation. Each magnetogram $x_i \in \mathcal{M}$ has a different number of sunspots denoted by N_i and can be represented as a flat n_i-dimensional vector where

$$n_i = n_i^m + n_i^c = 6 \cdot K \cdot N_i, \tag{5}$$

since

$$n_i^m = 2 \cdot K \cdot N_i \tag{6}$$

is the number of parameters that describe the means and

$$n_i^c = 4 \cdot K \cdot N_i \tag{7}$$

is the number of parameters that describe the collection of covariance matrices. In addition, the ordering of the GMM components within the vector has implications for projection, and it determines the pairing between components for interpolation. Nevertheless, the attributes preserved through the GMM abstraction are not affected by the arrangement of the components, and neighboring points are similar in this regard. Therefore, we have opted for an arbitrary ordering, where positive-polarity components are placed first, followed by the negative-polarity components, and the inner ordering is identical for all components. To make sure that all the representations can be projected onto the same vector space, vectors with a smaller number of entries than $n = \max_i\{n_i\}$ are padded with zeros. Once we project all $x_i \in \mathcal{M}$ onto the Euclidean space $\mathbb{R}^n$, we can use the SMOTE algorithm to sample new points between the minority samples. These new samples were then rendered as images 512×512 pixels in size, and we emphasize again that while this size may be different than the size of either HARP that generated the SMOTE samples, the normalization by HARP size guarantees that relative distance between oversampled sunspots is preserved. Figure 2 shows several new samples on the segment between the GMM representations of HARP 377 (from Figure 1) and HARP 4781.

3.3. Maintaining Physical Properties via Global Optimization

While physical rules that govern sunspots, ARs, and flares are far from being understood (Leka 2022), the obliviousness of the SMOTE process just described to the physical reality of the objects in the magnetogram images may lead to synthesized ARs that will mislead their eventual use. As a general rule, we

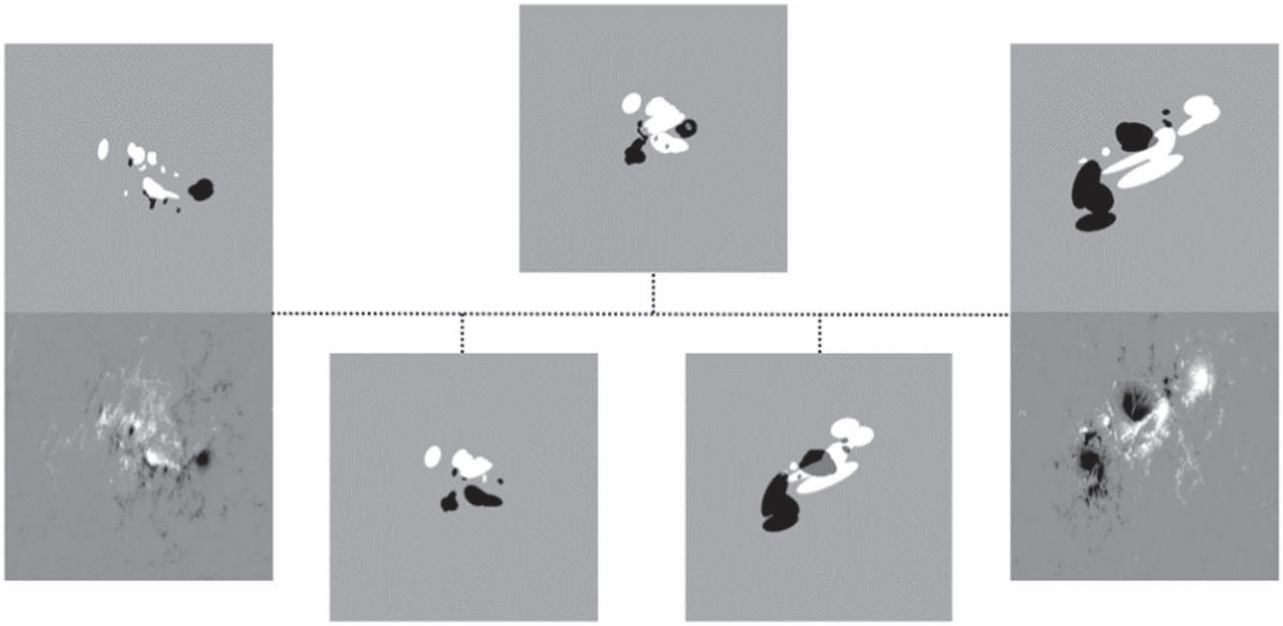


Figure 2. An illustration for SMOTE process and synthetic oversampled samples on the interpolation segment. The leftmost image shows HARP 4781 on 2014 November 6 at 17:36:00 TAI and its corresponding GMM model as confidence ellipses. The rightmost image shows HARP 377 on 2011 February 15 at 01:36:00 TA and its GMM representation (from Figure 1). The middle images illustrate synthetic oversampled GMM points sampled on the interpolation segment between the GMM representations of the two HARPs. The position of the samples along the segment represents how the properties of the two HARPs are weighted in the interpolation, including the sunspots' location, orientation, and size. Note that while HARPs depicted here have similar image sizes, this is not necessary for the SMOTE process, as normalization by HARP size is instrinsic to the process.

only wish to generate physically plausible AR representations, or what we call admissible samples. In practice, we seek to generate such new representations whose physical properties match reality. Hence, once the SMOTE process generates a sample, we now seek to perturb it as needed such that its physical properties converge to an admissible sample, and in particular to the properties of flaring magnetograms drawn from the training set. And while other properties could be used as well, here we attempted to preserve (1) the total absolute magnetic flux and (2) the total length of the PILs, which are the lines where the direction of the magnetic field changes, and previous work argued for their importance as a marker for a solar event (Wang et al. 2020). Searching the space of representations while preserving these properties amounts to solving an optimization problem, for which we used particle swarm optimization (PSO; Kennedy & Eberhart 1995).

PSO is an iterative optimization method used to seek the globally optimal solution to optimization problems by simulating the social behavior of a group of particles in the search space. Each particle represents a possible solution to the problem that updates its position based on its own experience and the experience of its peers. While there are different versions of how these updates are made, it is often defined based on three vectors that each particle i maintains: its current position p_i, its current velocity v_i, and its best position $p_{i,b}$ encountered so far. In addition, the global best position p_g of the swarm is updated in each iteration. The velocity vector v_i at each iteration is then updated as

$$v_i \leftarrow w \cdot v_i + c_1 \cdot r_1 \cdot (p_{i,b} - p_i) + c_2 \cdot r_2 \cdot (p_g - p_i), \quad (8)$$

where $w, c_1, c_2 \in \mathbb{R}$ are weights that combine the three factors. The parameters r_1 and r_2 are two random values generated from the uniform distribution $U(0, 1)$ at each iteration. Note that this definition formalizes the drive to move the particle from its current position based on the knowledge gained by the particle and by the swarm as a whole, keeping randomness in the process to balance between exploitation and exploration and provide yet another mechanism to escape local minima. With the new velocity determined in each iteration, an update of the particle position operation is performed in a straightforward way:

$$p_i \leftarrow p_i + v_i. \quad (9)$$

With PSO as the global optimization scheme, a population of synthetic oversampled points is generated and assigned to PSO particles, and the objective function we seek to optimize (i.e., minimize) involves the two physical features mentioned above. The total absolute magnetic flux is approximated by adding up the areas of the white and black ellipses, representing the positive and negative magnetic polarities, respectively. The total length of the PILs is obtained by summing the lengths of the skeletons of the areas where ellipses with opposing polarity intersect. Both features are computed from the GMM representation of several real AR images drawn randomly from the train set. Formally, given a positive magnetogram x and a synthetic oversampled point $\hat{x}$, the objective function we seek to minimize becomes

$$F(x, \hat{x}) = |g(x) - g(\hat{x})| + \lambda \cdot |l(x) - l(\hat{x})|, \quad (10)$$

where $g: \mathbb{R}^{W \times H} \rightarrow \mathbb{R}$ is a function that calculates the total absolute magnetic flux, while $l: \mathbb{R}^{W \times H} \rightarrow \mathbb{R}$ calculates the corresponding total PIL length, for either the magnetogram or the GMM representation. $\lambda \in \mathbb{R}$ is a weight parameter to balance the importance of both measures. The combination of SMOTE with PSO thus ensures the synthesis of physically admissible samples and can be extended to include any other computable features one wishes to preserve in the generation process. To ensure that the new samples adhere to the specified visual dissimilarity constraint in Equation (1), at the end of the

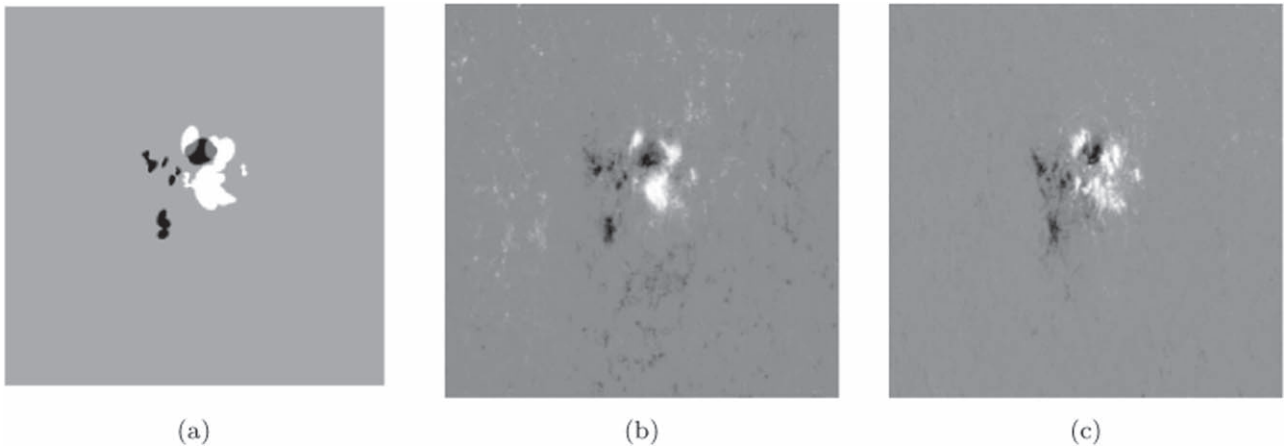


Figure 3. Generation of synthetic images from the label image. (a) The label image resulting from the PSO process (see Figure 2) and given to the image synthesis processes. (b) Output of the Pix2Pix model. (c) Output of the diffusion model.

synthesis procedure (with or without the optimization phase) the dissimilarity distance is tested and samples that do not satisfy it are disregarded. Figure 3(a) shows a result of the optimization process with 150 particles and the features of HARP 4781 from Figure 2.

3.4. Realistic AR Image Synthesis with Image-to-image Translation

As implied above, the modeling of the magnetograms, either before or after the SMOTE and the PSO optimization, provides a labeled image with four semantic categories (e.g., Figure 2): quiet-Sun regions (bright gray background), positive (white) and negative (black) magnetic concentrations, and overlapping areas (dark gray) where ellipses with opposing polarity overlap, often around PILs. This is not yet a realistic magnetogram image that can endow data sets, a result we obtain by an additional step of image translation. Following the review in Section 1, we explore and develop this mapping using one of two deep generative models. These two models effectively minimize the disparity between the distributions of generated data and the real data (Kwon et al. 2022), just as we sought in our formal optimization (see Equation (1)). Training on real magnetograms, it is thus possible to map labeled images to images that closely mirror the intricate visual features of real solar events. For both models, we used the same training data, namely magnetograms of ARs that produced X-class flares. The following sections provide a concise overview of the concepts behind the two models.

3.4.1. Image-to-image Translation Using cGAN

The objective of cGANs is to find a generator G^* such that for a label image $\boldsymbol{x}$, a corresponding magnetogram $\boldsymbol{y}$, and a random noise vector $\boldsymbol{z}$, it holds that

$$G^* = \arg\min_G \max_D \mathcal{L}_{\text{cGAN}}(G, D) + \lambda \mathcal{L}_{L1}(G), \tag{11}$$

where $\mathcal{L}_{\text{cGAN}}(G, D)$ is the binary cross-entropy loss,

$$\mathcal{L}_{\text{cGAN}}(G, D) = \mathbb{E}_{\boldsymbol{x}, \boldsymbol{y}}[\log D(\boldsymbol{x}, \boldsymbol{y})] + \mathbb{E}_{\boldsymbol{x}}[\log(1 - D(\boldsymbol{x}, G(\boldsymbol{x}, \boldsymbol{z})))], \tag{12}$$

and $\mathcal{L}_{L1}(G)$ is the L1-loss,

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{\boldsymbol{x}, \boldsymbol{y}}[\|\boldsymbol{y} - G(\boldsymbol{x}, \boldsymbol{z})\|_{L1}]. \tag{13}$$

The optimal generator essentially generates a data distribution that perfectly matches the distribution of real samples (Goodfellow et al. 2014). The model used for the cGAN-based mapping is Pix2Pix (Isola et al. 2017). The basic components of the generator and the discriminator are convolutional blocks consisting of a convolutional layer and rectified linear unit (ReLU) activation function followed by batch normalization. The generator structure resembles a Unet architecture (Ronneberger et al. 2015) and is composed of eight such convolutional blocks followed by a symmetric path of deconvolutional blocks (Zeiler et al. 2010), which serve to upscale and reconstruct finer details in the generated output. Additionally, skip connections are incorporated within the architecture to facilitate the gradient flow during backpropogation. The discriminator is composed of five convolutional blocks and outputs real/fake map based on its decision.

3.4.2. Image-to-image Translation Using Diffusion Models

Diffusion models learn to sample new images using an iterative denoising process. This denoising process, also called the reverse process, produces samples via a Markov chain, starting from a prior distribution. The objective of the reverse process is to recover the transitions of a perturbation process, known as the forward process, which is another Markov chain gradually perturbing the data and converting it to a different distribution (e.g., a Gaussian). Formally, and borrowing notation from Ho et al. (2020), the forward process is defined on $x_0 \sim p_X$ by progressively adding Gaussian noise to x_0, i.e.,

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{1 - \beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I})$$
$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) = \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}), \tag{14}$$

where the variance $\beta_t \in (0, 1)$ is increasing constantly from 10^{-4} to 0.02. For large enough T the image x_T is approximately pure isotropic Gaussian noise. Since estimating $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ requires all of the data distribution, it is practically intractable, and a neural network is used to approximate it. Then, the

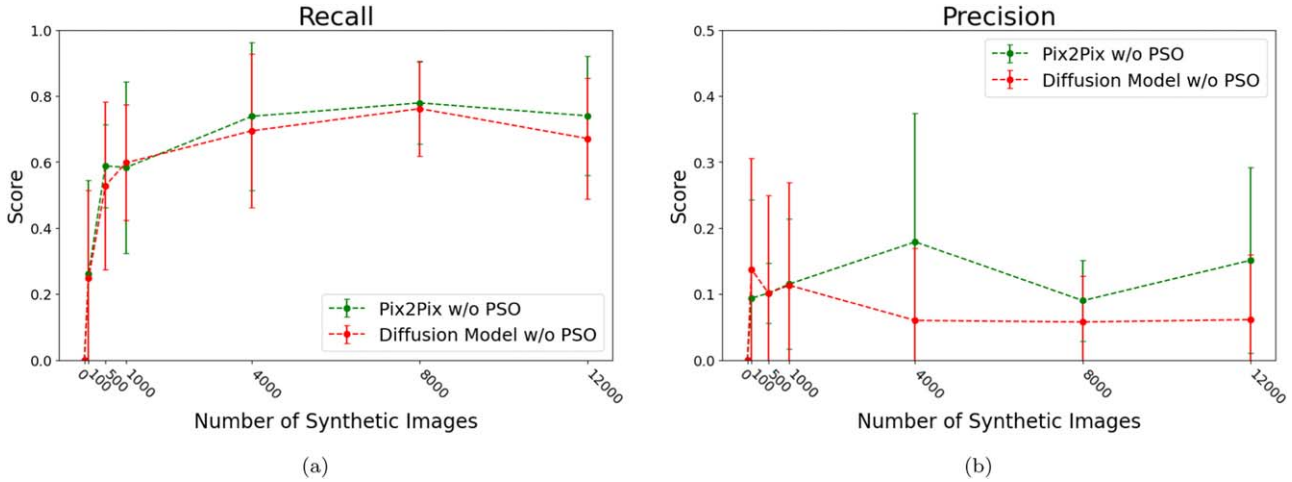


Figure 4. Evaluation of the prediction model varies depending on the number of synthetic images used during the training process. (a) The recall increases when adding more synthetic images to the training. (b) The precision improves with a small addition of synthetic images.

reverse process is defined as

$$p_\theta(\boldsymbol{x}_{0:T}) = p(\boldsymbol{x}_T)\prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$$
$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{x}_{t-1};\, \boldsymbol{\mu}_\theta(\boldsymbol{x}_t,\, t),\, \boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t,\, t)). \quad (15)$$

The initial image in the reverse process $x_T \sim \mathcal{N}(0, \boldsymbol{I})$. For each time step t the network tries to predict the mean $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,\, t)$ and the covariance matrix $\boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t,\, t)$. In practice, the covariance matrix $\boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t,\, t)$ is set to a constant using β_t, which gradually decreases over the course of the reverse process. The mean $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,\, t)$ is determined using a neural network that predicts the noise ϵ_t added at time step t. Then, the training objective is to minimize a mean-squared error loss:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{\epsilon}_t}[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t,\, t)\|^2]. \quad (16)$$

In order to condition the reverse process on the label images generated from the synthetic oversampled GMMs, the training process takes a label image $\boldsymbol{y}$ generated from a magnetogram $\boldsymbol{x}_0$. Then, the forward process is applied to $\boldsymbol{x}_0$ only as described above, and for the reverse process $\boldsymbol{x}_0$ and $\boldsymbol{y}$ are stacked over the channel axis and the network is trained to predict $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{y})$. Then, the loss can be reformulated as

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{y},\boldsymbol{\epsilon}_t}[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t,\, t,\, \boldsymbol{y})\|^2]. \quad (17)$$

Examples of outputs of the image-to-image process are presented in Figure 3.

4. Results

With the pipeline for generating synthetic training data fully defined, we tested its effect on solar flare forecasting using one selected neural model predictor from the prior art (Huang et al. 2018), though we emphasize that using such synthetic data is not limited to any specific data-driven prediction method. The neural architecture we selected integrates two convolutional layers, each succeeded by ReLU activation and pooling layers and followed by three fully connected layers. The network takes in input images sized at 100×100, accommodating variations in magnetogram resolutions through bilinear interpolation during preprocessing. The model yields two output values, representing the probabilities associated with the two classes—flare and no flare. For the optimization process we used stochastic gradient descent with cross-entropy loss. To facilitate this experimental evaluation, we independently trained the network multiple times, each time endowing the training data with a different number of synthetic images. The synthetic images were added at each training session while being drawn randomly from a larger pool of synthetic images that we created offline. The validation set and the test set did not consist of synthetic images, as the goal was to explore the effect of synthetic training data on the classification accuracy of real data. Moreover, the strategy of progressively adding synthetic data was applied several times with different (random) sets of such images, and results report both mean performance and standard deviation.

To evaluate the binary classification model, we express the confusion matrix in terms of the four raw measurements TP, FP, TN, and FN, namely the total number of true positives, false positives, true negatives, and false negatives, respectively. Two widely used performance metrics based on these raw measures are recall and precision. The recall,

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (18)$$

measures the model's capability to accurately identify all true flaring events, while precision describes the accuracy of the positive predictions generated by the model:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (19)$$

The results obtained by the prediction model are presented in Figure 4 and highlight the impact of using synthetic images in the training process, emphasizing, in particular, the relationship between the number of synthetic images and the model's performance. Figure 4(a) demonstrates that as the number of synthetic images increases, the recall metric of the model also improves, first rapidly and then more mildly until saturation, an indication of better identification of true positive samples and higher sensitivity. Note that this happens with either of the image-to-image generative components tested. At the same time, as depicted in Figure 4(b), the precision with no synthetic data is virtually zero but exhibits an initial improvement with

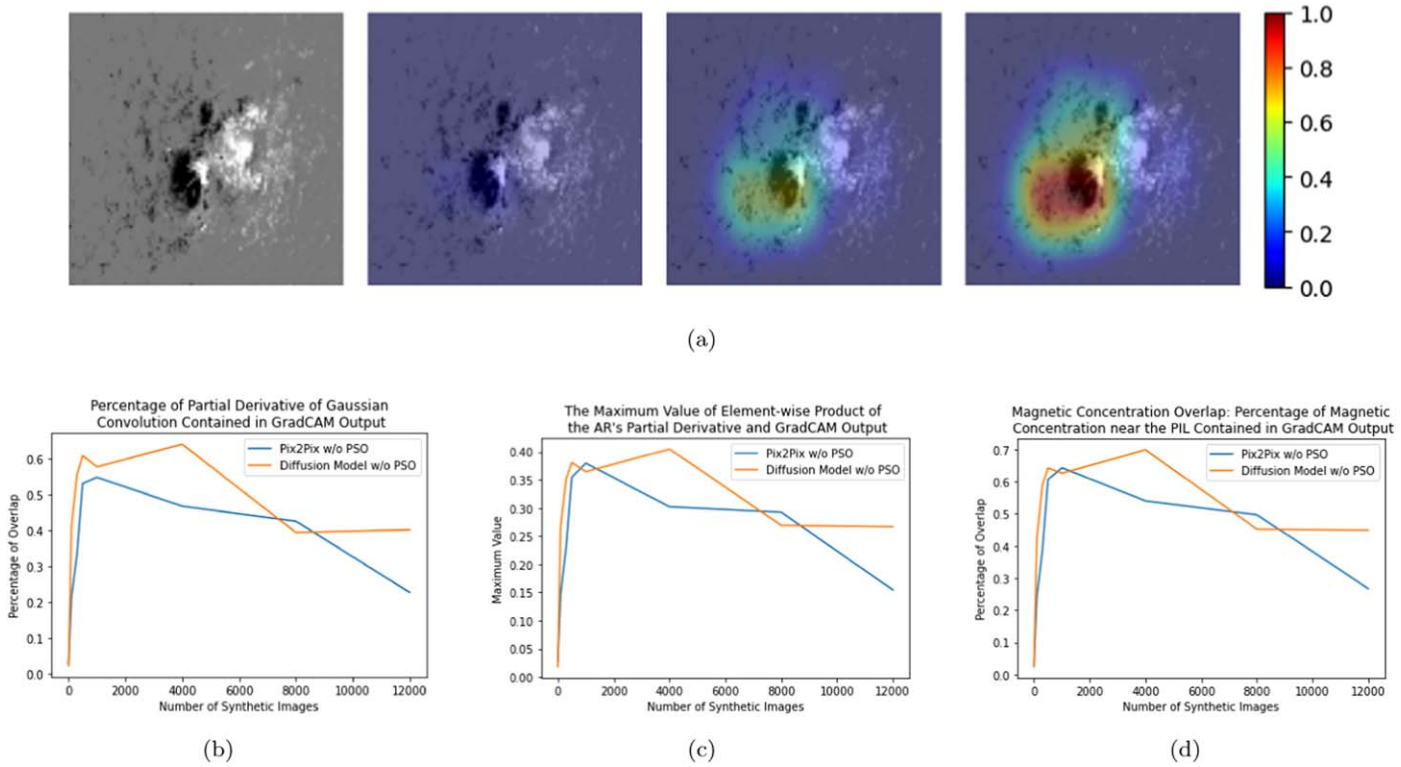


Figure 5. Effects of synthetic images on CNN attention: adding synthetic images to the training process directs the model's attention toward the sunspots next to the PIL. (a) Example of HARP 4698 on 2014 October 26 at 02:24:00 TAI (left), and the heatmaps generated with 0, 300, and 1000 synthetic images added to the training process. (b) The extent of overlap between Grad-CAM heatmaps and PILs was quantified in terms of intensity, represented as the percentage of PILs contained within the heatmaps. (c) The maximum value of the element-wise product between Grad-CAM output and the convolution output may indicate the correlation between high gradients in the AR and the CNN attention. (d) The extent of overlap between Grad-CAM heatmaps and the absolute total magnetic flux of sunspots near the PIL. Panels (b), (c), and (d) exhibit a similar trend: an initial increase followed by a decrease as more synthetic images are added.

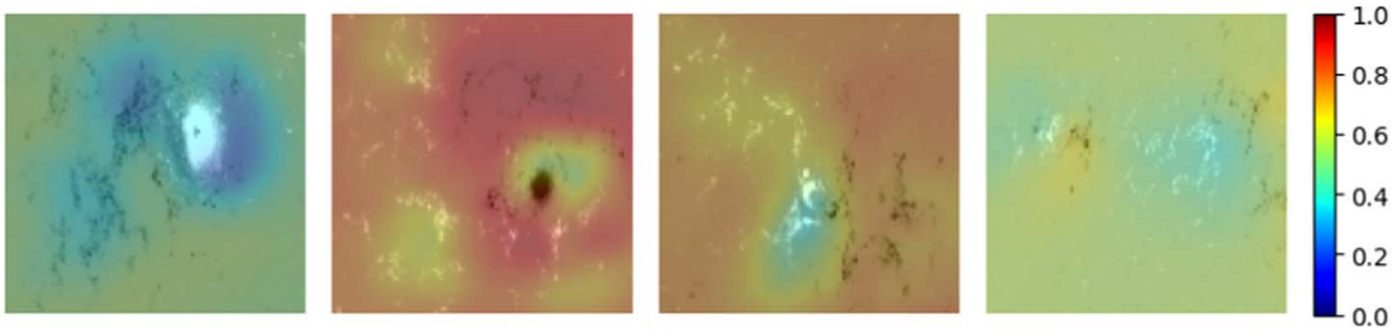
some synthetic images while remaining relatively stable thereafter. Indeed, in the absence of synthetic data integrated into the training set, while utilizing an operational data split, the model struggles to generalize on the unseen magnetograms. Moreover, it exhibited a bias toward the majority class, leading to the prediction of all samples as negative. Despite the relatively low precision owing to the high number of FP, the false-alarm rate (FAR), defined as

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{20}$$

is also low, achieving at max 0.04 ± 0.02. Besides the inherent bias toward the majority class, this means that while the model may occasionally generate FP, the overall rate at which it incorrectly identifies negatives as positives is relatively minimal, and adding synthetic positive samples does not increase the FP significantly.

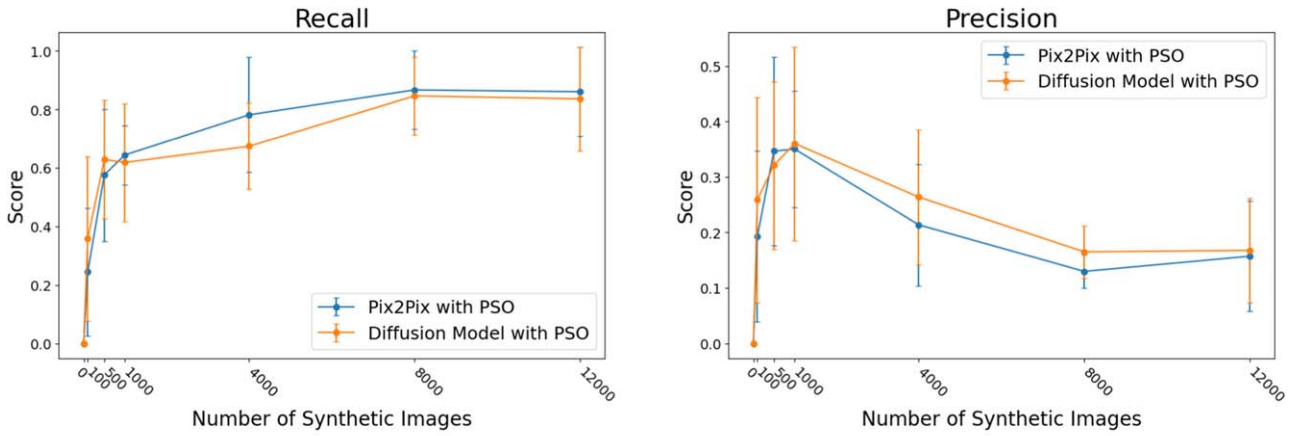
To seek better insights into the features learned by the neural network, we explored visual explanations similar to Yi et al. (2021). The Grad-CAM (Selvaraju et al. 2017) is a technique that calculates the gradients of the output with respect to the feature maps of the convolutional layers. Then, the output of the derivations is projected onto the input image to create a heatmap describing which regions in the image contributed to the output. To test the effect of the synthetic magnetograms on the CNN's training, we examined the Grad-CAM heatmaps created by passing positive samples to the CNNs and deriving them with respect to the positive prediction. Since several instances of the models were trained for each number of synthetic images, the heatmaps created using these models were averaged for each flaring magnetogram. It can be observed from the example in Figure 5 that by adding synthetic images the CNN directs its attention toward regions with magnetic field concentrations near the PILs. Interestingly, though, this effect is particularly pronounced when a small number of synthetic images are used and becomes less pronounced as more images are added. In order to quantify this observation, we ought to examine the effect on the classification of solar flares by features related to the PIL and the sunspots it separates. To do so, we first filtered the image with kernels of partial derivatives of a Gaussian. The resulting smoothed directional edge maps were thresholded to isolate the PILs. Then, we extracted several parameters from the resultant maps, in particular (1) the total absolute magnetic flux of the sunspots next to the largest PIL, (2) the maximum magnetic flux value in the edge map, and (3) the integral (sum) of magnetic activity in the edge map. Later we examined the regions in the negative samples that contribute the most to the negative prediction, and as can be seen in Figure 6, the CNN's attention in this case is concentrated at the quiet-Sun regions.

To get a better grasp on the quality of these parameters as solar flare precursors, we used the same data split as mentioned in Section 2, using the validation set to find the best classifier and to tune its hyperparameters. In our experiments, we noticed that the total absolute magnetic flux of the sunspots next to the largest PIL does not contribute to the prediction, and the results gained using a random forest algorithm with only three parameters are 1.00, 0.04, and 0.05 for the recall, precision, and FAR, respectively. As before, using these parameters as



(a)

Figure 6. Effects of synthetic images on CNN attention: examples of nonflaring magnetograms and their Grad-CAM outputs, with an emphasis on the negative prediction. The regions with the largest contribution to the output are the quiet-Sun regions.



(a) (b)

Figure 7. Evaluation of the prediction model with optimization of the GMM parameters using PSO. (a) The recall increases when adding more synthetic images to the training until saturated. (b) The precision improves with the incorporation of a small number of synthetic images but decreases as more synthetic images are introduced.

flare precursors exhibits a relatively high number of FP. Yet the FAR is low, and the model effectively identifies all positive instances in the test set. Considering these results, the synthetic images were created with emphasis on the total absolute magnetic flux and the PIL length, as described in Section 3.3. Then, the recall and precision were reevaluated by the prediction model with the optimized synthetic images. The results are presented in Figure 7, indicating that the use of synthetic images improves the maximal values of both recall and precision.

Since the validation and the test sets are imbalanced as well, these performance metrics might become highly biased by the frequency of negative samples for all instances. A common alternative under such conditions is the F1-score,

$$\mathrm{F1} = \frac{2 \cdot \mathrm{TP}}{2 \cdot \mathrm{TP} + \mathrm{FP} + \mathrm{FN}} = \frac{2 \cdot \mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}, \tag{21}$$

which is the harmonic mean of the precision and the recall. The F1-score ignores the TN, giving more attention to the positive class and providing a better assessment of both classes for an imbalanced data set. Another metric commonly used for solar flare prediction is the Heidke skill score (HSS; Bloomfield et al. 2012),

$$\mathrm{HSS} = \frac{2[(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FP} \times \mathrm{FN})]}{(\mathrm{TP} + \mathrm{FN})(\mathrm{FN} + \mathrm{TN}) + (\mathrm{TP} + \mathrm{FT})(\mathrm{FP} + \mathrm{TN})}. \tag{22}$$

Even though the HSS considers all elements of the confusion matrix, it converges to the F1-score when the TN approaches infinity. Indeed, in our case, when the number of TN is relatively large, the difference between the two metrics is insignificant.

In our experimental evaluation, the neural network is trained to explicitly maximize the F1-score by evaluating the model after each epoch on the validation set and saving the weights with the best performance. The results indicate that by using the data created without the optimization, the more images incorporated into the training process, the more confident the model tends to be in its positive predictions, resulting in improving the recall and decreasing the precision. By considering the F1-score, which combines both precision and recall metrics into a single score, the results show that using the PSO optimization process leads to a better assessment of the testing set.

Importantly, we evaluated the baseline performance of the selected model (Huang et al. 2018) while training it without any synthetic data. We then implemented several improvements,

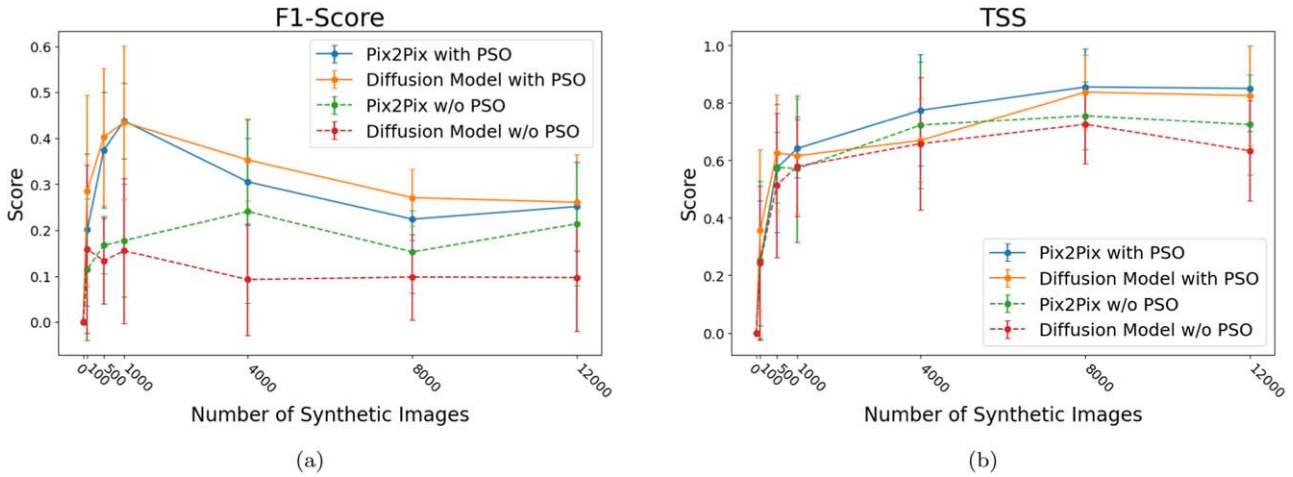


Figure 8. Comparison of the F1-score and the TSS as a function of the number of synthetic images used. Both metrics exhibit higher values when using the optimized synthetic images.

including the use of class weights for the gradient descent algorithm:

$$w_1 = 1 - \frac{\mathrm{P}}{\mathrm{P}+\mathrm{N}}, \quad w_2 = 1 - \frac{\mathrm{N}}{\mathrm{P}+\mathrm{N}}, \tag{23}$$

where w_1 and w_2 are the positive and the negative class weights, respectively. Oversampling of the positive class is also added with and without data augmentations. The augmentation included rotation, horizontal flips, and vertical flips. We found that none of these techniques had any impact on the model's performance.

Finally, we also implemented the true skill statistic (TSS) measure, which quantifies the ability of a binary classification system to forecast rare events by taking into account both recall and FARs:

$$\mathrm{TSS} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} - \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}}. \tag{24}$$

Since the number of TN is relatively large, the TSS is more sensitive to the recall than to the false alarms. Figure 8 summarizes the F1-score and the TSS obtained in all of the methods. Overall, the maximal F1-score observed was 0.43 ± 0.08 when employing 1000 perturbed synthetic images in the training set. At the same time, the TSS reached 0.64 ± 0.10.

5. Conclusion

In this work, we proposed an approach to generate synthetic positive data samples of LOS magnetograms to improve solar flare prediction using machine-learning algorithms. This method addresses the problem of imbalanced solar flare data and uses a combination of (1) GMM representation of ARs, (2) oversampling in the representation space, (3) global optimization technique to obtain and preserve desired features such as physical magnetic properties or flare precursors, and (4) image-to-image mapping back to image space to generate the synthetic data. Despite the supremacy of CNNs in various tasks, our experiments show that they fail to generalize when used under operational flare data splits, and that injecting the synthetic data into the training process of the neural network dramatically improves the results. In addition, our study has demonstrated that incorporating features derived from the interpretation of the CNNs into the synthetic data leads to improved forecasting outcomes.

It is important to note that the proposed process comes with certain limitations. One of the major challenges is that the use of synthetic images may also limit the prediction task in terms of flare class and preflare time window. In addition, despite the features extracted from the interpretation of the CNNs, the physical explainability of the predictions is still vague and the link to the physical flaring mechanism is unclear. Additionally, the generation of the data and the training of the models involved in the process require a considerable amount of time and memory. Changing the flaring prediction settings necessitates retraining of the neural networks, the generative model, and the prediction model, alongside the generation of a new pool of synthetic images. The time of all combined may take several days, not to mention the memory needed to store thousands of images. Indeed, further research is needed to optimize the generation process and explore more flare precursors and features to overcome these limitations and improve the accuracy, reliability, and explainability of solar flare prediction. However, with a proven synthesis method for admissible flare data, such future research may eventually yield completely practical and efficient space weather prediction tools.

Acknowledgments

The authors wish to thank the Institute for Future Technologies, a joint NJIT–BGU partnership of the New Jersey Institute of Technology and Ben Gurion University of the Negev, for its role in initiating this research. This work was supported in part by BGU–NJIT Joint Seed Research Fund, the Helmsley Charitable Trust through the Agricultural, Biological and Cognitive Robotics Initiative, the Marcus Endowment Fund, and the Frankel Center at Ben Gurion University.

References

Abd Elrahman, S. M., & Abraham, A. 2013, J. Netw. Innov. Comput., 1, 332
Ali-Gombe, A., & Elyan, E. 2019, Neurocomputing, 361, 212
Barnes, G., & Leka, K. 2008, ApJ, 688, L107
Blau, Y., & Michaeli, T. 2018, in Proc. of the 2018 IEEE Conf. on Computer Vision and Pattern Recognition (Piscataway, NJ: IEEE), 6228
Bloomfield, D. S., Higgins, P. A., McAteer, R. J., & Gallagher, P. T. 2012, ApJL, 747, L41
Bobra, M. G., & Couvidat, S. 2015, ApJ, 798, 135

Bobra, M. G., Sun, X., Hoeksema, J. T., et al. 2014, SoPh, 289, 3549
Branco, P., Torgo, L., & Ribeiro, R. P. 2016, ACM Comput. Surv., 49, 31
Camporeale, E., Wing, S., & Johnson, J. 2018, in Machine Learning Techniques for Space Weather, ed. Marisa LaFluer (Amsterdam: Elsevier)
Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. 2002, J. Artif. Intell. Res., 16, 321
Chen, Y., Manchester, W. B., Hero, A. O., et al. 2019, SpWea, 17, 1404
Cui, S., & Datcu, M. 2015, in 2015 IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS) (Piscataway, NJ: IEEE), 3719
Dani, T., Muhamad, J., Nurzaman, M., et al. 2022, JPhCS, 2214, 012016
Dash, A., Ye, J., & Wang, G. 2022, Ann. Data Science, in press
Deshmukh, V., Flyer, N., Van Der Sande, K., & Thomas, B. 2022, ApJS, 260, 9
Dhariwal, P., & Nichol, A. 2021, Advances in Neural Information Processing Systems 34, ed. M. Ranzato, (NeurIPS), 8780, https://proceedings.neurips.cc/paper_files/paper/2021/hash/49ad23d1ec9fa4bd8d77d02681df5cfa-Abstract.html
Fernández, A., Garcia, S., Herrera, F., & Chawla, N. V. 2018, J. Artif. Intell. Res., 61, 863
Frid-Adar, M., Diamant, I., Klang, E., et al. 2018, Neurocomputing, 321, 321
Georgoulis, M. K., Bloomfield, D. S., Piana, M., et al. 2021, JSWSC, 11, 39
Georgoulis, M. K., & Rust, D. M. 2007, ApJ, 661, L109
Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al. 2014, Advances in Neural Information Processing Systems 27, ed. Z. Ghahramani et al. (NeurIPS), https://papers.nips.cc/paper/5423-generative-adversarial-nets
Hesse, M., & Cassak, P. 2020, JGRA, 125, e2018JA025935
Ho, J., Jain, A., & Abbeel, P. 2020, Advances in Neural Information Processing Systems 33, ed. H. Larochelle et al. (NeurIPS), 6840, https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html
Huang, X., Wang, H., Xu, L., et al. 2018, ApJ, 856, 7
Iqbal, T., & Ali, H. 2018, J. Med. Syst., 42, 231
Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. 2017, in 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (Los Alamitos, CA: IEEE Computer Society), 5967
Jiao, Z., Sun, H., Wang, X., et al. 2020, SpWea, 18, e2020SW002440
Kennedy, J., & Eberhart, R. 1995, in Proc. of ICNN'95 - Int. Conf. on Neural Networks (Piscataway, NJ: IEEE), 1942
Krawczyk, B. 2016, Prog. Artif. Intell., 5, 221
Kwon, D., Fan, Y., & Lee, K. 2022, Advances in Neural Information Processing Systems 35, ed. S. Koyejo et al. (NeurIPS), 20205, https://proceedings.neurips.cc/paper_files/paper/2022/hash/7f52f6b8f107931127eefe15429ee278-Abstract-Conference.html
Lee, J., & Park, K. 2021, Pers. Ubiquitous Comput., 25, 121
Leka, K. 2022, BAAS, 54, 2022n7i111ap01
Leka, K., & Barnes, G. 2003a, ApJ, 595, 1277
Leka, K., & Barnes, G. 2003b, ApJ, 595, 1296
Leka, K., & Barnes, G. 2007, ApJ, 656, 1173
Leka, K., Park, S.-H., Kusano, K., et al. 2019, ApJS, 243, 36
Li, X., Kou, K., & Zhao, B. 2021, arXiv:2103.05422
Liu, C., Deng, N., Wang, J. T., & Wang, H. 2017, ApJ, 843, 104
Liu, H., Liu, C., Wang, J. T., & Wang, H. 2019, ApJ, 877, 121
Lugaz, N., Liu, H., Hapgood, M., & Morley, S. 2021, SpWea, 19, e2021SW003000
Mickey, D., Canfield, R., LaBonte, B., et al. 1996, SoPh, 168, 229
Mirza, M., & Osindero, S. 2014, arXiv:1411.1784
Moon, T. K. 1996, ISPM, 13, 47
Nichol, A. Q., & Dhariwal, P. 2021, in Proc. of Machine Learning Research 139, Proc. of the 38th Int. Conf. on Machine Learning, ed. M. Meila & T. Zhang (PMLR), 8162, https://proceedings.mlr.press/v139/nichol21a.html
Nie, D., Trullo, R., Lian, J., et al. 2018, ITBE, 65, 2720
Nishizuka, N., Sugiura, K., Kubo, Y., Den, M., & Ishii, M. 2018, ApJ, 858, 113
Nishizuka, N., Sugiura, K., Kubo, Y., et al. 2017, ApJ, 835, 156
Noguchi, A., & Harada, T. 2019, in Proc. of the 2019 IEEE/CVF Int. Conf. on Computer Vision (Piscataway, NJ: IEEE), 2750
Pesnell, W. D., Thompson, B. J., & Chamberlin, P. 2012, SoPh, 275, 3
Reynolds, D. A. 2009, Encyclopedia of Biometrics (New York: Springer), 659
Ronneberger, O., Fischer, P., & Brox, T. 2015, in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, ed. N. Navab et al. (Cham: Springer), 234
Saharia, C., Chan, W., Chang, H., et al. 2022, in ACM SIGGRAPH 2022 Conf. Proc., ed. M. Nandigjav, N. J. Mitra, & A. Hertzmann (New York: Association for Computing Machinery)
Salvatelli, V., Dos Santos, L. F., Bose, S., et al. 2022, ApJ, 937, 100
Scherrer, P. H., Bogart, R. S., Bush, R., et al. 1995, SoPh, 162, 129
Schrijver, C. J. 2007, ApJ, 655, L117
Selvaraju, R. R., Cogswell, M., Das, A., et al. 2017, in 2017 IEEE Int. Conf. on Computer Vision (ICCV) (Piscataway, NJ: IEEE), 618
Serra, J. 1986, CVGIP, 35, 283
Smith, A. R. 1979, in Proc. of the 6th Annual Conf. on Computer Graphics and Interactive Techniques, ed. T. DeFanti et al. (New York: Association for Computing Machinery), 276
Sun, W., Xu, L., Zhang, Y., Zhao, D., & Zhang, F. 2022, RAA, 23, 025003
Tang, R., Liao, W., Chen, Z., et al. 2021, ApJS, 257, 50
Wang, J., Zhang, Y., Webber, S. A. H., et al. 2020, ApJ, 892, 140
Xu, Z., Du, J., Wang, J., et al. 2019, in 2019 IEEE Int. Conf. on Communications (ICC) (Piscataway, NJ: IEEE)
Yi, K., Moon, Y.-J., Lim, D., Park, E., & Lee, H. 2021, ApJ, 910, 8
Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. 2010, in 2010 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (Piscataway, NJ: IEEE), 2528
Zheng, Y., Li, X., Si, Y., Qin, W., & Tian, H. 2021, MNRAS, 507, 3519
Zheng, Y., Li, X., & Wang, X. 2019, ApJ, 885, 73
Zhu, B., Pan, X., vanden Broucke, S., & Xiao, J. 2022, Inform. Sci., 609, 1397