# Surya: Foundation Model for Heliophysics

**Sujit Roy**[1,2,†,‡], **Johannes Schmude**[5,†,‡], **Rohit Lal**[1], **Vishal Gaur**[1], **Marcus Freitag**[5], **Julian Kuehnert**[5], **Theodore van Kessel**[5], **Dinesha V. Hegde**[3,4], **Andrés Muñoz-Jaramillo**[7], **Johannes Jakubik**[5], **Etienne Vos**[5], **Kshitiz Mandal**[1], **Ata Akbari Asanjan**[13], **Joao Lucas de Sousa Almeida**[5], **Amy Lin**[1], **Talwinder Singh**[6], **Kang Yang**[6], **Chetraj Pandey**[6], **Jinsu Hong**[6], **Berkay Aydin**[6], **Thorsten Kurth**[14], **Ryan McGranaghan**[8], **Spiridon Kasapis**[9], **Vishal Upendran**[10], **Shah Bahauddin**[11], **Daniel da Silva**[12], **Nikolai Pogorelov**[3,4], **Campbell Watson**[5], **Manil Maskey**[2], **Madhulika Guhathakurta**[15], **Juan Bernabe-Moreno**[5], **Rahul Ramachandran**[2]

[†]Equal Contribution,
[‡]sujit.roy@nasa.gov, Johannes.Schmude@ibm.com [*]

## ABSTRACT

Heliophysics is central to understanding and forecasting space weather events and solar activity. Despite decades of high-resolution observations from the Solar Dynamics Observatory (SDO), most models remain task-specific and constrained by scarce labeled data, limiting their capacity to generalize across solar phenomena. We introduce Surya, a 366M parameters foundation model for heliophysics designed to learn general-purpose solar representations from multi-instrument SDO observations, including eight Atmospheric Imaging Assembly (AIA) channels and five Helioseismic and Magnetic Imager (HMI) products. Surya employs a spatiotemporal transformer architecture with spectral gating and long–short range attention, pretrained on high-resolution solar image forecasting tasks and further optimized through autoregressive rollout tuning. Zero-shot evaluations demonstrate its ability to forecast solar dynamics and flare events, while downstream fine-tuning with parameter-efficient Low-Rank Adaptation (LoRA) adaptation shows strong performance on solar wind forecasting, active region segmentation, solar flare forecasting, and EUV spectra. We believe that this is the first foundation model designed on the native resolution of SDO data.

[*][1]Earth System Science Center, University of Alabama in Huntsville, AL, USA; [2]NASA Marshall Space Flight Center, Huntsville, AL, USA; [3]Department of Space Science, University of Alabama in Huntsville, AL, USA; [4]Center for Space Plasma and Aeronomic Research (CSPAR), University of Alabama in Huntsville, AL, USA; [5]IBM Research; [6]Georgia State University; [7]Southwest Research Institute; [8]NASA Jet Propulsion Laboratory; [9]Princeton University; [10]SETI Institute; [11]Laboratory for Atmospheric and Space Physics, University of Colorado Boulder; [12]NASA Goddard Space Flight Center; [13]Research Institute for Advanced Computer Science, Universities Space Research Association, USA [14]NVIDIA Corp., Santa Clara, USA Caltech, Pasadena, USA; [15]NASA Science Mission Directorate

## 1 INTRODUCTION

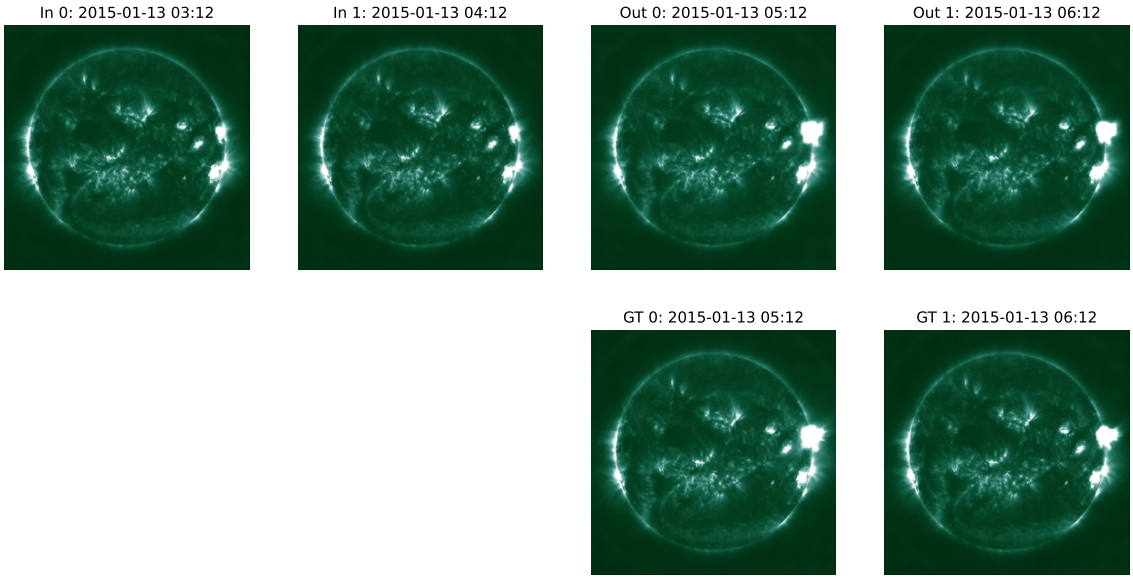


Figure 1: Solar flare on January 13, 2015, as predicted by Surya. AIA, 94Å. Top row, left two columns are model inputs ("In"). Top row, right two columns are model outputs ("Out"). Bottom row shows corresponding ground truth ("GT").

Heliophysics is the study of the Sun and its impact on the solar system. Its relevance goes significantly beyond a purely scientific interest in our star. Indeed, one of the most critical aspects of heliophysics is the study of space weather, which is driven by the Sun's activity, such as solar flares and coronal mass ejections (CMEs). These events can have significant effects on modern technology and consequently society (Boteler, 2001). Space weather can disrupt satellite communications and GPS signals, interfere with aviation navigation systems, and degrade the quality of radio transmissions. Strong geomagnetic storms can induce damaging currents in electrical grids, leading to widespread power outages and costly infrastructure repairs (Oughton et al., 2017). In space, increased radiation levels pose a severe hazard to astronauts and spacecraft electronics, potentially shortening mission lifespans or endangering human health. Even on Earth, high-latitude flights can be exposed to elevated radiation levels, and critical services that depend on precise timing, such as financial transactions, weather forecasting, and emergency response, can be severely impacted. These impacts create a broad network of stakeholders, including space agencies, satellite operators, aviation authorities, power companies, defense organizations, and emergency management agencies. Understanding and predicting solar activity is thus essential for safeguarding critical systems and ensuring resilience to space weather hazards. Beyond its practical importance, heliophysics offers a unique window into fundamental plasma physics in extreme environments and deepens our understanding of how solar activity shapes planetary atmospheres, informing studies of planetary habitability both on Earth and across distant exoplanets (Schrijver et al., 2019).

Foundation models (FMs) are large, pretrained models that learn general-purpose representations from vast datasets, and they have revolutionized practical applications, most notably in natural language processing and computer vision by capturing rich, transferable features that enable rapid adaptation to diverse down-

stream tasks with minimal task-specific training. Their potential is now increasingly recognized in scientific disciplines that generate complex, multimodal data, and heliophysics is not an exception Roy et al. (2024). Despite decades of extensive solar observations from a fleet of ground-based telescopes and space-based satellites, current ML applications in heliophysics research often depends on task-specific data and models trained from scratch (for a detailed review Asensio Ramos et al. (2023)), which can be inefficient, prone to overfitting, and limited by the scarcity of labeled data—especially for rare events, which are often the most interesting ones.

A heliophysics FM can address these limitations by learning generalized, more physics-aware representations from the wealth of high-resolution, multi-instrument solar data, enabling robust performance across a broad spectrum of predictive, diagnostic, and analytical tasks. By leveraging pre-training on large-scale solar observations, FMs can mitigate the supervision bottleneck, reducing the need for labeled data and improving real-world performance in forecasting rare or extreme events. Their versatility and adaptability allow fine-tuning for diverse downstream applications, including solar feature detection (e.g., coronal holes, sunspots, active regions), transient event forecasting, and heliospheric modeling, with minimal additional supervision. FMs also offer improved generalization, effectively handling data distribution shifts that hinder traditional models, and their scalability ensures compatibility with increasingly large and high-resolution datasets for multi-scale modeling of solar dynamics. Furthermore, their capacity for multi-modal integration can unlock more accurate and comprehensive predictions of solar activity and heliospheric conditions. Collectively, these capabilities position foundation models as a transformative step toward next-generation data-driven heliophysics.

In this paper, we introduce Surya (Sanskrit for Sun), our heliophysics foundation model trained on multi-channel native-resolution observations from Solar Dynamics Observatory (SDO) (Pesnell et al., 2012). We detail the pretext task design, pretraining protocols, and engineering approaches used to efficiently process and learn from large-scale, multimodal solar data. Our model architecture and training pipeline integrate modern self-supervised learning paradigms to capture high-fidelity, general-purpose solar representations. We demonstrate the model's versatility across a suite of downstream applications, including active region emergence prediction, active region segmentation, solar flare forecasting, and solar wind forecasting. The evaluations confirm that the model produces consistent, transferable representations that maintain competitive or improved performance across diverse tasks, underscoring its potential as a scalable backbone for both scientific discovery and practical forecasting applications.

## 2 SURYA FM

### 2.1 SDO DATA

SDO launched on February 11, 2010, as the first mission of NASA's Living With a Star Program (LWS). It is dedicated to advancing our understanding of solar variability and its impacts on Earth and near-Earth space. By observing the solar atmosphere at high spatial and temporal resolution across multiple wavelengths, SDO investigates how the Sun's magnetic field is generated, structured, and released, driving the solar wind, energetic particles, and irradiance variations. These observations aim to enable a predictive capability for solar variations that affect life and technological systems on Earth, collectively known as space weather.

For this study, we utilize observations from two primary imaging instruments onboard SDO: the *Atmospheric Imaging Assembly* (AIA)(Lemen et al., 2012), which records extreme ultraviolet (EUV) and ultraviolet (UV) photometric intensities, and the *Helioseismic and Magnetic Imager* (HMI)(Schou et al., 2012), which provides spectropolarimetric measurements for deriving vector magnetic fields and line-of-sight velocities. Both instruments utilize $4096 \times 4096$ pixel CCD detectors, capturing full-disk images with sub-arcsecond spatial resolution.

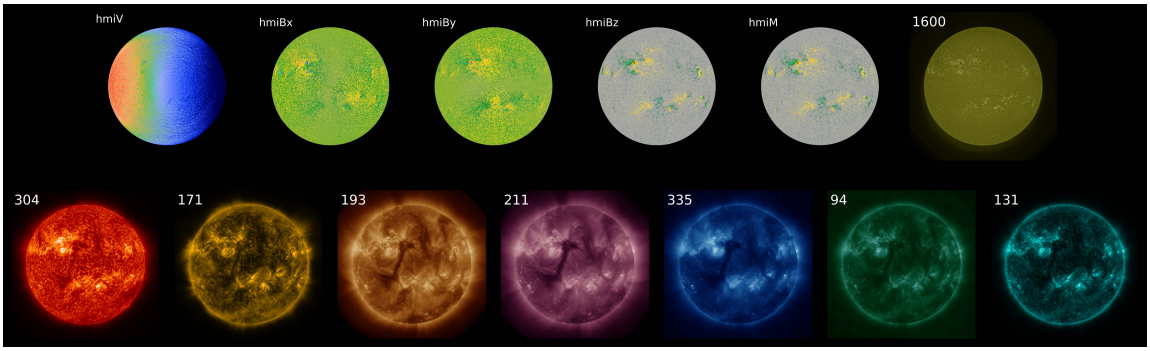


Figure 2: SDO solar imagery: solar coronal EUV images and solar surface magnetic field map (13 channels)

We selected SDO data from AIA and HMI because they uniquely provide the highest spatial resolution, full-disk solar observations with high cadence. SDO offers a nearly continuous and uniform dataset spanning 15 years, covering more than a full solar cycle. The consistent and simultaneous observations of the Sun by HMI and AIA make SDO a natural choice for building an FM, as its uniquely high spatial and temporal resolution, sustained coverage, and diverse data products provide a rich basis for addressing multiple data-driven learning problems in heliophysics and space weather, while enabling a wide spectrum of science use cases. Furthermore, as an ongoing mission, SDO ensures that the foundation model remains applicable to future unseen data, thereby extending its utility beyond historical records.

#### 2.1.1 Dataset Preparation

We construct our dataset by homogenizing multi-channel AIA EUV/UV imagery from the `aia.lev1_euv_12s` and `aia.lev1_uv_24s` series with HMI magnetograms from the `hmi.M_720s` (line-of-sight) and `hmi.B_720s` (vector) series. All data are downloaded from the Joint Science Operations Center (JSOC[1]). The temporal cadence is standardized to 12 minutes, matching the HMI vector magnetic field data series, with AIA observations temporally co-aligned within $\pm 2$ minutes where possible and subject to quality flag checks. Table 1 summarizes the properties of the individual channels used in this study, including their instrument-specific temporal and spatial resolutions as well as the dynamic range of the measurements.

The Level 1.0 AIA data are preprocessed by updating spacecraft pointing, aligning the y-axis of the image with solar north, rescaling to a uniform $0.6''/\text{pixel}$ grid, bringing solar disk center to the image center, and normalizing for exposure-time variability. Time-dependent instrument degradations are corrected using calibration factors, with values clamped to the instrument's dynamic range to avoid saturation artifacts. The Level 1.5 HMI data, with a pixel scale of $0.5''/\text{pixel}$, are preprocessed to align with the preprocessed AIA images at a $0.6''/\text{pixel}$ image scale by reprojecting using bilinear interpolation, which also corrects for the HMI roll angle. Finally, the solar disk radius in both datasets is fixed to a constant value to remove the nonphysical variations in the apparent solar disk radius due to the elliptical orbit of the Earth.

The resulting database consists of spatially and temporally aligned, preprocessed, multi-wavelength solar images that are well-suited for machine learning applications. This harmonized processing pipeline ensures that variations within the dataset are predominantly physical in origin rather than instrumental or geometric,

[1] `http://jsoc.stanford.edu`

thereby enabling robust downstream applications in heliophysics and space weather. Figure 2 presents an example set of preprocessed images from the eight AIA and five HMI channels used in this study.

Table 1: Key properties of the SDO/AIA and SDO/HMI instruments. Cadence values refer to both instrument-native and standardized dataset cadence.

| Instrument | Resolution | Cadence (instr./dataset) | Dynamic range | Channels / Measurements |
|---|---|---|---|---|
| AIA | $1.2''$ (≈725 km) | 12 s, 24 s / 12 min | 0–16,383 DN | 94, 131, 171, 193, 211, 304, 335, 1600 Å |
| HMI | $1.0''$ (≈870 km) | 45 s, 12 min / 12 min | $\pm$4,500 G (B), $\pm 10^4$ m/s (V) | $B_x$, $B_y$, $B_z$, $B_{\text{los}}$, $V_{\text{los}}$ |

### 2.1.2 Dataset Statistics

Our database contains ML-ready SDO data captured from May 13, 2010, to July 31, 2024. During this interval, there are about 2.9% data unavailable (18,261 out of 623,280 total timestamps) for *hmi.M_720s* series. The processed level 1.5 AIA and HMI data are stored in netCDF files (float32 format), with one file per hour. Each file contains data from five 12-minute timesteps within that hour, with the data shape of [5, 13, 4096, 4096]. Each netCDF file is about 2.2 GB, and the total size of the data for training is about 257 TB.

**Data details** We define a solar observation as a multi-channel, co-registered raster representing simultaneous measurements from AIA and HMI onboard SDO. Each observation is encoded as a three-dimensional tensor: $X \in \mathcal{R}^{13\times 4096\times 4096}$
Where:

- The first dimension indexes the physical measurement channel,
- The second and third dimensions represent the spatial domain, corresponding to the solar disk with a native resolution of $0.6\,''/\text{pixel}$.

Channel Composition Information: The 13 channels are composed as follows:

- 8 AIA Channels: 7 EUV (94 Å, 131 Å, 171 Å, 193 Å, 211 Å, 304 Å, 335 Å) and 1 UV (1600 Å).
- 5 HMI Channels: 1 LOS magnetic field map B_los, 3 vector magnetic field component maps (B_x, B_y, B_z), and 1 Doppler velocity map V_los.

Each AIA channel provides EUV or UV intensity measurements of the solar atmosphere, capturing coronal and chromospheric emission at distinct temperature regimes. HMI-derived channels capture photospheric vector magnetic field observations and line-of-sight plasma motion.

**Train/test split** For the train–test partition, observations from 2011 to 2019 were segmented by day-of-year. Days 1–14 and 32–45 of each year were excluded as temporal buffers to mitigate potential information leakage due to short-term temporal autocorrelation in solar activity. The interval spanning days 15–31 of each year was reserved exclusively for the test set, while all remaining days from day 46 onward were assigned to the training set.

**Normalization** SDO data is highly skewed due to the predominance of pixels depicting the quiet sun and the presence of a small number of pixels depicting extreme events such as solar flares. Modeling such data often benefits from log-transformation before applying the standard scaling. Specifically, we consider the signum-log transform $\text{sign}(x) \times \log 1\text{p}(|\mathbf{X}|)$, which can be applied equally to the strictly positive AIA channels and the HMI channels that contain both positive and negative values (where $log1p(x)$ is the natural

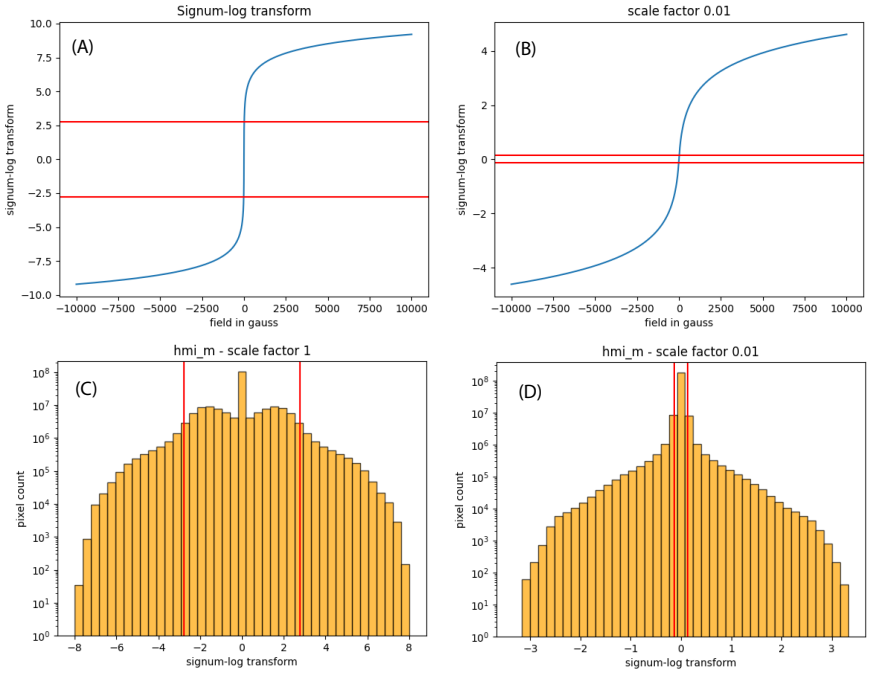


Figure 3: Scaled signum-log transform reduces the range over which noise dominates. (A) Standard signum-log transform. (B) Scaled signum-log transform with scale factor $10^{-2}$. (C) Histogram of HMI_m channel values after signum-log transform. (D) Histogram of HMI_m channel values after scaled signum-log transform with scale factor $10^{-2}$. Red lines indicate the limits of the range where the magnetic field magnitude is smaller than the noise level of 15 Gauss.

logarithm of $1 + x$). However, as figure 3 demonstrates, the standard signum-log transform comes with the downside that it stretches the inherently noisy low-magnetic field range of HMI channels at the expense of data points with good signal-to-noise ratio. Given the inherent noise level of the HMI_m channel of 15 Gauss, about 1/3 of the transformed scale would be occupied by data that is dominated by the inherent detector noise, while the dynamic range of extreme events may be compressed too much. As a compromise between applying the full signum-log or no transformation at all, we propose applying a scale factor of $10^{-2}$ to the raw data before applying the signum-log. Following this transformation, standard-scale the data per channel. Thus, the complete transformation per channel is

$$\frac{\text{sign}(\mathbf{X})\,\text{log1p}(|10^{-2}x|) - \mu}{\sigma + \epsilon}. \tag{1}$$

Here, $\mu$ and $\sigma$ are the per channel global means and standard deviations following the signum-log transformation; $\epsilon$ is a small constant, we use $10^{-8}$. A histogram of the resulting data distribution in 3 (D) shows that the inherently noisy values are now occupying a much narrower range and the model can focus on learning the important intermediate and extreme activity ranges of the sun.

### 2.2 PRETEXT TASK AND BASELINES

#### 2.2.1 PREDICTING FUTURE SDO IMAGERY AS A PRETEXT TASK

Prior to considering AI architectures, one needs to decide on a pretext task. Here one can take inspiration from prior work in computer vision, but also other scientific domains such as earth observation or atmospheric physics. Although there is a myriad of pretext tasks (i.e. self-supervised training methodologies) the most dominant ones are arguably masked reconstruction (He et al., 2022), contrastive objectives (Chen et al., 2020), and finally autoencoders. In addition, recent work in atmospheric physics moreover shows that AI models can learn complex temporal dynamics purely from data by regressing onto a future time step (Pathak et al., 2022; Lam et al., 2023). On the other hand, masked reconstruction has been successfully used in earth observation (Jakubik et al., 2023; Szwarcman et al., 2024) while Schmude et al. (2024) used a mixed objective combining reconstruction with forecasting. Yet in the end we note that our downstream tasks are frequently in nature and choose forecasting as a pretrainig objective. In heliophysics, Majid et al. (2024) trained with a 12-hour ahead forecasting objective while Walsh et al. (2024) explored both MAEs and autoencoders. Both papers did so at 512 by 512 pixel resolution.

For future work let us point out that a band-to-band translation pretext task, as was used with great success by Jakubik et al. (2025) in earth observation might be a strong alternative if combined with a temporal objective. Especially given the inherent multi-modality of SDO data.

In either case, we settle on using two timestamps, 60 minutes apart, as input to the model and train the model by regressing on SDO data 60 minutes in the future. As is now standard in models for atmospheric physics, this is followed by a second phase of pretraining where the outputs of Surya are used as inputs to predict 120, 180, ... minutes into the future. This is generally referred to as "autoregressive rollout tuning". During rollout tuning, we average the loss across all steps. See section 2.4.2 for details on our pretraining protocol.

To formalize this, we denote observed frames as $\mathbf{X}_t$ and the model as $f_\theta$. Then we train Surya with an MSE objective as follows:

$$\left[\mathbf{X}_{t+1} - f_\theta\left(\mathbf{X}_t, \mathbf{X}_{t-1}\right)\right]^2. \tag{2}$$

If we further denote model output as $\hat{\mathbf{X}}_{t+1} = f_\theta(\mathbf{X}_t, \mathbf{X}_{t-1})$, autoregressive prediction takes the form

$$\hat{\mathbf{X}}_{t+2} = f_\theta\left(\hat{\mathbf{X}}_{t+1}, \mathbf{X}_t\right) = f_\theta\left(f_\theta\left(\mathbf{X}_t, \mathbf{X}_{t-1}\right), \mathbf{X}_t\right). \tag{3}$$

The rollout loss is then

$$\frac{\left[\mathbf{X}_{t+1} - f_\theta\left(\mathbf{X}_t, \mathbf{X}_{t-1}\right)\right]^2 + \left[\mathbf{X}_{t+2} - f_\theta\left(f_\theta\left(\mathbf{X}_t, \mathbf{X}_{t-1}\right), \mathbf{X}_t\right)\right]^2}{2} \tag{4}$$

and equivalent for multi-step rollouts. At inference time, this scheme enables longer forecasts from two initial observations an hour apart.

#### 2.2.2 BASELINE SCORES

With the pretraining task decided, it is important to identify a series of baseline scores to compare against. The purpose is primarily to put model losses into context during training and development. The first of these is simply persistence. Using the notation from section 2.2.1, the persistence forecast is simply $\hat{\mathbf{X}}_{t+1} = \mathbf{X}_t$. Technically speaking, note that one can obtain an improved persistence forecast by averaging over multiple timestamps along the lines of $\hat{\mathbf{X}}_{t+1} = \left(\mathbf{X}_t + \mathbf{X}_{t-1}\right)/2$. The reason for this is that the averaging procedure smooths out sharp features. Especially due to the sun's rotation, sharp features lead to a double penalty on MSE loss of persistence. However, our persistence scores serve simply as a baseline, so we use $\hat{\mathbf{X}}_{t+1} = \mathbf{X}_t$.

The other relevant baseline is given by solar rotation. Rather than hard-coding rotation via a known equation, we decide to learn the effect from data. To do so, we assign coordinates $[-1, 1] \times [-1, 1]$ to the 4096 x 4096

Table 2: Baseline scores for 1 hour ahead forecasting. In model units.

| Baseline | Parameters | Loss (MAE) |
|---|---|---|
| Persistence | N/A | 0.594044030 |
| Learned flow | 642 | 0.337624282 |

pixels. That is, the bottom left pixel is $(-1, -1)$, the top right pixel $(1, 1)$ etc. Then we train a very small MLP that takes these coordinates of each pixel as input and yields a vector as output. The result is a learned vector field along which we interpolate the data. As the input is data independent, this trains a constant flow-field along which the data is moved to effectively learn the rotation. If we denote the coordinates of each pixel as $\mathbf{x}$, the MLP as $M_\theta$ and the operation to interpolate[2] along a vector as $\nearrow$, we train this baseline as follows:

$$\left[\mathbf{X}_{t+1} - M_\theta\left(\mathbf{x}\right) \nearrow \mathbf{X}_t\right]^2. \tag{5}$$

Our MLP consists of two linear layers and an internal dimension of 128 with GELU activation. Thus, it has a total of 642 parameters. Table 2 shows the baseline scores in the model units of equation equation 1.

## 2.3 Architecture

### 2.3.1 Overview of Surya Model Architecture

The Surya Foundation Model is a 2-D transformer architecture for high-resolution forecasting of SDO imagery and solar dynamics. It integrates frequency-domain filtering with efficient multi-scale attention to capture both fine-scale and global spatio-temporal dependencies. The architecture, shown in fig. 4, consists of two spectral gating blocks, eight long-short attention blocks, and a decoder block for reconstruction in the physical domain. See section A.1 for ablation studies.

**Tokenization** The raw input data to Surya is SDO data from 13 different channels scaled according to equation 1. Using two timestamps, the input data initially has the shape $13 \times 2 \times 4096 \times 4096$. To tokenize the data, we simply flatten the channel and temporal dimensions and use a simple linear layer. The internal dimension is $D = 1280$. Given a patch size of $16 \times 16$, we end up with $N = 65,536$ tokens:

$$\begin{gathered} C \times T \times H \times W \mapsto N \times D \\ (13 \times 2) \times (256 \times 16) \times (256 \times 16) \to 65,536 \times 1,280. \end{gathered} \tag{6}$$

Surya uses a Fourier position embedding.

**Spectral Gating Blocks** Let $\mathbf{X} \in \mathbb{R}^{B \times N \times D}$ denote the embedded spatiotemporal tokens, where $B$ is the batch size. Each spectral block reshapes $\mathbf{X}$ into $\mathbf{X}_s \in \mathbb{R}^{B \times H_p \times W_p \times D}$, where $H_p = W_p = 256$ are height and width in token space, and applies a 2-D real Fast Fourier Transform (rFFT):

$$\widetilde{\mathbf{X}} = \mathcal{F}(\mathbf{X}_s), \tag{7}$$

followed by modulation with a learnable complex-valued weight $W_c \in \mathbb{C}^{H_f \times W_f \times D}$:

$$\widetilde{\mathbf{X}}' = \widetilde{\mathbf{X}} \odot W_c. \tag{8}$$

[2]Implementation wise, we use the `F.grid_sample` method from PyTorch for interpolation. I.e. `F.grid_sample(x, flow_field, mode="bilinear")`, where `x` is the data $\mathbf{X}$ and `flow_field` the output of the MLP $M_\theta$.

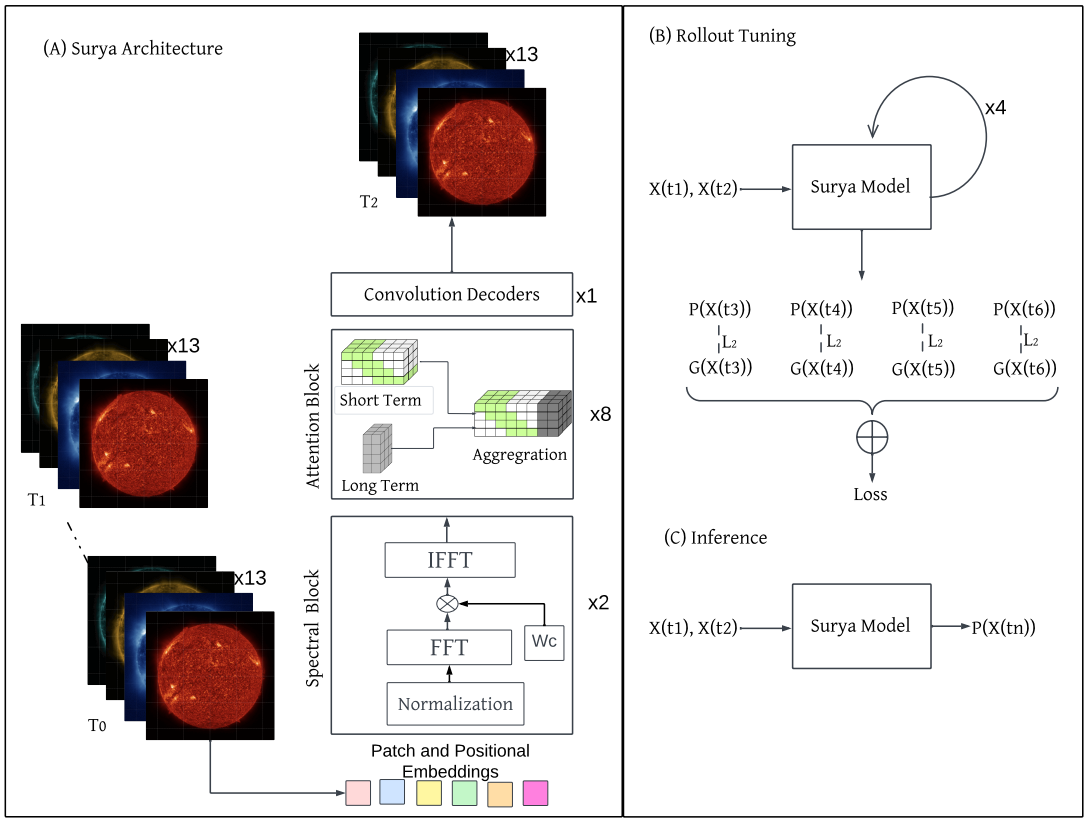


Figure 4: (A). The architecture of the Surya Foundation Model which uses 2 spectral blocks, 8 attention blocks based on long-short attention, and 1 decoder block. We are using a learnable weight parameter ($W_c$) after performing $FFT$ for estimating the weights of frequency components. We then perform an inverse $FFT$ to transform the information back into physical space, as described in Patro et al. (2025). The attention head is designed on the principle of long-short attention Zhu et al. (2021), where we calculate short-term attention by sliding window, and long-range attention by dynamic projections. (B). Overview of 4-step rollout finetuning, where the model is given a 2-timestep input and then predicts the next 4 steps autoregressively. C. Inference of the model with 2 time step input and an hour forecast.

Here, $H_f = 256$, $W_f = 129$, and $\odot$ denotes element-wise complex multiplication.[3] This adaptively re-weights frequency components to emphasize informative spectral bands and suppress noise. The result is transformed back to the physical domain via inverse rFFT:

$$\mathbf{X}'_s = \mathcal{F}^{-1}(\widetilde{\mathbf{X}}'), \tag{9}$$

[3]The weight matrices $W_c$ are large at $84,541,440$ real parameters. See section A.1 for a discussion. Note also that the imbalance between $H_f$ and $W_f$ arises from $\mathbf{X}$ being real and the use of the real FFT here.

and refined through a residual connection and feed-forward network (FFN):

$$\mathbf{X}_{\ell+1} = \mathbf{X}_\ell + \mathrm{FFN}(\mathrm{LN}(\mathbf{X}'))\,. \tag{10}$$

$\mathrm{LN}(\cdot)$ denotes Layer Normalization and $\mathbf{X}'$ is $\mathbf{X}'_s$ reshaped back to sequence order.

**Long-Short Attention Blocks** The attention backbone consists of $L = 8$ layers that fuse *local* and *global* attention pathways, following the principle of the Long–Short Transformer Zhu et al. (2021) adapted for 2-D spatiotemporal tokens.

Local (short-range) attention operates within non-overlapping spatial windows of size $w \times w$, capturing fine-scale dependencies. For each window $\Omega$, queries, keys, and values are restricted to $\Omega$, yielding:

$$\mathrm{Attn}_{\mathrm{short}}(\mathbf{X}) = \mathrm{Softmax}\!\left(\frac{Q_\Omega K_\Omega^\top}{\sqrt{d_k}} + \Delta_{\mathrm{rpe}}\right) V_\Omega, \tag{11}$$

where $\Delta_{\mathrm{rpe}}$ is an optional relative positional bias, and $d_k$ is the per-head dimension.

Global (long-range) attention uses dynamic low-rank projection: keys and values are content-adaptively compressed into a rank-$r$ basis,

$$\overline{K} = \alpha K, \quad \overline{V} = \alpha V, \tag{12}$$

where $\alpha \in \mathbb{R}^{B\times h\times r\times N}$ is a learned mixing weight obtained from the keys. Queries then attend to $(\overline{K}, \overline{V})$ across the entire sequence:

$$\mathrm{Attn}_{\mathrm{long}}(\mathbf{X}) = \mathrm{Softmax}\!\left(\frac{Q\overline{K}^\top}{\sqrt{d_k}}\right)\overline{V}. \tag{13}$$

The two branches are normalized to align their scales and then concatenated along the key-value dimension:

$$\mathbf{X}' = \mathrm{Concat}(\mathrm{Attn}_{\mathrm{long}}, \mathrm{Attn}_{\mathrm{short}})\,, \tag{14}$$

followed by a residual MLP:

$$\mathbf{X}_{\ell+1} = \mathbf{X}_\ell + \mathrm{MLP}(\mathrm{LN}(\mathbf{X}'))\,. \tag{15}$$

This design efficiently combines localized modeling with global context aggregation, achieving multi-scale representation learning at reduced complexity as proposed by Zhu et al. (2021).

**Decoder** The decoder is a lightweight projection that maps the final token representation $\mathbf{X}_L$ back to the spatial domain:

$$\mathbf{Y} \in \mathbb{R}^{B\times C\times H\times W} = \mathrm{Unembed}(\mathbf{X}_L), \tag{16}$$

where $C$ is the number of output channels.

### 2.4 Pretraining

#### 2.4.1 Scaling

In its final configuration, Surya comprises 366 million parameters. Considering that two timestamps of SDO data comprise

$$2 \times 13 \times 4096 \times 4096 \times 32\text{bit} = 1.7\text{GB} \tag{17}$$

of data, GPU memory is a primary concern. At a patch size of $16 \times 16$, we are dealing with $65,536$ tokens. To deal with memory pressure, we use FSDP, mixed precision as well as gradient checkpointing. The model's input and output layers operate in `float32`, yet the transformer layer uses `bfloat16`. Note that the spectral gating layers explicitly cast to `float32` for the FFT operations. Table 7 shows 1 hour ahead performance and memory consumptions of Surya as well as ablations and baselines. Here, the "No spectral gating" ablation replaces the two spectral gating blocks with additional long-short attention. As the table shows, the use of the spectral gating layers leads to the same loss at 6% less GPU memory.

### 2.4.2 PRETRAINING PROTOCOL

As outlined above, pretraining of Surya followed what has become a standard approach in AI forecasting models in atmospheric physics (Pathak et al., 2022; Lam et al., 2023). That is, Surya was trained with a two phase approach consisting of one step ahead forecasting and subsequent rollout tuning.

In phase one, we trained Surya for 160,000 gradient descent steps on 128 NVIDIA A100 GPUs. The model is trained with batch size 1 (per GPU), making an effective batch size of 128. Throughout training, we use cosine annealing to modify the learning rate from $10^{-4}$ to $10^{-5}$. Note that we did not find a need for a dedicated warm-up period to stabilize training. We clipped gradients at $0.1$. We use the `AdamW` optimizer from PyTorch with default values for its parameters `betas`, `eps`, and `weight_decay`.

The rollout tuning phase imposes additional demands on GPU memory. We now use gradient checkpointing after all ten spectral and transformer layers. First, we train two steps ahead – i.e., the initial step plus one autoregressive step – for 20,000 gradient descent steps at a constant learning rate of $10^{-5}$. Then we subsequently train three, four, and five steps ahead for 4,000 steps each at a learning rate of $10^{-6}$. This unbalanced schedule was partially motivated by the fact that a single gradient descent step takes longer and longer lead times. At the same time, loss curves showed continuous improvement for the 2 step ahead case until step 20,000. Generally, we used 64 GPUs for rollout tuning.

## 2.5 ZERO-SHOT EVALUATION

### 2.5.1 PREDICTING FUTURE SDO DATA (FORECASTING)

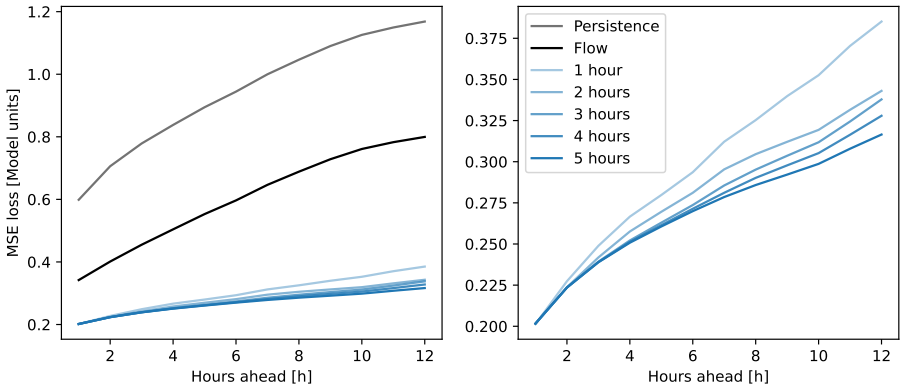


Figure 5: Forecasting performance of Surya. MSE across all channels in model units. Left hand side shows persistence and the flow model used to capture solar rotation. Both sides show the output of phase 1 of pretraining ("1 hour") as well as various stages of rollout tuning: 2 hour ahead rollout tuning to 5 hour ahead rollout tuning. The latter of these is the last stage of pretraining and thus shows the zero-shot performance of Surya.

We can get a first understanding of Surya's capabilities, as well as the effectiveness of the pretraining protocol, by predicting future SDO images. The result of this evaluation can be seen in Figure 5. Rollout tuning improves the performance at 12 hours ahead by $10.9$, $12.3$, $14.9$, and $17.8\%$ respectively when compared against the state of the model after phase 1. At the longer lead times, the performance improvements of Surya due to rollout tuning have not yet reached saturation. From a technical perspective, Surya could be tuned up to 24 hours ahead with no modifications to the code when using an 80 GB A100 GPU. The lim-

iting factor here was actually data loading speed with GPUs waiting for data. It might be worthwhile to do additional rollout tuning with longer lead times in the future.

### 2.5.2 VISUAL PREDICTION OF SOLAR FLARES

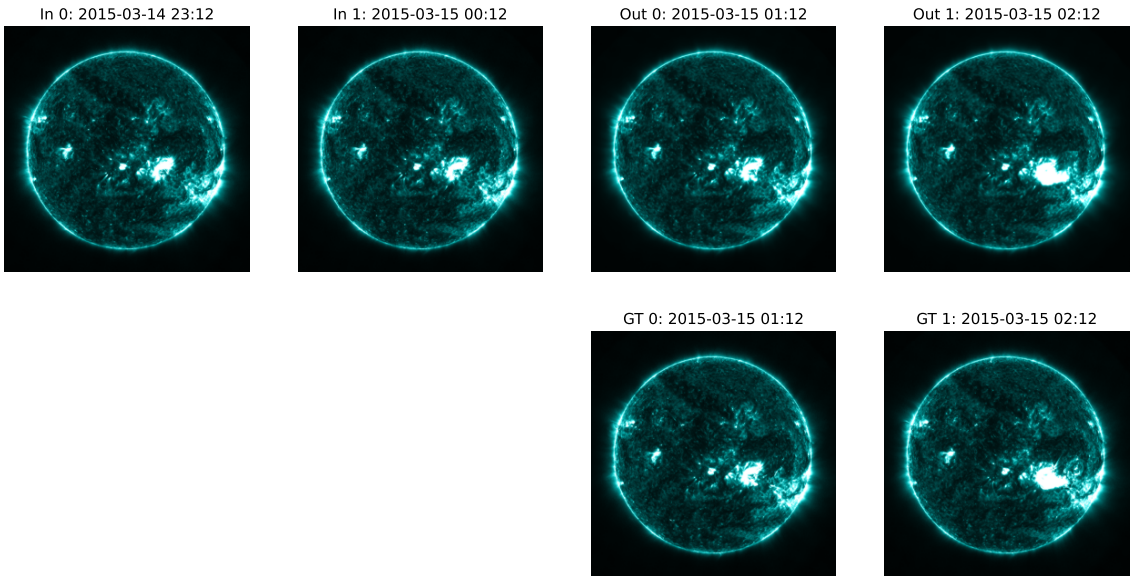


Figure 6: St. Patrick's Day Event as modeled by Surya. AIA, 131Å. Top row, left two columns are model inputs ("In"). Top row, right two columns are model outputs ("Out"). Bottom row shows corresponding ground truth ("GT").

To further test and evaluate Surya's capabilities to predict future SDO images – and thus the future state of the sun – we initialize Surya just prior to a solar flare event. To start, we do this on training data. I.e. imagery that Surya saw during its pretraining period. Figure 6 shows this for the St. Patrick's Day Event. As reported in Wu et al. (2016), "SOHO/LASCO C2 [...] recorded a CME [...] that erupted from the southwest at 01:48 UT." Compare the model output of Surya shown in figure 6. It exhibits a clear evolution of model output between 01:12 and 02:12 UTC, fitting the actual observation.

Naturally, this is training data. As a matter of fact, it stands to reason that a model pretrained with an MSE objective will at very least *remember* an event such as that of figure 6. After all, the flare is a clearly visible feature with much brighter pixels than the rest of the sun, having a strong impact on the MSE score. Thus, the critical question is whether we can make similar observations on testing data.

Figures 8 and 14 as well as 1, 12 and 13 answer this to the positive. They show model inputs, outputs, and ground truth at different initialization times on January 7 2014 and January 13 2015. While these are not exactly in our testing period of section 2.1.2 as they lie in the "temporal buffer", they are far enough from actual training data such that leakage and contamination is not a concern.

The above figures show a strong visual feature evolving in channels 94 Åand 131 Å. Yet that alone does not allow us to characterize these as "flare events". To do that, we plot the integrated extreme UV emissions for the January 13 2015 case in figure 7. Note that the time series here is a composite of multiple runs of

Surya. Each was initialized with two timestamps 60 minutes apart (as usual) and run two steps (2 hours) into the future. Notably, for all channels except 1600 Å, the integrated output of Surya tracks the ground truth closely. In particular, close to the rapid changes around 04:24 UTC. We consider the results discussed in this section as *visual* prediction of solar flare events. This is in contrast to the conventional approach discussed via fine-tuning in section 2.6.

#### 2.5.3 Blurring of sharp features

Surya is trained as a fully deterministic model with an MSE objective. A well-known characteristic of such models is that they tend to blur sharp features. Figures 9 and 10 show local crops of Surya's outputs AIA 171 Åas well as HMI_m compared to ground truth. Given the aforementioned choices regarding Surya's architecture and pretraining objective, we observe indeed a loss of the finest details for HMI data as well as some blurring in the AIA bands. In theory, one might be able to address this by transitioning to a probabilistic model. This could be done via diffusion techniques, or via noise injection and a suitable loss function as in Lang et al. (2024). We experimented with the latter, injecting noise into the model during the long-short attention blocks using adaptive layer norm and training on a CRPS objective. Yet we found that while it did yield improved detail, in particular in the HMI channels, it lead to occasional token-level artifacts.

### 2.6 Downstream Evaluation

#### 2.6.1 Fine-tuning architecture and fine-tuning protocol

**Fine-tuning architecture** As discussed in section 2.2.1, Surya was pretrained with a forecasting pretext task. This was motivated by the dynamical properties of the solar surface and atmosphere as well as the fact that some of our downstream tasks have a clear forecasting flavor. On the other hand, this can lead to a challenge when fine-tuning the model with frozen weights. The issue is that the representation learned by pretraining – the activations of the last long-short attention layer – can be assumed to be inherently *local*. After all, in pretraining, we apply a linear layer to these activations to regress on the image seen by SDO at this specific location in the future. This is in difference to a masked reconstruction approach. Here, the encoder learns a representation from which the decoder can reconstruct the entire image. This implies that each token learns a representation which – collectively with the other tokens – can be used for global reconstruction.[4] With this in mind, we implement multiple fine-tuning architectures for Surya.

In one category, we consider problems where the target is either a set of categories or a global classification or regression – think solar flare forecasting, solar wind forecasting, or EVE. Here, we enable global average pooling, global max pooling, attention pooling, transformer pooling, and finally the use of a global class token. Let us briefly discuss each of these:

Global average and max pooling are straightforward: One aggregates the activations of the last transformer block and applies one or several linear layers. It is here where the comments from the preceding paragraph most apply: For a frozen model trained on a forecasting task, global max pooling can be assumed to return the most prominent (brightest) pixels in the output. And indeed, if one uses global max or global average pooling with frozen weights for solar flare forecasting, one obtains reasonable performance quickly; yet said performance quickly reaches a maximum that can be surpassed by approaches that really consider patterns rather than the maximal local activation values.

[4]The reader might complain that the representation learned by a masked autoencoder is still local. And indeed, if one plots the activations returned by the encoder, they still contain the data at that location. So one should take the above with a grain of salt. Still, the main point is that a model that regresses on future state will inherently be forced to contain local information in its ultimate transformer layers.

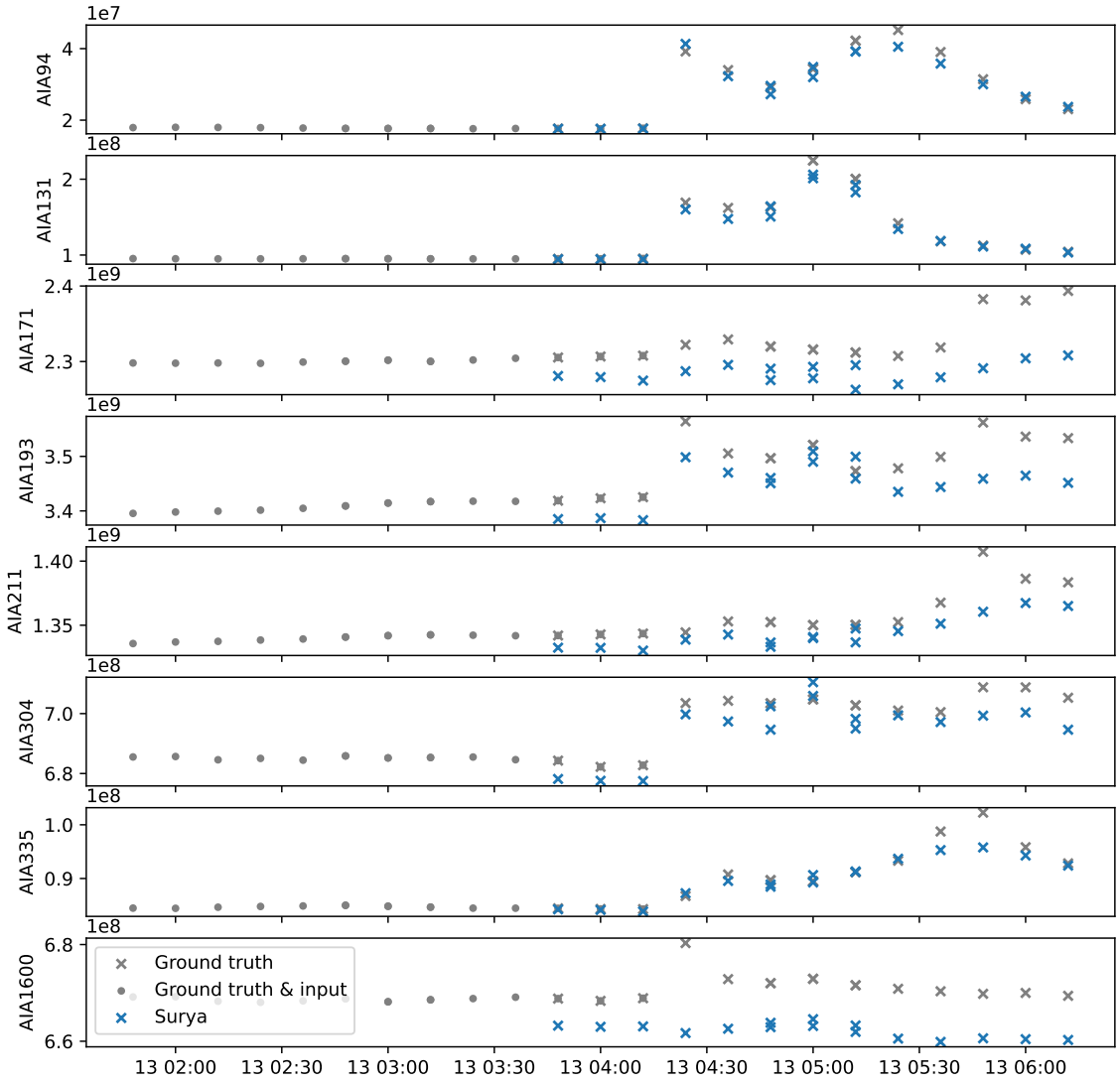


Figure 7: Integrated extreme UV emissions for a solar flare event on January 13, 2015. That is, the plot shows the per-channel sum over all pixels in model inputs and outputs. The model was initialized for multiple lead times on January 13, 2015, and ran up to 2 hours (2 steps) into the future. The time series is the join of all those runs (which is why certain times show multiple outputs). This matches the visual output of figures 1 and 8.

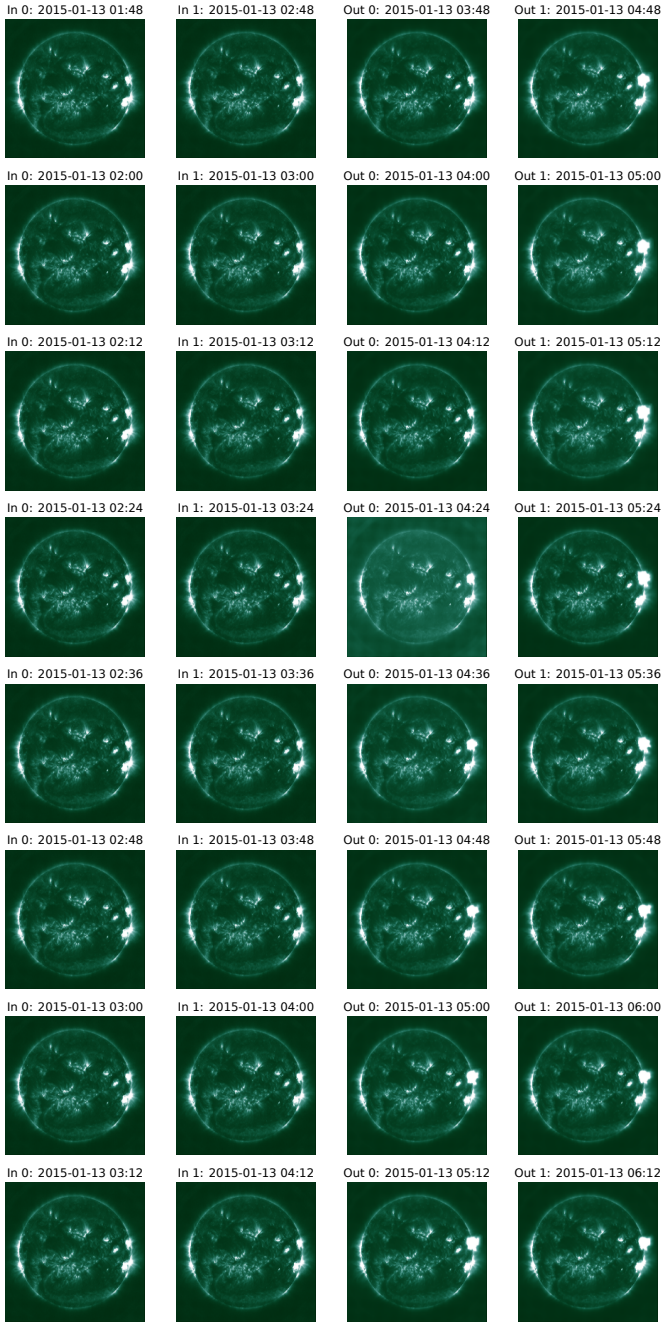


Figure 8: Surya inputs and outputs with different initializations for January 13, 2015.

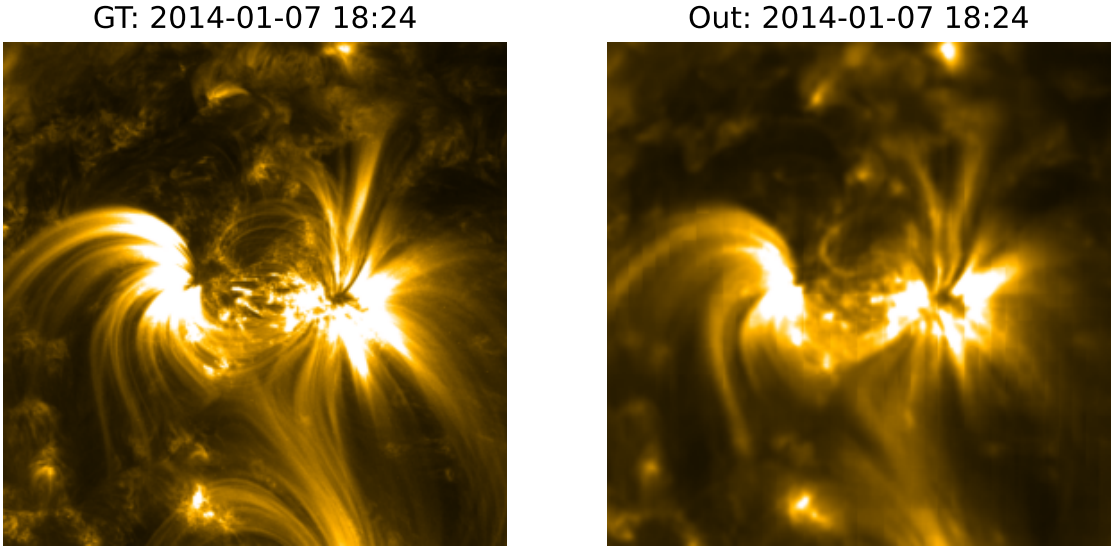


Figure 9: Ground truth data (left) and corresponding outputs from Surya for one hour ahead forecasting. 171Å. The images shows a local crop of 600 by 600 pixel.

Attention pooling applies another attention layer before summing activations. Transformer pooling introduces an additional attention block with a dedicated class token. For downstream tasks that concern rare events – see solar flare forecasting – we find that attention and transformer pooling introduce too many parameters and overfit heavily.

Finally, we consider the global class token. Here, we simply introduce a non-local token after the spectral gating layers. This token is initialized with learnable weights. In contrast to Swin-transformers, an advantage of the long-short attention codebase is that the introduction of this class token is relatively straightforward.

PARAMETER-EFFICIENT FINE-TUNING (LORA) Note that all of the above can be combined with LoRA fine-tuning of the model. And indeed, this is how we obtain our strongest fine-tuning results. In other words, to adapt *Surya* efficiently, selected linear maps (e.g., attention projections and MLP layers) are augmented with a low-rank residual while the pretrained weights remain frozen. For any targeted weight $W_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, LoRA parameterizes

$$W \;=\; W_0 \;+\; \frac{\alpha}{r}\, BA, \qquad A \in \mathbb{R}^{r \times d_{\text{in}}},\; B \in \mathbb{R}^{d_{\text{out}} \times r}, \tag{18}$$

with rank $r$ and scaling $\alpha$. During fine-tuning, only the adapter parameters $(A, B)$ and any task-specific head(s) are updated; $W_0$ is fixed. Let $\mathcal{S}$ index the set of adapted layers. The resulting optimization problem is

$$\min_{\theta_{\text{head}},\, \{A_\ell, B_\ell\}_{\ell \in \mathcal{S}}} \; \mathcal{L}_{\text{total}}\big(\theta_0, \theta_{\text{head}}, \{A_\ell, B_\ell\}\big) \;+\; \lambda_{\text{lora}} \sum_{\ell \in \mathcal{S}} \big(\|A_\ell\|_F^2 + \|B_\ell\|_F^2\big), \tag{19}$$

where $\lambda_{\text{lora}}$ regularizes the low-rank updates. This reparameterization can be interpreted as learning a task-specific, low-dimensional perturbation in the local tangent space of $W_0$, preserving the inductive biases of the pretrained model while controlling adaptation capacity via $r$ and $\alpha$.

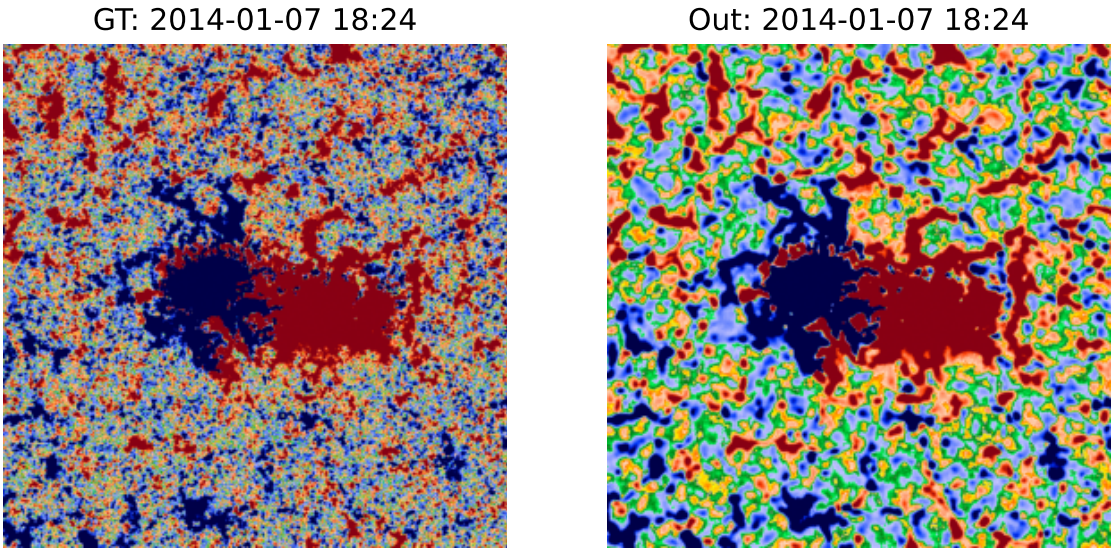


Figure 10: Ground truth data (left) and corresponding outputs from Surya for one hour ahead forecasting. HMI_m. The images shows a local crop of 600 by 600 pixels.

### 2.6.2 Active region Segmentation

Table 3: AR Segmentation results comparing baseline models with Surya

| Model | Params | IoU | Dice Coeff |
|---|---|---|---|
| Unet | 9.2 M | 0.688 | 0.801 |
| Surya | **4.1 M** | **0.768** | **0.853** |

Solar Active Regions (ARs) are magnetically complex structures associated with flares and CMEs. A key feature within ARs is the Polarity Inversion Line (PIL), the boundary separating opposite magnetic polarities, whose strong and sheared forms are robust precursors of eruptions Ji et al. (2023). Accurately segmenting ARs containing PILs is thus critical for space weather forecasting and advancing our understanding of solar magnetic complexity.

Traditional AR/PIL detection pipelines based on thresholding and morphology are interpretable but brittle—sensitive to noise, parameter choices, and unable to capture the thin, filamentary structures of PILs. This motivates a deep learning–based segmentation framework that learns robust, multi-scale representations directly from solar data.

We construct the ARPIL dataset using full-disk SDO/HMI line-of-sight magnetograms ($4096 \times 4096$) adapting the method in Cai et al. (2020). Positive and negative polarity maps are generated using $\pm 50$ G thresholds, filtered to remove regions smaller than 100 pixels, dilated with a 10-pixel kernel, and intersected to extract PILs. Only ARs with PILs are retained, yielding 119,454 binary masks spanning January 2011–December 2024.

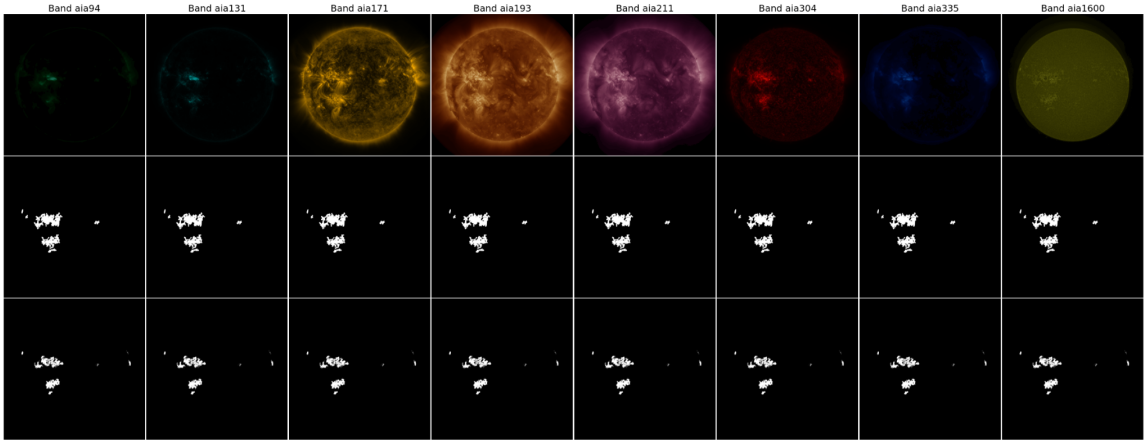


Figure 11: Active region Segmentation Results. Top Row: SDO data on Date: 2014-02-01, Time: 08:12, Middle Row: Model Output segmentation mask, Bottom Row: Ground Truth

As baselines, we compare a standard UNet with our finetuned Surya foundation model using LoRA adaptation. With just 4.1M trainable parameters, Surya achieves higher segmentation quality (IoU 0.768, Dice 0.853) than UNet (IoU 0.688, Dice 0.801).

### 2.6.3 Solar Flare Forecasting

As part of our downstream application evaluation, solar flare prediction is posed as a binary classification problem, where the objective is to determine whether a significant flare (M- or X-class) will occur within the next 24 hours following a time-point observation at $t$, similar to Pandey et al. (2023). The prediction window is defined as $[t, t + 24h)$ hours. The instance is labeled positive (flaring) if the peak X-ray flux of the strongest flare in $[t, t + 2)$ exceeds $\theta_{\text{max}} = 10^{-4}$ W/m$^2$.

Given solar observations $\mathbf{x}_t \in \mathbb{R}^{C\times H\times W}$ (or $\mathbb{R}^{T\times C\times H\times W}$ for temporal sequences), the classifier predicts:

$$f(\mathbf{x}_t) \to y_t \in \{0, 1\}, \tag{20}$$

with

$$y_t = \begin{cases} 1, & \text{if a strong flare occurs in } [t, t + 2) \\ 0, & \text{otherwise.} \end{cases}$$

We considered True Skill Statistic (TSS), Heidke Skill Score (HSS) and F1 score as evaluation metrics. TSS measures the ability to distinguish between flare and non-flare events. Ranges from -1 (inverse prediction) to +1 (perfect prediction), with 0 indicating no skill, and is defined as

$$\text{TSS} = \frac{TP}{TP + FN} - \frac{FP}{FP + TN}$$

HSS evaluates performance relative to random chance, considering both hits and false alarms. Ranges from -1 to 1, and is defined as:

$$\text{HSS} = \frac{2(TP \cdot TN - FP \cdot FN)}{(TP + FN)(FN + TN) + (TP + FP)(FP + TN)}$$

F1 Score is the harmonic mean of precision and recall of positive class (i.e., flaring), balancing both false positives and false negatives, defined as :

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2TP}{2TP + FP + FN}$$

Table 4: The results of flare forecasting evaluation comparing baseline models used in literature Pandey et al. (2023; 2024) with finetuned Surya-based ones

| Model | TSS | HSS | F1 |
|---|---|---|---|
| AlexNet | 0.358 | 0.398 | 0.454 |
| ResNet50 | 0.018 | 0.028 | 0.055 |
| Surya | **0.436** | **0.522** | **0.561** |

#### 2.6.4 SOLAR EUV SPECTRA PREDICTION

Accurately forecasting solar Extreme Ultraviolet (EUV) irradiance is essential for advancing space weather prediction, as it directly influences satellite functionality, communication infrastructures, and navigation systems. The challenge lies in modeling irradiance over 1343 spectral bands (Woods et al., 2012), which encode intricate spatial and temporal dependencies captured through solar observations.

Given multi-channel solar imagery $\mathbf{x}_t \in \mathbb{R}^{C \times H \times W}$ at time $t$, the objective is to estimate a continuous vector of EUV irradiance measurements $y_t \in \mathbb{R}^{1343}$:

$$y_t = f(\mathbf{x}_t). \tag{21}$$

Table 5: EVE spectra prediction results comparing baseline models with Surya

| Model | MSE | MAE | MAPE |
|---|---|---|---|
| AlexNet | 0.0001376225 | 0.0064941053 | 1.7340041399 |
| ResNet50 | 0.0008386846 | 0.0255825799 | 4.6996207237 |
| Surya | **0.0001287334** | **0.0042972970** | **1.4598474503** |

#### 2.6.5 SOLAR WIND FORECASTING

Solar wind forecasting aims to predict the solar wind speed at a given spatial point, specifically within a 4-day prediction window following an observation time $t$. Precise forecasting of solar wind speeds is fundamental for mitigating the adverse effects of space weather on satellite communication systems, navigation systems, and electrical grids on Earth.

This dataset comprises scalar measurements of solar wind speeds near Earth, recorded hourly from 2010-01-01 through 2023-12-31, resulting in a temporally rich dataset with substantial coverage of solar cycles (Gloeckler, 2023). The solar wind speed values exhibit significant variability, ranging from $2.4 \times 10^2$ km/s to $8.8 \times 10^2$ km/s.

Given solar observation data (such as AIA and HMI multi-channel solar imaging data) represented by $\mathbf{x}_t$ at observation time $t$, the task is to predict the scalar solar wind speed at time $t + \Delta t$, where $\Delta t = 4$ days:

$$y_{t+\Delta t} = f(\mathbf{x}_t),$$

where $\mathbf{x}_t \in \mathbb{R}^{C \times H \times W}$ represents the multi-channel, high-resolution input imagery data at time $t$, and $y_{t+\Delta t} \in \mathbb{R}$ represents the predicted scalar solar wind speed. We select a forecast horizon of $4$ days corresponding to the typical travel time of solar wind plasma (Rotter et al., 2012; Upendran et al., 2020). The table below mentions various metrics by AlexNet, trained on 18k steps, ResNet50 for 20k steps, and Surya for 10k steps. Surya outperformed the baseline, achieving the best results in least number of steps

Table 6: Solar Wind Prediction results comparing baseline models with Surya

| **Model** | **RMSE** (km $s^{-1}$) | **MAE**(km $s^{-1}$) | **Validation loss** ($km^2 s^{-2}$) |
|---|---|---|---|
| AlexNet | 118.6 | 95.7 km | 13839.49 |
| ResNet50 | 93.76 | 74.65 | 8547.924 |
| Surya | **75.92** | **58.06** | **5698.62** |

## 3 Discussion and conclusions

In this work, we presented Surya, a 366M-parameter foundation model for heliophysics, trained at the native 4096×4096 resolution of SDO's AIA and HMI instruments with a standardized 12-minute cadence. By pretraining on the task of forecasting, Surya learns general-purpose solar representations that capture both the fine-scale variability of magnetic fields and the large-scale dynamics of the solar atmosphere. This pretraining strategy enables the model to perform zero-shot forecasting of solar activity, including the visual evolution of flare events, while also providing transferable representations that can be adapted efficiently to various downstream applications. Surya thus represents a shift from narrowly focused, task-specific models to a more versatile and scalable approach for heliophysics.

A key result is Surya's capability to forecast solar dynamics without additional training. For example, in the case of the January 2015 flare, Surya's predicted integrated EUV emissions closely tracked observations, demonstrating sensitivity to the rapid changes associated with flare onset. Quantitatively, autoregressive rollout tuning improved long-range forecasting skill by up to 17.8% at 12 hours lead time compared to one-step pretraining. These results suggest that the model is not simply memorizing past patterns, but rather developing representations that are, to some extent, physics-aware. It also significantly outperforms persistence (MAE $\approx 0.59$) and learned-flow baselines (MAE $\approx 0.34$) in one-hour forecasts.

For finetuning on downstream tasks, we adapted parameter-efficient fine-tuning with LoRA. For active region segmentation, Surya achieved an IoU of 0.768 and a Dice coefficient of 0.853, outperforming a U-Net baseline (IoU 0.688, Dice 0.801). In solar flare forecasting, Surya obtained TSS of 0.436, HSS of 0.522, and F1 of 0.561, substantially improving over AlexNet (TSS = 0.358) and ResNet50 (TSS = 0.018). For EUV irradiance prediction, it reduced error with an MAE of 0.0043 compared to 0.0065 for AlexNet and 0.0256 for ResNet50. Finally, in solar wind speed forecasting, Surya achieved RMSE of 75.92 km $s^{-1}$ and MAE of 58.06 km $s^{-1}$, outperforming both AlexNet (RMSE = 118.6 km $s^{-1}$) and ResNet50 (RMSE = 93.76 km $s^{-1}$).

We also observed some limitations that can be improved in future work. As a deterministic model trained with an MSE objective, Surya exhibits blurring of sharp features in magnetograms and flare imagery, a common outcome in regression-based generative models. Probabilistic approaches, such as diffusion forecasting or training with alternative loss functions like CRPS, could mitigate this issue by producing sharper, more physically realistic predictions. Similarly, while forecasting proved effective as a pretraining task, other self-supervised strategies such as masked reconstruction or band-to-band translation may further enrich the

learned representations. From a technical perspective, training Surya required large-scale computational resources, with data throughput rather than model capacity becoming the primary bottleneck during rollout tuning, pointing to the need for more efficient data pipelines in future large-scale efforts.

In conclusion, Surya represents the first foundation model for heliophysics trained at the full resolution of SDO data, establishing a unifying framework that combines forecasting skill with transferable representations for a range of scientific and operational tasks. Its ability to generalize across segmentation, classification, regression, and forecasting problems illustrates the potential of foundation models to accelerate both discovery and operational space weather prediction. Looking forward, we can incorporate multimodal, multi-mission datasets and adopt probabilistic approaches for better and improved foundation models to support next-generation heliophysics and digital twin initiatives.

## CODE AND DATA AVAILABILITY

The model and datasets are publicly available on Huggingface: `https://huggingface.co/nasa-ibm-ai4science` Our code for model and downstream tasks are publicly available at `https://github.com/NASA-IMPACT/Surya`


### ACKNOWLEDGMENTS

We would like to thank Soumya Ranjan and WeiJi Leong from Development seed who contributed in the early stages of this project. We would also like to thank Shubha Ranjan from NASA Advanced Supercomputing (NAS) Division, and Mike Little from Goddard Spaceflight Center for their help and support. We would also like to thank David Hall for help and support with Nvidia computing resources.

The Authors acknowledge the National Artificial Intelligence Research Resource (NAIRR) Pilot and NVIDIA for providing support under grant no. NAIRR240178. The authors would also like to thank NASA Advanced Supercomputing (NAS) Division for their compute support. Vishal Upendran would like to acknowledge NASA for support under award number 80NSSC25K7956.


## REFERENCES


Andrés Asensio Ramos, Mark C. M. Cheung, Iulia Chifu, and Ricardo Gafeira. Machine learning in solar physics. *Living Reviews in Solar Physics*, 20(1):4, December 2023. doi: 10.1007/s41116-023-00038-x.

Cristian Bodnar, Wessel P Bruinsma, Ana Lucic, Megan Stanley, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan Weyn, Haiyu Dong, Anna Vaughan, et al. Aurora: A foundation model of the atmosphere. *arXiv preprint arXiv:2405.13063*, 2024.

D. H. Boteler. Space weather effects on power systems. *Geophysical Monograph Series*, 125:347–352, January 2001. doi: 10.1029/GM125p0347.

Xumin Cai, Berkay Aydin, Anli Ji, Manolis K. Georgoulis, and Rafal Angryk. A framework for detecting polarity inversion lines from longitudinal magnetograms. In *2020 IEEE International Conference on Big Data (Big Data)*, pp. 4175–4183. IEEE, December 2020. doi: 10.1109/bigdata50022.2020.9377808. URL `http://dx.doi.org/10.1109/BigData50022.2020.9377808`.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

George M. Gloeckler. Ace solar wind ion composition spectrometer (swics) solar wind plasma elemental and isotopic density, speed, thermal speed, charge state, and ratio parameters, level 2 (l2), 1 h data, 2023.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Johannes Jakubik, Sujit Roy, CE Phillips, Paolo Fraccaro, Denys Godwin, Bianca Zadrozny, Daniela Szwarcman, Carlos Gomes, Gabby Nyirjesy, Blair Edwards, et al. Foundation models for generalist geospatial artificial intelligence. *arXiv preprint arXiv:2310.18660*, 2023.

Johannes Jakubik, Felix Yang, Benedikt Blumenstiel, Erik Scheurer, Rocco Sedona, Stefano Maurogiovanni, Jente Bosmans, Nikolaos Dionelis, Valerio Marsocci, Niklas Kopp, et al. Terramind: Large-scale generative multimodality for earth observation. *arXiv preprint arXiv:2504.11171*, 2025.

Anli Ji, Xumin Cai, Nigar Khasayeva, Manolis K. Georgoulis, Petrus C. Martens, Rafal A. Angryk, and Berkay Aydin. A systematic magnetic polarity inversion line data set from sdo/hmi magnetograms. *The Astrophysical Journal Supplement Series*, 265(1):28, March 2023. ISSN 1538-4365. doi: 10.3847/1538-4365/acb43a. URL http://dx.doi.org/10.3847/1538-4365/acb43a.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, et al. Graphcast: Learning skillful medium-range global weather forecasting. *arXiv preprint arXiv:2212.12794*, 2022.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, et al. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023.

Simon Lang, Mihai Alexe, Mariana CA Clare, Christopher Roberts, Rilwan Adewoyin, Zied Ben Bouallègue, Matthew Chantry, Jesper Dramsch, Peter D Dueben, Sara Hahner, et al. Aifs-crps: ensemble forecasting using a model trained with a loss function based on the continuous ranked probability score. *arXiv preprint arXiv:2412.15832*, 2024.

James R. Lemen, Alan M. Title, David J. Akin, Paul F. Boerner, Catherine Chou, Jerry F. Drake, Dexter W. Duncan, Christopher G. Edwards, Frank M. Friedlaender, Gary F. Heyman, Neal E. Hurlburt, Noah L. Katz, Gary D. Kushner, Michael Levay, Russell W. Lindgren, Dnyanesh P. Mathur, Edward L. McFeaters, Sarah Mitchell, Roger A. Rehse, Carolus J. Schrijver, Larry A. Springer, Robert A. Stern, Theodore D. Tarbell, Jean-Pierre Wuelser, C. Jacob Wolfson, Carl Yanari, Jay A. Bookbinder, Peter N. Cheimets, David Caldwell, Edward E. Deluca, Richard Gates, Leon Golub, Sang Park, William A. Podgorski, Rock I. Bush, Philip H. Scherrer, Mark A. Gummin, Peter Smith, Gary Auker, Paul Jerram, Peter Pool, Regina Soufli, David L. Windt, Sarah Beardsley, Matthew Clapp, James Lang, and Nicholas Waltham. The Atmospheric Imaging Assembly (AIA) on the Solar Dynamics Observatory (SDO). *Solar Physics*, 275(1-2):17–40, January 2012. doi: 10.1007/s11207-011-9776-8.

Harris Abdul Majid, Pietro Sittoni, and Francesco Tudisco. Solaris: A foundation model of the sun. *arXiv preprint arXiv:2411.16339*, 2024.

Tung Nguyen, Rohan Shah, Hritik Bansal, Troy Arcomano, Romit Maulik, Rao Kotamarthi, Ian Foster, Sandeep Madireddy, and Aditya Grover. Scaling transformer neural networks for skillful and reliable medium-range weather forecasting. *Advances in Neural Information Processing Systems*, 37:68740–68771, 2024.

Edward J. Oughton, Andrew Skelton, Richard B. Horne, Alan W. P. Thomson, and Charles T. Gaunt. Quantifying the daily economic impact of extreme space weather due to failure in electricity transmission infrastructure. *Space Weather*, 15(1):65–83, January 2017. doi: 10.1002/2016SW001491.

Chetraj Pandey, Rafal A. Angryk, and Berkay Aydin. *Explaining Full-Disk Deep Learning Model for Solar Flare Prediction Using Attribution Methods*, pp. 72–89. Springer Nature Switzerland, 2023. ISBN 9783031434303. doi: 10.1007/978-3-031-43430-3_5. URL http://dx.doi.org/10.1007/978-3-031-43430-3_5.

Chetraj Pandey, Anli Ji, Jinsu Hong, Rafal A. Angryk, and Berkay Aydin. Embedding ordinality to binary loss function for improving solar flare forecasting. In *2024 IEEE 11th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10. IEEE, October 2024. doi: 10.1109/dsaa61799.2024.10722839. URL http://dx.doi.org/10.1109/DSAA61799.2024.10722839.

Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, et al. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

Badri N Patro, Vinay P Namboodiri, and Vijay S Agneeswaran. Spectformer: Frequency and attention is what you need in a vision transformer. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 9543–9554. IEEE, 2025.

W. Dean Pesnell, B. J. Thompson, and P. C. Chamberlin. The Solar Dynamics Observatory (SDO). *Solar Physics*, 275(1-2):3–15, January 2012. doi: 10.1007/s11207-011-9841-3.

T. Rotter, A. M. Veronig, M. Temmer, and B. Vršnak. Relation Between Coronal Hole Areas on the Sun and the Solar Wind Parameters at 1 AU. *Solar Physics*, 281:793–813, December 2012. doi: 10.1007/s11207-012-0101-y.

Sujit Roy, Talwinder Singh, Marcus Freitag, Johannes Schmude, Rohit Lal, Dinesha Hegde, Soumya Ranjan, Amy Lin, Vishal Gaur, Etienne Eben Vos, et al. Ai foundation model for heliophysics: Applications, design, and implementation. *arXiv preprint arXiv:2410.10841*, 2024.

Johannes Schmude, Sujit Roy, Will Trojak, Johannes Jakubik, Daniel Salles Civitarese, Shraddha Singh, Julian Kuehnert, Kumar Ankur, Aman Gupta, Christopher E Phillips, et al. Prithvi wxc: Foundation model for weather and climate. *arXiv preprint arXiv:2409.13598*, 2024.

J. Schou, P. H. Scherrer, R. I. Bush, R. Wachter, S. Couvidat, M. C. Rabello-Soares, R. S. Bogart, J. T. Hoeksema, Y. Liu, T. L. Duvall, D. J. Akin, B. A. Allard, J. W. Miles, R. Rairden, R. A. Shine, T. D. Tarbell, A. M. Title, C. J. Wolfson, D. F. Elmore, A. A. Norton, and S. Tomczyk. Design and Ground Calibration of the Helioseismic and Magnetic Imager (HMI) Instrument on the Solar Dynamics Observatory (SDO). *Solar Physics*, 275(1-2):229–259, January 2012. doi: 10.1007/s11207-011-9842-2.

Karel Schrijver, Fran Bagenal, Tim Bastian, Juerg Beer, Mario Bisi, Tom Bogdan, Steve Bougher, David Boteler, Dave Brain, Guy Brasseur, Don Brownlee, Paul Charbonneau, Ofer Cohen, Uli Christensen, Tom Crowley, Debrah Fischer, Terry Forbes, Tim Fuller-Rowell, Marina Galand, Joe Giacalone, George Gloeckler, Jack Gosling, Janet Green, Nick Gross, Steve Guetersloh, Viggo Hansteen, Lee Hartmann, Mihaly Horanyi, Hugh Hudson, Norbert Jakowski, Randy Jokipii, Margaret Kivelson, Dietmar Krauss-Varban, Norbert Krupp, Judith Lean, Jeff Linsky, Dana Longcope, Daniel Marsh, Mark Miesch, Mark Moldwin, Luke Moore, Sten Odenwald, Merav Opher, Rachel Osten, Matthias Rempel, Hauke Schmidt, George Siscoe, Dave Siskind, Chuck Smith, Stan Solomon, Tom Stallard, Sabine Stanley, Jan Sojka, Kent Tobiska, Frank Toffoletto, Alan Tribble, Vytenis Vasyliunas, Richard Walterscheid, Ji Wang, Brian Wood, Tom Woods, and Neal Zapp. Principles Of Heliophysics: a textbook on the universal processes behind planetary habitability. *arXiv e-prints*, art. arXiv:1910.14022, October 2019. doi: 10.48550/arXiv.1910.14022.

Daniela Szwarcman, Sujit Roy, Paolo Fraccaro, Þorsteinn Elí Gíslason, Benedikt Blumenstiel, Rinki Ghosal, Pedro Henrique de Oliveira, Joao Lucas de Sousa Almeida, Rocco Sedona, Yanghui Kang, et al. Prithvi-eo-2.0: A versatile multi-temporal foundation model for earth observation applications. *arXiv preprint arXiv:2412.02732*, 2024.

Vishal Upendran, Mark C. M. Cheung, Shravan Hanasoge, and Ganapathy Krishnamurthi. Solar Wind Prediction Using Deep Learning. *Space Weather*, 18(9):e02478, September 2020. doi: 10.1029/2020SW002478.

James Walsh, Daniel G Gass, Raul Ramos Pollan, Paul J Wright, Richard Galvez, Noah Kasmanoff, Jason Naradowsky, Anne Spalding, James Parr, and Atılım Güneş Baydin. A foundation model for the solar dynamics observatory. *arXiv preprint arXiv:2410.02530*, 2024.

T. N. Woods, F. G. Eparvier, R. Hock, A. R. Jones, D. Woodraska, D. Judge, L. Didkovsky, J. Lean, J. Mariska, H. Warren, D. McMullin, P. Chamberlin, G. Berthiaume, S. Bailey, T. Fuller-Rowell, J. Sojka, W. K. Tobiska, and R. Viereck. Extreme Ultraviolet Variability Experiment (EVE) on the Solar Dynamics Observatory (SDO): Overview of Science Objectives, Instrument Design, Data Products, and Model Developments. *Solar Physics*, 275(1-2):115–143, January 2012. doi: 10.1007/s11207-009-9487-6.

Chin-Chun Wu, Kan Liou, Ronald P Lepping, Lynn Hutting, Simon Plunkett, Russ A Howard, and Dennis Socker. The first super geomagnetic storm of solar cycle 24:"the st. patrick's day event (17 march 2015)". *Earth, Planets and Space*, 68(1):151, 2016.

Chen Zhu, Wei Ping, Chaowei Xiao, Mohammad Shoeybi, Tom Goldstein, Anima Anandkumar, and Bryan Catanzaro. Long-short transformer: Efficient transformers for language and vision. *Advances in neural information processing systems*, 34:17723–17736, 2021.

# A Architecture

## A.1 Ablation studies

The main body of the text already discusses the baseline scores: persistence as well as the flow model trained to capture differential rotation. In addition, we performed a number of ablation studies to validate Surya's architecture choices. All ablation experiments were trained for 10,000 gradient descent steps on 16 GPUs. The results can be found in table 7.

Table 7: Architecture ablations & baselines. The table shows MSE loss for 1 hour ahead forecasting in model units of equation 1. All baselines and ablations in this table – including the Surya configuration – were trained on 16 GPUs for 10,000 gradient descent steps.

| Type | Configuration | Parameters | Memory usage [MiB] | Loss (MSE) |
|---|---|---|---|---|
| Baseline | Persistence | N/A | N/A | 0.594044030 |
| | Learned flow | 642 | 24976 | 0.337624282 |
| Ablation | Single timestamp | 361.93 M | 55411 | 0.228553504 |
| | No spectral gating | 210.39 M | 59823 | 0.219618767 |
| | Perceiver | 351.52 M | 52721 | 0.234643415 |
| Surya | | 366.19 M | 56247 | 0.219778508 |

Let us start by discussing the spectral gating layers. In this ablation study ("No spectral gating"), we replace the two spectral gating layers with additional long-short attention layers. Within the context of the compute

budget used here, this led to a virtually identical loss, yet at 6% reduced GPU memory consumption. Interestingly, this happens although the conversion of long-short to spectral gating layers adds a huge number of parameters, each spectral block containing a weight matrix consisting of $84, 541, 440$ real parameters alone. Indeed, table 7 shows that a large number of Surya's parameters are in the two large weight matrices applied in Fourier space in those layers.

The next ablation we consider is training the model with one rather than two timestamps as input. Using two timestamps as input improves performance by 3.8%. On the one hand, one clearly expects two timestamps to do better than one as the model can infer a motion field. On the other one might be surprised that one timestamp as input still yields relatively strong performance. One reason for this might be that the tokenization in Surya is effectively a compression. Given two timestamps, 13 bands and a patch size of 16 by 16, each patch comprises $2 \times 13 \times 16 \times 16 = 6,656$ pixel. Our embedding dimension on the other hand is $1,280$. So we are effectively compressing our data by a factor of 5.2 with a very simple linear layer. Using one timestamp as input rather than two reduces this compression ratio.

Given this compression ratio, one might expect that using a more complex tokenization procedure would help model performance. Indeed, one can consider the case of atmospheric physics where there is a similar situation: Model inputs comprise many different variables at different vertical levels. Nguyen et al. (2024) used an attention mechanism to aggregate variables. Bodnar et al. (2024) extended this to the use of a perceiver. The latter was also use in Majid et al. (2024). With this in mind, we evaluated the use of a perceiver to aggregate input tokens and process outputs. As table 7, we obtained considerably worse performance. Note however that the memory consumption of the perceiver was such that we had to use a 32 by 32 token size in this case. Which of course makes the problem worse that the perceiver was supposed to address.

Let us conclude with a few remarks about the flow model. To start, it is remarkable how well it performs given its miniscule parameter count. For AI forecasting and foundation models in atmospheric physics, it is very common not to model the target directly, but to model the difference of the target from some known quantity. In Lam et al. (2022) the authors predicted the difference of the latest input timestamp $\mathbf{X}_t$ from the target $\mathbf{X}_{t+1}$. That is, the model was trained to predict the difference from persistence. Noting that in the presence of sharp features the model has to learn to exactly remove the sharp feature from $\mathbf{X}_t$ and add it at a new location in $\mathbf{X}_{t+1}$, Lang et al. (2024) improved on this by only considering the deviation from a smoothened version of $\mathbf{X}_t$. Schmude et al. (2024) was interested in the case of zero lead time. Here, predicting a difference from the presence becomes meaningless and the authors chose to instead model the delta from historical climate. In either case, given the parameter efficiency of the flow model, it is tempting to first train a flow model and then a transformer to model the discrepancy between the flow model and the actual target. In our ablation studies, we found this to perform worse than simply regressing directly onto the target.

# B Additional results

## B.1 Impact of rollout tuning

Table 8 shows the data corresponding to figure 5. Note that rollout tuning massively improves model performance at long lead times, we also see ever so slightly decreasing performance at the shortest ones. I.e. the best performing model at 1 hour ahead lead time is actually Surya before phase 2 of pretraining.

Table 8: Impact of rollout tuning. The table shows validation loss (MSE) per lead time for lead times up to 12 hours ahead (rows). Columns show persistence and flow baselines as well as Surya at various stages of pretraining. The 5 hour ahead tuned version (rightmost column) corresponds to Surya as released. The best score in each row is marked in bold.

| | **Persistence** | **Flow** | **1 hour (no rollout)** | **2 hours** | **3 hours** | **4 hours** | **5 hours** |
|---|---|---|---|---|---|---|---|
| 1 | 0.59868 | 0.34204 | **0.20133** | 0.20153 | 0.20159 | 0.20169 | 0.20172 |
| 2 | 0.70594 | 0.40124 | 0.22738 | 0.22371 | 0.22352 | 0.22359 | **0.22351** |
| 3 | 0.77862 | 0.45497 | 0.24889 | 0.24182 | 0.23928 | 0.23902 | **0.23889** |
| 4 | 0.83790 | 0.50416 | 0.26662 | 0.25759 | 0.25213 | 0.25125 | **0.25078** |
| 5 | 0.89444 | 0.55291 | 0.27976 | 0.26957 | 0.26295 | 0.26141 | **0.26064** |
| 6 | 0.94441 | 0.59671 | 0.29362 | 0.28111 | 0.27366 | 0.27140 | **0.26995** |
| 7 | 0.99986 | 0.64647 | 0.31217 | 0.29536 | 0.28557 | 0.28113 | **0.27854** |
| 8 | 1.04645 | 0.68856 | 0.32528 | 0.30458 | 0.29519 | 0.29018 | **0.28583** |
| 9 | 1.09013 | 0.72833 | 0.33984 | 0.31209 | 0.30361 | 0.29777 | **0.29216** |
| 10 | 1.12573 | 0.76106 | 0.35257 | 0.31934 | 0.31181 | 0.30522 | **0.29868** |
| 11 | 1.14940 | 0.78280 | 0.37037 | 0.33154 | 0.32443 | 0.31638 | **0.30789** |
| 12 | 1.16841 | 0.79964 | 0.38517 | 0.34304 | 0.33786 | 0.32791 | **0.31651** |

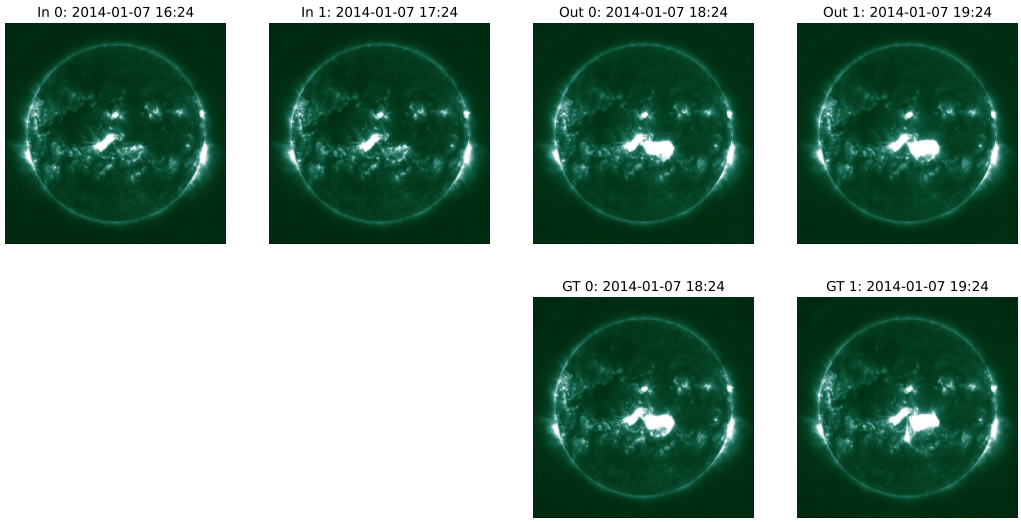


Figure 12: Surya inputs, outputs and ground truth for January 7 2014. AIA 94Å. Top row, left two columns are model inputs ("In"). Top row, right two columns are model outputs ("Out"). Bottom row shows corresponding ground truth ("GT"). The model is initialized slightly later than in 13 so the flare is already visible in its first output frame 60 minutes ahead.

## B.2 VISUAL PREDICTION OF SOLAR FLARES

### B.2.1 THE 2024-01-07 EVENT

Figures 12, 13 and 14 show model inputs and outputs as well as ground truth for a solar flare event on January 7 2014. This is complementary to figures 1 and 8. Note that both cases show testing data. See section 2.1.2 for a discussion of the train/test split.

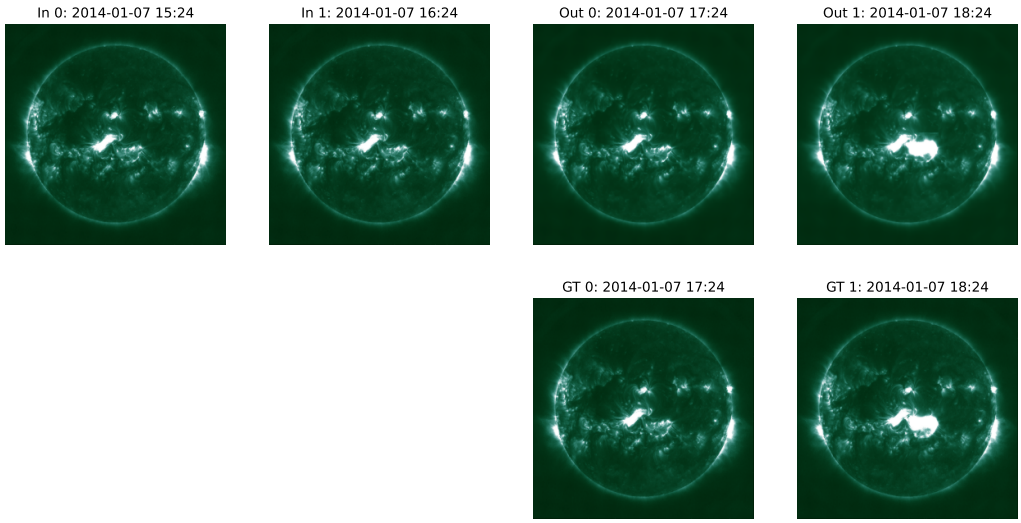


Figure 13: Surya inputs, outputs and ground truth for January 7 2014. AIA 94Å. Top row, left two columns are model inputs ("In"). Top row, right two columns are model outputs ("Out"). Bottom row shows corresponding ground truth ("GT").

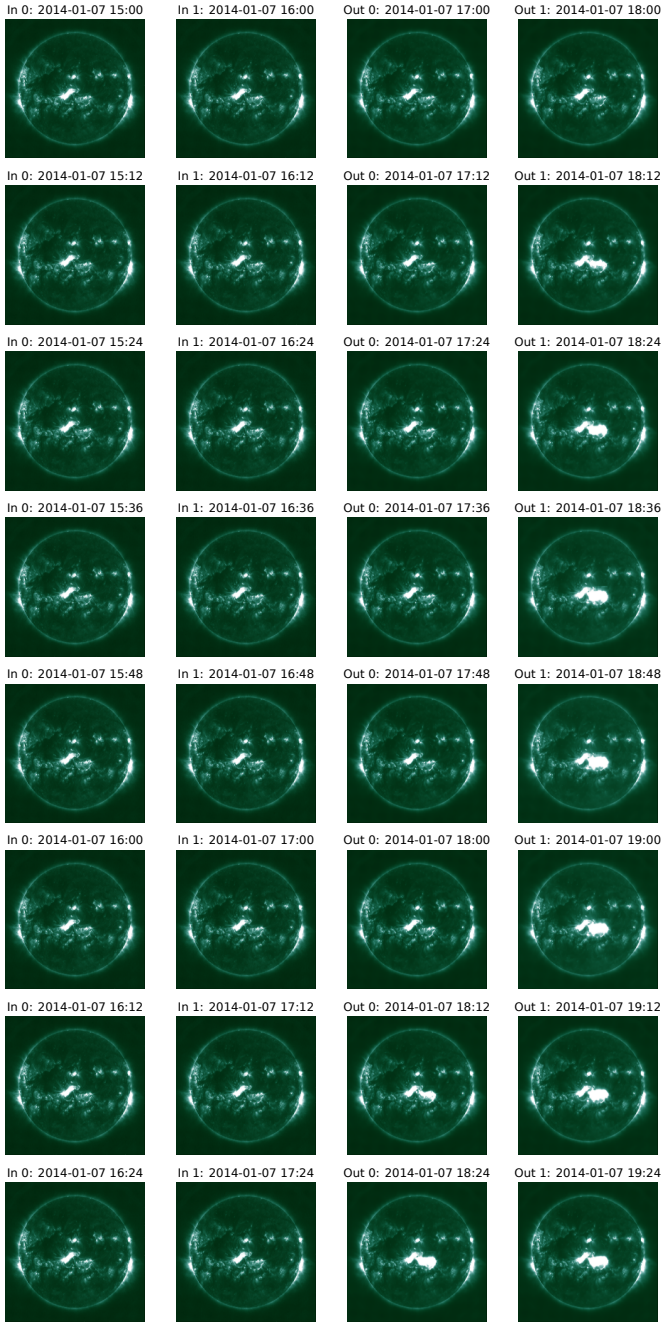


Figure 14: Surya inputs and outputs with different initializations for January 7 2014. AIA 94Å.