Author(s): Raita-Hakola, Anna-Maria; Pölönen, Ilkka

Title: Firefront Forecasting in Boreal Forests : Machine Learning Approach to Predict Wildfire Propagation

Year: 2024

Version: Published version

Please cite the original version:

Raita-Hakola, A.-M., & Pölönen, I. (2024). Firefront Forecasting in Boreal Forests : Machine Learning Approach to Predict Wildfire Propagation. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLVIII(3), 445-452. https://doi.org/10.5194/isprs-archives-xlviii-3-2024-445-2024

Firefront Forecasting in Boreal Forests: Machine Learning Approach to Predict Wildfire Propagation

Anna-Maria Raita-Hakola[1], Ilkka Pölönen[1]

[1] Faculty of Information Technology, University of Jyväskylä, Mattilanniemi 2, Jyväskylä Finland
(anna.m.hakola, ilkka.polonen)@jyu.fi

Keywords: Wildfire, Firefront, Forest Fire, Machine Learning, Propagation, Stochastic Cellular Automata

Abstract

Wildfires have become increasingly prevalent worldwide due to climate change, posing significant threats to human lives, property, and natural ecosystems. The rapid progression of wildfires necessitates predictive computational models to assist firefighters in effectively developing strategies to control firefronts. However, existing models often face challenges in computational complexity as the firefront expands. This study aims to develop a faster, more computationally efficient, deep-learning-based model for predicting wildfire spread. We hypothesise that firefront propagation can be modelled using stochastic cellular automata and that a deep-learning model can mimic this approach. With this in mind, we will first introduce our in-house stochastic cellular automata model, which is being validated with data from a known Finnish wildfire. After that, we propose a novel deep-learning model which uses the data generated by our cellular automata. The deep-learning-based model was based on Unet architecture, and it is capable of predicting firefront progression accurately and efficiently one time-step at a time. The model provided realistic simulations of firefronts with high computational efficiency, leaving future development needs to longer time series. One potential application of the developed model is in UAV-based real-time wildfire management systems.

1. Introduction

Global warming is causing droughts, increasing the risk of forest fires, which have become more common almost everywhere. Fire management and strategic firefighting planning require real-time information on where and how the firefront is advancing. Such information can be provided by a computational model that estimates the direction and speed of the firefront based on the inputs it receives. Computational firefront propagation models (FPMs) require multisource background information as input. A typical FPM inputdata includes the current location of the firefront, the amount of forest fire fuel in the terrain, the strength and direction of the wind, information about the shape of the terrain, and the humidity of the terrain. Models can be made more precise by separating different fuels (biomass types) from each other.

1.1 Real-time monitoring technology

Real-time monitoring technology is required to generate the information the models need to monitor the fire area. Drones can be used for the detection and mapping of wildfires (Raita-Hakola et al., 2023). For example, they can provide data suitable for FPMs if equipped with the right sensors, such as a thermal camera, an RGB camera, spectral imagers and lidars. The thermal camera can produce information about the location of the firefront. A spectral imager can be used to identify combustibles. A point cloud can be photogrammetrically produced with an RGB camera. Lidar produces a more accurate point cloud and possibly information about the understory vegetation. Wind direction and strength can be derived from the drone's autopilot data.

1.2 Real-time forecasting models

Real-time forecasting requires an efficient computational model. Many different firefront progression models have been proposed, which can be roughly divided into physical, empirical mathematical (Sullivan, 2009) and stochastic cellular automata models.

Physical models, such as ForeFire (Filippi et al., 2014) and FireTec (Linn et al., 2002), are designed to work with high spatial resolution. These models are based on fluid dynamics and heat transfer models.

Well-known empirical models are, for example, Prometheus (Tymstra et al., 2010) and Farsite (Finney, 1998). In these, the starting point of modelling has been to fit a suitable mathematical model to the empirically observed data. For example, Prometheus uses an elliptical propagation model that inputs environmental parameters, while Farsite is based on Huygens wavelet transforms.

Besides physical and empirical approaches, a classic way to model the progress of the firefront is stochastic cellular automata models (SCA). SCA models are intuitively easy-to-understand models so that clear rules can be written for the ignition probability of the cells. Rules increase or decrease the probability of ignition. The first cellular automata for modelling the firefront were already made in the 1970s (Kourtz and O'Regan, 1971), and their development and production are actively continuing (Trucchia et al., 2020, Mastorakos et al., 2023, Vanas, 2023).

What is typical for physical, empirical, and SCA models is that as the firefront expands, its computational complexity increases. This may reduce the implementation of these methods in real-time systems. Thus, more computationally efficient models are required to lower the threshold of enabling real-time systems.

1.3 Hypothesis and proposed approach

We hypothesise that since a real-time FPM requires a computationally efficient model, the efficiency might be accelerated by

https://doi.org/10.5194/isprs-archives-XLVIII-3-2024-445-2024

developing a deep-learning (DL) model that learns the operation of firefront propagation models from a traditional mathematical model, resulting in a computationally effective DL-mimic of the original propagation model.

With this in mind, we developed two models. 1.) An in-house implementation of stochastic cellular automata (SCA-FPM), adapted to the Kalajoki wildfire’s progression by adjusting the parameters using a differential evolution optimisation algorithm. The SCA-FPM was fitted into publicly accessible remote sensing data: fuel map, moisture levels and digital elevation models collected from the Kalajoki burning site. Kalajoki wildfire occurred in 2021 in Finland, where over 200 hectares of forest and dried marshland burned (Puustinen, 2022).

2.) The SCA-FPM was used as a basis for our DL-based firefront propagation model (DL-FPM). After optimising the SCA-FPM for the Finnish environment, we randomly selected 1400 coordinates from Finland and simulated 75,000 separate fire events. Each fire event had 16 time steps, each lasting 15 minutes. Each time step provided firefront, fuel and intensity prediction maps after the step.

The proposed DL-FPM utilised 100,000 SCA-FPM time steps as its training data. The selected architecture was based on Unet. For each to-be-predicted firefront parameter map, we trained a separate model.

2. Material and methods

2.1 SCA-FPM simulation input data description

Simulating wildfires requires understanding landscape characteristics, especially fuel, moisture and topography details. For example, it is necessary to get an approximate impression of the flammable material in the landscape that feeds the fire. A fire will be extinguished either when it runs out of fuel or when it encounters areas where there is no fuel.

Another critical element is the presence of moisture and water within the landscape. Dry terrains are more prone to ignition than their moist counterparts, and a fire will be extinguished upon encountering an area with sufficient moisture or a water body. The topography significantly influences the direction of fire spread, exhibiting a marginally higher propensity for upward rather than downward propagation.

In Finland, the aforementioned environmental characteristics are publicly available, and our SCA-FPM used this processed remote sensing data for its simulation. The quantification of fuel within the terrain is accomplished by utilising a nationwide tree stand carbon inventory. This carbon stock estimation is derived from the National Forest Inventory Programme, which employs lidar data and a comprehensive network of tree stand plots for forest inventories. Moisture levels were determined utilising a topographic moisture index calculated from the laser scanning data. Additionally, the terrain configuration is readily available in a 25x25 m elevation model, also derived from laser scanning data. The utilised materials are publicly accessible at (CSC, 2024).

Figure 1 shows an example of Finnish data. A fuel, height model and topographic water index maps describe the Kalajoki wildfire burning area, further discussed in subsection 2.3.

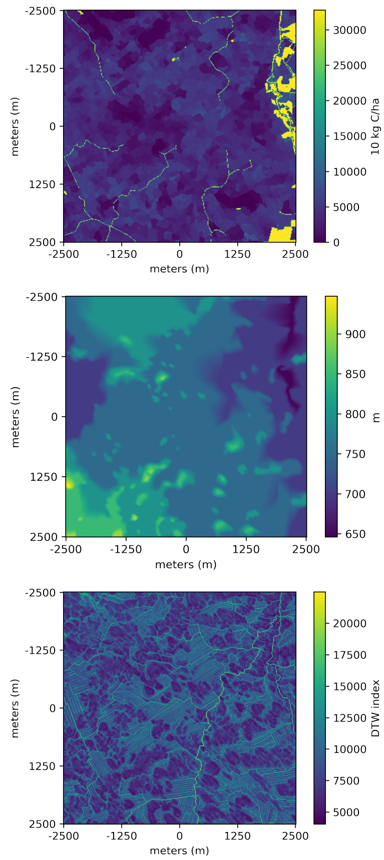

Figure 1. The inputs for the stochastic model are the wood carbon content as a fuel estimate, the height model and the topographic water index from the Kalajoki 2021 burning site.

2.2 SCA-FPM input data sampling and output

Besides fuel, moisture, and topography information, simulation requires the determination of wind direction and velocity, with the direction expressed in radians and velocity expressed in meters per second.

The input data was randomly sampled from 1400 distinct locations across Finland for SCA-FPM simulation. A 5000×5000 m window was gathered around each coordinate with a ground sampling distance (GDS) of 16 m. This procedure generated 313x313 pixel rasters for fuel, moisture, and the elevation model. The resolutions of the maps were corrected to match each other

Wind direction underwent variation by estimating five distinct directions and intensities, which were then applied within the simulation. The simulation spanned 16 time steps, with each step lasting 15 minutes. This resulted in a total of 75,000 distinct fire events, with a firefront moving through over 1.2 million incidents. From this data set, we randomly sampled 100,000 incidents to train and test DL-FPM.

As an output, the stochastic fire simulator yields data on the location of the firefront, fire intensity, and remaining fuel quantity. This information is crucial for simulating subsequent time steps.

2.3 SCA-FPM validation: the Kalajoki wildfire in 2021

To validate the simulation, its results was compared against actual events. In Finland, significant wildfires are infrequent; the most recent extensive wildfire occurred during the summer of 2021 in Kalajoki, located in western Finland, covering an area exceeding 200 hectares. The wildfire is thoroughly documented, with available sources providing detailed information on the progression of the firefront in half-hour intervals. This particular wildfire had two distinct fronts, each with its own timeline. One front was utilised to calibrate the stochastic model to align with accurate parameterisation. The efficacy of both the stochastic propagation model and the machine learning model were evaluated within this wildfire scenario. Figure 2 depicts both fires and the progression of their respective fronts in 30-minute intervals.

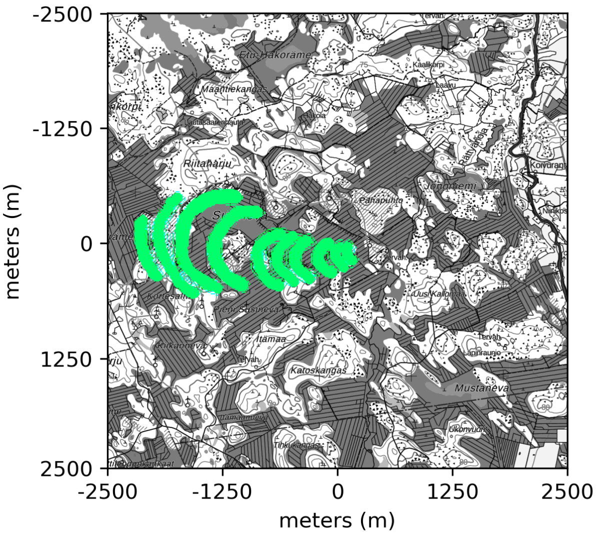

Figure 2. Progression of Kalajoki 2021 wildfire: The burn investigators' assessment of the propagation of the firefront in green in 30-minute intervals. The ignition point of the fire is in the middle of the Figure (EPSG:3067 N: 7100439, E: 361200).

2.4 Operating principles of SCA-FPM data generator

The Stochastic cellular automata-based firefront propagation model (SCA-FPM) uses Python, and utilises libraries such as NumPy, Matplotlib, and the random module. The simulation is modelling the spread of a wildfire over a grid-based landscape, considering factors like fuel availability, moisture levels, wind direction and strength, and elevation.

The input and output data of a SCA-FPM is presented in Table **??**. Figure 1 is an example of the Finnish remote-sensing material used as an input in our simulation starting points. The top figure represents the fuel, middle figure is a height model and bottom figure is a moisture level map (A., B. and C. in Table **??**) For each time step, the model updates the firefront, fuel and fire intensity maps as output (H, I and J in Table **??**).

In this study, the SCA-FPM generates training and validation data for the proposed DL-FPM approach. For each randomly selected fire location, the first values of D-G (Table **??**) are randomly generated and values A-C are gathered from remote sensing data. The time step is 15 minutes. SCA-FPM runs four hours lasting wildfire events, generating related updated output maps (H-J).

Inputs
A. Fuel map (wood carbon content)
B. Height model (digital elevation model)
C. Moisture level (topographic water index)
D. Current firefront or ignition point
E. Fire intensity
F. Wind direction
G. Wind strength
Outputs
H. Updated firefront map
I. Updated fuel map
J. Updated fire intensity map

Table 1. Inputs and outputs of SCA-FPM.

The stochastic approach allows the simulation to incorporate the inherent unpredictability of fire spread, where even under similar conditions, the outcome can vary. This randomness is crucial for simulating real-world scenarios where many small, unpredictable factors can significantly influence the spread of a wildfire. The stochastic element in the simulation primarily comes from the random number, which determinates whether the fire spreads to a neighboring cell based on calculated total ignitation factor from input parameters.

Let the probability of fire spreading from cell (i, j) to a neighboring cell (ni, nj) to be

$$P_{\text{spread}}(i, j \to ni, nj). \tag{1}$$

Let now

- $W(i, j, ni, nj)$ be the wind influence on the probability of fire spread from cell (i, j) to cell (ni, nj), considering the wind direction and strength,
- $M(ni, nj)$ be the moisture influence at cell (ni, nj), inversely related to the moisture level,
- $L(i, j, ni, nj)$ be the landscape influence based on the elevation difference between cell (i, j) and cell (ni, nj) and
- $I(i, j)$ be the flame intensity at cell (i, j).

The total influence $T(i, j, ni, nj)$ on the transition probability can be expressed as the product of these factors, scaled by the flame intensity of the active cell:

$$T(i, j, ni, nj) = W(i, j, ni, nj)M(ni, nj)L(i, j, ni, nj)I(i, j). \tag{2}$$

The stochastic step to determine if the fire spreads to the neighboring cell is modeled by comparing a random number R from a uniform distribution to the total influence:

$$P_{\text{spread}}(i, j \to ni, nj) = \begin{cases} 1 & \text{if } R < T(i, j, ni, nj) \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

If $P_{\text{spread}}(i, j \to ni, nj) = 1$ fire spreads to cell (ni, nj). In case $P_{\text{spread}}(i, j \to ni, nj) = 0$ the fire does not spread to cell (ni, nj).

2.5 SCA-FPM optimisation

In developing the stochastic simulator, the environmental parameters were calibrated using the empirical data from the Kalajoki wildfire, Finland, in 2021 (Puustinen, 2022). The differential evolution optimisation method was used to achieve an optimal fit of these parameters, which is known for its efficacy in handling complex optimisation problems (Storn and Price, 1997).

This approach adapted the parameters of the simulation model to accurately track the environmental conditions prevailing during a wildfire, thus improving the predictive accuracy and reliability of the simulator in modelling such events. In the cost function, we utilised the structural similarity index

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{,}(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2) \tag{4}$$

where x is Canny edge of propagated firefront from SCA-FPM and y is corresponding firefront from figure 2. The μ_x and μ_y are the means (average pixel values) of images x and y, respectively. The σ_x^2 and σ_y^2 are the variances of images x and y. The σ_{xy} is the covariance of images x and y. And c_1 and c_2 are small constants added to stabilise the division with weak denominators . Here

$$c_1 = (k_1 L)^2$$

$$c_2 = (k_2 L)^2$$

where L is the dynamic range of the pixel values , and k_1 and k_2 are constants (here $k_1 = 0.01$ and $k_2 = 0.03$). The SSIM index ranges from -1 to 1, where 1 indicates perfect similarity. By employing SSIM, we could quantitatively evaluate the reliability of our models' predictions in replicating the complex dynamics observed in actual wildfire propagation. Thus, actual cost function

$$\mathcal{L}_{SSIM} = 1 - \mathrm{SSIM}.$$

2.6 Deep-learning-based firefront propagation model architecture

Our deep-learning-based firefront propagation model (DL-FPM) aims to predict these events similarly to SCA-FPM. Unet-type architecture has been used promisingly in previous research with similar types of multi-dimensional image data. Therefore, a natural choice for this DL-based simulator was Unet architecture (Al-Dabbagh and Ilyas, 2023).

The Unet architecture, originally designed for biomedical image segmentation (Ronneberger et al., 2015), is characterised by its U-shaped design, which consists of a contracting path to capture context and a symmetrically expanding path that enables precise localisation. This structure suits it, particularly for the semantic segmentation task, where the goal is to assign a label to each pixel in an image.

In the context of wildfire simulation, the Unet model can be adapted to analyse spatial data, assimilating various environmental inputs to accurately delineate and predict the firefront's evolving boundaries and intensity.

DL-FPM implementation has an encoder-decoder structure with skip connections. Both the encoder and decoder stages apply padding to maintain the spatial dimensions of the feature maps throughout the network.

The encoder consists of four stages of double convolutional layers, each followed by batch normalisation and ReLU activation. The number of filters starts at 16 and doubles at each stage, reaching up to 256 in the deepest layer. This structure allows the model to capture increasingly complex features while preserving the spatial resolution due to the padding applied in the convolutional layers.

Double convolutional blocks, comprising two convolutional layers with batch normalisation and ReLU activation, ensure consistent and stable learning throughout the network. The double convolutional block is visualized in figure 3.

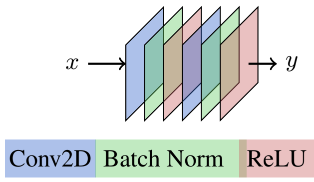

Figure 3. The double convolutional block. The block includes two convolutional layers, two batch normalisation layers, and two ReLU activation layers. Here x is the input of the block and y is output of the block

The decoder mirrors the encoder's structure, including padding to maintain the spatial dimensions. The number of filters decreases at each stage, to maintain symmetry with the encoder. The final layer is a 1x1 convolution that maps the 16 feature maps from the last decoder stage to a single output channel, producing the final prediction. The whole architecture is described in figure 4.

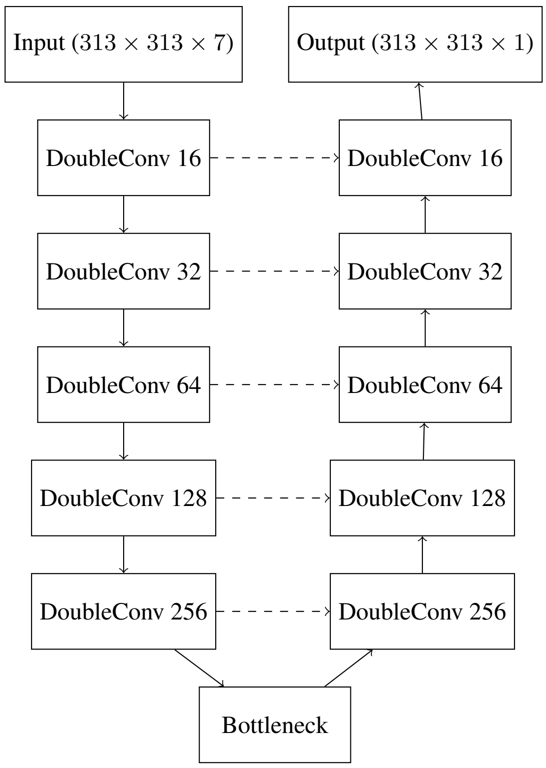

Figure 4. Modified Unet architecture.

2.7 DL-FPM loss function and optimiser

The masked mean squared error (MSE) loss function focuses on relevant regions of the input data while including a regularisation term. Therefore, it was a natural choice for the DL-FPM, and it was implemented as follows:

Let $\mathbf{y}_{\text{true}}$ be the ground truth and $\mathbf{y}_{\text{pred}}$ be the predicted output. The mask $\mathbf{M}$ is defined as:

$$\mathbf{M} = (\mathbf{y}_{\text{true}} > 0) \tag{5}$$

The masked mean squared error (MSE) is then computed as:

$$\text{MSE}_{\text{masked}} = \frac{\sum_{i,j} \left((\mathbf{y}_{\text{true}} - \mathbf{y}_{\text{pred}})^2 \cdot \mathbf{M} \right)}{\sum_{i,j} \mathbf{M}} \tag{6}$$

Additionally, a regularisation term was included to penalise the overall MSE across the entire image. The final masked MSE loss function is given by:

$$\mathcal{L} = \text{MSE}_{\text{masked}} + \alpha \cdot \frac{\sum_{i,j} (\mathbf{y}_{\text{true}} - \mathbf{y}_{\text{pred}})^2}{N \times M} \tag{7}$$

where α is a regularization coefficient (in this case, $\alpha = 0.1$) and N and M are the spatial dimensions of the image.

MSE loss function ensured that the model focused on accurately predicting the relevant regions while also considering the overall error across the image.

The Adam optimiser was employed to minimise the loss function (with a learning rate of 0.0001), chosen for its adaptive learning rate capabilities that help efficiently converge to a solution.

2.8 Training strategies

The training of the DL-FPM involved a few key steps. At first, separate models were trained to achieve high accuracy for each desired output (firefront, fire intensity and fuel), of which the most interesting in this study was the firefront since it is the only map that can be compared to real ground truth observations of Kalajoki fire event.

Since we were utilising a large amount of multi-dimensional data in the training process, custom data generators were employed to load and preprocess the data and corresponding ground truth labels, which were divided into training and validation sets. These sets were loaded in batches (batch size 8) using PyTorch's DataLoader.

The training loop iterated over a predefined number of epochs. In each epoch, the model was set to training mode, and for each batch of data, the input images and labels were loaded onto the GPU. The model performed a forward pass to generate predictions, the loss was computed using the masked MSE function, and backpropagation was performed to compute gradients, allowing the optimiser to update the model weights. The running loss for the epoch was accumulated for monitoring.

After each epoch, the model was evaluated against the validation set. During this validation loop, the model was set to evaluation mode, predictions were made for each batch of validation data, and the accuracy of the predictions was calculated by comparing the predicted firefronts to the ground truth labels. The models were trained within 10 epochs (each epoch lasted approximately 10 hours). The final models had no obvious marks of overfitting. The training process was executed on a GPU (NVIDIA RTX A4000, 16GB).

3. Results

3.1 SCA-FPM results

Figure 5 represents the SCA-FPM prediction results of the Kalajoki wildfire event propagation against the actual fire event. The green pixels represent the ground truth based on the official fire propagation reports. The simulation is drawn with red pixels. The firefront propagation direction is from right to left. The top figure represents the situation after one hour, the middle after two hours, and the bottom figure is the situation after three hours. The SCA-FPM seems to predict well the the Kalajoki fire event.

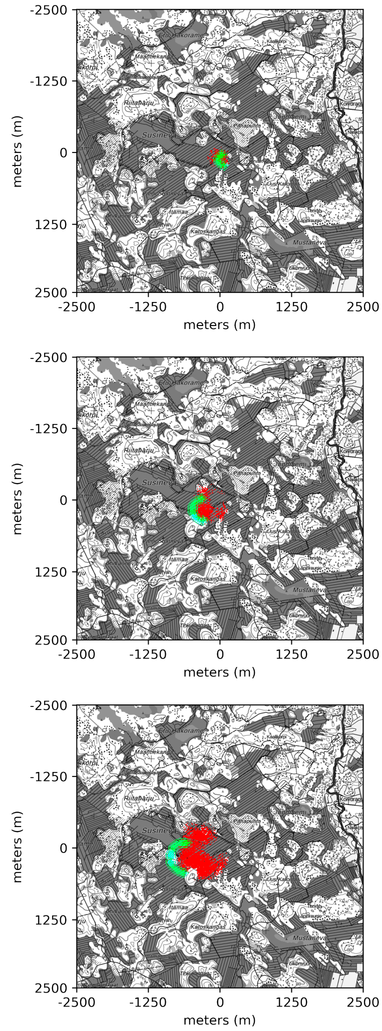

Figure 5. The propagation of the firefront inone, two and three hours after it has started. The green represents the ground truth, the red shows the prediction produced by SCA-FPM.

3.2 DL-FPM firefront prediction results

One fire event at a time, the DL-FPM produces a probability distribution-like firefront map of the fire progression. Figure

6 represents the results of one fire-event time-step. The top figure visualises the ground truth prediction (SCA-FPM), the middle figure is the DL-FPM prediction and the mean absolute error of the SCA-FPM simulated and DL-FPM predicted firefront propagation is seen in the bottom figure.

Here, it is noticed that most of the error is generated at the edge of the firefront, but the prediction error is mostly small (values mainly < 1), indicating that the model is predicting the firefront relatively right.

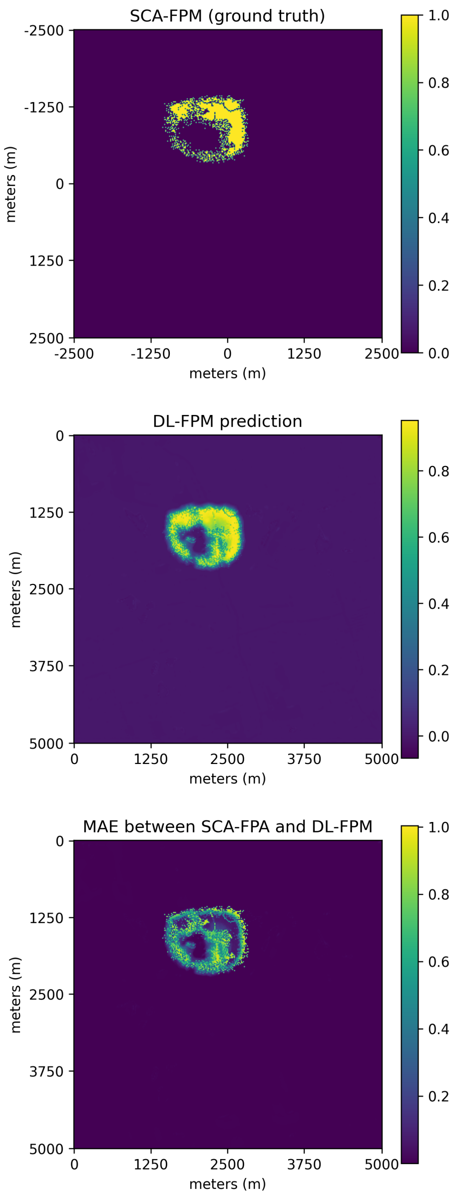

Figure 6. DL-FPM firefront prediction results of one simulated fire event. Top: SCA-FPM simulation ground truth, middle: DL-FPM prediction, bottom: mean absolute error.

To evaluate the DL-FPMs’ capability to predict accurate fire propagation direction, we compared the SCA-FPAs firefront simulation map one time-step earlier to the DL-FPM prediction of the next time-step of the same fire event.

In figure 7, the absolute difference between the previous DL-FPM input (SCA-FPM generated firefront) and the DL-FPL model-generated prediction is calculated. The figure shows that the firefront propagates in the direction of the given wind input (yellow) and along the fire zone’s boundaries. The firefront has been burned out from the centre (dark area), where the fire has had the longest time to burn. This indicates that the DL-FPM follows accurate fire propagation direction.

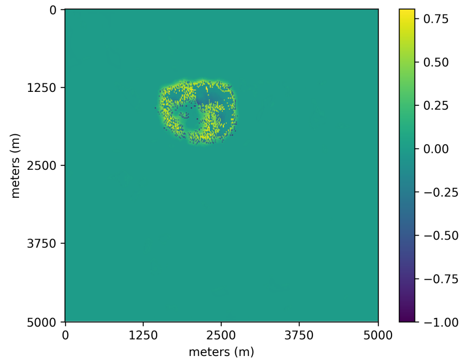

Figure 7. DL-FPM prediction compared to input firefront of the model, absolute difference.

While the DL-FPM performs well in predicting the first step, the model’s performance could be enhanced in the time-series prediction of the complete four hours of fire events.

Errors were found in the simulations of the four-hour fire event. In an experiment using Kalajoki event as ground truth, where the first 15-minute time step was based on the SCA-FPM simulation and the remaining time steps were simulations generated by DL-FPM, the accuracy of the results deteriorated as the time steps progressed.

If the same event was simulated by letting the DL-FPM model predict the fire progression time-step by time-step with SCA-FPM data, the model could produce accurate predictions for each time-step for four hours. Thus, the model’s ability to predict iterations independently requires further development.

Figure 8 shows DL-FPM predictions for the same three-hour firefront progression time-steps as Figure 5. Here, DL-FPM uses SCA-FPM-generated input data for each step.

4. Discussion

4.1 Pros and cons

SCA-FPM can model the progression of the Kalajoki firefront. In this study, the SCA-FPMs input data, which was collected from a Finnish database, and the predicted results were working as expected in Kalajoki Fire event validation tests.

The input data required for the SCA-FPM are simple remote sensing products, ie. surface models, biomass estimates and moisture models, which could be obtained globally from remote sensing satellites. Much of the forest resource data is collected on a 16 m grid, making the model applicable beyond Finland.

The SCA-FPM is robust and can only be used for rough predictions. For example, more detailed models, such as Prometheus and the Wise, take into account different fuel types ranging from tree species to terrain. Another major shortcoming of SCA-FPM is that it cannot simulate different fire types. For example, dangerously fast-moving crown fires are not given special attention here.

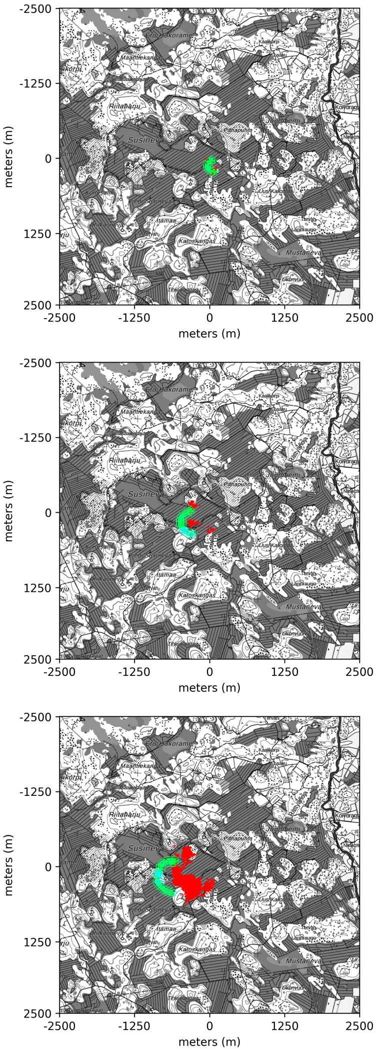

Figure 8. The propagation of the firefront in one, two and three hours after it has started. The green represents the ground truth, and red shows here the prediction produced by DL-FPM.

DL-FPM can learn single-time step prediction well. Unfortunately, the straightforward training strategy used here did not make it easily generalisable to multiple steps, such that it could have used its own generated prediction as input. Training the model itself is computationally expensive, but running the trained model on larger firefronts was a couple of orders of magnitude faster than running a simple stochastic cellular automata.

4.2 Future development

In conclusion, the SCA-FPM is accurate but robust. It serves as a starting point for developing DL models. Our proof-of-concept shows that DL architecture can be used to train a mimic model for fire prediction. However, in further research, it would be more sensible to use more advanced simulators instead of SCA-FPM. Therefore, in our following studies, we will use data from the Prometheus simulator to generate train data for DL-FPM. We will also use more in-detailed fuel models, which are gathered in the eastern part of Finland.

There are several options to change the architecture of the model. The simplest is to take the time step to one input and train DL-FPM with the SCA-FPM prediction of that time step. This way, it would be possible to have better temporal propagation model. In this case, the wind direction has to be assumed to be constant over several steps.

Another option is to regularise the cost function with forecasts over several time steps. This makes the script used to load and train the data more complex, but the end result should be that the model is better able to handle inputs, which are produced by DL-FPM.

One possible direction of development instead of the reasonably simple Unet-based model are attention-based transformer models. For example, (Bodnar et al., 2024) has developed a weather prediction model that is quite predictive. The model uses SWIN Unet (Cao et al., 2022) in its structure, which allows the input of different sizes to the model.

Eventually, DL-FPM could be implemented into a real-time system that gathers similar data from forests. The motivation for this research stems from the potential future application of drone swarms to map the locations of wildfires and estimate essential parameters. These parameters are critical for the machine learning model to assist information in strategising the containment of wildfire spread.

5. Conclusions

The integration of stochastic cellular automata with deep learning represents a novel methodology for wildfire modelling, combining the strengths of rule-based and data-driven approaches. The use of a real-world wildfire event for validation underscores the model’s applicability and accuracy. Additionally, the successful utilisation of publicly available remote sensing data for wildfire prediction demonstrates the accessibility and scalability of such models.

Practically, the efficiency of the DL model points towards its feasibility for real-time application in wildfire management, potentially optimising firefighting strategies. The computational efficiency of the model allows for rapid updates and forecasts, essential in dynamic wildfire scenarios. Moreover, the prospect of integrating this model into UAV-based systems for real-time data collection and prediction could transform wildfire monitoring and management, reducing response times and improving safety outcomes. While further refinement and validation are necessary, eventually, deep-learning-based models may offer promising tools for enhancing wildfire management and mitigation efforts.

6. Acknowledgements

This project has received funding from the European Union – NextGenerationEU instrument and is funded by the Academy of Finland under grant number 348009.

References

Al-Dabbagh, A. M., Ilyas, M., 2023. Uni-temporal Sentinel-2 imagery for wildfire detection using deep learning semantic segmentation models. *Geomatics, Natural Hazards and Risk*, 14(1), 2196370.

Bodnar, C., Bruinsma, W., Lucic, A., Stanley, M., Brandstetter, J., Garvan, P., Riechert, M., Weyn, J., Dong, H., Vaughan, A., Gupta, J., Thambiratnam, K., Archibald, A., Heider, E., Welling, M., Turner, R., Perdikaris, P., 2024. Aurora: A foundation model of the atmosphere. Technical Report MSR-TR-2024-16, Microsoft Research AI for Science.

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M., 2022. Swin-unet: Unet-like pure transformer for medical image segmentation. *European conference on computer vision*, Springer, 205–218.

CSC, 2024. Paituli, finnish spatial data download service. https://paituli.csc.fi/download.html.

Filippi, J.-B., Bosseur, F., Grandi, D., 2014. ForeFire: open-source code for wildland fire spread models. *Advances in forest fire research'.(Ed. DX Viegas)(Imprensa da Universidade de Coimbra: Coimbra, Portugal)*.

Finney, M. A., 1998. *FARSITE, Fire Area Simulator–model development and evaluation*. US Department of Agriculture, Forest Service, Rocky Mountain Research Station.

Kourtz, P. H., O'Regan, W. G., 1971. A model a small forest fire... to simulate burned and burning areas for use in a detection model. *Forest science*, 17(2), 163–169.

Linn, R., Reisner, J., Colman, J. J., Winterkamp, J., 2002. Studying wildfire behavior using FIRETEC. *International journal of wildland fire*, 11(4), 233–246.

Mastorakos, E., Gkantonas, S., Efstathiou, G., Giusti, A., 2023. A hybrid stochastic Lagrangian–cellular automata framework for modelling fire propagation in inhomogeneous terrains. *Proceedings of the Combustion Institute*, 39(3), 3853–3862.

Puustinen, A., 2022. Kalajoen Raution Metsäpalo 2021. *Kokemuksia ja oppeja metsäpalo-osaamisen kehittämiseen*, 3, 75.

Raita-Hakola, A.-M., Rahkonen, S., Suomalainen, J., Markelin, L., Oliveira, R., Hakala, T., Koivumäki, N., Honkavaara, E., Pölönen, I., 2023. Combining YOLO V5 and Transfer Learning for Smoke-Based Wildfire Detection in Boreal Forests. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 1771–1778.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, Springer, 234–241.

Storn, R., Price, K., 1997. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11, 341–359.

Sullivan, A. L., 2009. Wildland surface fire spread modelling, 1990–2007. 1: Physical and quasi-physical models. *International Journal of Wildland Fire*, 18(4), 349–368.

Trucchia, A., D'Andrea, M., Baghino, F., Fiorucci, P., Ferraris, L., Negro, D., Gollini, A., Severino, M., 2020. PROPAGATOR: An operational cellular-automata based wildfire simulator. *Fire*, 3(3), 26.

Tymstra, C., Bryce, R., Wotton, B., Taylor, S., Armitage, O. et al., 2010. Development and structure of Prometheus: the Canadian wildland fire growth simulation model. *Natural Resources Canada, Canadian Forest Service, Northern Forestry Centre, Information Report NOR-X-417.(Edmonton, AB)*.

Vanas, L., 2023. Kalajoen 2021 metsäpalon mallintaminen stokastisella soluautomaatilla.