



UNIVERSITY OF MELBOURNE

COMP90055 RESEARCH PROJECT

Convolutional Neural Networks in Resource Constrained Satellites for Wildfire Segmentation in Australia

Supervisor: Prof. Richard Sinnott

Author: Oliver Bestel de Lezongard

Student ID: 914956

I certify that this thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person where due reference is not made in the text. The thesis is 6489 words in length (excluding text in images, table, bibliographies and appendices).

Contents

1	Introduction	3
2	Literature Review	4
3	Methodology	8
3.1	Data	8
3.2	Data Annotation	10
3.3	Model Setup	13
3.3.1	Feature Extractors into XG Boosted Trees	14
3.3.2	Established Encoder-Decoder Frameworks	15
3.4	Performance and Computational Resource Evaluation	18
3.5	Experimental Setup	18
4	Results	18
4.1	Summary of Results	18
4.2	Model Mask Predictions	19
5	Discussion	24
5.1	Analysis of Results	24
5.2	Challenges	25
5.3	Class Imbalance	25
5.4	Limited Dataset Size	25
5.5	Cloud Cover	26
5.6	Benchmark Dataset	26
5.7	Compression Methods	27
5.8	Implications for Non-Deep Learning Based Techniques	28
5.9	Future Directions	28
6	Conclusion	28

List of Figures

1	Coastal Region of New South Wales used to Generate the Dataset	9
2	A Satellite Image of a Wildfire	10
3	Annotations using [33]	11
4	Cloud Mislabelled as Active Fire Points	11

5	Annotations using [36]	12
6	Examples of Images Removed from Manual Inspection	12
7	224 by 224 Pixel Image Patch in the Dataset	14
8	ResNet50 Features Extracted for Image in Figure 7	14
9	Xception Features Extracted for Image in Figure 7	14
10	Training for Models without Transfer Learning	17
11	Training for Xception-DeepLabV3+ with Transfer Learning and fine-tuning	17
12	Predictions for Feature Extractors into XGBoosted Tree	19
13	Predictions using ResNet50 Encoder with Transfer Learning	20
14	Predictions using Xception Encoder with Transfer Learning	21
15	Predictions for ResNet50 Encoder without Transfer Learning	22
16	Predictions using Xception Encoder without Transfer Learning	23

List of Tables

1	Parameter Bounds for Bayesian Optimization	15
2	ResNet50-XGBoost Hyperparameters	15
3	Xception-XGBoost Hyperparameters	15
4	Encoder-Decoder Model Hyperparameters	16
5	Feature Extractors into XG Boosted Tree Classifier Performance Metrics	18
6	Model Performance Metrics	18
7	GFLOPS for Encoder-Decoder Pairs	19

Abstract

Recent academic literature demonstrates a keen interest in combining satellite data with convolutional neural networks (CNNs) for wildfire monitoring, yet has an absence of research into active fire point segmentation constrained to visible spectrum input data. This study investigates deep learning-based approaches for active fire point segmentation using satellite data from the Sentinel-2 mission. The study aims to examine whether active fire point segmentation is feasible when hosted on a CubeSat nanosatellite with a Jetson Nano for computing resources, and visible spectrum images for model input. These limited computing resources highlight the need for creating lightweight architectures. The approaches used are feature extraction into XG-Boosted Trees and using established encoder-decoder models with and without transfer learning. Xception DeepLabV3+ is the top-performing model that can be hosted on a Jetson Nano, achieving an F1 score of 0.08788. Significant challenges such as class imbalance, limited dataset size, cloud cover, and the target variable's inherent sparsity and sporadic nature are credited for poor performance. Outcomes suggest alternate target variables such as smoke or wildfire areas, and emphasise the need for a benchmark dataset that would allow for objectively comparing approaches in the literature.

1 Introduction

Wildfires in Australia pose significant threats to environments, communities, and socio-economic stability. The 2019 - 2020 Australian bushfires were responsible for over 32 fatalities, 3094 homes destroyed, 19 million hectares of burnt land, A\$4-5 billion in economic losses to the Australian food system, and impacting approximately 3 billion animals [1, 2]. During the Australian bushfires, a large volume of CO₂ was released, negatively impacting air quality [3]. Increased airborne particulate matter and gaseous combustion products have been linked to increased incidence of vascular and respiratory disorders [4]. Swift and precise detection of active fire points are crucial as wildfires typically spread at an average pace of 23km per hour [5]. Timely identification enables the prioritisation of relief efforts and aids in mitigating the impacts mentioned above. Using satellite imagery to observe fire landscapes has rich history, dating back to the 1980s [6]. In recent years, the fusion of artificial intelligence, specifically convolutional neural networks (CNNs), and satellite data, has emerged as a promising approach to wildfire segmentation. Ghali et al. in [7] provide a comprehensive review of 59 deep learning approaches for wildland fire detection, mapping, and prediction over 2018 to 2022 and is a testament to the interest in this area. Additionally, satellite-based fire detection methods reduce the manual effort and cover vast areas in contrast to alternative fire detection methods such as fire observation towers and cameras, ground observation, air surveillance, and local reports. This research aims to investigate the semantic segmentation of wildfires in raster data within CubeSats. Wildfire semantic segmentation was chosen over fire detection as it provides more detailed and valuable information. While detection methods can determine the presence or absence of fire, semantic segmentation provides further insights by delineating active fire borders. This additional information indicates the severity and impact of an active wildfire, thus allowing for more informed resource allocation for relief efforts.

CubeSats are a form of spacecraft classified as a nanosatellite, characterised by a maximum mass of 10kg [8]. They are traditionally used as Low Earth Orbit (LEO) satellites in applications such as remote sensing and communications and typically are operational for 3 to 12 months [9]. A single 'unit' is 10cm in height, width, and breadth and weighs approximately 1kg, depending on the instruments onboard. CubeSats can be used as a single unit or in groups of up to 24. Depending on build specifications, costs can range from a few thousand to a few million dollars. NASA's Terra satellite cost approximately USD\$1.3 billion for the spacecraft, US instruments, and launch in 1999, while the Sentinel-2 mission costs are not publicly disclosed yet are estimated to cost several hundred million euros [10]. CubeSats can use launch providers or rideshare programs to reach orbit. SpaceX provides rideshare options, and a US-based company called NanoRacks charges approximately USD\$90000 per single CubeSat unit to be deployed from the International Space Station [11]. Thus the relatively low cost and small size of the CubeSat make space missions significantly more accessible to the common public. Mission objectives can vary across science research, educational projects, commercial use, and technology demonstrations for space instruments. In the context of this project, CubeSats provide a cost-effective and scalable solution for monitoring large areas. In contrast to current remote sensing methods for wildfire detection, developing models for wildfire detection deployed on CubeSats will subsequently improve the ability to map and detect wildfires closer to real-time, allowing for the potential improvement of wildfire detection and response. The small size and relatively low cost of the CubeSats enable dedicating a single CubeSat or a small group to a specific role. Alternatively, larger satellites are required to serve multiple functions. However, considering the limited computational resources of CubeSats, it is crucial to examine the implications of this resource-constrained environment to determine whether they are suitable to host CNNs for wildfire segmentation. For this research, we use computational capabilities of the Jetson Nano as a benchmark. This device delivers 472 giga floating-point operations per second (FLOPS) [12].

In addition to computational resource constraints, data constraints need consideration. For wildfire semantic segmentation inference on the CubeSat, the input data is constrained to the visible spectrum to simulate a realistic environment.

2 Literature Review

Wildfire areas and active fire points are referred to with two distinct definitions throughout this research. Wildfire area segmentation encompasses any area that has been burned or is currently burning. Wildfire area segmentation refers to segmenting the whole area surrounded by a wildfire perimeter. Active fire point segmentation refers to segmenting points in an image experiencing an active fire.

Ghali et al. in [7] perform a comprehensive review of 59 deep learning techniques from 2018 to 2022 for wildfire detection, mapping, and prediction using satellite data. The detection methods discussed are the most relevant to this project and are further classified into classification and segmentation approaches. Similarly,

in a second paper, Ghali et al. in [13] perform a deeper investigation into detection, emitting the sections on mapping and prediction in [7]. [7, 13] discuss common challenges associated with wildfire-related deep learning projects, such as class imbalances and a lack of annotated datasets. Both reviews demonstrate that visible spectrum images yield satisfactory results in image classification tasks. [7] discusses differing forms of wildfire segmentation. Namely, segmentation of wildfire areas in satellite imagery, segmentation of active fire points in satellite-imagery, segmentation of wildfires in non-satellite imagery, and segmentation of smoke in satellite and non-satellite imagery. Visible spectrum images yield promising results in wildfire area segmentation using satellite data. At the same time there is an absence of research into active fire point segmentation using satellite imagery input constrained to the visible spectrum (which is the purpose of this report). All methods discussed in these reviews for active fire point segmentation use inputs outside visible spectral bands. [13] subdivides fire detection into classification, detection, and non-satellite-based segmentation, emitting satellite-based wildfire area and active fire point segmentation entirely. These reviews are a testament to the interest in combining deep learning techniques with wildfire satellite (and non-satellite) data

[14, 15] use CNNs with raster data for semantic segmentation of natural disasters, extending beyond fire-related scenarios. [15] aimed to produce an early oil spill identification system using intersection-over-union (IoU) and mean IoU (mIoU) as evaluation metrics. Oil spills, look-alikes, ships, land, and sea were considered classes. Using a variety of epochs, they achieved a maximum mIoU of 0.606 and an IoU of 0.503 for oil spills. Similarly, [14] aimed to generate natural disaster insights by considering changes in man-made features, allowing for a more general model that can be used for different types of natural disasters. Additionally, they developed a novel measure of disaster impact, namely the Disaster Impact Index (DII). The article emphasised that model in conjunction with the DII, would allow disaster responders to concentrate relief effort priority based on impact. The Santa Rosa Fire and Hurricane Harvey Flood were used as validation data pieces, achieving F1 scores of 0.812 and 0.854 ,respectively.

In [5], James et al. implemented a classification system using MobileNetV2 to distinguish between images with and without fire. They synthesised a dataset of 614 images by superimposing fire onto satellite imagery. Their model was trained using transfer learning methods. They investigated implications of variability in input image resolution, network depth multiplier, number of neurons in the dense layer, and dropout rates, on performance and computing resources. Flash usage, peak RAM usage, and inference timing were used as computing resource measures. Their findings indicate that input image resolution and network depth multiplier are the significant parameters to consider when evaluating a trade-off between classification accuracy and computational resources required. Keeping depth multiplier constant, they observe a test error increase slightly below 6% to 13% when transitioning from 160 by 160 pixel to 96 by 96 pixel input data.

Several studies have investigated segmentation techniques for satellite-based wildfire area segmentation. [16] utilised a U-Net architecture with data collected from the infrared, microwave, and visible spectrum sources. Sources include the C-SAR instrument on Sentinel-1, the multispectral instrument from Sentinel-2, the sea and

land surface temperature instrument from Sentinel-3, and the Moderate Resolution Imaging Spectroradiometer (MODIS) instrument from the Terra and Aqua satellites. They created single-data source models from the sources mentioned above and investigated combinations of these models. A combination of Sentinel-1 and Sentinel-2 resulted in the best performance in cloudy conditions, with an F1 score of 0.72. A combination of Sentinel-2 and Sentinel-3 achieved the best result in clear conditions, with an F1 score of 0.87. [17] employed a U-Net-based wildfire area segmentation approach and achieved maximum F1 scores of 0.465 and 0.321 on the Planet and Resurs-P datasets, respectively. An emphasis was placed on data augmentation techniques due to the limited dataset sizes of 1850 images. [3] conducted a wildfire damage assessment in Australia using Sentinel-2, MODIS land use land cover product, and Google Earth Engine. They employed pixel subtraction of pre- and post-wildfire imagery to detect changes in vegetation cover and utilised feature selection of spectral band combinations through the Harris Hawks Optimisation algorithm [18]. Feeding these inputs into a Random Forest Classifier yielded the highest accuracy for burned area mapping at 0.9102. These studies collectively demonstrate the effectiveness of satellite-based wildfire area segmentation.

[19] curated a novel dataset comprising 146,213 image patches from Landsat-8 using the OLI sensor during August and September 2020. They explored three algorithms for active fire annotation, namely those discussed in [20, 21, 22]. The study considered three U-Net architectures: U-Net(10c), U-Net(3c), and U-Net-Lite(3c), using 10, 3, and 3 spectral bands as inputs, respectively. In the (3c) versions, spectral bands 2, 6, and 7 were used. A maximum F1 score of 0.942 was achieved using a U-Net (10c) with a voting scheme. The voting scheme is used for mask annotations where a pixel is considered a fire pixel if at least two of the three methods demonstrated in [20, 21, 22] mark it as a fire pixel. They observed minimal performance differences between the largest and smallest networks, with U-Net-Lite(3c) achieving a maximum F1 score of 0.932.

[23] discusses alternate approaches to satellite-based fire detection using Faster Region-based CNN (RCNN) with ground-based video. They emphasise the decreased response times in contrast to satellite-based methods. Additionally, they introduce synthetic smoke to images to expand their data. However, [14] suggests that satellite-based methods avoid the manual effort involved in alternative methods. Intuitively, satellite-based methods cover larger areas with a single process.

[24, 25] investigate the use of smoke as a proxy for wildfire. [24] presents a deep learning model for smoke classification, SmokeNet. Their model incorporates spatial and channel-wise attention for smoke scene detection using MODIS data from the Terra and Aqua satellites. The dataset is comprised of 6,225 manually labelled 1km spatial resolution visible spectrum images, with six approximately equally-balanced classes: cloud, dust, haze, land, seaside, and smoke. The proposed model achieves an accuracy of 0.9275. Alternatively, [25] proposes a novel deep learning model for smoke segmentation in satellite video, namely the attention-guided optical satellite video smoke segmentation network model (AOSVSSNet). The model uses a U-Net++-derived architecture, MobileNetV2 as the convolution unit to reduce training parameters, a Convolutional Block Attention Module (CBAM), and a combined loss function comprising four error terms. They note challenges due to a lack of

smoke datasets with segment annotations, and thus synthesise a dataset based on the optical imaging principle. The generalisation performance of the model is validated on real-world data. The synthetic dataset consists of 10,000 256 by 256 pixel images, while the real dataset contains 200 images. Testing on real-world data, the model achieves a mean intersection-over-union (mIoU) of 0.7284, outperforming DeepLabV3+, UNet, and FCN, with scores of 0.7214, 0.6567, and 0.2963, respectively.

[6] reviews satellite remote sensing for detecting and monitoring active fires via non-deep learning techniques. Active fire detection algorithms are categorised into fixed threshold algorithms, contextual algorithms, multi-temporal algorithms, and non-thermal methods. Certain traditional method instrumentation (such as MODIS) allows fire radiative power calculations, which are not commonly mentioned in the literature or deep learning related approaches. This paper is valuable for understanding the capabilities and limitations of non-deep learning methods in active fire detection. Furthermore, the inaccuracies associated with active fire products are discussed, which is relevant to this research as these methods generate masks. For example, reflective non-burning surfaces are sources of commission errors with short-wave infrared (SWIR) methods.

[26] emphasises the effectiveness of SWIR Sentinel-2 satellite data in active fire detection and proposes a CNN-based method to upscale the spatial resolution from 20m to 10m, aiming to achieve more detailed active fire detection maps. They propose a novel super-resolution technique, SRNN₊, in comparison to common Single Image Super Resolution (SISR) and Super Resolution Fusion (SRF) methods. To evaluate the performance of the SRNN₊ method, they use Spectral Angular Mapping (SAM), Universal Image Quality Index (UIQI), Relative Dimensionless Global Error, High-frequency Correlation Coefficient (HCC), precision, recall, and IoU. Their proposed CNN-based super-resolution technique results in superior performance for all metrics excluding IoU and recall.

[27] covers a series of loss functions applicable to semantic segmentation problems. Distribution-based, region-based, boundary-based, and compounded loss functions are covered. The paper summarises the advantages and limitations of each loss function and highlights their impact on the performance of semantic segmentation models. Loss functions were evaluated using Dice coefficient, sensitivity, and specificity. Binary Cross-Entropy, Focal, and Dice loss are notable losses discussed that were used in this report, and achieve Dice coefficients of 0.968, 0.970, and 0.936, respectively. Focal loss and Dice loss are discussed as approaches to problems with a significant class imbalance within the dataset. The authors acknowledge the absence of a universally effective loss function for all semantic segmentation tasks, emphasising that its suitability relies on the specific characteristics of the dataset.

[28] performs a survey on state-of-the-art model compression techniques. Namely, model pruning, parameter quantisation, low-rank decomposition, knowledge distillation, and lightweight model design. These compression techniques are justified in the context of well-established frameworks (AlexNet, GoogLeNet, ResNet etc.) and their computational complexity on ImageNet. Lightweight model design via efficient architectures is the most prominent technique in recent years.

3 Methodology

3.1 Data

The following satellite missions were initially considered for data gathering.

NASA's Terra and Aqua satellites with the MODIS sensor [29].

- 1km spatial resolution for thermal anomalies/fire.
- Revisit time of 1-2 days.

NASA's Landsat 8 and 9 satellites [30].

- 30m spatial resolution for fire detection.
- Revisit time of 8 days.

ESA's Sentinel-2 satellites (Sentinel-2A and Sentinel-2B) [31].

- 20m spatial resolution for fire detection.
- Revisit time of 5 days.

Ultimately the Sentinel-2 satellites were selected with the intuition that more finely detailed imagery would translate into superior model performance. Sentinel-2 is a high-resolution, multispectral earth observation imaging mission consisting of twin satellites sharing a polar orbit phased at 180°. The instruments on board capture thirteen spectral bands and have a revisit time of five days at the Equator [31]. Its primary objective is land monitoring. Spectral bands 2, 3, 4, 11, and 12 were collected for this project. Corresponding to red, green, blue, and two SWIR bands, respectively. The visible spectrum bands have a spatial resolution of 10m, while the SWIR bands selected have a spatial resolution of 20m. Thus a 20m spatial resolution was used during this project.

Dataset images were collected using SentinelHub's process API via Python [32]. Each image encompassed an area of approximately 2000 square kilometres, resulting in images with dimensions ranging from approximately 2000 to 2500 pixels in width and height at a 20m spatial resolution. Images were filtered for less than 1% cloud cover, yet the received images do not adhere strictly to this filtering. Images were subsequently cropped to a standardised size of 2000 by 2000 pixels. From the cropped images, non-overlapping patches of size 224 by 224 pixels were created. These patches served as the inputs for following analysis and modelling.

An initial dataset was constructed using seven bounding boxes that contained significant fires during the 2019-2020 Australian bushfire season. A total of 270 images were collected over the period ranging from 1 November 2019 to 1 March 2020. From these images, 9991 patches were extracted, which were subsequently filtered into 189

patches containing an arbitrary threshold of 100 or more fire pixels. Model performance using this dataset was unsatisfactory, and thus it was deemed unsuitable for wildfire segmentation and was subsequently abandoned.

In an attempt to build a more extensive dataset, 1590 images were collected from a coastal region of New South Wales spanning from 1 September 2019 to 1 January 2020. The aforementioned Australian 2019-2020 bushfires rampaged before, throughout, and after this period. These images were subsequently filtered into 174 images containing active fire points. 11264 patches were extracted from these selected images. Once again, a filtering criterion of 100 or more active fire pixels was applied, resulting in 524 patches meeting this threshold. Following this, a manual inspection was conducted to ensure data quality. This inspection revealed occasionally anomalous annotations for coastal regions and other irregular images. These images are depicted in Figure 6, Section 3.2. Consequently, the dataset was further refined, resulting in 484 image patches. The following coordinates bound the area of interest: : longitude 150°to 153.5°and latitude -30°to -35°, as depicted in Figure 1. The largest fire during the 2019-2020 Australian bushfire season occurred in this region, namely the Gospers Mountain Fire. The selection of this region was deliberate, as it features numerous cities and towns. We aimed to produce a model which could effectively detect fires in densely populated regions, thus helping to potentially mitigate infrastructure damage and the loss of life during wildfires.

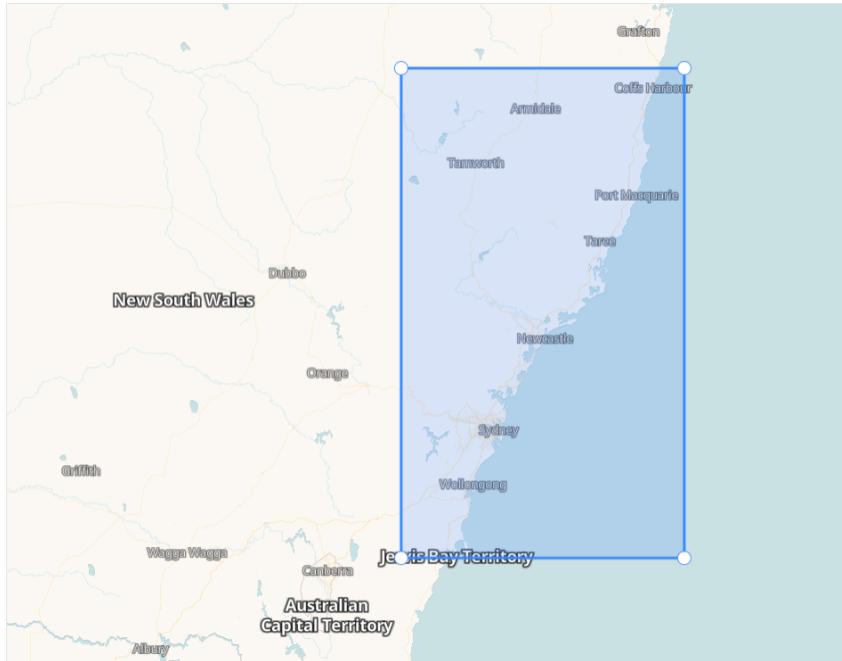


Figure 1: Coastal Region of New South Wales used to Generate the Dataset

The dataset exhibits a significant class imbalance, as illustrated by the pixel-label counts below.

- 24 031 651 background pixels.
- 253 533 active fire point pixels.

3.2 Data Annotation

The annotation of active fire points in wildfire segmentation poses a significant challenge. Annotation methods typically use SWIR light. Evaluating these methods is challenging as it is unknown which pixels are truly active fire points; thus, a qualitative evaluation of whether annotations look appropriate is used. [20, 21, 22] propose methods for active fire detection using Landsat-8 OLI data, resulting in differing masks, as demonstrated in [19], illustrating the variability and subjectivity inherent in the choice of annotation methods. Figure 2 below illustrates that the manual annotation of visible spectrum satellite wildfire data segments is infeasible. While having an approximate understanding of active fire point locations is possible, specifying the exactl locations through smoke and cloud is simply impossible.



Figure 2: A Satellite Image of a Wildfire

Thus we resort to alternative methods that utilise SWIR light to annotate active fire points. Four approaches to annotating wildfires were investigated.

- Versions one to three of Pierre Markuse's wildfire detection scripts [33, 34, 35].
- A script using a normalised green-red difference index and a spectral angle mapper index [36].

As previously discussed, the approach to evaluating these scripts was to inspect whether annotated areas aligned with the expected locations of active fire points and to identify any potential mislabeling of non-fire components as fire points. This approach is not highly scientific, yet it serves its purpose of filtering poor-quality annotations. The results of the third version of the Pierre Markuse scripts can be seen in Figure 3. This script is the inbuilt SentinelHub script for wildfire annotation.



(a) SentinelHub built-in Wildfire Annotations



(b) Sentinel build-in Wildfire Annotations using Single Colour

Figure 3: Annotations using [33]

Although the results in Figure 3 appear promising for annotating the data, it often annotates cloud pixels as active fire points for certain instances, as depicted in Figure 4 below.

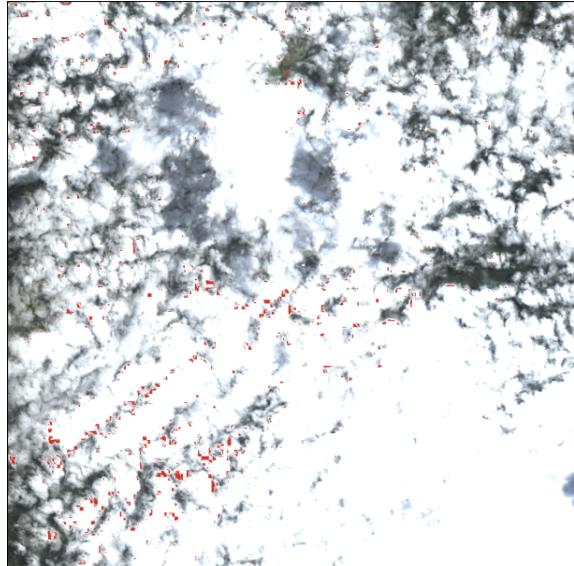
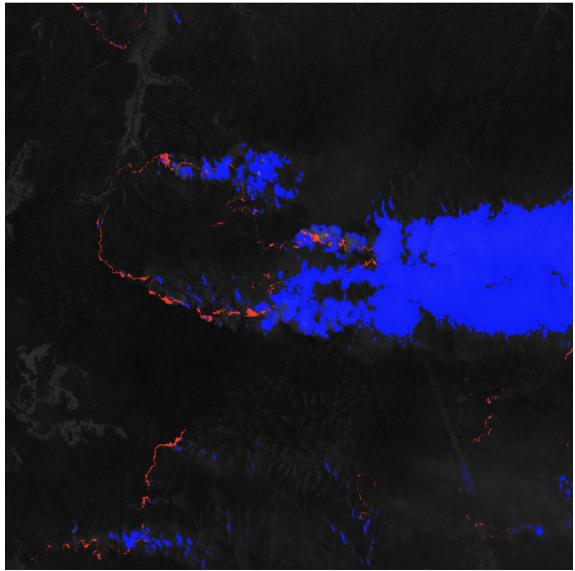
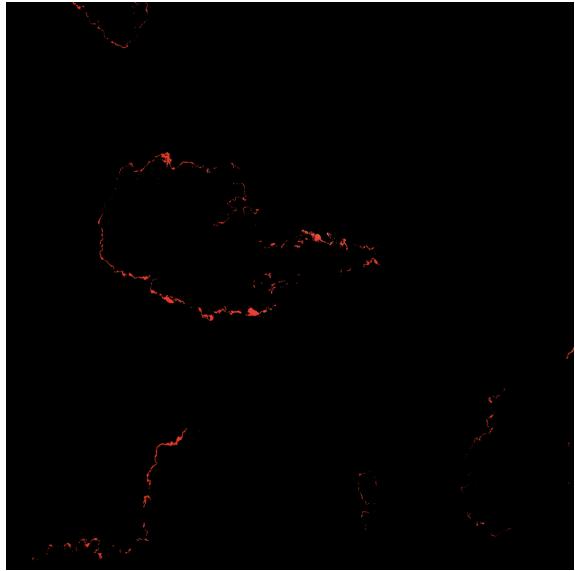


Figure 4: Cloud Mislabelled as Active Fire Points

The second method actively annotates both active fire points and clouds and thus prevents clouds from being labelled as active fire points. The cloud masks can be removed, leaving an active fire point mask, as depicted in Figure 5.



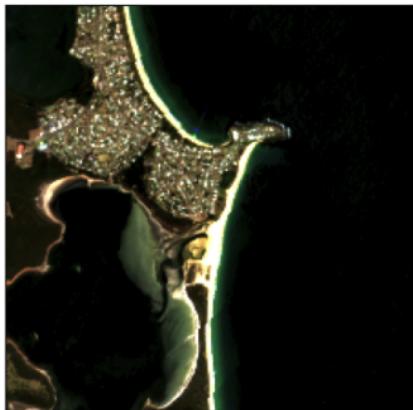
(a) Cloud and Fire Annotations



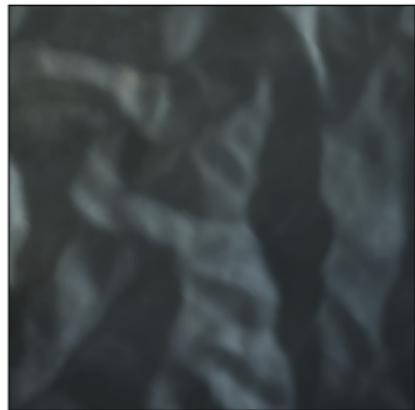
(b) Fire Mask

Figure 5: Annotations using [36]

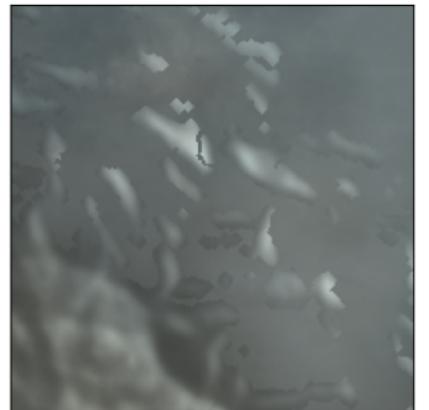
As mentioned in 3.1, the second method occasionally mislabels certain instances of coastline as active fire points. Additionally, we have irregular images with unnatural artefacts that need to be removed. Examples of images like these are shown in Figure 6



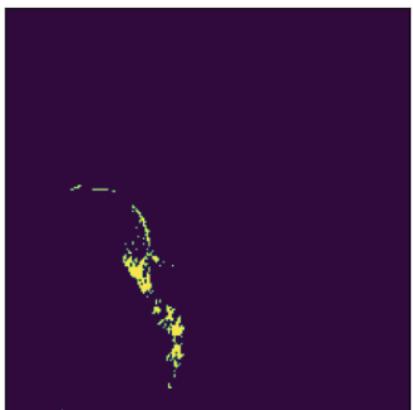
(a) Coastal Image



(b) Haze Image



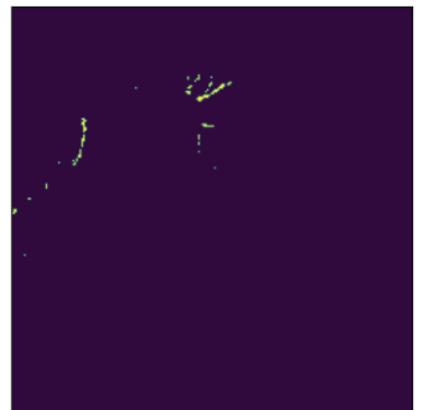
(c) Irregular Image



(d) Coastal Image Mask



(e) Haze Image Mask



(f) Irregular Image Mask

Figure 6: Examples of Images Removed from Manual Inspection

3.3 Model Setup

Following the data gathering process, standard machine learning methods were implemented. Input images were preprocessed according to the backbone network used.

- ResNet50: The images are zero-centred according to the ImageNet dataset.
- Xception: Input pixel values are scaled between -1 and 1, sample-wise.

The dataset was subsequently split into:

- 70% for training.
- 20% for validation.
- 10% for testing.

Data was augmented in real-time during training using random horizontal and vertical flips. These augmentations were chosen as their outputs mirror real-world data.

Various loss functions were considered to approach the class imbalance in the data [27].

- Binary Cross-entropy: Common loss for a binary classification problem. Explored as a benchmark.
- Binary Focal Cross-entropy: Down weights easy examples and focuses more on difficult examples.
- Dice Loss (Sørensen-Dice coefficient): A common approach to highly unbalanced data.
- A combination of Dice and Binary Focal Cross-entropy loss.

The above loss functions were evaluated on a ResNet50-DeepLabV3+ model trained on 100 epochs. F1 score was the evaluation metric. Dice loss resulted in the best performance and was used in all following models.

The model methodologies used in this project can be split into two main sections:

- Using established frameworks as feature extractors and then feeding these into an XGBoosted Tree.
- Using established encoder-decoder frameworks with and without transfer learning.

3.3.1 Feature Extractors into XG Boosted Trees

ResNet50 and Xception preloaded with ImageNet weights were chosen as feature extractors [37]. The output of the first convolutional layers was taken as the features fed into an XGBoosted Tree. This is an instance of transfer learning, in which the objective was to use the knowledge of established architectures to extract high-level features, and was used to mitigate the effects of a small dataset. The output of these feature extractors is demonstrated in Figures 8 and 9 below.



Figure 7: 224 by 224 Pixel Image Patch in the Dataset

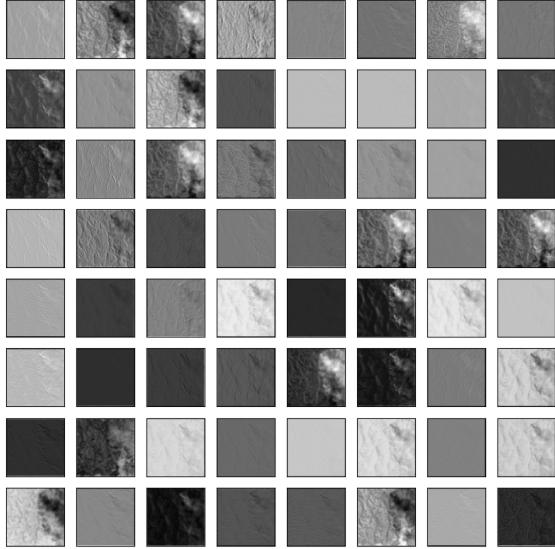


Figure 8: ResNet50 Features Extracted for Image in Figure 7

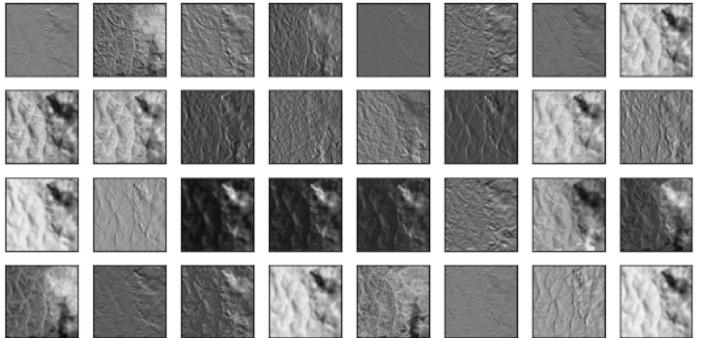


Figure 9: Xception Features Extracted for Image in Figure 7

Bayesian optimisation was employed to tune the hyperparameters of the XGBoost classifier. Using cross-validation, the optimisation was defined to optimise the mean area under the precision-recall curve (AUCPR) score. The optimisation process takes several hyperparameters as input and returns the mean AUCPR score.

The XGB Classifier used a binary logistic objective function. The hyperparameters to be optimised and their respective bounds are as follows:

Hyperparameter	Bounds
<code>max_depth</code>	(3, 10)
<code>learning_rate</code>	(0.01, 0.3)
<code>n_estimators</code>	(100, 1000)
<code>gamma</code>	(0, 5)
<code>min_child_weight</code>	(1, 10)
<code>subsample</code>	(0.5, 1.0)
<code>colsample_bytree</code>	(0.5, 1.0)

Table 1: Parameter Bounds for Bayesian Optimization

Following Bayesian optimisation, the hyperparameters used can be found in the tables below:

Parameter	Value
<code>colsample_bytree</code>	0.57353
<code>gamma</code>	4.66499
<code>learning_rate</code>	0.10981
<code>max_depth</code>	8
<code>min_child_weight</code>	4.86660
<code>n_estimators</code>	240
<code>subsample</code>	0.66570

Table 2: ResNet50-XGBoost Hyperparameters

Parameter	Value
<code>colsample_bytree</code>	0.76084
<code>gamma</code>	1.49139
<code>learning_rate</code>	0.02882
<code>max_depth</code>	8
<code>min_child_weight</code>	3.24148
<code>n_estimators</code>	659
<code>subsample</code>	0.72225

Table 3: Xception-XGBoost Hyperparameters

Further investigation into this approach was halted due to poor model performance, as depicted in the results section.

3.3.2 Established Encoder-Decoder Frameworks

ResNet50 and Xception were chosen as backbones to be used in the DeepLabV3+ and UNet architectures [38, 39, 40, 41]. These four combinations were further split into training the model without preloaded weights and using transfer learning techniques with ImageNet weights. Transfer learning with fine-tuning involved training the decoder with an encoder that consists of frozen weights derived from ImageNet training, unfreezing the encoder weights, and then retraining the model with a low learning rate. Plots for a given backbone architecture pair can be found at [42]. The models are as follows:

- ResNet50-UNET
- ResNet50-UNET with transfer learning and fine-tuning
- ResNet50-DeepLabV3+
- ResNet50-DeepLabV3+ with transfer learning and fine-tuning

- Xception-UNET
- Xception-UNET with transfer learning and fine-tuning
- Xception-DeepLabV3+
- Xception-DeepLabV3+ with transfer learning and fine-tuning

A backbone-architecture pair that does not use transfer learning was trained as follows. The model without any preloaded ImageNet weights is built. Bayesian optimisation using five initial points, ten iterations, and 100 epochs was used to select appropriate learning and dropout rates. A model with these best-performing parameters from Bayesian Optimisation was then trained using 500 epochs to determine when the model becomes overfit. A model is considered to be overfit once our validation loss deviates from our training loss. Once this point is established, the final model is trained using combined training and validation data and evaluated on the test set.

The training for a model using transfer learning with fine-tuning is as follows. The model is built with preloaded ImageNet weights in the encoder. These preloaded model weights are frozen. Bayesian optimisation using five initial points, ten iterations, and 100 epochs was used to select appropriate learning and dropout rates. A model with these best-performing parameters from Bayesian optimisation was then trained using 500 epochs to determine when the model becomes overfit. Once this point is established, models are trained to the epoch before they become overfit, have the encoder weights unfrozen, and then retrained with a low learning rate, specifically 0.00001. This retraining period is then analysed to find at which epoch the models become overfit during the fine-tuning phase. Once the optimal epochs are found, the models are trained using the combined training and validation set and evaluated on the test set.

The optimal hyperparameters calculated for each model are shown below.

Model	Learning Rate	Dropout	Epochs	Epochs(Fine-Tune)
ResNet50-UNET	0.00566	0.09018	100	-
ResNet50-UNET (TL)	0.00046	0.26142	50	40
ResNet50-DeepLabV3+	0.00928	0.13858	100	-
ResNet50-DeepLabV3+ (TL)	0.00001	0.13858	60	50
Xception-UNET	0.00145	0.12823	100	-
Xception-UNET (TL)	0.00001	0.08699	50	0
Xception-DeepLabV3+	0.00433	0.16521	100	-
Xception-DeepLabV3+ (TL)	0.00044	0.33549	30	45

Table 4: Encoder-Decoder Model Hyperparameters

It should be noted that if the epoch at which models became overfit was unclear, 100 epochs were used as a default, as is the case for all models that have not used transfer learning. The training for these models is illustrated in Figure 10 below and is discussed further in Section 5. We contrast this to the training process for the Xception-DeepLabV3+ model illustrated in Figure 11.

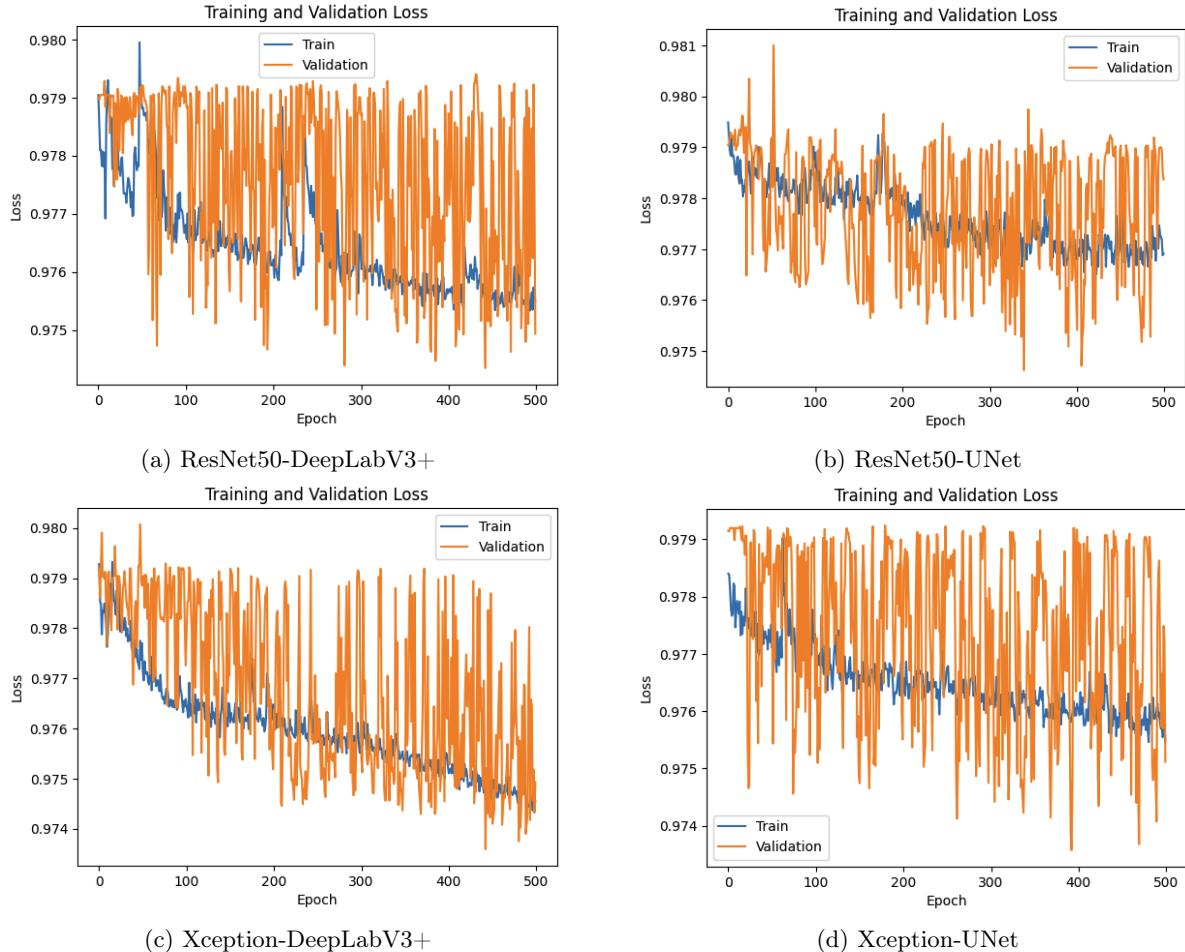


Figure 10: Training for Models without Transfer Learning

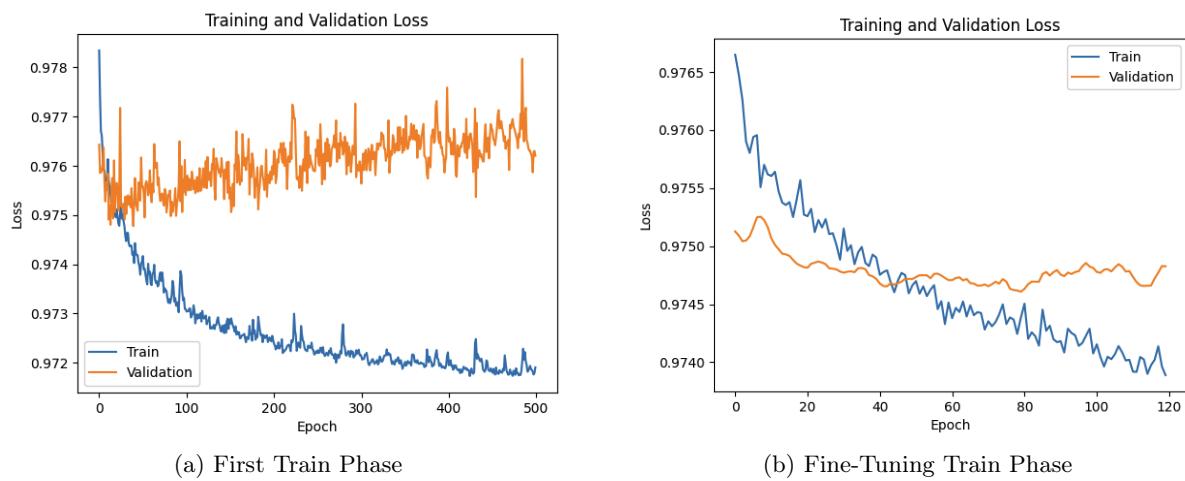


Figure 11: Training for Xception-DeepLabV3+ with Transfer Learning and fine-tuning

3.4 Performance and Computational Resource Evaluation

The evaluation of model performance relied on F1 score, precision, and recall. Accuracy was used to show further information regarding model performance. However, it was not a metric that was targeted for improvement. These metrics were chosen due to the significant class imbalance present in the dataset.

The computational resources required for models were evaluated through floating-point operations per second (FLOPS). FLOPS measures the quantity of floating-point operations used within a network. FLOPS provide insights into the computational requirements. As discussed in the introduction, this project considers a Jetson Nano as the computational resource on a CubeSat satellite with the capacity for 472 FLOPS [12]. FLOPS measures the quantity of floating-point operations used within a network.

3.5 Experimental Setup

The Tensorflow and Keras libraries were the primary libraries used for model construction. The models were hosted across the Google Colab T4, A100, and V100 GPUs. All methods used an Adam optimiser with default arguments excluding learning rate [43]. Learning rates and dropout rates were tuned using Bayesian optimisation. The batch size was set to 16.

4 Results

4.1 Summary of Results

Model Name	F1	Precision	Recall	Accuracy
Xception Feature Extractor	0.02764	0.01421	0.44444	0.99097
ResNet50 Feature Extractor	0.08173	0.04491	0.44012	0.98904

Table 5: Feature Extractors into XG Boosted Tree Classifier Performance Metrics

Architecture	Backbone	Transfer Learning	F1	Precision	Recall	Accuracy
DeepLabV3+	RESNET50	No	0.02889	0.59474	0.01479	0.61652
DeepLabV3+	RESNET50	Yes	0.08172	0.73282	0.04380	0.84409
DeepLabV3+	Xception	No	0.01995	0.98582	0.01005	0.06929
DeepLabV3+	Xception	Yes	0.08788	0.61192	0.04715	0.87777
U-Net	RESNET50	No	0.02598	0.98727	0.01322	0.29341
U-Net	RESNET50	Yes	0.04324	0.78839	0.02228	0.66639
U-Net	Xception	No	0.00100	0.00050	0.00186	0.98785
U-Net	Xception	Yes	0.09305	0.67655	0.04997	0.87362

Table 6: Model Performance Metrics

Architecture	Backbone	GFLOPS
DeepLabV3+	ResNet50	251
DeepLabV3+	Xception	267
UNet	ResNet50	778
UNet	Xception	863

Table 7: GFLOPS for Encoder-Decoder Pairs

4.2 Model Mask Predictions

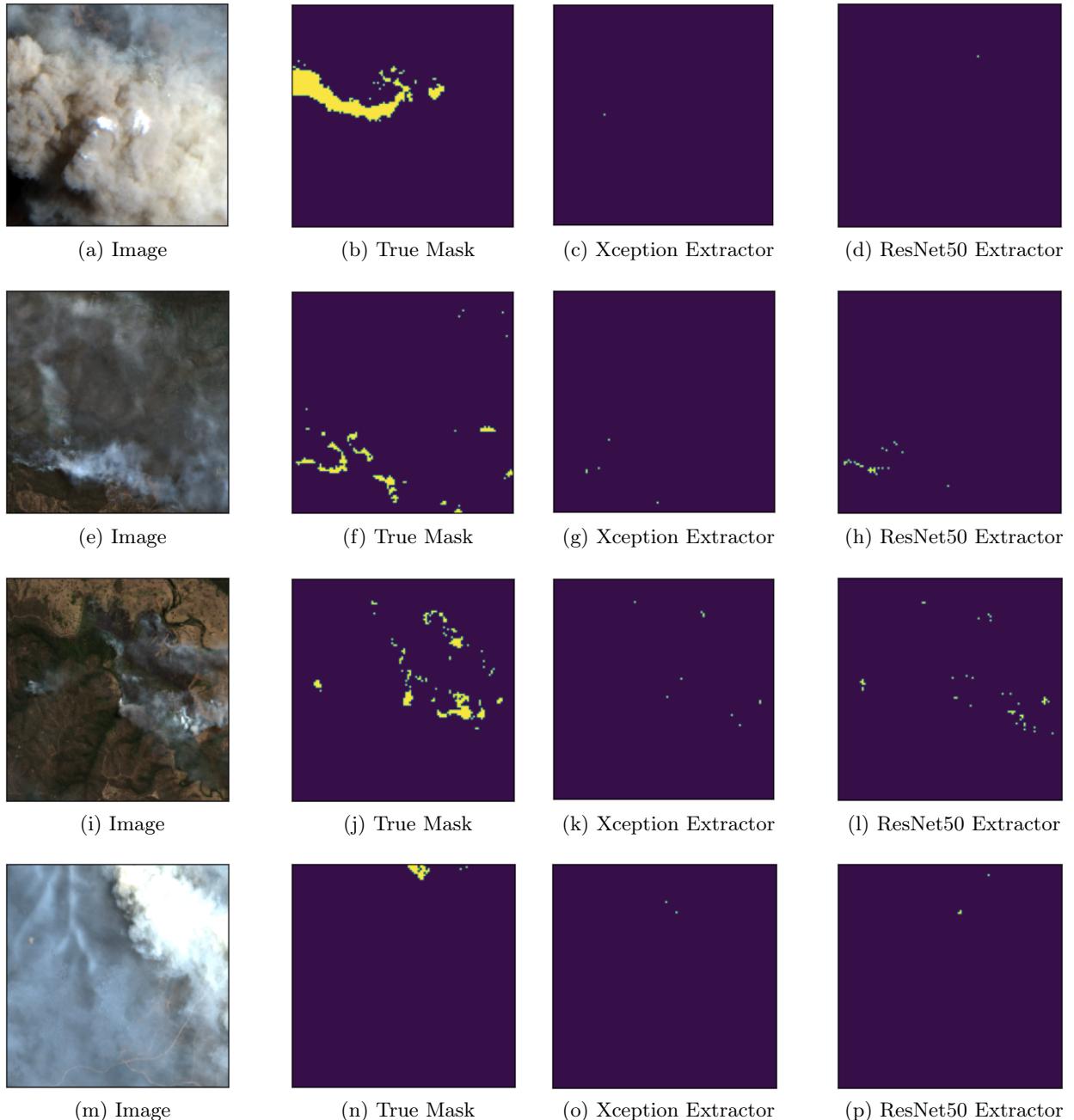


Figure 12: Predictions for Feature Extractors into XGBoosted Tree

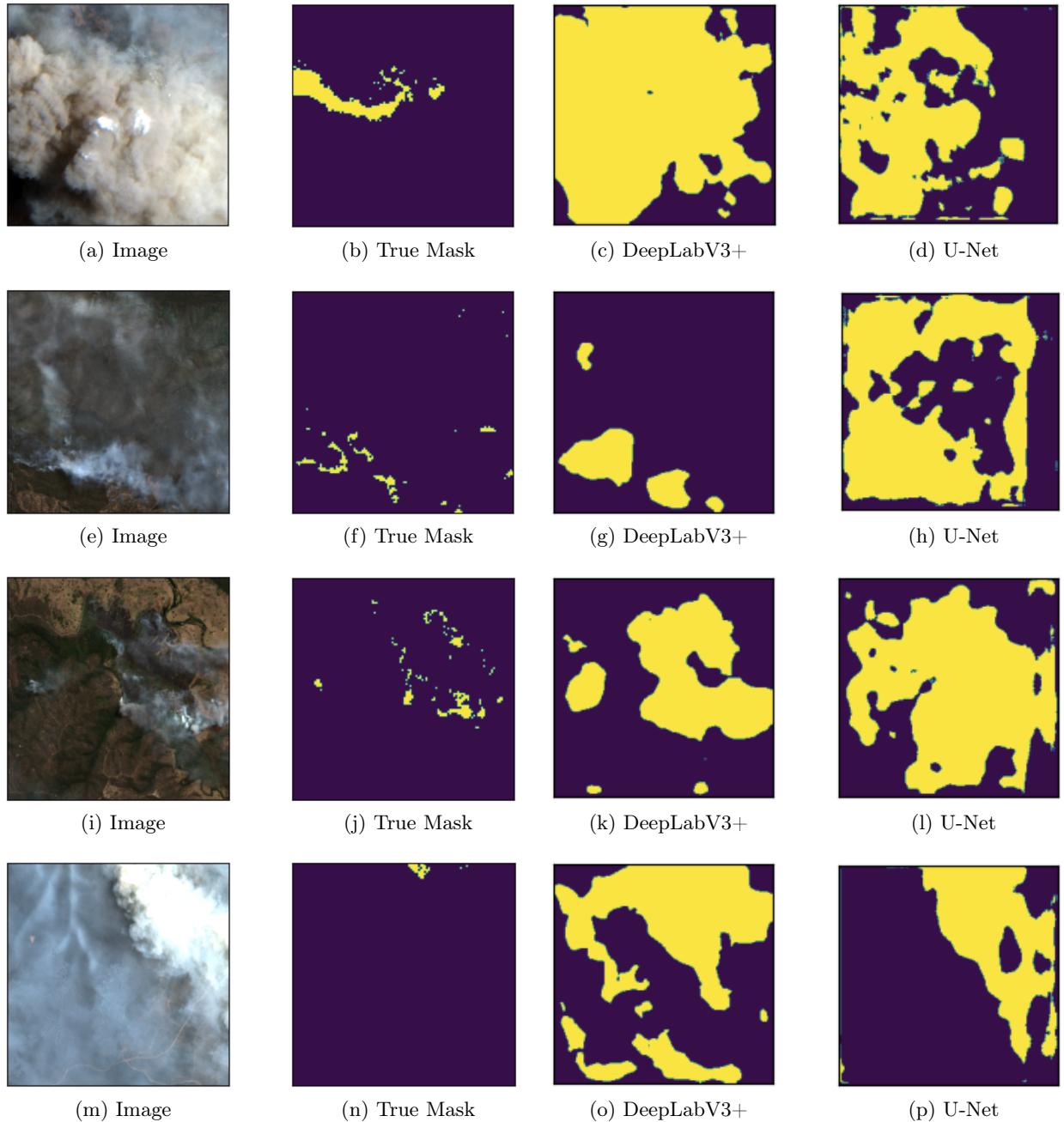


Figure 13: Predictions using ResNet50 Encoder with Transfer Learning

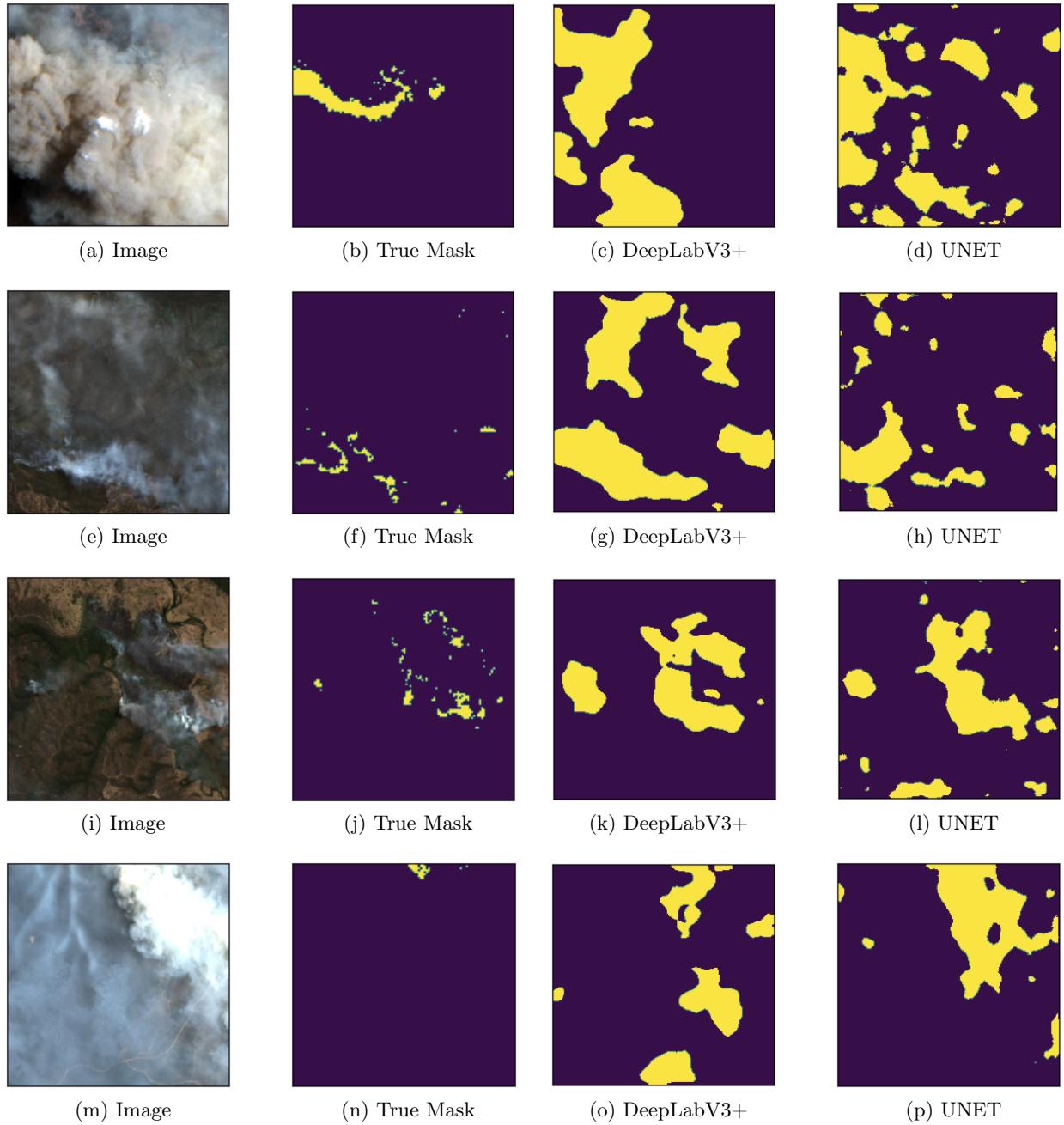


Figure 14: Predictions using Xception Encoder with Transfer Learning

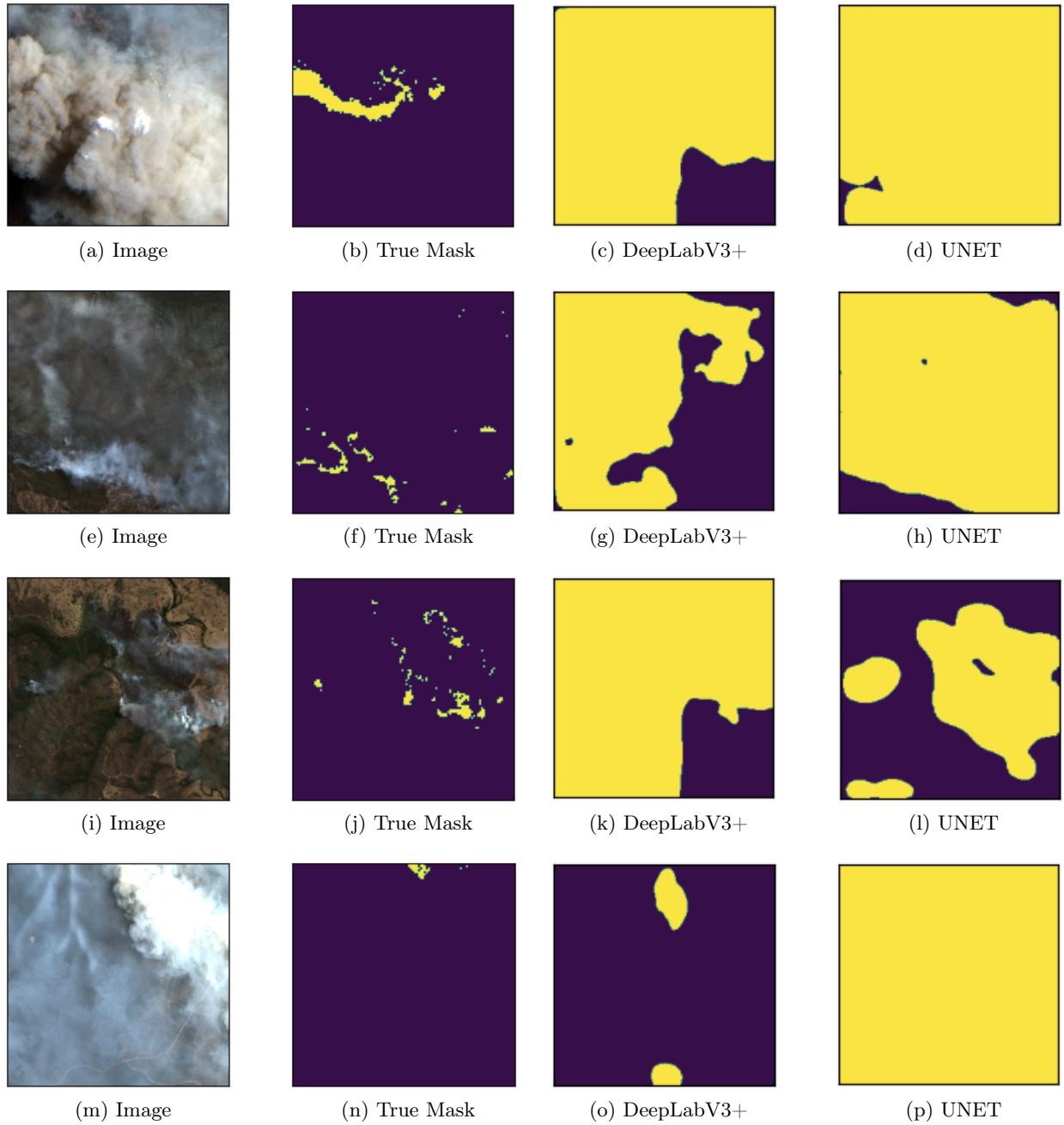


Figure 15: Predictions for ResNet50 Encoder without Transfer Learning

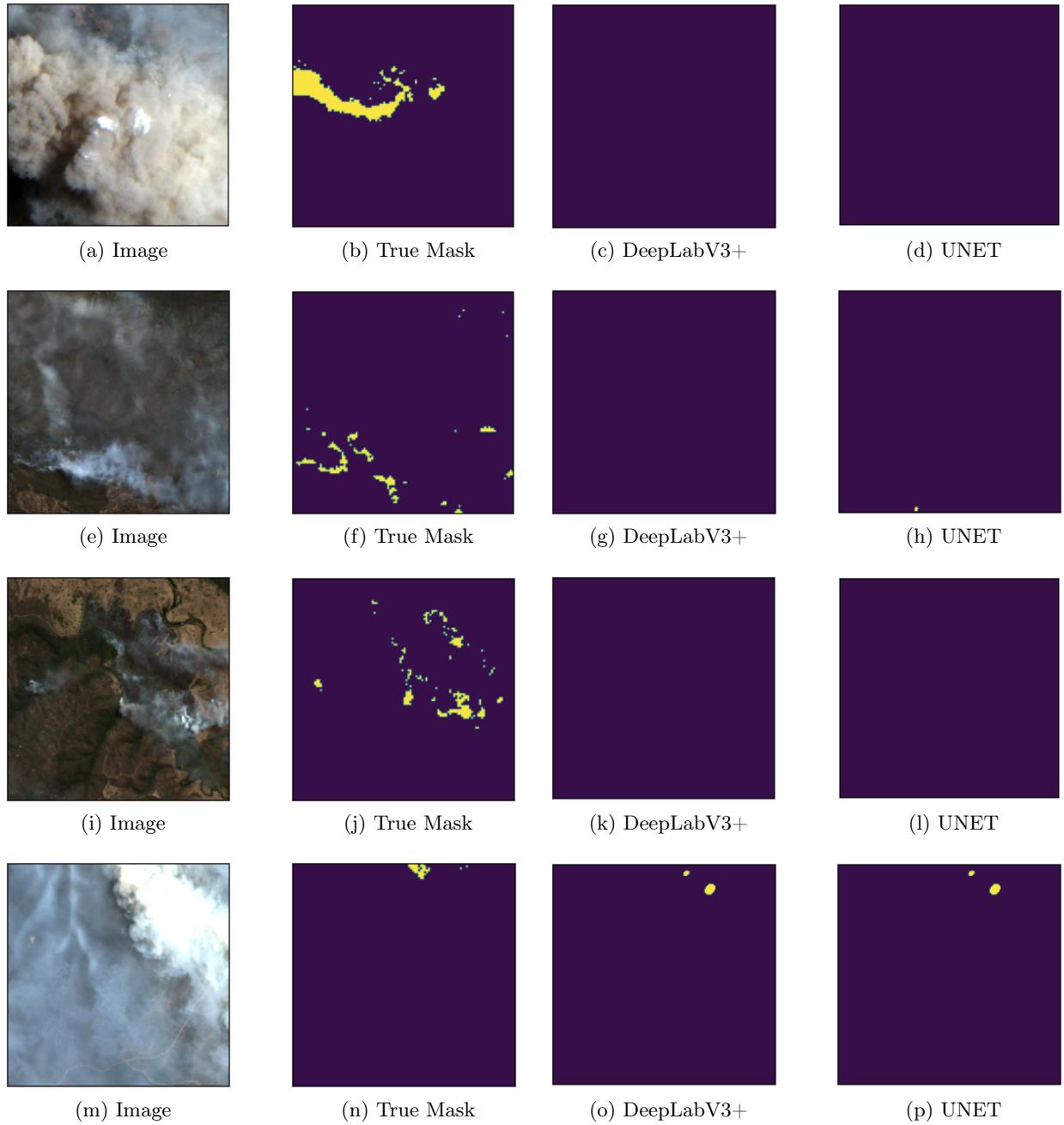


Figure 16: Predictions using Xception Encoder without Transfer Learning

5 Discussion

Once again, we remind the readers of this research that wildfire areas and active fire points are referred to with two distinct definitions throughout this investigation. Wildfire area segmentation refers to segmenting the whole area encompassed by a wildfire perimeter. In contrast, active fire point segmentation refers to segmenting points in an image that are experiencing an active fire. Wildfire area segmentation encompasses any area that has been burned or is currently burning.

5.1 Analysis of Results

In this research, we have examined several models for active fire point segmentation and evaluated their ultimate performance using F1 scores. The obtained results indicate that the F1 scores achieved by the investigated models are very low, leading us to conclude that the methods employed in this study are not satisfactory for the effective segmentation of active fire points in satellite imagery. However, despite this limitation, the research has provided valuable insights that can contribute to the advancement of this field.

Contrasting the two main modelling approaches, we observe that the ResNet50-based feature extractor approach achieves an F1 score of 0.08173, which is only marginally lower than the Xception DeepLabV3+ model with an F1 score of 0.08788. Interestingly, the feature-extraction-based approach yielded high recalls and lower precision, while this trend is reversed for our top-performing encoder-decoder models, demonstrating the well-known precision-recall trade-off.

Considering the encoder-decoder approach, models with the Xception encoder demonstrate the highest performance for both the DeepLabV3+ and U-Net architectures, attaining F1 scores of 0.08788 and 0.09305, respectively. Of these models, only the DeepLabV3+ Xception model’s 267 GFLOPS measurement does not surpass the 472 GFLOPS threshold of the Jetson Nano [12]. Consequently, if a choice had to be made among the models explored in this paper, the DeepLabV3+ Xception architecture would be the preferred option. Despite this model being able to be hosted using a Jetson Nano on a CubeSat, poor results suggest that further research should be conducted in an attempt to gain wildfire insights via alternative methods. Suggestions for alternative methods are discussed further in this research.

Additionally, we note that models using the DeepLabV3+ architecture achieve significantly lower GFLOPS measurements when compared to their U-Net counterparts, as seen in Table 7. The DeepLabV3+ and Xception frameworks leverage depthwise separable convolutions to reduce computation costs and the number of parameters while maintaining competitive (or better) performance [39, 40]. This is a testament to the importance of model structure choices when using CNNs in resource-constrained environments. Other notable architectures, such as MobileNet, MobileNetV2, and EfficientNet, similarly employ depthwise separable convolutions making them well-suited to resource-constrained scenarios [44, 45, 46].

5.2 Challenges

As previously mentioned in Section 3.2, the accurate classification of active fire points through smoke is a task in which we can not expect near-perfect accuracy without the use of electromagnetic wavelengths outside the visible spectrum. Although a rough general area for active fire points can be estimated, it is not clear what distinguishes an active fire point from a neighbouring non-active fire point when they are both shrouded by smoke. This lack of clarity evidently poses difficulties and is reflected in poor results. Of the true images shown in Figures 12 to 16, images (a) and (m) are examples of such cases in which it would be impossible to pinpoint active fire points. Consequently, our predicted masks from all models provide little insight into the active fire points. Conversely, images (e) and (i) in Figures 12 to 16 provide clearer visibility of active fire points. If we observe the predicted masks for the models that make use of transfer learning in Figures 13 and 14, we notice that they begin to roughly predict shapes that somewhat coincide with active fire points. This is a positive indication that a dataset of clearly visible, active fire data may be used for active fire point segmentation. However, it is crucial to acknowledge the limited generalisation ability of this task to conditions typically encountered when using satellite data. Using active fire points as a target segment results in significant inherent challenges, including but not limited to a lack of annotated datasets, data annotation variability, cloud cover, class imbalances within the data, low evaluation metric scores, and the inherently sparse and sporadic nature of the target segments themselves. Similar dataset-related challenges and limitations are highlighted in previous studies on combining deep learning with wildfire imagery [7, 13, 25].

5.3 Class Imbalance

The methodology employed throughout this study aimed to mitigate several of these challenges. One strategy implemented to address the class imbalance was the investigation of different loss functions, namely Binary Cross-entropy loss, Dice loss, Focal loss, and a combination of Dice and Binary Focal loss. Dice loss was observed to yield the highest F1 score when training a ResNet50 DeepLabV3+ model over 100 epochs. Additionally, filtering image patches based on an arbitrary minimum threshold of 100 fire pixels per patch served as another approach to mitigate the effects of inherently class-imbalanced data.

5.4 Limited Dataset Size

In Figure 10, there are large fluctuations in the validation dataset loss, distinguishing it from the training observed in our models that use transfer learning, depicted in Figure 11. This is a potential indicator that the models without preloaded ImageNet weights immediately begin to overfit, use learning rates that are too large, or that our validation dataset does not accurately reflect the characteristics of our training dataset. Several pieces of evidence support the notion that our validation dataset may not be an effective representation of our training and test datasets.

- We observe large fluctuations in validation loss.
- Training, validation, and test splits are more prone to not be representative of each other due to the limited size of our dataset.
- During training, we observe high recall and low precision. During evaluation on the test dataset, we observe the inverse with high precision and low recall metrics.

It is intriguing to observe that transfer learning models do not experience the same volatility during training. This suggests that the incorporation of preloaded weights has successfully introduced a significant level of generalisation. This is an endorsement of the use of transfer learning, as it was intentionally introduced to mitigate the effects of a limited dataset. Additional approaches to mitigate the limited dataset included on-the-fly random horizontal and vertical flip data augmentations, which are commonplace in the literature [7]. [47] augments its dataset using a CycleGAN data augmentation, demonstrating further approaches to limited dataset size.

5.5 Cloud Cover

Cloud cover is a significant factor in input image quality. In this study, input images with less than 1% cloud cover were sourced from SentinelHub’s Process API. This is not a true reflection of real-world conditions and severely reduces the quantity of data that can be used. [16] demonstrate the negative effect of cloudy conditions on their wildfire area segmentation models, with a best F1 score of 0.87 in clear conditions, and 0.72 in cloudy conditions. [16]’s methods use SWIR and achieve poorer results in cloudy conditions, thus we expect that methods constrained to the visible spectrum would experience harsher performance reductions in similar conditions. Methods with smoke based targets that can be distinguished from cloud could potentially mitigate the challenges due to cloud cover.

5.6 Benchmark Dataset

Creating a benchmark dataset specifically for wildfire-related deep learning projects would significantly aid research. This dataset should have a variety of classification and segment labels (e.g. smoke, active fire, cloud, wildfire area, land, sea). The lack of a benchmark dataset for wildfire segmentation poses significant challenges and implications for research and development in this field. A benchmark dataset serves as a standardised reference point that can be used to compare and evaluate the performance of different algorithms, models, and techniques within wildfire segmentation (and classification, detection, mapping, prediction) using raster data. Variations in differing annotation methods introduce further challenges to model comparison across data sources [19, 20, 21, 22]. Additionally, a wildfire benchmark dataset would encourage competition and innovation to produce results that ultimately have life-saving implications, among a host of other benefits. [7, 13, 14, 19, 25]

report relatively high-performing results, yet an objective comparison of these methods across differing datasets is challenging. Datasets typically vary across factors such as spatial resolutions, annotation methods, patch sizes, geographical locations, and spectral band inputs.

In [7, 13], there is an absence of investigation into active fire point segmentation using purely the visible spectrum. [16, 17, 25] perform wildfire segmentation on wildfire areas, while [24, 25] investigate the use of smoke as a proxy for wildfire. The absence of active fire point segmentation studies constrained to the visible spectrum within the existing literature could serve as a warning regarding the inherent difficulties and challenges associated with this task. Furthermore, this absence results in a lack of accurate comparison of method performance. Hopefully, future reviews and studies will include further research in this area, explicitly stating the challenges encountered in active fire point segmentation using the visible spectrum and providing a set of results for comparison.

5.7 Compression Methods

The original intent of this research was to establish a wildfire segmentation framework and then explore model compression methods and their performance implications. However, this objective has not been achieved as compression techniques outside of lightweight model architectures were rendered unnecessary by the following factors:

1. The DeepLabV3+ Xception models use depthwise separable convolution and are compatible with the Jetson Nano’s GFLOPS resource constraints. We contrast this to other U-Net-based encoder-decoder models that do not satisfy this constraint. Li et al. in [28] state lightweight architectures to be the most prominent technique for model compression in recent years, as demonstrated in the model structures used in this study that solely use lightweight architectures as a compression technique.
2. The overall poor performance of the active fire point segmentation models renders them non-viable for practical use.

Additional potential considerations that need to be considered are input data quality. In [5], James et al. vary input resolution for a MobileNetV2 wildfire classifier, noting increases in input resolution share a significant positive relation with classifier accuracy and computational resources required. Imaging instruments on the CubeSat will generate significantly lower quality images than that of Sentinel-2, and we suspect this has a large negative impact on performance, in accordance with [5]. However, this will need to be investigated. Other potential compression methods, such as quantisation and pruning, effectively reduce computing and memory resources required and may be relevant for future investigations [28, 48].

5.8 Implications for Non-Deep Learning Based Techniques

Traditional techniques use multispectral methods for active fire point annotation. They additionally can be used to estimate radiative fire power [6]. The poor results yielded by methodologies in this research strengthen the use case for traditional methods.

5.9 Future Directions

Considering the poor metrics and notable challenges encountered throughout this research, it is imperative to explore alternative avenues. Potential options include:

- Binary classification of patches with 'fire' and 'no fire' labels. Segmentation of the smoke and cloud within the image is then used to indicate fire size. If low cloud cover data is used, the smoke and cloud should give a relatively accurate indication of the size of the fire.
- Segmentation of wildfire areas rather than active fire points. A less sparse and sporadic target variable appears to translate to improved evaluation metrics. [3, 16, 17] observe better performance than our study through this approach. [16]'s top performing model uses Sentinel-2 spectral bands, including SWIR, and achieves a kappa coefficient of 0.61. A similarly decent kappa coefficient of 0.54 was achieved by their model constrained to the visible spectrum, suggesting that segmenting wildfire areas using the visible spectrum can yield promising results.
- Segmentation of smoke as a proxy for wildfire activity. Smoke segmentation reflects environmental conditions present in the fire scene, aiding rescuers in identifying the source of the fire and predicting spread trends and speed [25]. If possible, extending the smoke versus cloud classification in [24] to segmentation models would possibly mitigate the significant challenge of cloud cover in deriving wildfire insights from CNNs in satellite imagery.

6 Conclusion

This research has investigated active fire point segmentation using satellite imagery data constrained to the visible spectrum. Visible spectrum and floating-operations per second (FLOPS) constraints were introduced to simulate wildfire segmentation performed by a Jetson Nano on a CubeSat nanosatellite. The models used in this paper to achieve this objective yield unsatisfactory results, achieving a maximum F1 score of 0.08788 for a model compatible with the Jetson Nano capabilities. Despite the unsatisfactory results, the research has provided valuable insights for deep learning with wildfire satellite imagery. Significant challenges such as limited datasets, cloud cover, and class imbalance translated into poor results, despite varied approaches to mitigate these challenges. The need for a benchmark dataset is strongly emphasised in the context of these challenges and for comparing approaches present in the literature. Suggested future research directions involve reframing the input data target, further investigating constraints to input data, and model compression techniques. Smoke and wildfire areas are two notable target alternatives.

References

- [1] *Australian Bushfires of 2019/2020 / Australian Bushfires / WWF Australia.* URL: <https://wwf.org.au/what-we-do/australian-bushfires/>.
- [2] *Fire on the farm: Assessing the impacts of the 2019-2020 bushfires on food and agricultures in Australia.* URL: <https://www.preventionweb.net/publication/fire-farm-assessing-impacts-2019-2020-bushfires-food-and-agricultures-australia>.
- [3] Seyd Teymoor Seydi et al. “Wildfire Damage Assessment over Australia Using Sentinel-2 Imagery and MODIS Land Cover Product within the Google Earth Engine Cloud Platform”. In: *Remote Sensing* 13.2 (Jan. 2021), p. 220. ISSN: 2072-4292. DOI: 10.3390/rs13020220. URL: <https://www.mdpi.com/2072-4292/13/2/220>.
- [4] Pier Mannuccio Mannucci et al. “Effects on health of air pollution: a narrative review”. In: *Internal and Emergency Medicine* 10.6 (Sept. 2015), pp. 657–662. ISSN: 1828-0447, 1970-9366. DOI: 10.1007/s11739-015-1276-7. URL: <http://link.springer.com/10.1007/s11739-015-1276-7>.
- [5] George L. James et al. “An Efficient Wildfire Detection System for AI-Embedded Applications Using Satellite Imagery”. In: *Fire* 6.4 (Apr. 2023), p. 169. ISSN: 2571-6255. DOI: 10.3390/fire6040169. URL: <https://www.mdpi.com/2571-6255/6/4/169>.
- [6] Martin J. Wooster et al. “Satellite remote sensing of active fires: History and current status, applications and future requirements”. In: *Remote Sensing of Environment* 267 (Dec. 15, 2021), p. 112694. ISSN: 0034-4257. DOI: 10.1016/j.rse.2021.112694. URL: <https://www.sciencedirect.com/science/article/pii/S0034425721004144>.
- [7] Rafik Ghali and Moulay A. Akhloufi. “Deep Learning Approaches for Wildland Fires Using Satellite Remote Sensing Data: Detection, Mapping, and Prediction”. In: *Fire* 6.5 (May 7, 2023), p. 192. ISSN: 2571-6255. DOI: 10.3390/fire6050192. URL: <https://www.mdpi.com/2571-6255/6/5/192>.
- [8] Alén Space. *A Basic Guide to Nanosatellites*. Alén Space. URL: <https://alen.space/basic-guide-nanosatellites/>.
- [9] Canadian Space Agency. *CubeSats in a nutshell*. Canadian Space Agency. Apr. 12, 2017. URL: <https://www.asc-csa.gc.ca/eng/satellites/cubesat/what-is-a-cubesat.asp>.
- [10] NASA. *Terra: Flagship of the Earth Observing System*. NASA, Nov. 1999. URL: https://www.nasa.gov/pdf/156293main_terra_press_kit.pdf.
- [11] SatCatalog. *CubeSat Launch Costs*. SatCatalog. URL: <https://www.satcatalog.com/insights/cubesat-launch-costs/www.satcatalog.com>.

- [12] *Jetson Benchmarks*. NVIDIA Developer. Aug. 11, 2020. URL: <https://developer.nvidia.com/embedded/jetson-benchmarks>.
- [13] Rafik Ghali and Moulay A. Akhloufi. “Deep Learning Approaches for Wildland Fires Remote Sensing: Classification, Detection, and Segmentation”. In: *Remote Sensing* 15.7 (Jan. 2023), p. 1821. ISSN: 2072-4292. DOI: 10.3390/rs15071821. URL: <https://www.mdpi.com/2072-4292/15/7/1821>.
- [14] Jigar Doshi, Saikat Basu, and Guan Pang. *From Satellite Imagery to Disaster Insights*. Dec. 17, 2018. DOI: 10.48550/arXiv.1812.07033. arXiv: 1812.07033[cs]. URL: <http://arxiv.org/abs/1812.07033>.
- [15] Marios Krestenitis et al. “Early Identification of Oil Spills in Satellite Images Using Deep CNNs”. In: *MultiMedia Modeling*. Ed. by Ioannis Kompatsiaris et al. Vol. 11295. Cham: Springer International Publishing, 2019, pp. 424–435. ISBN: 9783030057091 9783030057107. DOI: 10.1007/978-3-030-05710-7_35. URL: http://link.springer.com/10.1007/978-3-030-05710-7_35.
- [16] Dmitry Rashkovetsky et al. “Wildfire Detection From Multisensor Satellite Imagery Using Deep Semantic Segmentation”. In: (2021). URL: [https://sadil.ws/handle/123456789/\\$%7Bsadil.baseUrl%7D/handle/123456789/3284](https://sadil.ws/handle/123456789/$%7Bsadil.baseUrl%7D/handle/123456789/3284).
- [17] Vladimir Khryashchev and Roman Larionov. “Wildfire Segmentation on Satellite Images using Deep Learning”. In: *2020 Moscow Workshop on Electronic and Networking Technologies (MWENT)*. 2020 Moscow Workshop on Electronic and Networking Technologies (MWENT). Mar. 2020, pp. 1–5. DOI: 10.1109/MWENT47943.2020.9067475.
- [18] Ali Asghar Heidari et al. “Harris hawks optimization: Algorithm and applications”. In: *Future Generation Computer Systems* 97 (Aug. 1, 2019), pp. 849–872. ISSN: 0167-739X. DOI: 10.1016/j.future.2019.02.028. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X18313530>.
- [19] Gabriel Henrique de Almeida Pereira et al. “Active Fire Detection in Landsat-8 Imagery: a Large-Scale Dataset and a Deep-Learning Study”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 178 (Aug. 2021), pp. 171–186. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2021.06.002. arXiv: 2101.03409[cs]. URL: <http://arxiv.org/abs/2101.03409>.
- [20] S. S. Kumar and D. P. Roy. “Global operational land imager Landsat-8 reflectance-based active fire detection algorithm”. In: *International Journal of Digital Earth* 11.2 (Feb. 2018), pp. 154–178. ISSN: 1753-8947, 1753-8955. DOI: 10.1080/17538947.2017.1391341. URL: <https://www.tandfonline.com/doi/full/10.1080/17538947.2017.1391341>.
- [21] Wilfrid Schroeder et al. “Active fire detection using Landsat-8/OLI data”. In: *Remote Sensing of Environment*. Landsat 8 Science Results 185 (Nov. 1, 2016), pp. 210–220. ISSN: 0034-4257. DOI: 10.1016/j.rse.2015.08.032. URL: <https://www.sciencedirect.com/science/article/pii/S0034425715301206>.

- [22] Sam W. Murphy et al. "HOTMAP: Global hot target detection at moderate spatial resolution". In: *Remote Sensing of Environment* 177 (May 1, 2016), pp. 78–88. ISSN: 0034-4257. DOI: 10.1016/j.rse.2016.02.027. URL: <https://www.sciencedirect.com/science/article/pii/S0034425716300554>.
- [23] Qi-xing Zhang et al. "Wildland Forest Fire Smoke Detection Based on Faster R-CNN using Synthetic Smoke Images". In: *Procedia Engineering*. 2017 8th International Conference on Fire Science and Fire Protection Engineering (ICFSFPE 2017) 211 (Jan. 1, 2018), pp. 441–446. ISSN: 1877-7058. DOI: 10.1016/j.proeng.2017.12.034. URL: <https://www.sciencedirect.com/science/article/pii/S1877705817362574>.
- [24] Rui Ba et al. "SmokeNet: Satellite Smoke Scene Detection Using Convolutional Neural Network with Spatial and Channel-Wise Attention". In: *Remote Sensing* 11.14 (Jan. 2019), p. 1702. ISSN: 2072-4292. DOI: 10.3390/rs11141702. URL: <https://www.mdpi.com/2072-4292/11/14/1702>.
- [25] Taoyang Wang et al. "AOSVSSNet: Attention-Guided Optical Satellite Video Smoke Segmentation Network". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 15 (2022), pp. 8552–8566. ISSN: 2151-1535. DOI: 10.1109/JSTARS.2022.3209541.
- [26] M. Gargiulo et al. "A CNN-based Super-resolution Technique for Active Fire Detection on Sentinel-2 Data". In: *2019 PhotonIcs & Electromagnetics Research Symposium - Spring (PIERS-Spring)*. 2019 PhotonIcs & Electromagnetics Research Symposium - Spring (PIERS-Spring). ISSN: 1559-9450. June 2019, pp. 418–426. DOI: 10.1109/PIERS-Spring46901.2019.9017857.
- [27] Shruti Jadon. "A survey of loss functions for semantic segmentation". In: *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. Oct. 27, 2020, pp. 1–7. DOI: 10.1109/CIBCB48159.2020.9277638. arXiv: 2006.14822[cs, eess]. URL: <http://arxiv.org/abs/2006.14822>.
- [28] Zhuo Li, Hengyi Li, and Lin Meng. "Model Compression for Deep Neural Networks: A Survey". In: *Computers* 12.3 (Mar. 2023), p. 60. ISSN: 2073-431X. DOI: 10.3390/computers12030060. URL: <https://www.mdpi.com/2073-431X/12/3/60>.
- [29] GISGeography. *MODIS: Moderate Resolution Imaging Spectroradiometer*. GIS Geography. Dec. 2, 2019. URL: <https://gisgeography.com/modis-satellite/>.
- [30] *Landsat 9 / U.S. Geological Survey*. URL: <https://www.usgs.gov/landsat-missions/landsat-9>.
- [31] *Sentinel-2: Satellite Imagery, Overview, And Characteristics*. Dec. 15, 2021. URL: <https://eos.com/find-satellite/sentinel-2/>.
- [32] *Sentinel Hub*. URL: <https://www.sentinel-hub.com/>.
- [33] *QuickFire 1.0 - Visualizing Fires in the Sentinel Hub EO Browser - Pierre Markuse*. Sept. 21, 2022. URL: <https://pierre-markuse.net/2022/09/21/quickfire-1-0-visualizing-fires-in-the-sentinel-hub-eo-browser/>.

- [34] *Visualizing (Wild)Fires in Sentinel-2 imagery through EO Browser* - Pierre Markuse. Aug. 7, 2017. URL: <https://pierre-markuse.net/2017/08/07/visualizing-wildfires-sentinel-2-imagery-eo-browser/>.
- [35] *Visualizing Wildfires and Burn Scars with the Sentinel Hub EO Browser V2* - Pierre Markuse. Apr. 30, 2018. URL: <https://pierre-markuse.net/2018/04/30/visualizing-wildfires-burn-scars-sentinel-hub-eo-browser/>.
- [36] Sentinel-Hub by Sinergise. *Active Fire Detection*. Sentinel Hub custom scripts. URL: https://custom-scripts.sentinel-hub.com/custom-scripts/sentinel-2/active_fire_detection/.
- [37] Jia Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009 IEEE Conference on Computer Vision and Pattern Recognition. ISSN: 1063-6919. June 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.
- [38] Kaiming He et al. *Deep Residual Learning for Image Recognition*. Dec. 10, 2015. DOI: 10.48550/arXiv.1512.03385. arXiv: 1512.03385[cs]. URL: <http://arxiv.org/abs/1512.03385>.
- [39] François Chollet. *Xception: Deep Learning with Depthwise Separable Convolutions*. Apr. 4, 2017. DOI: 10.48550/arXiv.1610.02357. arXiv: 1610.02357[cs]. URL: <http://arxiv.org/abs/1610.02357>.
- [40] Liang-Chieh Chen et al. *Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation*. Aug. 22, 2018. DOI: 10.48550/arXiv.1802.02611. arXiv: 1802.02611[cs]. URL: <http://arxiv.org/abs/1802.02611>.
- [41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. May 18, 2015. DOI: 10.48550/arXiv.1505.04597. arXiv: 1505.04597[cs]. URL: <http://arxiv.org/abs/1505.04597>.
- [42] Oliver Bestel de Lezongard. *obestel|model_structure*. Github. URL: https://github.com/obestel/model_structure.
- [43] Keras Team. *Keras documentation: Adam*. URL: <https://keras.io/api/optimizers/adam/>.
- [44] Mingxing Tan and Quoc V. Le. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. Sept. 11, 2020. DOI: 10.48550/arXiv.1905.11946. arXiv: 1905.11946[cs,stat]. URL: <http://arxiv.org/abs/1905.11946>.
- [45] Andrew G. Howard et al. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. Apr. 16, 2017. DOI: 10.48550/arXiv.1704.04861. arXiv: 1704.04861[cs]. URL: <http://arxiv.org/abs/1704.04861>.
- [46] Mark Sandler et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. Mar. 21, 2019. DOI: 10.48550/arXiv.1801.04381. arXiv: 1801.04381[cs]. URL: <http://arxiv.org/abs/1801.04381>.

- [47] Minsoo Park et al. “Wildfire-Detection Method Using DenseNet and CycleGAN Data Augmentation-Based Remote Camera Imagery”. In: *Remote Sensing* 12.22 (Jan. 2020), p. 3715. ISSN: 2072-4292. DOI: 10.3390/rs12223715. URL: <https://www.mdpi.com/2072-4292/12/22/3715>.
- [48] *Pruning in Keras example / TensorFlow Model Optimization*. TensorFlow. URL: https://www.tensorflow.org/model_optimization/guide/pruning/pruning_with_keras.