Early Forest Fire Detection System
SPROJ Report

Sharjeel Ahmed Shaikh 23100214

Daim Armaghan 23100221

Advisor: Dr. Murtaza Taj
School of Science and Engineering
Lahore University of Management Sciences
Submission Date: 30/04/2023

Acknowledgment and Dedication

We would like to express our heartfelt gratitude to our supervisor, Dr.Murtaza Taj who has supported and helped us throughout the project. His unwavering support and guidance have been instrumental in the successful completion of this project. Dr. Murtaza provided support, feedback, and valuable insight, which helped us to enhance the quality of our work. His constant encouragement and trust in our abilities have motivated us to perform better and exceed expectations.

We are also grateful to our collaborators Umair Yousef and Ahmad Zubair for their hard work, dedication, and efforts in dataset generation through web scraping and synthetic methods. Their contributions have been invaluable in achieving our project goals and objectives. We would also like to acknowledge the efforts and support of Research Assistants Muzammil and Muqsit and appreciate their willingness to share their expertise, knowledge, and experiences.

Finally, we would like to thank Fatima Naweed for her dedicated efforts in the research process, and Asad Soomro for his constant support in the Development Phase. We are very thankful to everyone, who has directly or indirectly supported us throughout this project.

Table of Contents

Abstract

Forest Fires can pose a significant threat to human resources and can disturb the natural ecological balance. They require a sizable amount of resources and manpower to control. In our project, we first worked on researched methods to perform early forest fire detection using deep learning techniques. During our research, we worked on dataset generation, annotation, and augmentation. We then researched state-of-the-art models and analyzed their performance on our dataset. The findings of this research were submitted to IEEE ICASSP 2023. In the second phase of our project, we developed an Early Forest Fire Detection System currently deployed in KPK, Pakistan to detect and localize impending fires using live feed coming from PTZ monitor cameras.

1. Introduction

In our project, we explore an early warning system for detecting forest fires. Due to the lack of sizeable datasets and models tuned for this task, existing methods suffer from missed detection. Hence our first step was compiling and annotating a dataset spanning 3000+ images of fire, smoke, and NULL (no smoke or fire) images. Unlike existing image corpora that contain images of widespread fire, our dataset consists of multiple instances of smoke plumes and fire indicating fire initiation. We then explored several state-of-the-art models like YOLO and Deformable Detr and submitted a paper noting the model's performance and shortcomings. We chose Yolo Tiny since it was easy to run on the existing Jetson-Nano systems and had comparable performance. Once the dataset and model were finalized, we began developing the system, including its Pipelines, Frontend, Backend, APIs, and Database.

1.1. Roadmap

```
                      |--> Dataset Generation  —-> Data Annotation and Augmentation -> Dataset Finalization
                      |                                      |
Research Problem —                          Model Comparison —-> Research Paper  —>Dashboard Development
                      |                                      |
                      |--> Model Experimentation —> Model Selections
```

2. Related Work

Algorithms for the automatic identification of fire or smoke in the images can be categorized into image classification [9], segmentation [6], and localization [7, 10, 11]. SmokiFi-CNN [9] is an image classification-based technique, that tends to classify images into either normal, smoke, fire, or fire with smoke classes. During pre-processing, segmentation between the background and foreground is performed, along with white balancing to further highlight areas with smoke which help in reducing the number of missed detections. The flames of wildfires can almost be invisible from long distances. However, the rising smoke plumes generated by the fire can usually be seen in the viewing field of the camera. Therefore, it would be more practical to detect the smoke plumes instead of detecting flames for the purpose of long-distance wildfire detection.

A wildfire smoke detection method based on local extremal region segmentation [6] was proposed to localize smoke pixels. This method used a linear time Maximally Stable Extremal Regions (MSERs)[12] detection algorithm to make the initial segmentation less dependent on motion and color information.

Most of the localization algorithms perform both classifications as well as extraction of bounding boxes. These include object detectors such as variants of Region CNN [13], YOLO [14], and detection transformers [15]. A comparison of these methods for the application of forest fire detection has been performed which reveals that YOLOv3 achieved better accuracy at low compute cost [7]. Many of the recent methods are thus based on variants of YOLO. Since a practical forest fire surveillance system requires continuous monitoring of the site via a live camera feed, thus real-time performance is usually considered an important aspect [10]. This work focuses on both outdoor and indoor detection. Simultaneously, it reduces the need for storage and computing via YOLOv2, which has a less complex model and a reduced number of weights. This allows its deployment on smaller devices while providing results within 2 seconds, which is almost 25 times faster than the R-CNN [13]. Similarly, a variant of YOLOv5 is used for the localization of fire in both ground and aerial imagery [11]. This method attempts to reduce the error rate by improving the anchor box clustering via a variant of the K-means algorithm. It also prunes the network head of YOLOv5 to further improve the detection speed. A more comprehensive survey of fire detection can be found in [8].

A major shortcoming of existing methods is, although they detect widespread wildfires in the vicinity of the camera, they are inapplicable for early detection of forest fires via the identification of smoke plumes. Major reasons include a lack of sizeable datasets of smoke from the initial stages of fire, inherent ambiguities between smoke, cloud, fog, and smog as well as changes in sky appearances at sunset and sunrise (see Fig. 1). We first propose a dataset for early identification of forest fires through visual analysis. Unlike, existing image corpora that contain images of widespread fire, our dataset consists of multiple instances of smoke plumes and fire that indicate the initiation of fire. We obtained this dataset synthetically via the game simulator Red Dead Redemption 2 [16]. We also combined our dataset with published images to obtain a more comprehensive set. Finally, we compared image classification and localization methods on the proposed dataset.

3. Dataset

Although there are many cameras installed around the world to watch forest areas, since fire is a rare event, most of the existing datasets from real forest watch towers contain the vast majority of normal images (i.e. images without any fire or smoke). While the other datasets contain images from intense fire, or fire and smoke augmented in the scene.

Thus a major challenge is to find a sizeable, well-balanced dataset to train our models. We addressed this problem by generating synthetic data from 3 main regions, Cumberland Forest, Grizzlies East, and Grizzlies West which are regions in the computer game titled Red Dead Redemption 2. The game features meticulous, true-to-life graphics. These 3 regions were selected due to there being several mountains and extensive vegetation and trees. Mods were used to place smoke within the trees to simulate a fire in its initial stages, as well as to place a large number of campfires to simulate a fire that had been burning for some time. Screen recording of these scenarios from various angles and focal lengths was done. Frames were extracted from these recordings and a synthetic dataset was generated for smoke and fire classes. We combined this with other resources to form our final dataset.

	Train		Valid		Test	
Class	Im.	Inst.	Im.	Inst.	Im.	Inst.
Smoke	1257	1448	403	467	378	432
Fire	1592	2657	532	909	497	808
Normal	2656	2656	907	907	937	937

Train-Validation-Test split in dataset along each class (Im.: Images, Inst.: Instances).

(a) Smoke Augmented (b) Fire (c) Fire (d) Smoke Synthetic

(e) Smoke (f) Fire (g) Fire Synthetic (h) Smoke

Sample images from dataset including samples of both synthetic, augmented and real images.

	Ann.	Imgs	Inst.	R/Sy/A	Views	Type	Used
Mivia (Fire)[19]	Class	Videos	106	R	8	S	Y
Mivia (Smoke)[20]	Class	Videos	300	R	1	S	Y
Kaggle 1[21]	Class	1900	1900	R	Mult	S	Y
Kaggle 2[22]	Class	15734	15734	Aug	Mult	S	Y
Mendeley [23]	Class	1900	1900	R	Mult	S	N
Images.cv [24]	Class	948	948	R	Mult	S	Y
HPWREN [25]	None	Videos	-	R	50+	S	Y
RoboFlow [26]	BB	737	737	Aug	Few	FE	N
NIST[27]	Class	Videos	-	R	Mult	360	N
HPWREN[28]	BB	2192	2192	R	Mult	FE	Y
Read Dead 2[16]	Class	1531	1531	Synth	Mult	S	Y

Summary of existing and generated datasets for fire and smoke detection (Ann.: Annotations, Class: Classification, BB: Bounding Box, Imgs.: Images, R: Real, Sy: Synthetic: A: Augmented, S: Street view, FE: Fish Eye, Mult: Multiple).

4. Model

We selected four state-of-the-art object detectors in this work. We for the first time introduced transformer-based networks such as Detector Transformer (DETR) [15] and its variant Deformable DETR [29] and compared their performance with variants of YOLO (YOLOv7 [17] and YOLOv7-tiny). YOLOv7: YOLO-v7 [17] belongs to a series of state-of-the-art models, YOLO (You Only Look Once). As opposed to other object detectors which focus on specified regions of an image, YOLO views the complete image at once. YOLO processes images faster than most object detectors and is therefore commonly used for real-time detections. Fine-tuning was performed on pre-trained YOLO-v7. We trained the model weights for 50 epochs with a batch size of 8 and a learning rate of 0.01. It utilizes an SGD optimizer with Nestrov momentum for backpropagation and SiLU as its activation function. YOLOv7-tiny: It is a version of YOLO-v7 optimized for edge GPU computation. Compared to the YOLO-v7 standard model, it only uses 16% of the number of parameters (See Table 4). Consequently, it leads to faster inference and can run on lightweight devices that are much more suitable for real-time deployment.

Fine-tuning was performed on pre-trained YOLO-v7-tiny model weights. We trained the model weights for 50 epochs with a batch size of 8 and a learning rate of 0.01. It utilizes an SGD optimizer with Nestrov momentum for backpropagation and ReLU as its activation function. DETR: DETR (Detection Transformer) [15] views object detection as a direct set prediction problem, incorporating a set-based global loss that forces unique predictions via bi-partite matching and a transformer encoder-decoder architecture. We used the DETR pre-trained model and fine-tuned it on our dataset without making any changes to their hyperparameters. Deformable DETR: Deformable DETR[29] is based on DETR and aims to speed up the model's convergence and increase feature spatial resolution caused by the limitation of Transformer attention modules in processing image feature maps. Deformable DETR achieves better performance than DETR with fewer training epochs. We used the Deformable DETR (single scale) pre-trained model and fine-tuned it on our dataset without making any changes to their hyper-parameters.

—	**YOLOv7**	**YOLOv7 -tiny**	**Def. DETR**	**DETR**
GPU	Tesla T4	Tesla T4	Tesla T4	Tesla T4
LR	0.001	0.001	0.002	0.001
Epochs	50	50	37	65
Batch	8	8	4	4
CT	0.1	0.1	0.5	0.5

Summary of training and testing conditions for the four models. (LR: Learning rate, Batch: the batch size of each iteration, CT: Confidence Threshold).

—	YOLOv7	YOLOv7-tiny	Def. DETR	DETR
Accuracy	0.65	0.88	0.81	0.62
Precision	0.72	0.89	0.94	1.0
Recall	0.67	0.89	0.65	0.17
F1	0.64	0.88	0.74	0.29
mAP@ 0.5	0.40	0.41	0.46	0.36
mAP@ 0.5:0.95	0.17	0.17	0.21	0.13
Params (M)	37.20	6.02	40.80	41.28
Inf. time	0.15	0.2	2.1	2.4

Comparison of different performance metrics of the four different models. (Row 1-4) Classification results (Inf. time: Inference time for one image in milliseconds(ms), M: 1 million parameters).

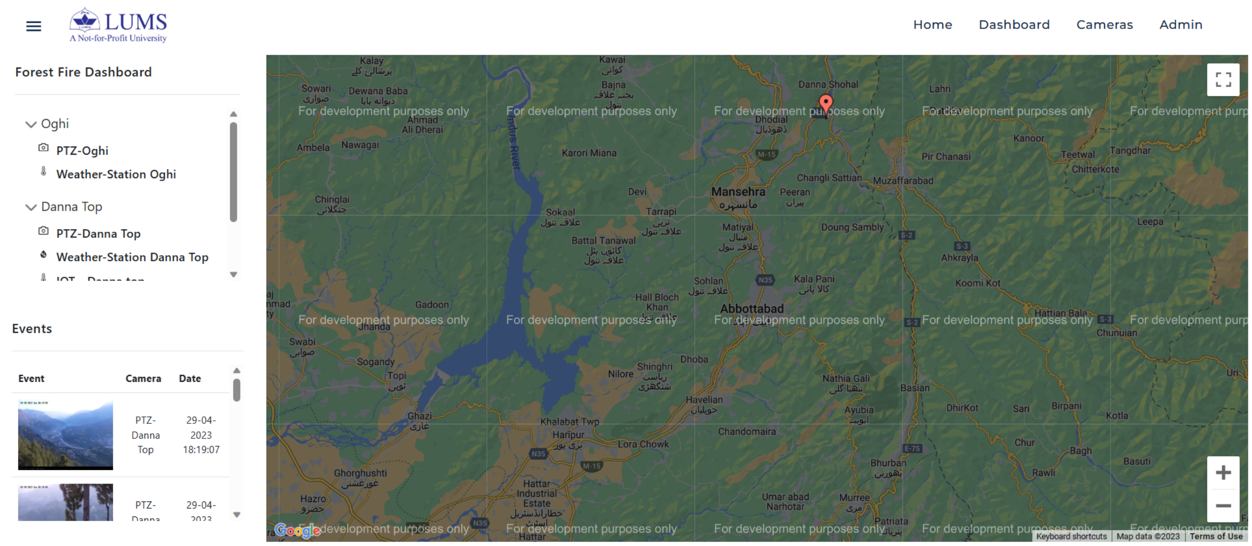
5. Dashboard

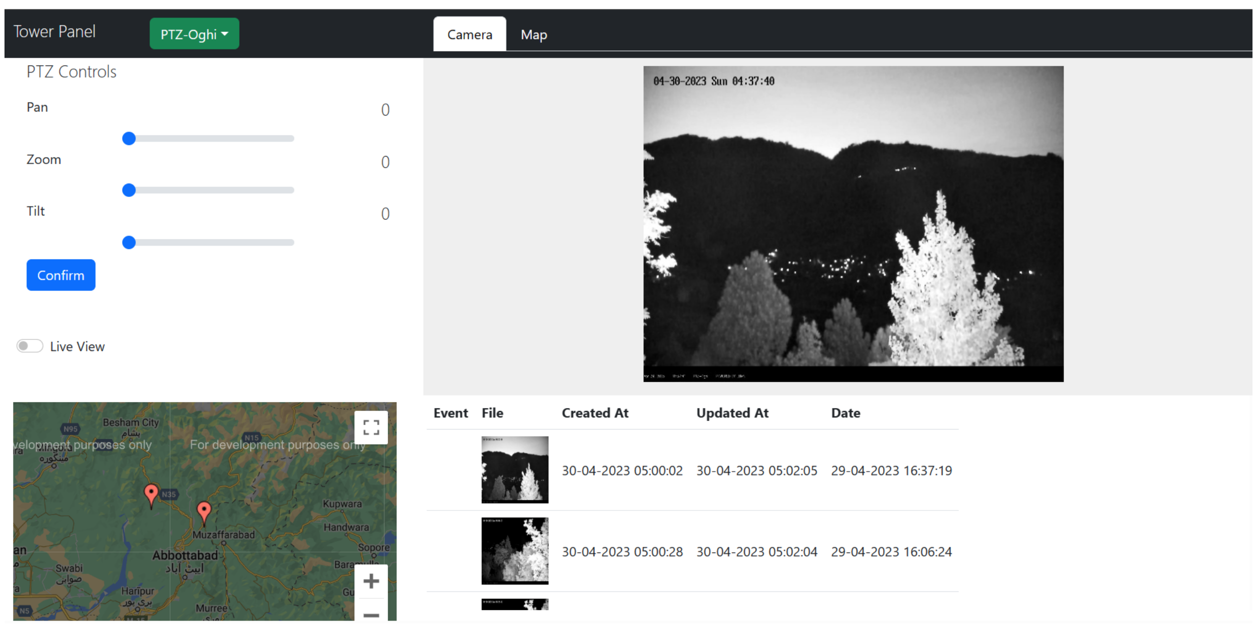
During the Senior Spring semester, our team started the development of the dashboard system using React.js, Django rest framework, and MySQL. Our initial objective was to design various components, including Map, Arc, and LiveView. With guidance from Dr. Murtaza and his Research Assistants Muzaamil and Muqsit, we worked collaboratively to construct a comprehensive dashboard for the system. The following figures show the landing screen, camera, and sensor page.

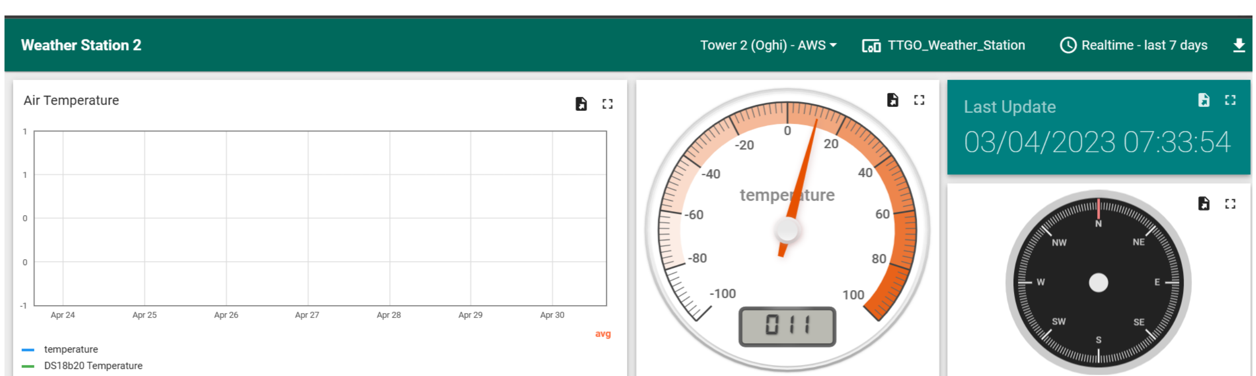
5.1. Main Features of the Dashboard

- Map displayed on landing screen along with markers of active cameras.
- Ability to zoom into individual markers and look at their coverage and sensors nearby.
- Sensor page displaying live sensor reading via Thingsboard.
- Camera page displaying a live feed of the camera, events reported for this particular area, controls for the camera, etc.
- PTZ Controls sending live commands to Pan, tilt, and zoom PTZ cameras.
- Secure system with layered authentication.
- An interactive Admin Dashboard to change, update, add, or remove devices.

- Real-Time streaming of PTZ Cameras through YouTube endpoint.







6. Conclusions & Future Work

6.1. Research Conclusion

In this work we first filled the gap in the existing dataset by generating fire and smoke events synthetically via a game engine. We then combined our data with existing datasets and used it for performance evaluation of state-of-the-art algorithms such as YOLOv7, YOLOv7-tiny, and variants of Detection Transformers. We for the first time performed forest fire prediction via transformers for which we introduced DETR [15] and Deformable DETR [29] for solving the problem of generating early warnings for wildfires. Low inference time and low missed detections are some of the important constraints associated with early warning systems. Since there is typically a human in the loop thus some amount of false positives can be accepted. Considering these constraints we found YOLOv7-tiny to be the most suited algorithm for the problem. Deformable DETR, on the other hand, gives fewer false positives but at the cost of miss-detection. It also has a higher inference time and is computationally expensive due to its model complexity. YOLOv7, on the other hand, gives comparatively higher false positives, however, it has the lowest inference time and the highest accuracy out of the all models. In addition, it has a low memory footprint and can be executed on embedded devices such as Jetson Nano. It is also computationally light due to the low number of parameters which helps run it 24/7. In summary, out of the 4 models, YOLOv7-tiny is the best, most suited, with Deformable-DeTr as the runner-up.

6.2. Development Conclusion

In conclusion, we developed a dashboard that can perform early forest fire detection employing knowledge from our research. Dashboard was built through rapid development and main functionalities were completed within a month. This dashboard has wide applications and can accommodate new features easily. World Wildlife Fund is currently operating this system, in a dedicated monitoring station. We hope that this system will make valuable contribution in the worldwide combat against Forest Fires.

6.3. Future Work

Add more features to the system such as

- Path suggestions for firepeople based on fire/smoke trajectory

- Whatsapp Integration for community interaction where people can report fire, smoke events.
- A community moderation system, to filter community-based events.
- Better performance especially on sunset/sunrise and clouds
- Haze detection and removal
- Finetune model once we get real data

References

[1]. “State of the climate: Monthly global climate report for annual 2021,” Tech. Rep., NOAA National Centers for Environmental Information, January 2022.

[2]. Giorgio Vacchiano and Davide Ascoli, “An implementation of the rothermel fire spread model in the r programming language,” Fire Technology, vol. 51, 01 2014.

[3]. Suwei Yang, Massimo Lupascu, and Kuldeep S. Meel, “Predicting forest fire using remote sensing data and machine learning,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 17, pp. 14983–14990, May 2021.

[4]. Pasquale Foggia, Alessia Saggese, and Mario Vento, “Real-time fire detection for video-surveillance applications using a combination of experts based on color, shape, and motion,” IEEE Transaction on circuits and systems for video technology, vol. 25, no. 9, pp. 1545–1556, 2015.

[5]. Yingshu Peng and Yi Wang, “Real-time forest smoke detection using handdesigned features and deep learning,” Computers and Electronics in Agriculture, vol. 167, Dec 2019.

[6]. Zhiqiang Zhou, Yongsheng Shi, Zhifeng Gao, and Sun Li, “Wildfire smoke detection based on local extremal region segmentation and surveillance,” Fire Safety Journal, vol. 85, pp. 50–58, 2016.

[7]. Pu Li and Wangda Zhao, “Image fire detection algorithms based on convolutional neural networks,” Case Studies in Thermal Engineering, vol. 19, pp. 100625, 2020.

[8]. Suha Berk Kukuk and Zeynep Hilal KiLiMci, “Comprehensive Analysis of Forest ¨ Fire Detection using Deep Learning Models and Conventional Machine Learning Algorithms,” International Journal of Computational and Experimental Science and Engineering, July 2021.

[9]. A Robert Singh, Suganya Athisayamani, S Sankara Narayanan, and S Dhanasekaran, “Fire detection by parallel classification of fire and smoke using convolutional neural network,” in Computational Vision and Bio-Inspired Computing, pp. 95–105. Springer, 2021.

[10]. Sergio Saponara, Abdussalam Elhanashi, and Alessio Gagliardi, “Real-time video fire/smoke detection based on cnn in antifire surveillance systems,” Journal of Real-Time Image Processing, vol. 18, no. 3, pp. 889–900, 2021.

[11]. Qing An, Xijiang Chen, Junqian Zhang, Ruizhe Shi, Yuanjun Yang, and Wei Huang, “A robust fire detection model via convolution neural networks for intelligent robot vision sensing,” Sensors, vol. 22, no. 8, 2022.

[12]. Jiri Matas, Ondrej Chum, Martin Urban, and Tomas Pajdla, “Robust wide-baseline ´ stereo from maximally stable extremal regions,” Image and vision computing, vol. 22, no. 10, pp. 761–767, 2004.

[13]. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” Advances in neural information processing systems, vol. 28, 2015.

[14]. Joseph Redmon and Ali Farhadi, “Yolo9000: Better, faster, stronger,” in IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[15]. Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko, “End-to-end object detection with transformers,” in European Conference on Computer Vision. Springer, 2020, pp. 213–229.

[16]. “Red dead redemption 2 - cumberland forest,” https://www.rockstargames.com/games/reddeadredemption2, Accessed: 2022-10- 26.

[17]. Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao, “Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” 2022, arXiv.

[18]. “Large detailed map of red dead redemption 2 world,” https://www.mapsland.com/games/large-detailed-map-of-red-dead-redemption-2- world.

[19]. Mivia, “Fire detection dataset,” https://mivia.unisa.it/datasets/video-analysisdatasets/fire-detection-dataset/).

[20]. Mivia, “Smoke detection dataset,” https://mivia.unisa.it/datasets/video-analysisdatasets/smoke-detection-dataset/.

[21]. Alik05, “Forest fire dataset,” https://www.kaggle.com/datasets/alik05/forest-firedataset, Apr 2022.

[22]. Kutay Kutlu, “Forest fire,” https://www.kaggle.com/datasets/kutaykutlu/forest-fire, Mar 2021.

[23]. Ali Khan and Bilal Hassan, “Dataset for forest fire detection,” https://data.mendeley.com/datasets/gjmr63rz2r/1, Aug 2020.

[24]. “Image datasets for computer vision and machine learning,” https://images.cv/search-labeled-image-dataset.

[25]. “Cameras from various hpwren related sites,” http://hpwren.ucsd.edu/cameras/.

[26]. AI For Mankind, “Wildfire smoke object detection dataset - raw,” https://public.roboflow.com/object-detection/wildfire-smoke/1/download, Aug 2022.

[27]. “Fire,” https://www.nist.gov/fire, Apr 2021.

[28]. Aiformankind, “Aiformankind/wildfire-smoke-dataset: Open wildfire smoke datasets,” https://github.com/aiformankind/wildfire-smoke-dataset.

[29]. Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai, “Deformable DETR: deformable transformers for end-to-end object detection,” in 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. 2021, OpenReview.net.

[30]. Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and C. Lawrence Zitnick, “Microsoft coco: Common ob- ´ jects in context,” in Computer Vision – ECCV 2014, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Eds., Cham, 2014, pp. 740–755, Springer International Publishing