Evaluating Deep Neural Network-based Fire Detection for Natural Disaster Management

Matthaios D. Tzimas
Aristotle University of Thessaloniki
Thessaloniki, Greece
mtzima@csd.auth.gr

Christos Papaioannidis
Aristotle University of Thessaloniki
Thessaloniki, Greece
cpapaionn@csd.auth.gr

Vasileios Mygdalis
Aristotle University of Thessaloniki
Thessaloniki, Greece
mygdalisv@csd.auth.gr

Ioannis Pitas
Aristotle University of Thessaloniki
Thessaloniki, Greece
pitas@csd.auth.gr

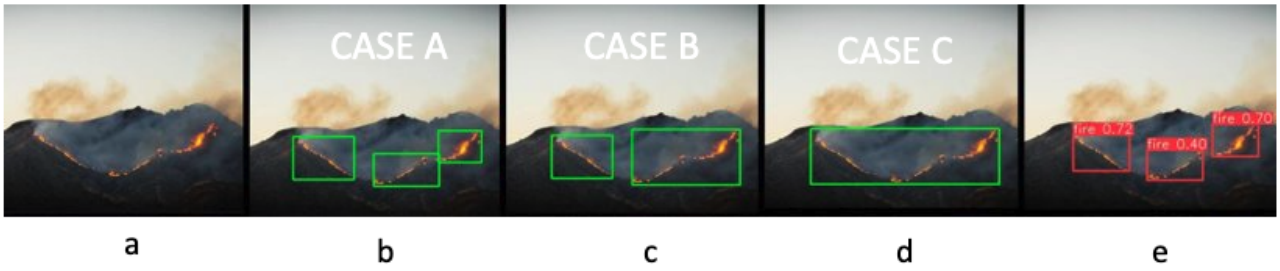

Figure 1: a) Raw image [3] , b, c, d) Different annotation strategies e) Fire prediction

ABSTRACT

Recently, climate change has led to more frequent extreme weather events, introducing new challenges for Natural Disaster Management (NDM) organizations. This fact makes the employment of modern technological tools such as deep learning and Deep Neural Networks (DNNs) a necessity, as they can assist such organizations manage these extreme events more effectively. One of the most important tasks in which DNN-based algorithms could be invaluable is fire detection, where the goal is to identify and localize fires by predicting bounding boxes, typically using as input video frames. Selecting the most suitable DNN model to be utilized in such algorithms is vital to NDM, since inaccurate predictions can adversely affect disaster response, thus highlighting the importance of a reliable fire detection evaluation metric. In this work, we argue that the mean Average Precision (mAP) metric that is commonly used to evaluate typical object detection algorithms can not be trusted for the fire detection task, due to its high dependence on the employed data annotation strategy. This means that the mAP score of a fire detection algorithm may be low even when it predicts fire bounding boxes that accurately enclose the depicted fires. In this direction, a new evaluation metric for fire detection is proposed, denoted as Image-level mean Average Precision (ImAP), which reduces the dependence on the bounding box annotation strategy by rewarding/penalizing bounding box predictions on image level, rather than on bounding box level. Experiments using different Convolutional Neural Network (CNN)-based and Transformer-based object detection algorithms have shown that the proposed ImAP metric reveals the true fire detection capabilities of the tested algorithms more effectively.

UCC '23, December 4–7, 2023, Taormina (Messina), Italy

ACM ISBN 979-8-4007-0234-1/23/12…$15.00
https://doi.org/10.1145/3603166.3632082

CCS CONCEPTS

• **Computing methodologies** → **Object detection**.

KEYWORDS

Fire Detection, Object Detection, Natural Disaster Management, Deep Learning

ACM Reference Format:
Matthaios D. Tzimas, Christos Papaioannidis, Vasileios Mygdalis, and Ioannis Pitas. 2023. Evaluating Deep Neural Network-based Fire Detection for Natural Disaster Management. In *2023 IEEE/ACM 16th International Conference on Utility and Cloud Computing (UCC '23), December 4–7, 2023, Taormina (Messina), Italy*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3603166.3632082

1 INTRODUCTION

Climate change has resulted in a notable upsurge in the frequency and severity of natural disasters, particularly wildfires and floods, posing significant threats to both ecosystems and human lives. Since

2000 the recorded average number of fires per year was 70,600 [11]. These climatic events are projected to persist in the future, requiring significant improvements in the field of Natural Disaster Management (NDM). One pivotal facet of NDM pertains to emergency response, which focuses on human lives safety, immediate relief provision, and the restoration of stability in disaster-stricken areas. Within the ambit of emergency response, the deployment of advanced fire detection mechanisms emerges as a critical task. This technology not only detects fires at their early stages, preventing uncontrolled conflagrations, but also provides useful outputs that can be used to optimize the allocation and utilization of available firefighting resources.

Traditional image processing-based approaches for fire detection primarily relied on processing video frames using wavelet transformations [28] or combining color and motion detection to identify fire pixels [26]. More recent fire detection methodologies [9, 27] typically rely upon Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs) to identify and localize fires within images or video frames. DNNs and CNNs have the capacity to undergo training for fire detection across various scales and in a diverse range of environmental conditions, offering a more effective solution when compared to conventional sensors. However, they also face some challenges, primarily due to their reliance on both the quality and quantity of available data. In order to achieve good generalization ability, DNNs/CNNs typically require annotated datasets that encompass a wide array of scenes and numerous fire-related scenarios. Additionally, the development of larger, high-accuracy models usually entails an increased computational demand, posing a greater challenge for real-time fire detection.

Nowadays, the prevailing approaches for fire detection [2, 20, 21] leverage advanced object detection algorithms built on CNNs [13, 15] , which are great at handling spatial information. Conversely, some methods opt for Transformer-based approaches [1, 31], which, despite being slower, utilize architectures capable of capturing global context of an image.

All these approaches utilize the mean Average Precision (mAP) metric to evaluate their performance on detecting objects/fires, which awards/penalizes object/fire bounding box predictions based on their alignment with the corresponding ground-truth boxes. In most objects such as cars, numerous "children" objects that belong to different classes (e.g., car wheel, car window) collectively contribute to creating the "parent" object (car). Consequently, each "parent" object corresponds to exactly one ground-truth bounding box. However, in the case of objects like fire, "children" objects belong to the same class as the "parent" object (fire), which creates uncertainty regarding whether each "child" is, in fact, a "parent" object. This uncommon property of fire entities introduces uncertainty for both human annotators and DNNs/CNNs concerning the number of bounding boxes required to represent a fire object accurately. This is illustrated in Fig.(1), where despite the fact that all annotation styles are deemed correct, it is probable that only one of them will align with the predicted bounding boxes (case A, sub-figures) . In cases like the ones depicted in sub-figures c) and d), the mAP scores do not represent the actual object/fire detection performance of the detectors. To tackle this, we propose a new evaluation metric for fire detection, namely Image-level mean Average Precision (ImAP). Instead of looking at each predicted bounding box separately, ImAP evaluates the fire detection models on their ability to predict fire object bounding boxes in the whole image. Experiments using different object detectors show that the proposed metric is more suitable for evaluating these models in the fire detection task.

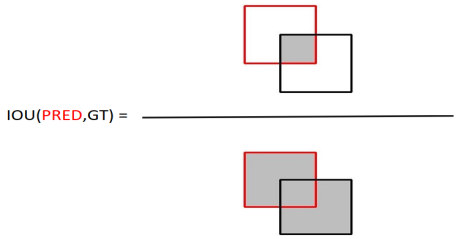

Figure 2: Intersection over Union (IoU)

2 RELATED WORK

Object detection involves identifying and localizing numerous distinct objects within an image. Training DNNs to identify specific objects, typically requires a manually annotated dataset, where each object of interest is outlined by its corresponding ground-truth bounding box and labeled with its associated class. During testing time, object detections algorithms typically output the predicted bounding box coordinates (in a pre-defined format), the object class and the corresponding prediction confidence score. Therefore, the correctness of a bounding box prediction ($pred_{BB}$) with respect to its corresponding ground-truth (gt_{BB}) is measured using the Intersection over Union (IoU) metric. This metric computes their overlapping area divided by their union area as depicted in Fig. 2 and it is defined as:

$$IoU(pred_{BB}, gt_{BB}) = \frac{pred_{BB} \cap gt_{BB}}{pred_{BB} \cup gt_{BB}}. \quad (1)$$

Based on Eq. (1) and the ground-truth and predicted classes gt_{cls}, $pred_{cls}$ respectively, True Positives (TP), False Positives (FP) and False Negatives (FN) are defined as follows:

- True Positive (TP): A prediction for which the IoU of the predicted bounding box with the corresponding ground-truth is higher than a threshold τ and both of them belong to the same class, $IoU(pred_{BB}, gt_{BB}) > \tau$ AND $pred_{cls} = gt_{cls}$.
- False Positive (FP): A prediction for which the IoU of the predicted bounding box with the corresponding ground-truth is lower than a threshold τ, or the predicted box and the ground-truth do not belong to the same class, $IoU(pred_{BB}, gt_{BB}) < \tau$ OR $pred_{cls} \neq gt_{cls}$.
- False Negatives (FN): A ground-truth bounding box which the DNN fails to detect.

It is important to highlight the fact that if there is a bunch of predictions that match the conditions to be counted as TP for a particular ground-truth bounding box, we mark as TP only the one with the highest confidence score, and we classify the remaining as FP. Then, in order to calculate the average precision metric, the

precision and recall metrics are utilized.

$$Precision = \frac{TP}{TP + FP}, \tag{2}$$

$$Recall = \frac{TP}{TP + FN}. \tag{3}$$

The precision metric Eq. 2 signifies the percentage of accurate predictions made by the model. A higher precision value implies a greater likelihood that a given prediction is correct. On the other hand, the recall metric Eq. 3 calculates the percentage of ground-truth bounding boxes that the model successfully identifies. A low recall value indicates that the model lacks the ability to detect the objects of interest in an image.

Both of these metrics provide information about the weaknesses and strengths of detection algorithms. However, selecting the best among them becomes a challenging task when we lack a single scalar metric. Additionally, the evaluation results are not influenced by the confidence scores of the predicted bounding boxes. These disadvantages have been addressed by the Average Precision (AP) metric, which calculates the area under the precision-recall curve (PR-curve) depicted in Fig. 3. In Fig. 3, the X-axis represents the recall rate, while the Y-axis denotes the corresponding precision values. In order to generate the PR-curve, all predictions must be arranged in descending order based on their confidence scores. Subsequently, each TP prediction is given a value DT=1, while each FP one is given a value of DT=0. The X-axis and Y-axis values of the PR-curve for the n-th prediction are then defined as:

$$P_n = \frac{\sum_{i=1}^{n} DT_i}{n}, \tag{4}$$

$$R_n = \frac{\sum_{i=1}^{n} DT_i}{N_{GT}}, \tag{5}$$

where N_{gt} is the total number of ground-truth bounding boxes. However, drawing the PR-curve based on linear interpolation of the points produced by Eq.(4, 5), is causing many "zigzags" on the curve as shown in Fig. 3 , which may lead to inaccurate evaluation results [17]. This phenomenon arises due to the fact that when consecutive FP predictions are followed by a TP, the precision value of the TP is higher than the minimum of the FPs (consecutive FPs have the same recall and different precision values). All AP variations utilize the approach of selecting the maximum precision from the right, with the aim to eliminate the error that "zigzag" curve produces. Therefore, the precision for a specific recall value r, is the highest precision achieved among all recalls r' where $r' \geq r$ [6]. Based on the updated piece-wise constant curve Fig. (3), object detection challenges [6, 7, 14] utilize either N-point [19] or all-point [30] interpolation for the AP metric computation.

The N-point interpolation creates a set of N equal spaced recall values $R'_N = \{\frac{1}{N}, \frac{2}{N}, ..., \frac{N-1}{N}, \frac{N}{N}\}$ in order to compute the average of their corresponding precision values [17].

$$AP_N = \frac{1}{N} \sum_{r \in R'_N} P_{max}(r), \tag{6}$$

where $P_{max}(r) = \max_{k:R_k \geq r} P_k$.

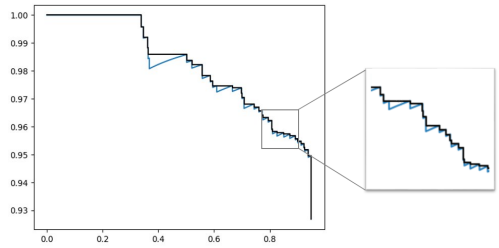

Figure 3: Precision-Recall Curve

All-point interpolation calculates the AP across all recall values generated by Eq. 5. While this approach offers enhanced accuracy relative to N-point interpolation techniques, it may present computational inefficiencies when applied to expansive datasets [29].

$$AP_{all} = \sum_{n=0}^{N_{predictions}} (R_{n+1} - R_n) \cdot P_{max}(R_{n+1}), \tag{7}$$

where $P_{max}(R_n) = \max_{k:R_k \geq R_n} P_k$, $R_0 = 0$ and $N_{predictions}$ the total number of predictions made by the model.

In both approaches, the AP metric is computed for each class separately. So the mean Average Prevision (mAP) metric evaluates the performance of the model across all classes: $mAP = \frac{1}{N_{cls}} \sum_{i=1}^{N_{cls}} AP_i$.

In the Pascal VOC 2007 challenge [8], the authors initially proposed the 11-point interpolation method using an IoU threshold $\tau = 0.5$. This evaluation system remained consistent for the competition until 2010 when there was a transition from 11-point interpolation to all-point interpolation, still utilizing $\tau = 0.5$. MS COCO challenge [14] introduced the Average Precision (AP) with 101-point interpolation, coupled with evaluations across 10 different IoU thresholds $\{0.5, 0.55, 0.6, ..., 0.9, 0.95\}$, calculated using the following equations:

$$mAP@\tau = \frac{1}{N_{cls}} \sum_{i=1}^{N_{cls}} AP_i@\tau, \tag{8}$$

$$mAP@[0.5 : 0.05 : 0.95] = \frac{1}{11} \sum_{\tau \in [0.5:0.05:0.95]} mAP@\tau, \tag{9}$$

where $AP_i@\tau$ is the average precision of the class i at IoU threshold τ.

3 IMAGE-LEVEL MEAN AVERAGE PRECISION

In the field of object detection, each image is linked to a set of fire predictions (Preds) and a set of ground truths (Gts). Each element of Gts comprises Bounding Box coordinates (BB) and their corresponding Class (CLS) labels, while Preds also include the Confidence Score variable (CS). Unlike most objects, fire objects can be represented in various BB combinations, leading to discrepancies between $Preds^{BB}$ and Gts^{BB} (cases b, c, of Fig. 1). Preventing these scenarios necessitates employing an IoU between the sets $Preds$ and Gts, to evaluate the overall fire prediction performance within an image. Image-level Intersection over Union (ImIoU) is a modification of the IoU, which measures how well the union of $Preds$ fit

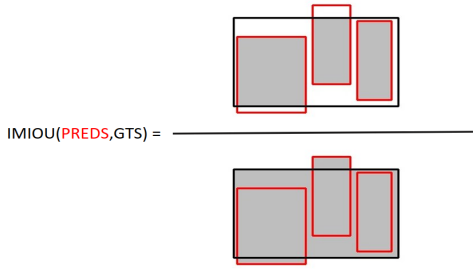

Figure 4: A visualization of the Image-level Intersection over Union

the Gts union Fig. 5.

$$ImIoU(Preds, Gts) = \frac{(\cup_{i=1}^{N_{Preds}} Preds_i^{BB}) \cap (\cup_{i=1}^{N_{Gts}} Gts_i^{BB})}{(\cup_{i=1}^{N_{Preds}} Preds_i^{BB}) \cup (\cup_{i=1}^{N_{Gts}} Gts_i^{BB})}. \quad (10)$$

where N_{Preds}, N_{Gts} are the lengths of the prediction and ground-truth sets, respectively. Therefore, we redefine True positives, False Positives, False Negatives and True Negatives as follows:

- True Positive (TP): Images for which $N_{Preds} > 0$ AND ImIoU(Preds, Gts) $> \tau$,
- False Positive (FP): Images for which $N_{PREDS} > 0$ AND ImIoU(Preds, Gts) $< \tau$
- False Negative (FN): Images for which $N_{Preds} = 0$ AND $N_{Gts} > 0$
- True Negative (TN): Images for which $N_{Preds} = 0$ AND $N_{Gts} = 0$

Each TP prediction is given a value DT=1, while each FP one is given a value of DT=0. Then the image-level detection results must be arranged based on the mean Confidence Score of their predictions (mCS Eq. 11), in order the Eq.(4,5) to compute the new Precision Recall points. The number of these points is equal to the images with $N_{Preds} > 0$ and lower than the number of BB predictions. This fact led us choose All-point interpolation as the method to compute the Image-level Average precision (IAP). Like MS-COCO challenge we provide the new Image-level mean Average Precision (ImAP) metric which evaluates image-level predictions across different ImIoU thresholds $t \epsilon [0.5 : 0.05 : 0.95]$.

$$mCS = \frac{1}{N_{Preds}} \sum_{i=1}^{N_{Preds}} Preds_i^{CS} \quad (11)$$

$$ImAP@\tau = \frac{1}{N_{cls}} \sum_{i=1}^{N_{cls}} IAP_i@\tau \quad (12)$$

$$ImAP@[0.5 : 0.05 : 0.95] = \frac{1}{11} \sum_{\tau \in [0.5:0.05:0.95]} ImAP@\tau \quad (13)$$

When two small $pred^{BB}$ correspond to a bigger gt^{BB}, the area of the union that is not in the intersection one, is causing a drop on ImIoU value. Comparing theoretically $ImAP@[0.5 : 0.05 : 0.95]$ with $ImAP@[0.5]$, the last one can handle these 'error areas' due to low ImIoU threshold τ while the metric with large thresholds incorrectly evaluate image-level predictions as FP. So $ImAP@[0.5]$ fulfil the purposes of image-level evaluation while $ImAP@[0.5 : 0.05 : 0.95]$ tends to behave like a combination of the box-to-box mAP with ImAP.

DNN-based object detection models often predict more objects than what actually exists. High-confidence predictions typically align closely with the gt^{BB}. In contrast, low-confidence predictions may not correspond to any target and often exhibit low or zero IOU with the corresponding gt. These incorrect predictions expand the "error area", resulting a large number of false positives due to low ImIoU. To optimize the evaluation performance of our metric, we need to identify the CS threshold that maximizes the ImAP. By filtering predictions based on this threshold, we retain only the essential predictions that best match the union of ground-truths.

For natural disaster management, visualizing and analyzing fire detections results is crucial. Setting the confidence score threshold to zero often results in poor visualizations due to a high number of false positives. Moreover, manually selecting the threshold can be imprecise. By setting the threshold to the value that maximizes ImAP, we can filter out false positives while retaining the predictions that best capture fire within an image. mAP metric can not detect this threshold due to the fact that removing detections is decreasing its value. When the AP algorithm evaluates the low confidence score predictions in order to draw the furthest right points of the PR-curve, FPs detections do not affect the metric as much as possible TPs that will be removed after the filtering. The reason for this based on Eq.(4,5), is that a low confident TP extent the limits of RP-curve in the same amount as a high confident TP (denominator of Eq. 5 has a constant value equal to the number of the ground-truths). In contrast to recall, the changes in precision are small regardless of the prediction result, as the denominator of Eq. 4 is equal to n.

4 EXPERIMENTS

4.1 Dataset

In order to train deep neural networks across a wide range of fire scenarios and environments like cities, forests, and aerial images, we combine three datasets: dfire [3], jhope [12], and crossican [22, 23]. Crossican, initially a segmentation dataset for forest fires, undergoes a transformation into a detection dataset using image processing. Jhope, sourced from roboflow, contains diverse fire types. Dfire, a fire-smoke dataset, has smoke boxes removed, retaining all images for training without ground-truth. This approach aids deep neural networks in distinguishing fires with like-fire objects. We created a test set, emphasizing scenarios of forest fires and wildfires, serving the purpose of natural disaster management.

4.2 DNN-based Object Detection

The choice of DNN models wasn't solely guided by the latest real-time object detectors. Models process image information in diverse ways, resulting in differences between their detections. CNN-based models [10, 18, 25] have been dominant in computer vision, due to their ability extracting rich spatial information. Yolo-v8 [13] is a powerful CNN based real-time detector which have the best accuracy compared to other architectures within the YOLO [15, 25] family. It is extracting 3 feature maps of different scales produced

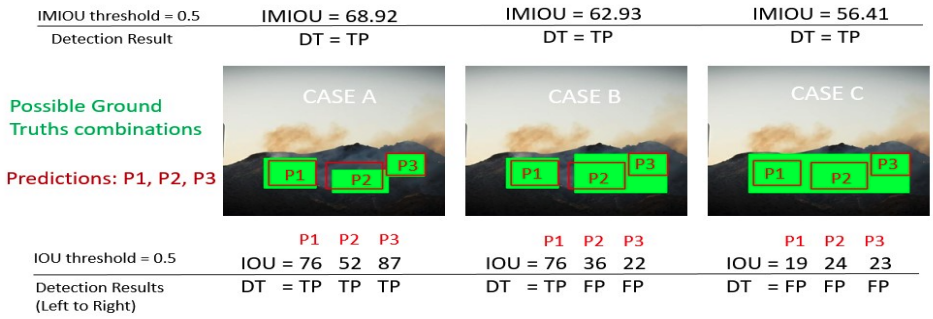

Figure 5: Comparison of the IOU with ImIoU for three different cases. Above the images are the ImIoU results along with the detection results for each scenario. Below the images are the IOU of each prediction P1, P2, P3 with their corresponding ground-truth. We observe that ImIoU predict the images as True Positives regardless of the annotation style. In contrast to ImIoU, IOU in most cases will result many incorrect FPs affecting the mAP

by the backbone and transfer information from one map to another via down-sampling and up-sampling. Subsequently, predictions are generated from each new feature map. In contrast to Yolo-v8, Faster-RCNN [18] generate its predictions based on the Region Proposal Network (RPN). The RPN, for every vector of the last feature map, predicts the coordinates of many bounding boxes along with their confidences scores. Then, for every proposal, Faster-RCNN extracts the corresponding part of the feature map and feeds it to a network in order to classify it.

The rise of Natural Language Processing, thanks to transformer-based [4, 24] architectures with excellent long-range dependency detection capabilities, also made an impact in the field of the computer vision. Soon enough, architectures like Visual Transformer (VIT) [5] and Detection Transformers (Detr) [1] demonstrated superior performance compared to traditional CNN-based approaches. RT-DETR [16] is state of the art real-time object detector. The RT-DETR architecture consists of a ResNet backbone, a hybrid encoder that transfers information between the last three feature maps produced by the backbone, and a decoder comprising several stacked transformer decoder layers. From the output of the decoder, RT-DETR predicts the bounding boxes along with their associated classes.

4.3 Experimental Setup

All models were trained for 72 epochs, 640 input image size and with its recommended setup. The Faster R-CNN was trained using the Stochastic Gradient Descent (SGD) optimizer, with a learning rate of 10^{-3} and a batch size of 4. In contrast, the RT-DETR employed the AdamW optimizer, set at a learning rate of 10^{-4} and 10^{-5} , and also maintained the same batch size of 4. Lastly, the YOLOv8 was trained using SGD, but with a higher learning rate of 10^{-2} and a larger batch size of 16.

4.4 Experimental Results

Fig. 6 depict $ImAP[0.5 : 0.05 : 0.95]$ and $ImAP@[0.5]$ scores in relation to the Confidence Score (CS) threshold. It is crucial to emphasize that maintaining a zero or constant CS threshold across

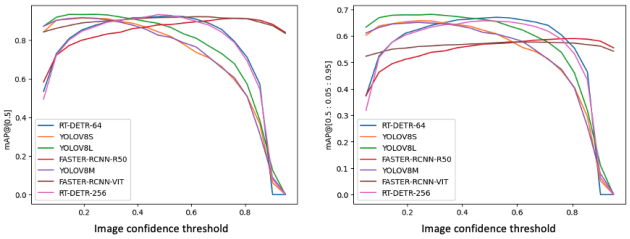

Figure 6: ImAP@[0.5] and ImAP[0.5:0.05:0.95] with respect to image confidence threshold

MODEL	mAP	$mAP_{0.5}$	ImAP	$ImAP_{0.5}$
F-RCNN[18]-VIT [5]	47.6	79.8	57.57	91.84
F-RCNN [18]-R50 [10]	51.04	82.35	59.02	91.74
YOLOV8-Large [13]	61.8	86.5	67.76	93.05
YOLOV8-Medium [13]	59	84.6	64.98	91.39
YOLOV8-Small [13]	58.9	85.2	65.57	91.53
RT-DETR-R50-256 [16]	58.06	84.36	65.25	93.02
RT-DETR-R50-64 [16]	59.1	84.82	67.06	93.07

Table 1: Comparative evaluation of Faster-RCNN, YOLO-v8, and RT-DETR DNN methods using mAP and ImAP evaluation metrics

	mAP	$mAP_{0.5}$	ImAP	$ImAP_{0.5}$
mAP	1			
$mAP_{0.5}$	0.9812	1		
ImAP	0.9846	0.9519	1	
$ImAP_{0.5}$	0.4248	0.4031	0.5249	1

Table 2: Correlation between the Table 1 columns

all models, as observed in Fig. 6, can lead to an unjust and biased comparison of object detectors. By selecting the CS threshold that maximizes ImAP, we obtain a robust metric that reveals the maximum fire detection performance of the models.

The maximum ImAP values for each detector are presented in Table 1, accompanied by mAP metric results for comprehensive assessment. Notably, the strong correlation between $ImAP@[0.5 : 0.05 : 0.95]$ and mAP (Table 2) proves that the ImAP metric, as the ImIoU threshold increases, mirrors the behavior of mAP, eliminating any margin for "error" areas in fire detection. However, $ImAP@[0.5]$ shows a lower correlation with mAP due to its ability to overcome the incorrect discrepancies between predicted and ground-truth bounding boxes (Fig. 5, case C). Consequently, based on its results, we can identify the model that excels in predicting fire within an image.

5 CONCLUSIONS

In this work a new metric for evaluating fire detection algorithms, called Image-level mean Average Precision (ImAP) is proposed. Due to the particular nature of the fire detection task, the proposed metric measures how well the overall fire is detected in the whole image, extending the bounding box-per-bounding box evaluation protocol followed by the typical mAP metric. Experiments using a wide variety of object detection algorithms and a challenging fire detection dataset have shown that the proposed metric can more accurately capture and represent the actual performance of the fire detectors. As a result, it can serve as a very useful tool for a wide range of NDM applications. Through additional experiments it is also shown that for increased threshold values the proposed ImAP metric behaves similar to the typical mAP one. Finally, it is shown that ImAP with a threshold value $\tau = 0.5$ provides very useful insights for selecting the most appropriate fire detection model.

6 ACKNOWLEDGEMENT

The research leading to these results has received funding from the European Commission - European Union (under HORIZON EUROPE (HORIZON Research and Innovation Actions) under grant agreement 101093003 (TEMA) HORIZON-CL4-2022-DATA-01-01). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union - European Commission. Neither the European Commission nor the European Union can be held responsible for them.

REFERENCES

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*. Springer, 213–229.

[2] Chuangmao Chen, Jie Yu, Yuqing Lin, Fuqiang Lai, Guoqiang Zheng, and Youxi Lin. 2023. Fire detection based on improved PP-YOLO. *Signal, Image and Video Processing* 17, 4 (2023), 1061–1067.

[3] Pedro Vinicius AB de Venancio, Adriano C Lisboa, and Adriano V Barbosa. 2022. An automatic fire detection system based on deep convolutional neural networks for low-power, resource-constrained devices. *Neural Computing and Applications* 34, 18 (2022), 15349–15368.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

[6] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. 2010. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* 88, 2 (2010), 303–338. http://dblp.uni-trier.de/db/journals/ijcv/ijcv88.html#EveringhamGWWZ10

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision* 88 (2010), 303–338.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. [n. d.]. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[9] Rafik Ghali and Moulay A Akhloufi. 2023. Deep Learning Approaches for Wildland Fires Remote Sensing: Classification, Detection, and Segmentation. *Remote Sensing* 15, 7 (2023), 1821.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[11] Katie Hoover and Laura A Hanson. 2015. *Wildfire statistics*. Congressional Research Service.

[12] ichrak. 2022. jhope Dataset. https://universe.roboflow.com/ichrak/jhope. https://universe.roboflow.com/ichrak/jhope visited on 2023-10-12.

[13] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. 2023. *YOLO by Ultralytics*. https://github.com/ultralytics/ultralytics

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 740–755.

[15] Xiang Long, Kaipeng Deng, Guanzhong Wang, Yang Zhang, Qingqing Dang, Yuan Gao, Hui Shen, Jianguo Ren, Shumin Han, Errui Ding, and Shilei Wen. 2020. PP-YOLO: An Effective and Efficient Implementation of Object Detector. *CoRR* abs/2007.12099 (2020). arXiv:2007.12099 https://arxiv.org/abs/2007.12099

[16] Wenyu Lv, Shangliang Xu, Yian Zhao, Guanzhong Wang, Jinman Wei, Cheng Cui, Yuning Du, Qingqing Dang, and Yi Liu. 2023. Detrs beat yolos on real-time object detection. *arXiv preprint arXiv:2304.08069* (2023).

[17] Rafael Padilla, Sergio L Netto, and Eduardo AB Da Silva. 2020. A survey on performance metrics for object-detection algorithms. In *2020 international conference on systems, signals and image processing (IWSSIP)*. IEEE, 237–242.

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

[19] G. Salton and M. J. McGill. 1986. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.

[20] Hartmut Surmann, Artur Leinweber, Gerhard Senkowski, Julien Meine, and Dominik Slomma. 2023. UAVs and Neural Networks for search and rescue missions. *arXiv preprint arXiv:2310.05512* (2023).

[21] Fatma M Talaat and Hanaa ZainEldin. 2023. An improved fire detection approach based on YOLO-v8 for smart cities. *Neural Computing and Applications* (2023), 1–16.

[22] T. Toulouse, L. Rossi, A. Campana, T. Celik, and M.A. Akhloufi. [n. d.]. jhope Dataset. https://feuxdeforet.universita.corsica/article.php?id_art=2133&id_rub=572&id_menu=0&id_cat=0&id_site=33&lang=en. https://feuxdeforet.universita.corsica/article.php?id_art=2133&id_rub=572&id_menu=0&id_cat=0&id_site=33&lang=en accessed on 5 January 2023.

[23] Tom Toulouse, Lucile Rossi, Antoine Campana, Turgay Celik, and Moulay A Akhloufi. 2017. Computer vision for wildfire research: An evolving image dataset for processing and analysis. *Fire Safety Journal* 92 (2017), 188–194.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[25] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. 2023. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7464–7475.

[26] Hao Wu, Deyang Wu, and Jinsong Zhao. 2019. An intelligent fire detection approach through cameras based on computer vision methods. *Process Safety and Environmental Protection* 127 (2019), 245–256.

[27] Yi Yang, Mengyi Pan, Pu Li, Xuefeng Wang, and Yun-Ting Tsai. 2023. Development and optimization of image fire detection on deep learning algorithms. *Journal of Thermal Analysis and Calorimetry* 148, 11 (2023), 5089–5095.

[28] Shiping Ye, Zhican Bai, Huafeng Chen, Rykhard Bohush, and S Ablameyko. 2017. An effective algorithm to detect both smoke and flame using color and wavelet analysis. *Pattern Recognition and Image Analysis* 27 (2017), 131–138.

[29] Haodi Zhang, Alexandrina Rogozan, and Abdelaziz Bensrhair. 2022. An enhanced N-point interpolation method to eliminate average precision distortion. *Pattern Recognition Letters* 158 (2022), 111–116.

[30] Mu Zhu. 2004. Recall, precision and average precision. *Department of Statistics and Actuarial Science, University of Waterloo, Waterloo* 2, 30 (2004), 6.

[31] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. 2020. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159* (2020).