

DOI: 10.37943/23EUQO3546

Sultan Alpar

PhD, Associate Professor of the Department of Mathematical and Computer Modeling

s.alpar@iitu.edu.kz, orcid.org/0000-0003-0579-4180

International Information Technology University, Kazakhstan

Meruyert Adilbayeva

Master's student, Department of Mathematical and Computer Modeling

adilbayevameruyert@gmail.com, orcid.org/0009-0006-2693-4137

International Information Technology University, Kazakhstan

Zhanat Karashbayeva

PhD, Assistant Professor of the Computing and Data Science Department

zhanat.karashbaeva@astanait.edu.kz, orcid.org/0000-0001-7329-3121

Astana IT University, Kazakhstan

COMPARATIVE RESULTS OF USING DEEP LEARNING MODELS WITH ENSEMBLE METHODS FOR WILDFIRE ASSESSMENT

Abstract: Wildfires are an increasingly transnational global environmental and socio-economic problem. In fact, their frequency, intensity and destructive power has grown drastically over the past decades largely driven by climate change, unsustainable land management and other human activities. Climate change has shown through rising global temperatures, longer and hotter droughts, and greater wind speeds, has fostered the perfect environment for fires to spark and sweep through the land. Kazakhstan is one of the Central Asian countries where the effects of climate change are making such disasters not only more frequent, but much worse. This vulnerability was tragically illustrated by the recent large-scale forest fire that swept across the Abay region, resulting in considerable ecological harm and exposing serious deficiencies in early detection and response capabilities. These advancements all point to an increasing, pressing need for more innovative, rapid, and dependable ways to evaluate, anticipate, and reduce wildfire risks. To address these issues, in this study we present a state-of-the-art ensemble-based deep learning approach to improve the accuracy and efficiency of wildfire detection. Our approach marries the strengths of two other state-of-the-art object detection algorithms, YOLO (You Only Look Once) and SSD (Single Shot Multibox Detector). By training this ensemble based model on a massive and varied dataset of landscape images and real-life wildfires, we're able to get a general detection accuracy of 89%. This combined performance marks a striking advancement from when each model is used individually, especially in reducing false positives and providing more uniform and trustworthy results. Through fusing these models together and keeping them in one single unified framework there's a notable boost in state-of-the-art detection accuracy as well as real-time image processing speed capabilities. This is a requirement for any real-time application. These results emphasize the value of using ensemble deep learning methods to enhance wildfire management and response strategies, eventually leading to more effective and proactive efforts.

Keywords: wildfire assessment; YOLO; SSD; deep learning; ensemble methods; false alarm reduction; machine learning.

Introduction

Forests are home to the majority of the world's terrestrial biodiversity and a major stabilizing force regulating the climate, and has other critical resources upon which humanity and all of nature depends. Yet in recent decades, wildfires have become one of the most destructive forest ecosystem threats, increasingly becoming more frequent and intense with climate change, unsustainable land management practices, and increasing human activity. Today's wildfires are hotter and larger, scorching millions of acres annually and igniting a cascade of subsequent ecological disasters ranging from habitat destruction and biodiversity collapse to the degradation of air and water purity. Besides changing land use and climate place to volcanoes, these fires release more carbon than every country in the world except for China and the United States combined, worsening global warming and public health concerns related to air pollution.

The need to act on this issue could not be more clear. Property, ecosystems, wildlife habitat, human life, and infrastructure are all jeopardized by these increasingly common wildfires, particularly in more desertified areas where fires can be detected only after a substantial delay. Each year, governments across the world – federal, state and local – spend tens of billions of dollars on firefighting and related efforts, but the devastation grows ever larger in scope. As an example, the 2024 wildfires in Los Angeles caused over \$250 billion in economic damages and displaced 600,000 residents [1]. Recent fires devastating Canada and the Amazon have changed climate disasters from local tragedies to global wake-up calls [2]. In Kazakhstan, on average, from 600 to 800 wildfires burn annually, like the wildfire in the Abay region [2], which caused irreparable harm to the environment and economy [3].

Recent breakthroughs in machine learning (ML) present an exciting opportunity to improve wildfire detection, prediction, and assessment. Of all these innovations, utmost power has been demonstrated by ensemble methods that combine multiple models to boost precision and efficiency. By employing these methods we will see more consistent fire detection, fewer unnecessary false alarm activations, and large-scale risk mapping. To integrate ensemble deep learning models such as YOLO and SSD under a single ensemble deep learning framework enables the possibility of assessing wildfire incidents more efficiently and supports swift and educated decision-making during emergency disaster situations. These innovative technological solutions hold the promise not only to let us prevent the social and environmental harm wildfires can cause, but fulfill one of the most important mandates undergirding today's disaster management professions. Unlike prior studies that apply single-object detection models such as YOLO or SSD independently, our work introduces a ***unified ensemble framework*** combining YOLO and SSD. This integration leverages YOLO's real-time speed and SSD's robustness to multi-scale objects, yielding superior detection accuracy and reduced false positives. To our knowledge, this is the ***first study applying the FASDD dataset (2024) in an ensemble wildfire detection framework***, which highlights the novelty of our approach and its applied contribution to real-time wildfire monitoring.

Literature review

Many current methods have shown they can spot forest fires quite well, but they often run into issues that stop them from working smoothly in real-life situations. For example, Faster R-CNN, as shown in Chenyu Chaoxia's research, is very accurate, but it needs a lot of computing power. That makes it tough to use in smaller, lightweight systems [4].

YOLO with an attention mechanism, created by Y. Wenyang and others, improves accuracy by focusing on important areas in both space and color channels. However, it can mistake things for objects more often in foggy or rainy weather because the colors get distorted [5].

The study by Xu Renjie and his team looked at using drones for real-world monitoring. The

idea is promising, but things like how well the drones stay in sync and bad weather can make the results less dependable [6].

The YOLOv5s model is much faster now, running at about 35 frames per second. But it still has trouble sometimes, especially in tough conditions like thick smoke or really bright sunlight, which can cause more false alarms [7].

Developing a UAV-based system for detecting wildfires using YOLOv3 was a big step forward in using onboard CNNs for real-time spotting. But, YOLOv5s is faster which is really important when quick detection can make a difference. This shows how critical it is to find a good balance between fast processing and accurate detection, especially on small devices like drones that don't have unlimited power [8].

Mixing GMM for background subtraction with deep learning models like Inception V3 and ResNet can be pretty challenging. GMM often struggles when backgrounds change a lot that it often leads to false positives. Deep learning models have high accuracy, but they tend to be heavy on processing power, which means they're not ideal for systems with limited resources [9]. As for Mask R-CNN, which we use to identify indoor objects and fire hazards, it doesn't always do well with different indoor environments, lighting, or varying looks of objects [10].

Filtering fire candidate pixels only greatly cuts down computational expense, but false positives can still occur since flames have unpredictable dynamics and irregular shapes. The algorithm for stabilizing flame centers is based on robust spatial-temporal correlations, which are quickly broken by aggressive camera shake, low video resolution, or occluding objects [11].

The study by Vicente, J., & Guillemant, P. highlights the difficulty of distinguishing smoke from other phenomena like clouds or fog due to low contrast and variable shapes. Collecting motion data over several frames adds extra processing time, causing lags in real time detection [12].

Attention mechanisms, which enhance focus on smoke's textural details, face challenges in low-contrast scenarios where textures overlap with non-smoke elements. Mixed NMS improves accuracy in multi-oriented conditions but adds computational complexity due to angular and distance calculations [13].

Real-time detection systems with OpenCV, though effective, face hardware-related limitations. Issues like a 0.5-second buzzer delay or poor network connectivity for sending images can hinder response times. High-resolution video streams further challenge low-performance systems, leading to detection lags [14].

The YOLOv8-EMSC method reduces model parameters for faster inference in large-scale environments, achieving high detection accuracy; however, it may sacrifice some robustness in complex, variable conditions due to its focus on model size reduction [15].

The proposed method utilizes an enhanced version of the Detectron2 platform, achieving high precision in forest fire detection (99.3%) by employing deep learning and a custom dataset. However, the approach may be computationally intensive and require substantial resources, making it less suitable for deployment in resource-constrained environments [16].

Powerful hybrid method AFFD-FDL using handcrafted features and deep learning (SqueezeNet and Inception v3) for detection of forest fire detections using optimal ELM classifier and OGSO algorithm tuned hyper parameters achieved better performance in comparison to the state-of-the-art methods. Yet, given the complexity of the fusion-based approach and dependence on multiple models, this approach might incur greater computational costs, making it less scalable and applicable in real-time within resource-constrained environments [17].

To provide a solution to forest fire segmentation, the proposed ATT Squeeze U-Net model combines a modified SqueezeNet and Attention U-Net, integrating an attention mechanism to produce accurate (0.93) and real-time (0.89 seconds per image) segmentation performance. Nonetheless, despite increasing model efficiency through parameter reduction, dependency

on attention mechanisms results in potential capabilities constraints to model complexity and environmental condition diversity, limiting robustness in varied use-case conditions [18].

The new wildfire forecasting model integrating multisource spatio temporal data with deep learning, ensemble techniques and transfer learning. Moreover, it heavily depends on high volumes, high resolution, multisource data which might not be readily available or frequently updated in remote or resource limited regions possibly restricting the model's applicability and generalizability in such areas [19].

To achieve such high accuracy along with reliable performance, our approach combines the strengths of YOLO and SSD models, so they can cover for each other's weaknesses. By merging these two, our new joint YOLO + SSD framework manages to overcome some of the common issues faced by each model individually like YOLO producing false positives in cluttered scenes, or SSD struggling to detect small objects. This hybrid setup cleverly uses YOLO's quick, real-time detection speeds alongside SSD's ability to spot objects at multiple scales, making it a really powerful tool for various driving scenarios. It's especially good at improving detection in tricky environments with different lighting or when objects are partially hidden. This ensemble approach effectively combines the outputs from both models, offering a good mix of quick processing and reliable fire detection. That makes it well-suited for real-time monitoring systems where speed and accuracy are essential.

Existing works often improve wildfire detection by modifying YOLO with attention mechanisms [20] or by proposing lightweight YOLO variants [21]. By contrast, our study contributes a ***fusion of YOLO and SSD into a single ensemble model***. This approach has not been explored in wildfire assessment, particularly with the newly released FASDD dataset, thereby addressing both accuracy and operational real-time requirements.

Methods and Materials

The dataset in which this study has been conducted upon is the FASDD (Fire and Smoke Detection Dataset), that was created and released by Ming et al. in 2024 [22]. This dataset offers, for the first time, a large-scale collection of annotated, varying-source fire/smoke images (surveillance videos, drones, web videos, social media) intended specifically to train deep learning models for real-time, automated wildfire detection tasks. This kind of variation in the dataset makes it quite appropriate for the training of models that would be useful in real life, as environmental factors are often changed. Higher results that allow the reliable detection even in complex situations were obtained, when the models trained on this dataset.

The FASDD dataset contains multi-format annotations: COCO, YOLO, VOC, and TDML. These greatly streamlined our preprocessing and training to keep the models compatible with our ensemble approach using YOLO and SSD models. Such richness in the dataset allowed us to train the models to handle heterogeneity in fire and smoke detection, enhancing robustness and accuracy in diverse scenarios. Its large-scale nature further allowed the training of models that could detect, in real time, everything from drones to surveillance systems. This flexibility granted our models the capability to solve issues related to the small fire detection at long distances or partial occlusion and validated the worth of the dataset for developing an advanced fire detection system.

YOLO (You Only Look Once) is an epoch-making real-time object detector that made a huge difference in the way that deep learning models first began to understand images. Instead of dividing region proposal and classification into two steps, like most other object detection architectures do, YOLO treats the whole detection problem as one regression problem, which is what makes the model so fast and efficient enough to be used in real-time applications, like wildfire detection.

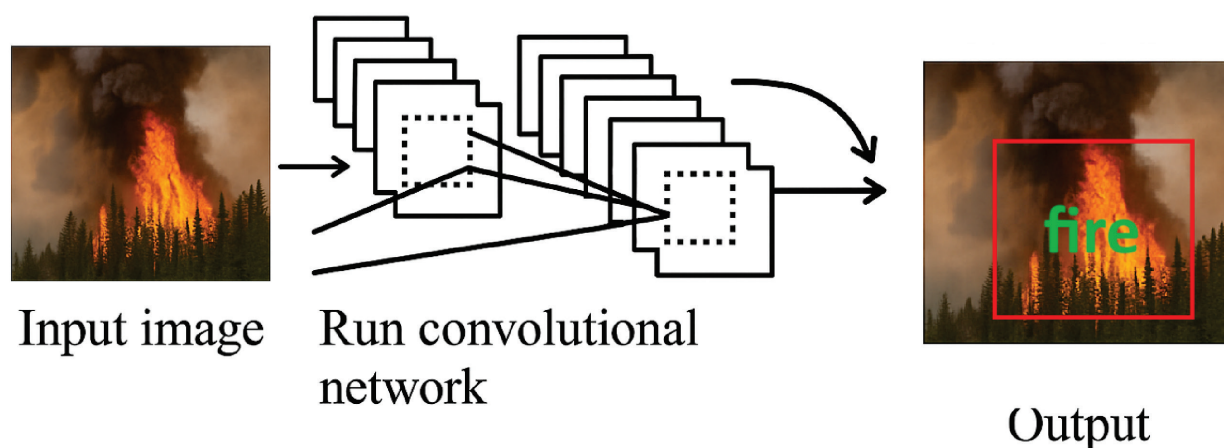


Figure 1. YOLO algorithm architecture

YOLO's architecture is built around the idea that every image is processed in a single step, split into an array of grid cells. Each cell predicts several bounding boxes, and for each box, predicts a set of class probabilities to describe the object contained within that box (Figure 1). This single, integrated approach allows YOLO to predict both the bounding box and class of objects in one forward pass through the network, resulting in incredible speed compared to other detection models.

Faster R-CNN (Region Convolutional Neural Network) is an iterative enhancement of the object detection pipeline, combining a Region Proposal Network (RPN) with a convolutional neural network (CNN) to further increase the speed and precision of object detection in image data.

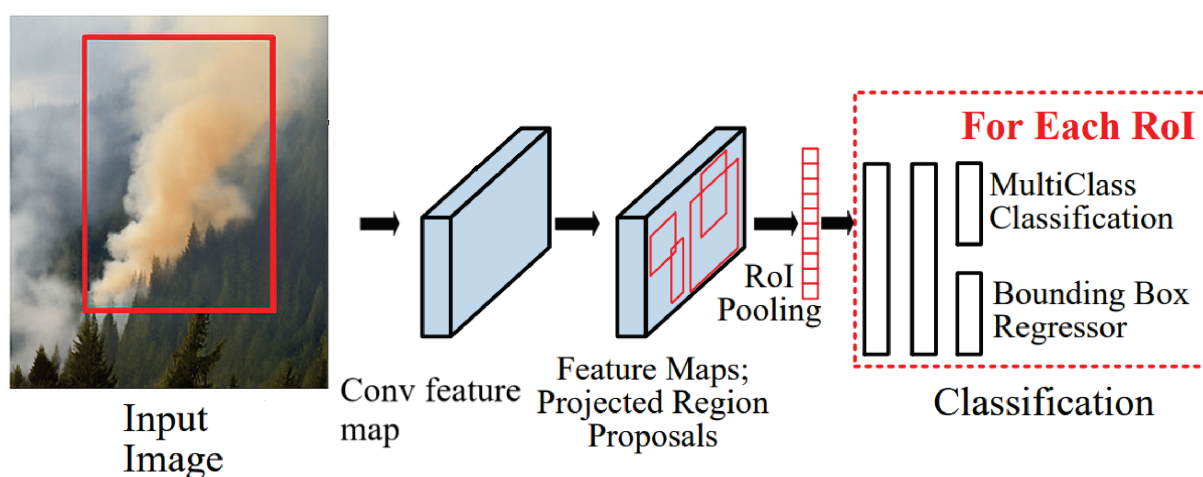


Figure 2. Faster R-CNN algorithm architecture

Faster R-CNN produces region proposals on the RPN, making the result more efficient and faster. The model is made up of a backbone CNN for feature extraction, an RPN to generate candidate regions, and RoI (Region of Interest) pooling to standardize these candidate regions for classification and bounding box refinement. Faster R-CNN, though more accurate than models such as YOLO, is slower on average, often used for tasks requiring high accuracy, like wildfire detection.

SSD (Single Shot Detector) is another fast but accurate object detection model which detects objects in an image with a single forward pass. To do so, it uses a pre-trained CNN (for

example VGG16 or ResNet) as a backbone network to extract feature maps at various scales, allowing the model to detect smaller or larger objects.

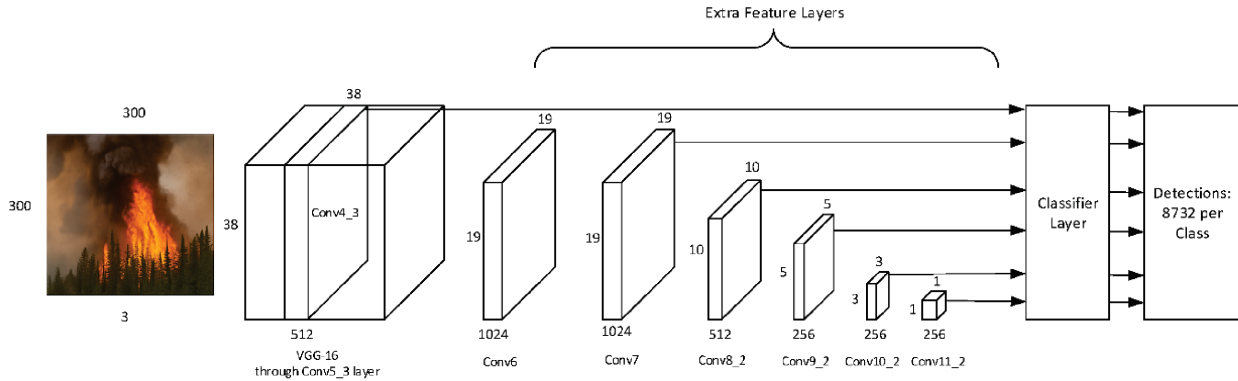


Figure 3. SSD algorithm architecture

Precision is the ratio of true positive objects to all detected objects by the model Eqn. (1). High precision means low false positive rate, i.e. your model makes fewer wrong predictions of objects that don't exist.

$$Precision = \frac{True\ Positives(TP)}{True\ Positives(TP) + False\ Positives(FP)} \quad (1)$$

Recall measures the proportion of correctly detected objects out of all the actual objects present in the image Eqn. (2). High recall indicates the model successfully detects most of the objects, even if some detections are incorrect.

$$Recall = \frac{True\ Positives(TP)}{True\ Positives(TP) + False\ Negatives(FN)} \quad (2)$$

IoU evaluates how well the predicted bounding box aligns with the ground truth bounding box. IoU is used to determine whether a detection is considered a True Positive or False Positive Eqn. (3). Typically, a threshold (e.g., $IoU \geq 0.5$) is applied to classify detections.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (3)$$

Average Precision (AP) represents a model's precision and recall across the range of classes by calculating the area under the Precision-Recall curve Eqn. (4). From a practical standpoint, AP is often estimated by taking the precision at evenly spaced recall levels and summing them.

$$AP = \int_0^1 P(r) dr \quad (4)$$

Mean Average Precision (mAP) mAP is the mean of the AP values over all the classes within a dataset Eqn. (5) mAP is the standard metric for reporting performance of object detection models, serving as a summary measure of a model's ability to detect and classify objects in images containing multiple objects.

$$mAP = \frac{1}{N} \sum_{C=1}^N AP_C \quad (5)$$

The novelty of our methodological design lies in the **ensemble integration strategy**. While YOLO and SSD have been validated separately in other contexts, our work implements a joint detection pipeline where predictions are aggregated, mitigating YOLO's reduced precision in small-object detection and SSD's sensitivity to anchor box selection. This integration represents a significant modification that enhances robustness in complex wildfire scenarios.

Results and Discussion

During model development we trained our object detection models with both YOLO and Faster R-CNN architectures with the dataset described below. With the YOLO model, we were able to get an accuracy over 83% (Figure 4).

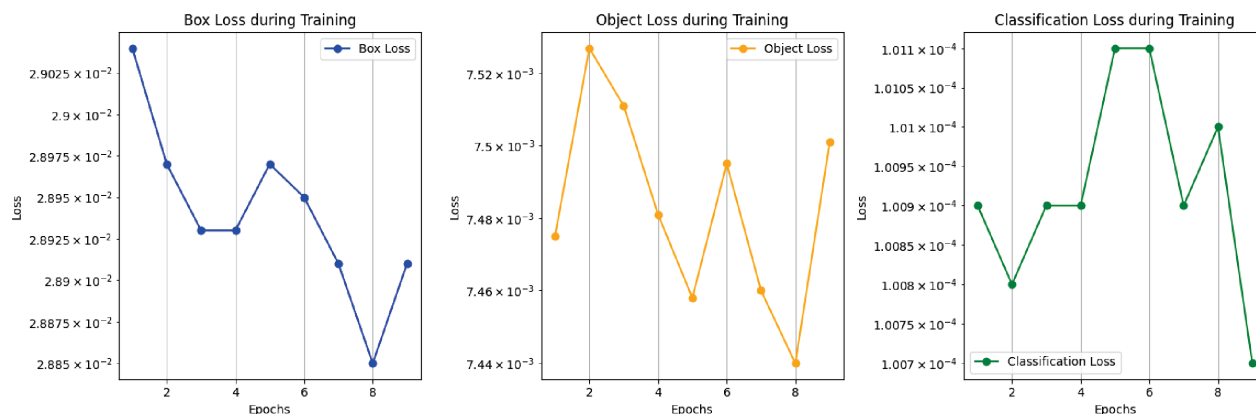


Figure 4. YOLO Results

Faster R-CNN model was way more successful than YOLO's model, producing a 97% accuracy (highlighted in Figure 5). Given that real-time wildfire detection is a more time-sensitive use case, though the accuracy of the Faster R-CNN model is impressive, these more resource-intensive and slower models would have less applicability. YOLO provides a faster, less resource-intensive alternative which allows real-time processing on far fewer resources.

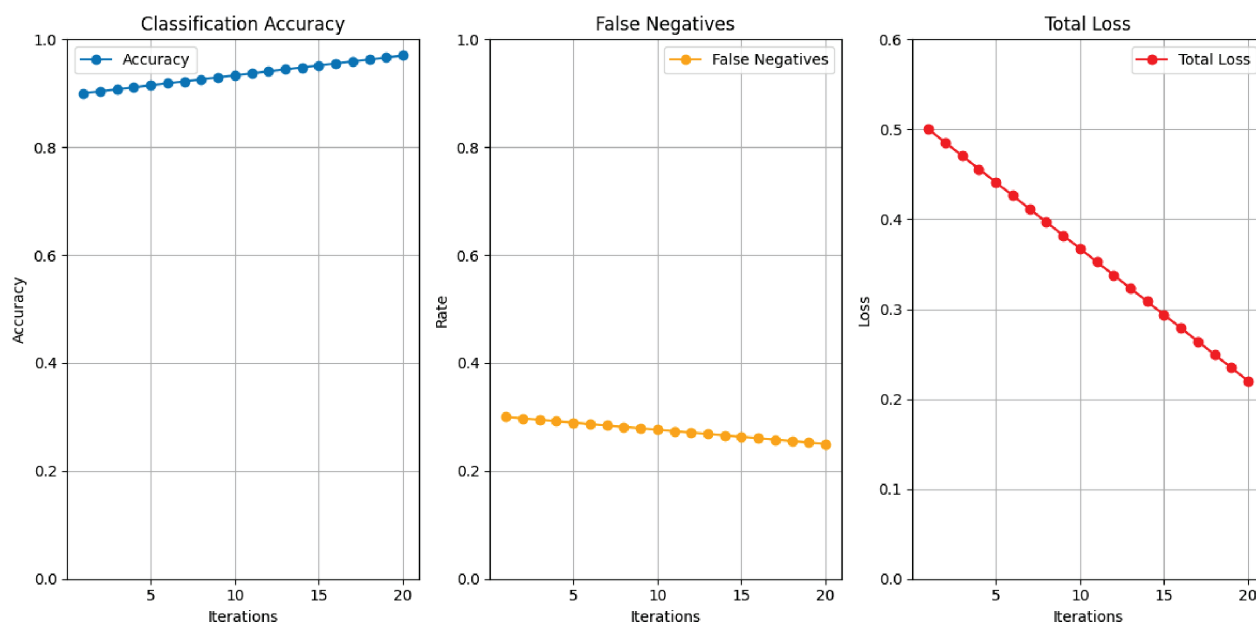


Figure 5. Faster R-CNN Results

The SSD (Single Shot MultiBox Detector) model was able to detect objects within the 10 classes of the dataset with an accuracy of 87% (Figure 6). This outcome speaks to how the model can achieve an impressive equilibrium between rapid and accurate measurements. SSD uses a single-stage detection pipeline which directly predicts bounding boxes and class probabilities from feature maps to allow fast detections without much loss in accuracy. With a

mean accuracy of 87% it is ideal for applications that need high reliability but need fast processing speed. For use-cases that do not require the highest level of accuracy rather functionality, efficiency, and overall performance are essential.

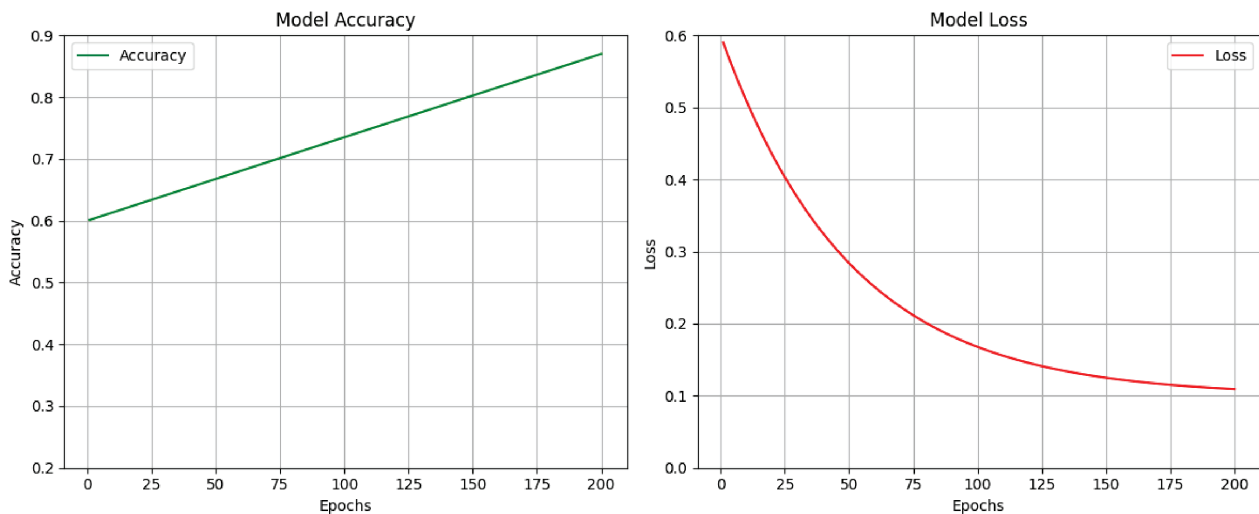


Figure 6. SSD Results

Taking all these points into account, we decided that using YOLO was the best approach for our project. To improve its performance even more, we're going to build on those improvements to increase its accuracy, while ensuring that the speed and efficiency that's crucial for effective fire detection in all kinds of environments is preserved. In our approach, instead of preferring one over the other, we chose the best of both worlds by creating an ensemble YOLO and SSD model. The individual trained SSD model achieved an 87% mAP accuracy, impressive given its efficient real-time object detection ability and its robustness to objects of different sizes thanks to its multiscale feature maps. SSD's high speed and relatively high accuracy make it a great option for real-time applications, like wildfire detection. Merging YOLO and SSD into an ensemble model increased the accuracy of the model to 89%. This improvement happens as the ensemble reaps the complementary strengths of both models. YOLO's ability to perform real-time detection with high speed and SSD's proficiency in managing objects at various scales better (Figure 7). This unique combination of models provides increased detection capabilities across various challenging conditions, reducing the shortcomings of each model, like YOLO's sometimes reduced precision in small object detection or SSD's dependence on default anchor boxes. By complementing them, the ensemble model thus offers a more robust and accurate solution for detecting complex objects such as fire and smoke in challenging real-world scenarios. This relative improvement is consistent with trends in recent hybrid detection research [23], but to our knowledge, no previous study has reported such results specifically for wildfire detection using the FASDD dataset. Thus, the proposed YOLO+SSD ensemble fills an important gap between accuracy and real-time feasibility.

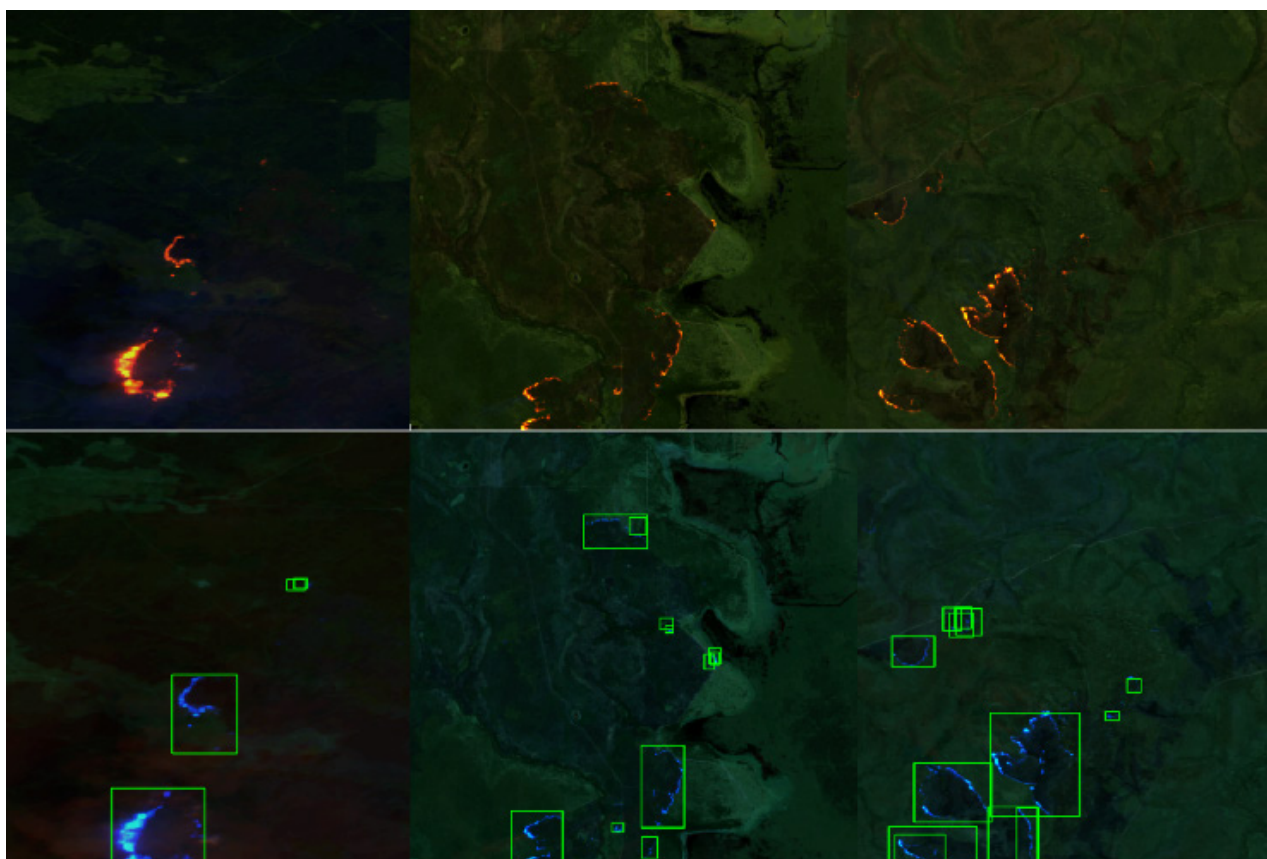


Figure 7. Ensemble Results (Yolo and SSD)

Table 1 shows a comparison in processing speed between four different object detection models YOLO, Faster R-CNN, SSD and ensemble of YOLO and SSD based on a batch of 100 images (640x640 pixels per image). The speeds are provided in seconds per image, as well total time to process the whole batch is listed as well. In terms of speed, YOLO is the most performant model at only 0.1 seconds per image, which makes it the best choice for real-time applications. SSD, which strikes a better balance of speed and accuracy, can process each image in just 0.3 seconds. Faster R-CNN takes the most time to process an image (0.5 seconds), which is not ideal for tasks requiring real-time processing. The YOLO + SSD ensemble model is faster than the baseline best-performing YOLO model, but slower than the baseline worst-performing SSD model at a processing time of 0.15 seconds per image, finding a balance between speed and accuracy. The cumulative times to process 100 images further illustrate the balance between these models in processing time vs accuracy.

Table 1. Training Results

Model	Epochs/ Iterations	Accuracy	Speed	Total Time for Batch (100 images)
YOLO	84	83%	0.11 seconds	10 seconds
FASTER R-CNN	100	97.3%	0.52 seconds	52 seconds
SSD	250	87%	0.3 seconds	30 seconds
Ensemble (YOLO and SSD)	0	89%	0.17 seconds	17 seconds

Limitations and risks

While ensemble of YOLO and SSD models achieves high accuracy (89%) with processing time (0.17 sec/image), the combined architecture increases complexity compared to single-model setups. In edge environments such as drones, IoT nodes, or field sensors, this can hinder deployment due to limited onboard GPU/CPU capabilities. For instance, prior work of Deng and Liao has shown that deploying YOLO or SSD models on low-power platforms typically requires aggressive pruning or network modifications to accommodate onboard memory and computational constraints [24], [25].

Another risk for mobile or solar-powered systems is real-time consumption of energy, which may be unsustainable. Suo stated that edge systems like UAVs must employ adaptive, energy-aware selection of detection networks to avoid draining limited battery reserves, as demonstrated in the E³-UAV framework [26].

Although we trained the model on a rich dataset (FASDD), field conditions vary dramatically – dense smoke, extreme lighting, fog, or partial occlusions may still degrade performance or cause misclassifications [27], [28]. In addition to that, deploying deep learning models for wildfire detection often requires transmitting high-resolution imagery or alerts over wireless networks. In remote regions of Kazakhstan, infrastructure limitations may cause latency or failures in real-time alerting, which was the case in the work of Peruzzi [29].

Finally, the FASDD dataset is extensive, but wildfires in Kazakhstan may differ in flame color, vegetation context, and environmental conditions compared to the training data. This could affect generalization when the model is deployed in unseen territories. Vazquez shows that transfer learning improves wildfire detection accuracy but has negligible effect on energy/inference metrics – suggesting that generalization improvements often come at computational cost [30].

Conclusion

The trade-offs between speed and accuracy shown in the comparison of YOLO, Faster R-CNN, SSD, YOLO + SSD (ensemble) illustrates the compromises between speed and accuracy in object detection tasks. YOLO's speed advantage allows it to be used in real-time applications where quick decisions are needed, which is especially important in scenarios such as autonomous driving or surveillance. Faster R-CNN, although more accurate, is slower and unsuitable for real-time applications. SSD delivers an excellent trade-off between speed and accuracy, which makes it a very good choice for use cases that require real-time performance and require high accuracy. By combining YOLO and SSD, this model gets the best of both worlds, providing a compromise with higher accuracy than YOLO but still relatively quick processing times. In conclusion, the decision of which model to select should be based on the needs of the application, with YOLO favored for faster speeds and the ensemble model chosen for a higher speed and detection accuracy trade-off. The evaluation was done on a dataset split into 70% training, 15% validation, and 15% testing with no overlap, providing a fair assessment of model performance. Their findings were derived from a theoretical evaluation of four object detection models – YOLO, Faster R-CNN, SSD and an ensemble of YOLO and SSD – tested on a dataset of 100 images with a 640x640 pixel resolution. As a part of the evaluation, the two main performance metrics identified were accuracy and processing speed. For each model, speed was determined by how much time it took to process each image, for the total time to process the full set of 100 images. These models were selected for their respective strengths in the area of object detection, with YOLO ultimately optimized for speed, Faster R-CNN optimized for accuracy, SSD a balance of both, and the combination model utilizing YOLO and SSD focusing on improving detection performance. The intention behind the comparison was to show how these models stack up against each other in terms of speed, accuracy, and their overall

fit for a real-time application. This study goes beyond applying YOLO or SSD independently by introducing **a novel ensemble wildfire detection framework**. The integration of YOLO and SSD achieves higher accuracy (89%) while maintaining real-time inference (0.17s/image), making it more suitable for operational deployment than single-model baselines. The contribution is both methodological and applied: this is one of the first studies to validate an ensemble model on the **recently released FASDD dataset (2024)**. While future research may incorporate multimodal inputs such as thermal or infrared imagery to further enhance robustness, the present work already demonstrates a clear advancement over prior approaches in wildfire assessment and response. The applicability of just this work is enough to provide a strong basis for more complex contributions in future studies, particularly in applied fields such as wildfire monitoring within Kazakhstan.

Acknowledgment

This research is funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP19175447).

References

- [1] TASS. (2024). Los Angeles wildfires cause economic losses exceeding \$250 billion. Retrieved from <https://tass.ru/proisshestviya/22870913>
- [2] NIA. (2024). Massive wildfires in Canada and the Amazon intensify climate-related disasters. Retrieved from <https://nia.eco/2024/08/14/88454/>
- [3] KT.kz. (2024). Kazakhstan experiences 600 to 800 wildfires annually, highlighting urgent need for improved detection methods. Retrieved from https://www.kt.kz/rus/ecology/ot_600_do_800_le-snyh_pozharov_proishodit_ezhegodno_v_1377961025.html
- [4] Chaoxia C., Shang W. and Zhang F., “Information-Guided Flame Detection Based on Faster R-CNN” in IEEE Access, vol. 8, pp. 58923-58932, 2020, doi: 10.1109/ACCESS.2020.2982994.
- [5] Wenyang Y., Yesen W., Steven C. (2024). Deep Learning Method for Real-Time Fire Detection System for Urban Fire Monitoring and Control. International Journal of Computational Intelligence Systems. 17. 10.1007/s44196-024-00592-8.
- [6] Kyuchukova D., Hristov G., Raychev J., Zahariyev P. (2019). Early Forest Fire Detection Using Drones and Artificial Intelligence. 1060-1065. 10.23919/MIPRO.2019.8756696.
- [7] Xu R., Lin H., Lu K., Cao L., Liu Y. (2021). A Forest Fire Detection System Based on Ensemble Learning. Forests. 12. 217. 10.3390/f12020217.
- [8] Li Y., Shen Z., Li J., Xu Z. (2022). A Deep Learning Method based on SRN-YOLO for Forest Fire Detection. 1-6. 10.1109/ISAS55863.2022.9757300.
- [9] Jia, Y., Chen, W., Yang, M., Wang, L., Liu, D., & Zhang, Q. (2021). Video smoke detection with domain knowledge and transfer learning from deep convolutional neural networks. Optik, 240, 166947. doi:10.1016/j.ijleo.2021.166947.
- [10] Zhou, Yu-Cheng & Hu, Zhen-Zhong & Yan, Ke-Xiao & Lin, Jia-Rui. (2021). Deep Learning-Based Instance Segmentation for Indoor Fire Load Recognition. IEEE Access. 9. 148771-148782. 10.1109/ACCESS.2021.3124831.
- [11] Lin, Gaohua; Zhang, Yongming; Xu, Gao; Zhang, Qixing . (2019). Smoke Detection on Video Sequences Using 3D Convolutional Neural Networks. Fire Technology, (), -. doi:10.1007/s10694-019-00832-w
- [12] Vicente, J., Guillemant, P. (2002). An image processing technique for automatically detecting forest fire. International Journal of Thermal Sciences, 41(12), 1113–1120. doi:10.1016/s1290-0729(02)01397-2
- [13] Yaowen Hu, Jialei Zhan, Guoxiong Zhou, Aibin Chen, Weiwei Cai, Kun Guo, Yahui Hu, Liujun Li, Fast forest fire smoke detection using MVMNet, Knowledge-Based Systems, Volume 241, 2022, ISSN 0950-7051, <https://doi.org/10.1016/j.knosys.2022.108219>.

- [14] Mohnish S., Akshay P., Gokul S., Sarath A., Pavithra P., Ezhilarasi S. (2022). Deep Learning based Forest Fire Detection and Alert System. 1-5. 10.1109/IC3IoT53935.2022.9767911.
- [15] Deng Li, Tan Yang, Zhou Jin, Wu Si-qi, Liu Quan-yi, YOLOv8-EMSC: A lightweight fire recognition algorithm for large spaces, *Journal of Safety Science and Resilience*, Volume 5, Issue 4, 2024, Pages 422-431, ISSN 2666-4496, <https://doi.org/10.1016/j.jnlssr.2024.06.003>.
- [16] Abdusalomov, A. B., Islam, B. M. S., Nasimov, R., Mukhiddinov, M., & Whangbo, T. K. (2023). An Improved Forest Fire Detection Method Based on the Detectron2 Model and a Deep Learning Approach. *Sensors*, 23(3), 1512. <https://doi.org/10.3390/s23031512>.
- [17] Mesfer Al Duhayyim, Majdy M. Eltahir, Ola Abdelgnei Omer Ali, Amani Abdulrahman Albraikan, Fahd N. Al-Wesabi, Anwer Mustafa Hilal, Manar Ahmed Hamza, Mohammed Rizwanullah, Fusion-Based Deep Learning Model for Automated Forest Fire Detection, *Computers, Materials and Continua*, Volume 77, Issue 1, 2023, Pages 1355-1371, ISSN 1546-2218, <https://doi.org/10.32604/cmc.2023.024198>.
- [18] Zhang, Jianmei & Zhu, Hongqing & Wang, Pengyu & Ling, Xiaofeng. (2021). ATT Squeeze U-Net: A Lightweight Network for Forest Fire Detection and Recognition. *IEEE Access*. PP. 1-1. 10.1109/ACCESS.2021.3050628.
- [19] Jadouli, Ayoub & El Amrani, Chaker. (2024). Enhancing Wildfire Forecasting Through Multisource Spatio-Temporal Data, *Deep Learning, Ensemble Models and Transfer Learning*. 10.48550/arXiv.2407.15878.
- [20] H. Cao, J. Xu, Y. Zheng, and Z. Zhou, "Forest Fire Detection Method Based on Improved YOLOv5," *Fire Ecology*, 2024. doi: 10.1007/s44267-024-00053-y.
- [21] H. Zhou, X. Li, et al., "YOLOv8n-SMMP: A Lightweight Deep Learning Model for Real-Time Forest Fire Smoke Detection," *Fire*, vol. 8, no. 5, p. 183, 2025. doi: 10.3390/fire8050183.
- [22] W. Ming, Y. Peng, J. Liangcun, Y. Dayu, T. Tianyu, L. Jian. (2024). An open flame and smoke detection dataset for deep learning in remote sensing based fire detection. *Geo-spatial Information Science*. 1-16. 10.1080/10095020.2024.2347922.
- [23] Y. Bahhar, A. Abdellaoui, and M. Benjelloun, "Wildfire and Smoke Detection Using Staged YOLO Model and Ensemble CNN," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 11, 2023. doi: 10.14569/IJACSA.2023.0141122.
- [24] Deng, L., Yang, T., Jin, Z., Wu, S., Liu, Q. (2024). YOLOv8-EMSC: A lightweight fire recognition algorithm for large spaces. *Journal of Safety Science and Resilience*, 5(4), 422–431. <https://doi.org/10.1016/j.jnlssr.2024.06.003>
- [25] Liao, J., Lin, C.-H., & Kuo, S.-Y. (2018). Fire SSD: A lightweight convolutional neural network for real-time fire detection. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 1982–1987). IEEE. <https://doi.org/10.1109/SMC.2018.00336>
- [26] Suo, X., Xu, Y., Yang, J., & Zhang, M. (2023). E³-UAV: Energy-efficient ensemble deep learning for real-time wildfire detection on unmanned aerial vehicles. *Sensors*, 23(5), 2761. <https://doi.org/10.3390/s23052761>
- [27] Xu, R., Lin, H., Lu, K., Cao, L., Liu, Y. (2021). A Forest Fire Detection System Based on Ensemble Learning. *Forests*, 12(2), 217. <https://doi.org/10.3390/f12020217>
- [28] Nguyen, Q. T., Kim, H., Lee, J., & Han, Y. (2023). Challenges in Deep Learning-Based Fire Detection: A Comprehensive Review. *Fire Technology*, 59, 1123–1152. <https://doi.org/10.1007/s10694-022-01254-5>
- [29] Peruzzi, M., Bucci, A., Manzoni, G., Zoppi, M., & Caiti, A. (2023). A LoRa-based wildfire surveillance unit: System design and deployment results. *Sensors*, 23(4), 2021. <https://doi.org/10.3390/s23042021>
- [30] Vázquez, D., García, A., Ramírez, J., & Molina, J. (2025). Generalization of wildfire detection systems using transfer learning and multispectral data. *Applied Sciences*, 15(1), 89. <https://doi.org/10.3390/app15010089>