Early Fire Detection and Segmentation Using Frame Differencing and Deep Learning Algorithms with an Indoor Dataset

John Paul Q. Tomas
MAPUA University
jpqtomas@mapua.edu.ph

Jean Isaiah Dava
MAPUA University
jidava@mymail.mapua.edu.ph

Tia Julienne Espejo
MAPUA University
tjpespejo@mymail.mapua.edu.ph

Hanna Katherine M. Medina
MAPUA University
hmmedina@mymail.mapua.edu.ph

Bonifacio T. Doma
MAPUA University
btdoma@mapua.edu.ph

ABSTRACT

Deep learning models, such as YOLOv5, well-known for object detection, and U-Net, used for segmentation, are known for their respective capabilities within computer vision tasks. In this study, the researchers introduced a novel framework that uses YOLOv5 and U-Net models, in combination with frame differencing techniques, to achieve early fire detection in an indoor setting. YOLOv5 was trained on a diverse dataset consisting of fire, smoke, and non-fire scenarios, while U-Net was exclusively trained on fire data. Motion detection was then implemented using frame differencing that allowed to identify fire movements effectively. The developed framework achieved an overall accuracy of 88%, outperforming the standalone YOLOv5 model with its 81% accuracy. This improvement of 7% in detection performance was influenced by the incorporation of fire motion analysis which effectively reduced false positive results. In summary, the study presents a robust framework that significantly improves fire detection in indoor environments with the help of motion analysis alongside the used deep learning models.

CCS CONCEPTS

• **Computing methodologies**; • **Artificial Intelligence**; • **Computer Vision**; • **Computer Vision Problems**; • **Object Detection**;

KEYWORDS

Deep Learning, Object detection, YOLOv5, Segmentation, U-Net, Early Fire Detection, Indoor Setting, Frame differencing, Motion detection

ACM Reference Format:
John Paul Q. Tomas, Jean Isaiah Dava, Tia Julienne Espejo, Hanna Katherine M. Medina, and Bonifacio T. Doma. 2024. Early Fire Detection and Segmentation Using Frame Differencing and Deep Learning Algorithms with an Indoor Dataset. In *2024 The 8th International Conference on Machine Learning and Soft Computing (ICMLSC 2024), January 26–28, 2024, Singapore, Singapore*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3647750.3647775

ICMLSC 2024, January 26–28, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-1654-6/24/01
https://doi.org/10.1145/3647750.3647775

1 INTRODUCTION

Recent statistics show that about 13k fire incidents occur in the year 2022 causing 700 million worth of property damage. With that said early fire detection models have existed such as traditional alert systems which have limitations due to human interference as well as computer vision which uses color features, texture classification etc. which may result in high accuracy but still be limited in the image's characteristics. However, detecting instances of early fire can be achieved by classifying smoke, the color of fire, and shape through deep learning models as well as the motion of fire through the motion detection method.

The paper [1] which is research done on utilizing both YOLOv5 and U-Net on detecting forest fires scored 71.8% on mean Average Precision on YOLOv5 fire detection while high accuracy on fire segmentation using U-Net which was applied on forest fire datasets. Thus, researchers propose to utilize both YOLOv5 and the U-Net model to determine its performance on fire detection in an indoor scenario. However, color and shape will not be sufficient characteristics to produce an accurate model, therefore, researchers propose to classify through additionally detecting smoke and the motion of fire using deep learning models and frame differencing method.

The purpose of this study is to alter and apply an existing high-accuracy forest fire detection framework [1] to a modified dataset and test the incorporation of motion as a feature to enhance accuracy [2].

The study aims to answer the following: (1) What are the effects of hyperparameter tuning on the performance and robustness of deep learning models like YOLOv5 and U-Net? (2) How did the incorporation of motion feature detection affect the overall performance and effectiveness of the proposed framework?

In this study, researchers propose to develop a new indoor dataset and adapt an existing framework to include motion detection. The primary objective is to optimize the model through hyperparameter tuning and compare its performance with the original framework, as well as classify the presence of early fire through its features such as the presence of smoke, motion of fire, color of fire, and its shape. To achieve the aims of the research the researchers have outlined the following objective statements: (1) To analyze the effects of hyperparameter tuning on deep learning models, aiming to achieve model optimization and efficiency. (2) To assess the impact of motion feature integration on the performance of the proposed framework.

This study proposes an image-based fire detection system using computer vision, which offers more accurate results compared to traditional systems that can produce false alarms [3]. With the increasing use of CCTV surveillance for security, fire detection systems are considered as early fire detectors, provided that they reduce response time, aiding in early fire detection and risk mitigation [4]. This system benefits those people within the area where a fire might take place, providing early fire detection and evacuation warnings for safety [5].

This study focuses on enhancing an existing indoor fire detection system by combining motion detection and deep learning algorithms. It utilizes the frame difference method to identify motion, utilizes YOLOv5 and U-Net models for fire detection and segmentation respectively [1], and adds smoke detection for accuracy in complex environments [6]. Publicly available image and video datasets will be used to create scenarios of fire, smoke and non-fire in indoor settings. To extract fire motion features accurately, videos used for testing would consist of fire, smoke, and non-fire scenarios using stationary cameras to ensure the effectiveness of the frame differencing method.

2 REVIEW OF RELATED LITERATURE

2.1 Indoor/Urban Fire Detection Research

Indoor and urban fire detection studies commonly used YOLOv4 and YOLOv5 as their fire detection algorithm [7, 8] and the used image dataset mainly consist of publicly available fire image dataset from scenarios such as indoor fires, small-scale fires, etc. [7–9]. Different papers have also used other deep learning models such as VGGNet in indoor fire detection [9], but the majority still used some version of the YOLO framework.

2.2 Forest Fire Detection Research

Early detection of forest fire was implemented by utilizing mostly the Corsican Fire Dataset, Self-built dataset, or webcam images or videos to be used for real-time detection [1, 3, 10]. Moreover, it can be observed that the frameworks used consist of deep learning models that were optimized to improve the current model, or add novelty to the framework, with this usage of different classifiers were included in the researcher's model such as the Haar Cascade classifier Algorithm, SVM, etc. [1, 3, 10].

2.3 Smoke and Motion Feature

Additional features of smoke and motion better ensure the timeliness and accuracy of fire warnings in different scenarios. [6] uses fire static feature extraction that is based on color by establishing RGB and HSI criterion models. For the fire dynamic feature extraction, dynamic features of fire are extracted by moving target monitoring technology, particularly using the background difference method. Different background removal and motion detection methods to get the region of interest are widely used such as the ViBe Method [4] and the Frame Differencing Method [11]. The addition of smoke and motion features in detection greatly increase the accuracy of the model as seen in [12].

3 METHODOLOGY

Figure 1 below the conceptual framework consists of three distinct sections, the YOLOv5 section, the U-Net section, and the Proposed Framework section. The first two sections focus on the preparation of the data for training and testing the two deep-learning models used for the framework. Finally, the Proposed Framework section focuses on the creation of the testing set which would be used for evaluation of the framework and the usage of the different models and algorithms in detecting fires. Each section will be further discussed below.

Figure 1 below the conceptual framework consists of three distinct sections, the YOLOv5 section, the U-Net section, and the Proposed Framework section. The first two sections focus on the preparation of the data for training and testing the two deep-learning models used for the framework. Finally, the Proposed Framework section focuses on the creation of the testing set which would be used for evaluation of the framework and the usage of the different models and algorithms in detecting fires. Each section will be further discussed below.

3.1 Dataset

The research utilized a dataset comprising 3,000 images from online sources, including fire, smoke, and non-fire scenes in indoor settings. The dataset was split into a 70-20-10 split strategy with a balanced distribution of images across classes (Fire, Smoke, Fire and Smoke, and Non-Fire) that was based on the study [13]. All images were standardized to a 640x640 resolution [14] to facilitate model training and performance in classification and segmentation tasks.

3.1.1 *Fire Dataset Profile.*

Researchers categorized flames in their dataset based on the colors (yellow, orange, white) observed in the collected images and videos which according to [15] corresponds to its heat and chemical. Based on [15], fire goes through four phases: ignition, growth, steady burning, and decay. Ignition starts the fire, growth makes it spread, steady burning maintains it, and decay is when it decreases due to fuel shortage. Researchers categorized fire size based on these phases. Studies [7] and [16] guided the size categorization: small-scale fires had bounding boxes less than 10%, large-scale fires were over 35%, and medium-scale fires fell between 10% and 34.99% of the bounding box. This led to classifying fire datasets as small, medium, or large as seen in Figure 2.

Medium fires have the most images (550), while large fires have the fewest (458). In terms of color, yellow fires are the most abundant (626), and white fires are the least represented (403). Overall, the dataset contains 1,500 fire images across these profiles, enabling a comprehensive analysis of the model's performance in different fire scenarios.

3.1.2 *Smoke Dataset Profile.*

Following [17], which describes the types of smoke and the combustion gases affecting its color, researchers categorized the smoke dataset into light, gray, and dark smoke based on the shades observed. Figure 3 shows the different categories of smoke color used in the study.

Similarly, to the fire profile and [16, 18] researchers adopted a size of small, medium, and large. Large smoke instances have the most images (800), while small instances have the fewest (444) based on size. In terms of color, light smoke instances have the most images (867), and dark smoke instances are the least represented (258). In total, the dataset contains 1,500 smoke images across these profiles,

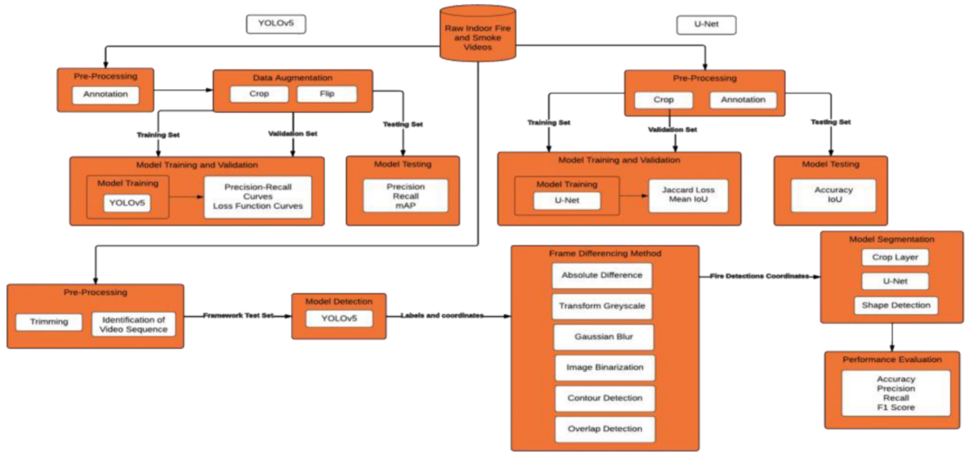

Figure 1: Conceptual Framework

Figure 2: Sample Fire Images Based on Size

Figure 3: Sample Smoke Images Based on Color

providing insights into how the model performs in different smoke scenarios.

3.2 Data Acquisition

A dataset comprising fire, smoke, and non-fire images and videos in indoor settings, was acquired from online databases and recorded at the National Fire Training Institute. This raw data was processed and transformed into specialized datasets for the models in the study.

3.2.1 Data Pre-Processing. Pre-processing involved isolating relevant fire segments from the raw indoor fire and smoke video datasets. These segments were converted into images at a 1-frame-per-second rate. For YOLOv5, images were annotated with object labels and bounding boxes for fire and smoke instances. For U-Net, fire images were cropped to the region of interest of the object and resized images to 640x640 before segmentation with no data augmentation for data integrity.

3.2.2 Data Augmentation. To enhance model performance and reduce the occurrence of overfitting, data augmentation techniques were applied. For this study, image mosaic and horizontal flipping techniques were selected to align with the nature of fire flames. An example of the Mosaic Augmentation can be seen on the left part of Figure 4 and the Flipping Augmentation Technique on the right side.

To ensure that the data augmentation properly produced different images, the SSIM score is looked at the different augmentations done on the image. SSIM values determine the retention of augmentation techniques wherein as SSIM approaches one, it indicates minimal differences between images. Results show, that the Mosaic augmentation technique will be retained due to a 0.29 SSIM Value, while the Flipping augmentation technique will be removed due to a 0.75 SSIM Value.

3.3 Model Training

3.3.1 YOLOv5. To obtain an optimized YOLOv5 model capable of detecting early indoor fire instances, a training procedure was done aiming to accurately detect fire on video datasets which is set up using Google Colab with T4 GPU. Diverse video and image datasets were collected consisting of different classes such as fire, non-fire, and smoke datasets which are then split into training, validation and test sets. Researchers adjusted hyperparameters such as batch size, epochs, image size, and learning rate, similar to [7, 14]. The model was trained over the dataset in a consistent epoch but with modifications in the dataset, such as increasing the data through the addition of false positives and removing images that cause inaccuracies. Through this training procedure, optimizations were done on the model as researchers assessed the progress during the validation phase. Furthermore, evaluating the model's performance on a validation dataset was done. YOLOv5 uses a 20% validation dataset with annotations to evaluate performance, using metrics like Precision, Recall, and mAP. Optimization involves modifying the training dataset by adding false positive images, removing incorrectly labeled ones, adjusting learning rates, and epoch sizes to enhance model performance and detection accuracy during validation.

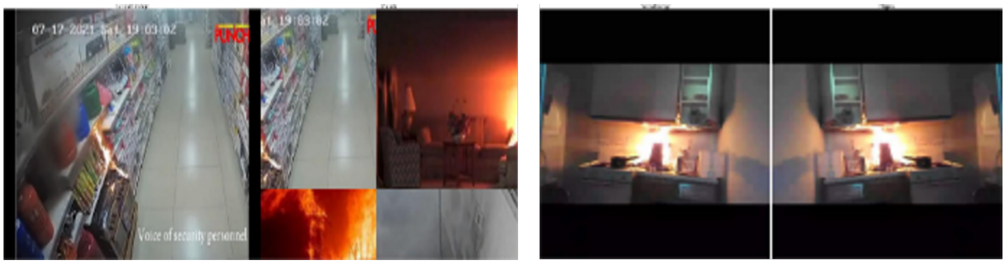

Figure 4: Mosaic & Flipping Augmentation

3.3.2 U-Net. Researchers conducted the U-Net training using Colab Pro with the T4 GPU and High-Ram wherein the training consists of subjecting the training dataset to different types of loss functions with consistent epochs, batch size, and learning rates hyperparameters and from there determining the best possible loss function to utilize. Overall, researchers experimented with various loss functions and adjusted the hyperparameters along the way to achieve notable segmentation results. Researchers monitored the model and made adjustments in the learning rate, epoch, and loss function during the validation phase based on its metric results. The validation process for the U-Net architecture consists of the validation dataset which comprises cropped fire images with a corresponding ground truth binary mask. Jaccard Index or IOU is utilized to measure the performance of the segmentation model, as it measures the similarity among the ground truth image and target segmented image, thus evaluating segmentation tasks with the equation as follows:

$$\begin{aligned} JL &= 1 - J\,(Gt, Pr) \\ &= 1 - \frac{Gt \cap Pr}{Gt \cup Pr}. \end{aligned} \tag{1}$$

3.4 Framework Development

The overall framework uses deep learning models for fire detection and fire segmentation respectively, it also uses the Frame Differencing Method in order to verify the detection results of the deep learning model. The framework will be tested using a different dataset containing video clips that are curated by the researchers.

3.4.1 Testing Videos. The performance of the proposed method is evaluated using video datasets of fire, smoking and non-fire seen from an indoor scenario acquired from online and researcher-generated videos from local fire departments in indoor settings. Videos will be pre-processed such as video curation and trimming wherein, a total of 100 videos, each lasting 5 seconds, were used as the test dataset. Overall, 50 videos were allocated to testing fire, 20 videos were for smoke, and the remaining 30 videos were for non-fire.

3.4.2 YOLOv5. Firstly, the YOLOv5 model will create bounding box detections based on what it detects as fire and smoke in the test videos from the dataset. The model will output a detection video created by YOLOv5 and different text files for each frame of the video with the corresponding bounding box and class of the detection on that frame. A sample YOLOv5 Detection is seen below, the input frames and output frames can be seen in Figure 5 below.

Figure 5: Sample YOLOv5 Detection

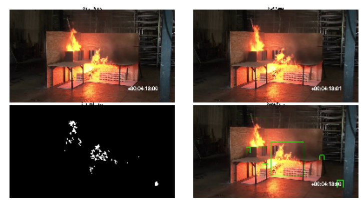

Figure 6: Sample Frame Differencing Method

3.4.3 Frame Differencing Method. The motion feature of the video would be extracted using the Frame Differencing Method by getting the absolute difference of the pixel values between two consecutive frames after which Gaussian blur and binarization image would be done to extract the motion feature from the frames [19]. The bounding box location of the motion is then saved and used to compare to the bounding box of the YOLOv5 detections. To find the best thresholds for the image transformation techniques, experimentation would be done [20].

The bounding box of the two YOLOv5 and Frame differencing must overlap to ensure there is fire detected. A sample of the frame differencing method can be seen in Figure 6 below, wherein the two consecutive frames are taken as an input and processed to create a binarized image and bounding box of the motion feature. Each frame would be tested by the framework to check the detections.

3.4.4 U-Net. The resulting bounding box would be cropped and resized and be fed into the U-NET model to localize the presence of fire and detect its precise location on the cropped images which would result in a binary mask of fire pixels, given these bounding lines would be applied on the precise location of the original images to enable shape [1]. Figure 7 below presents a U-Net segmentation result, showing shape detection.

3.5 Evaluation Metrics

To evaluate the performance of the framework, metrics such as confusion matrix will be done to determine Accuracy, Precision,

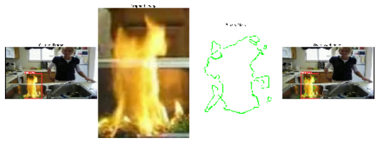

Figure 7: Sample U-Net Segmentation

Recall, and F1-Score values of the framework. Also, the overall Accuracy of the framework can be evaluated by calculating all the true positives determined in all three classes over the entire instances, thus having the following equation:

$$\begin{aligned} &Overall\ Accuracy \\ &= Overall\ Accuracy \\ &= \frac{AA+BB+CC}{Total\ Instances} \end{aligned} \tag{2}$$

Other than this, the framework is also evaluated using 5 number of videos that show the continuous phases of fire, where researchers evaluated whether the model could detect fire in its Ignition Phase, Growth Phase, and Steady Burning.

4 RESULTS

4.1 YOLOv5

Researchers aimed to determine the best parameter and found that reducing the learning rate from 0.03 to 0.01 improved performance. A lower learning rate allows the model to update its weights more slowly and provides more stability during training for the model to learn more effectively. Adjusting the batch size from 12 to 16 had minimal improvements while increasing the epoch size had significant results. The parameters that significantly improved YOLOv5's object detection performance are a lower learning rate and a higher epoch size, thus utilizing a learning rate of 0.01, batch size of 16 and an epoch size of 250. The model was tested on unseen data, and the results in Table 1 revealed that precision for fire instances was high at 0.882, while precision for smoke was lower due to its complex shape and occurrence. The model excelled in classifying fire and presented challenges in detecting smoke instances, especially in overlapping scenarios. Despite this, the optimized YOLOv5 model consistently showed its ability to detect both fire and smoke instances in various scenarios.

4.2 U-Net

Researchers observed the slower learning rate allowed for fine-grained weight adjustments and was effective in preventing overfitting and facilitating better generalization of the data. Further, as the epochs of the model increased to 50 epochs metrics and segmentation performance improved. However, through a further increase in epochs to 100, some metrics improved but segmentation performance decreased. It is also observed that using low learning rates on large epoch sizes did not further improve the model but increased training error due to overfitting as the capability of the model to memorize the data can occur especially when trained on the minimal dataset. Given that the learning rates utilized in the runs of this phase are consistently a learning rate of 1e-6 with an epoch size of 50. Table 2 shows the researcher's tuning on a different loss function such as Dice Loss and Jaccard Loss, which evidently shows a higher IoU metric value on Run 10 which utilizes Jaccard Loss.

A higher IoU pertains to better segmentation performance, thus showing Run 10 to have better segmentation performance than Run 7 with only 0.3993, thus concluding that Jaccard Loss performed better than Dice Loss. Although, utilizing Jaccard Loss resulted in an IoU of 0.60 still indicates that its prediction is able to overlap with the ground truth to some degree although not fully, thus emphasizing that further improvements can still be made to improve its performance.

4.3 Developed Framework

The developed framework has undergone parameter tuning for the different parameters such as the confidence levels of the classifier, the threshold values for the binarization of the images and the threshold values for the contour area. Each of these parameters is tested and experimented upon to find the best combination for detecting fires and smoke. For the confidence value of the classifier model, three confidence levels are tested, 0.1, 0.2, and 0.3 specifically. The 0.1 confidence reaching an 87% accuracy, the 0.2 reaching 86%, and the 0.3 reaching 85% accuracy.

Since the best parameter for the confidence values was already discovered, the different thresholding values for the binarization of the image and for the contour area would be adjusted next. These two thresholds affect the detection of the movement from the frame differencing method. For the Binary Image Thresholds, 4 threshold groups were used namely the, Very High, High, Normal, and Low Spreads. This would create a clear binary image to determine movement. The higher the range of the threshold, the more sensitive the framework is to movements. As seen in Figure 8 below, as the threshold range widens, a better binary image is produced as the framework is it becomes more sensitive to movement. A better binary image in which the motion feature is extracted would greatly affect the accuracy of the developed framework.

Additionally, the threshold for the contour area is tested with 1000 and 500 sizes which has a minimal impact on the performance of the developed framework. The Binary Image Threshold has a great impact on the overall accuracy of our framework. A difference of 9% accuracy between the worst and best-performing frameworks

Table 1: YOLOv5 Testing Results

Class	Images	Instances	Precision	Recall	mAP50	mAP50-95
ALL	300	405	0.853	0.691	0.773	0.523
Fire	300	253	0.882	0.751	0.844	0.56
Smoke	300	152	0.824	0.632	0.702	0.486

Table 2: Loss Function Trial Results

Run #	Learning Rate	Epochs	Loss Function	Accuracy	Test Loss	Test Accuracy	IOU
7	1e-6	50	Dice	0.80	0.52	0.75	0.3993
10	1e-6	50	Jaccard	0.77	0.72	0.72	0.60

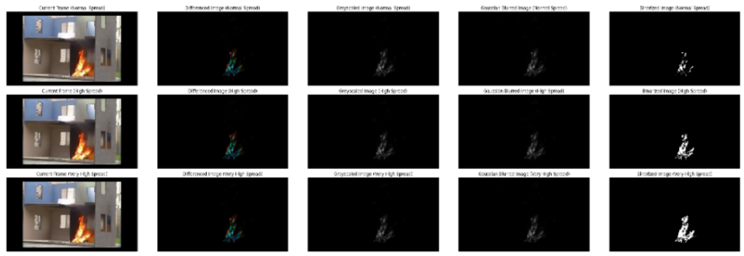

Figure 8: Motion Extraction of Different Spreads

Table 3: Optimized Developed Framework Parameters

Threshold Values for Binary Image	Threshold Values for Contour Area	YOLOv5 Confidence	Overall Accuracy
22-255	1000	0.1	88%

Table 4: YOLOv5 and Developed Framework Test Accuracy Scores

	True Positives	Overall Accuracy
YOLOv5	81	81%
Proposed Framework	88	88%

was identified. The optimized parameters for the framework can be seen in Table 3 below.

Table 4 below provides the results of the YOLOv5 and the developed framework in terms of True Positives and Overall Accuracy metrics that are based on testing with a dataset of 100 videos. In terms of True Positives, the developed framework outperforms the YOLOv5 model, with 88 true positives compared to 81. This is an important finding because True Positives represent the number of correctly identified instances of the target classes: fire, smoke and non-fire. A higher number of True Positives indicates that the proposed framework is more effective in correctly detecting these instances in the video dataset. Both the YOLOv5 model and the developed framework achieved impressive overall accuracies of 81% and 88%, respectively. This means that they are effective at correctly classifying the majority of the video frames. However, the developed framework achieves a slightly higher accuracy rate of 7%, suggesting that the integration frame differencing method for motion detection contributes to the improvement in the overall classification performance.

Evidently, the developed framework shows good performance as it was able to detect in its Growth phase and steady Burning phase, given the YOLOv5 model was trained mostly on video image frames wherein a fire is already burning in a steady or the fire exponentially increasing in its growth. Figure 9 below shows the framework's performance in detecting fire its different phases such as ignition, growth, and steady burning.

5 SUMMARY, CONCLUSIONS, RECOMMENDATIONS

5.1 Summary

In this research study researchers combines deep learning models such as Yolov5 and U-Net models with frame differencing method for motion feature detection capable of detecting indoor fire scenarios. Researchers also adjusted the different hyperparameters of Yolov5 and U-Net wherein Yolov5 was able to output average results with an mAP50 of 0.773. in the overall classes and the U-Net model with a 0.60 IoU value. Researchers also compared the performance of the proposed framework and yolov5 using the framework 100 videos test dataset with 5 second duration and continuous fire

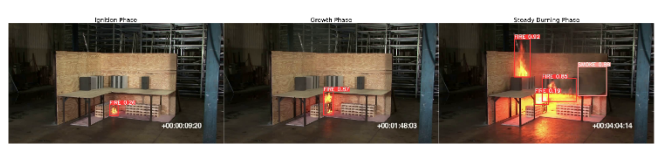

Figure 9: Sample Detection of Framework on Different Phases of Fires

videos, wherein researchers determined that fire could detect fire in its different phases. Ultimately, Yolov5 was only able to output an overall accuracy of 0.81, which is lower than the results of the proposed framework resulting in an overall accuracy of 0.88.

5.2 Conclusion

In conclusion, the study combining YOLOv5 and U-Net with frame differencing outperformed the standalone YOLOv5 model. Through thorough parameter exploration and fine-tuning, both YOLOv5 and the framework resulted in effective indoor fire and smoke detection. Incorporating motion features improved accuracy by considering fire movement alongside color and light characteristics. However minor performance variations were observed in specific categories like medium orange fire and medium gray smoke. YOLOv5 had moderate scores for smoke detection due to its complex features and a better prediction performance in detecting fire instances. U-Net faced difficulties due to the limited size of the fire dataset for precise fire shape segmentation.

5.3 Recommendation

To enhance the performance of the developed YOLOv5 model and U-Net framework for indoor fire and smoke detection, expanding the dataset is essential by using a wider array of images and videos and ensuring a more balanced representation of fire and smoke categories. For categories with low prediction scores, optimization techniques can be used. In smoke detection, a thorough exploration of feature extraction methods is recommended. Additionally, increasing image data in the U-Net model's training set and exploring experimentation with different losses may raise Intersection over Union (IoU) values. Finally, implementing a sampling technique for improvement can significantly improve the overall performance of the developed framework to make it a more effective system for indoor fire and smoke detection.

REFERENCES

[1] Mseddi, W.S., et. al. 2021. Fire Detection and Segmentation using YOLOv5 and U-NET. (2021). DOI: http://dx.doi.org/10.23919/EUSIPCO54536.2021.9616026
[2] Shiwei Wang, Feng Yu, Changlong Zhou, and Minghua Jiang. 2020. Straw burning detection method based on improved frame difference method and deep learning. 2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC) (2020). DOI: http://dx.doi.org/10.1109/icivc50857.2020.9177456
[3] Kumar, K., Varma, K., Sujihelen, L., Jancy, S., Aishwarya, R. and Yogitha, R., 2022. Computer Vision-Based Early Fire Detection Using Machine Learning. 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT). DOI: http://dx.doi.org/10.1109/IC3IOT53935.2022.9767886
[4] Sthevanie, F., Kosala, G., Ramadhani, K.N., and Putra, K.S. (2021) Fire Detection on Video Using ViBe Algorithm and LBP-TOP. Jurnal RESTI: Rekayasa Sistem dan Teknologi Informasi. DOI: https://doi.org/10.29207/resti.v6i6.4164
[5] Ryu, J. and Kwak, D., 2021. Flame Detection Using Appearance-Based Pre-Processing and Convolutional Neural Network. Applied Sciences, [online] 11(11), p.5138. DOI: http://dx.doi.org/10.3390/app11115138
[6] Zhang, B., Sun, L., Song, Y., Shao, W., Guo, Y. and Yuan F. (2020) DeepFireNet: A real-time video fire detection method based on multi-feature fusion. Aimpress. Mathematical Biosciences and Engineering. DOI: 10.3934/mbe.2020397
[7] Hongyu, H., Ping, K., Fan, L., Huaxin, S. 2020. An Improved Multi-Scale Fire Detection Method based on Convolutional Neural Network. (2020). DOI: http://dx.doi.org/10.1109/ICCWAMTIP51612.2020.9317360
[8] An, Q., Chen, X., Zhang, J., Shi, R., Yang, Y. and Huang, W., 2022. A Robust Fire Detection Model via Convolution Neural Networks for Intelligent Robot Vision Sensing. Sensors, [online] 22(8), p.2929. DOI: http://dx.doi.org/10.3390/s22082929
[9] Mwedzi N.A., Nwulu N.I., and Gbadamosi S.L. 2019. Machine Learning Applications for Fire Detection in a Residential Building. IEEE. [Online]. DOI: http://dx.doi.org/10.1109/ICETAS48360.2019.9117318
[10] Chen, K., et. al. 2019. Research on Image Fire Detection Based on Support Vector Machine. (2019). DOI: http://dx.doi.org/10.1109/ICFSFPE48751.2019.9055795
[11] K. Chen, Y. Cheng, H. Bai, C. Mou and Y. Zhang, "Research on Image Fire Detection Based on Support Vector Machine," 2019 9th International Conference on Fire Science and Fire Protection Engineering (ICFSFPE), Chengdu, China, 2019, pp. 1-7, doi: 10.1109/ICFSFPE48751.2019.9055795
[12] Guohua Wang, Juncong Li, Yongsen Zheng, Qi Long, and Wanrong Gu. 2020. Forest smoke detection based on Deep Learning and background modeling. 2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS) (2020). DOI: http://dx.doi.org/10.1109/icpics50287.2020.9202287
[13] Gonçalo Perrolas, Milad Niknejad, Ricardo Ribeiro, and Alexandre Bernardino. 2022. Scalable fire and smoke segmentation from aerial images using convolutional neural networks and quad-Tree Search. Sensors 22, 5 (2022), 1701. DOI:http://dx.doi.org/10.3390/s22051701
[14] Xue, Z., Lin, H. and Wang, F., 2022. A Small Target Forest Fire Detection Model Based on YOLOv5 Improvement. Forests, [online] 13(8), p.1332. DOI: http://dx.doi.org/10.3390/f13081332
[15] Hurley, Morgan J. and Richard W. Bukowski. "Fire Hazard Analysis Techniques." (2019).
[16] Sida Dai, Tao Han, Mingchuan Yang, Yuan Zhang, and Lichuan Wang. 2022. Towards Small Scale Urban Fire Detection Dataset. 2022 IEEE 9th International Conference on Cyber Security and Cloud Computing (CSCloud)/2022 IEEE 8th International Conference on Edge Computing and Scalable Cloud (EdgeCom) (2022). DOI:http://dx.doi.org/10.1109/cscloud-edgecom54986.2022.00026
[17] Yuanpan Zheng, Zhenyu Wang, Boyang Xu, and Yiqing Niu. 2022. Multi-scale semantic segmentation for fire smoke image based on global information and U-Net. Electronics 11, 17 (2022), 2718. DOI: http://dx.doi.org/10.3390/electronics11172718
[18] Yafei Gong, Xuanchao Ma, Hongyan Liu, and Ke Gu. 2022. Multiple categories of visual smoke detection database. 2022 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS) (2022). DOI:http://dx.doi.org/10.1109/ispacs57703.2022.10082827
[19] Yessi Jusman, Lentera Hinggis, Rama Okta Wiyagi, Nor Ashidi Mat Isa, and Faaris Mujaahid. 2020. Comparison of Background Subtraction and Frame Differencing Methods for Indoor Moving Object Detection. 2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE) (2020). DOI:https://doi.org/10.1109/icitamee50454.2020.9398484
[20] Faming Gong *et al.* 2019. A real-time fire detection method from video with Multifeature Fusion. Computational Intelligence and Neuroscience 2019 (2019), 1–17. DOI:http://dx.doi.org/10.1155/2019/1939171