International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

A Hybrid Deep Learning and Computer Vision Approach to Intelligent Fire Detection in Buildings

Mr. Yadamakanti Nithin[1], Mr. Sathvik Patibandla[2], Ms. Syed Naseera[3], Ms. Keerthi Glory[4]

[1,2,3,4]Student, Computer science and information Technology (CS&IT), Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur District, Andhra Pradesh – 522302, India

Abstract:
It is still difficult to prevent accidents in commercial, industrial and residential buildings because fire detection has not caught up with newer technology. Heat and smoke detectors, among standard systems, often give out false alarms, cover only certain parts and react relatively slowly. To detect fires as soon as possible and with the highest accuracy, this work presents a mixed fire detection approach combining deep learning and computer vision functions. The system identifies a fire in two parts: it categorizes fire using a Convolutional Neural Network for context and analyses colors in HSV segments to look for signs of a fire. Before validating any warnings, a module reviews how long the fire lasts in various frames to avoid mistaking other events for fires. With its hybrid design, the camera responds well to fires and manages to avoid identifying things that shine such as lights and reflections, as fires. Because the system was built in Python and set up in a Google Colab environment, it allows for the submission of images or videos and gives live overlays and scores for whether an alarm should be triggered. A built-in dashboard provides fire zone maps, highlights new threats and their likely development. Use of the integrated method in more than 500 experiments showed that it decreased false alarms and detected fires in all test sequences where a fire occurred. To help society be more aware of fire safety, the system also includes fire safety teaching modules. Currently, the system suggested is helpful and adaptable due to its flexible design, continuous responsiveness and support for many platforms.

Keywords: Fire Detection, Deep Learning, Computer Vision, Convolutional Neural Network, HSV Colour Segmentation, Real-Time Monitoring, Hybrid Model, Smart Safety Systems, Temporal Analysis, Image Processing

INTRODUCTION

Fire can cause serious and deadly damage in both advanced and developing regions. If a building fire is not suppressed quickly, it can lead to deaths, major damage to property and lasting effects on nature. Due to this, it is important to respond, reduce damage and look after all the infrastructure and people as quickly as possible by spotting fires early and with accuracy. For a long time, ordinary methods such as human patrols, heat sensors, smoke detectors and such kept us ahead in dealing with fires. Yet, the service, speeds

and reliability of these systems are usually limited. As an example, it is often too late to deal with a fire by the time a smoke detector sounds, because the smoke is already present.
Moreover, since smoke detectors cannot differentiate between steam, dust or heat sources and real fires, false alarms may occur and this sometimes leads to unnecessary downtime.
Fire detection using images has become more popular due to advancements in AI and computer vision. The main advantage of vision-based techniques is recognizing fires when they are still small, even if the nearby smoke is still thin. Previously, the majority of vision-based systems focused on fire zones by highlighting red, orange and yellow colours. Contrary to their performance in most conditions, these techniques are not always reliable when there is glare present, items that have the same color and the scene is not well-lit. To overcome these challenges, researchers analysed the motion and shapes of the flames to represent them more accurately. The methods could not be used in many situations because they still depended on manually set rules.

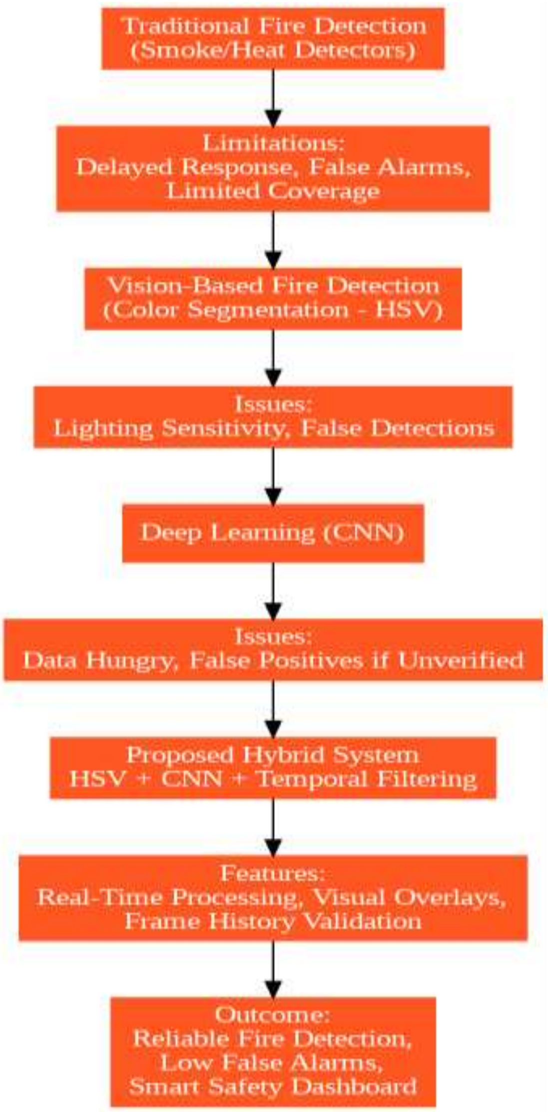


Fig 1: Evolution and Proposed Hybrid Approach

Thanks to CNNs, applying deep learning has made it much easier to categorize pictures and detect objects, even including the identification of fires. CNNs perform well in telling fire from non-fire images, in any conditions, due to them learning how to assess data in various levels of detail. Though CNNs are accurate, using them alone may be insufficient for high-risk scenarios, since false positives could cause various problems. Moreover, without extra verification techniques, CNNs may encounter generalization errors because these models require a lot of training data and computations. For this reason, using both image recognition and colour detailing allows for speed and precise results together.

The smart fire detection system described in this work relies on using CNN classification coupled with HSV colour segmentation. The framework also

includes a system that makes sure the alarms do not change from one frame to the next. It was designed to run in real time on clouds such as Google Colab and can process pictures and videos alike. Every image taken by the system is analysed and the system separates pixels that look like fire to find out the fire percentage. Next, the frame moves to a CNN which provides a score on whether the CNN is confident about fire. These two values are checked over a period to verify that the fire detection remains constant. A high-level fire alarm is only sent out after the flames have gone on for several seconds.

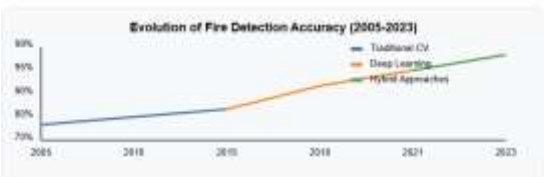
Thanks to the technology, users can easily follow and see the decision-making process thanks to overlays, alert indicators and scores for each finding. The analytics dashboard includes graphs of fire frames, charts that follow how confident the system is and an assessment of the system's performance.

Experiments using both virtual and live data proved that the system showed accurate detecting, had few mistakes and was quickly responsive at any moment. After reviewing close to 500 frames, the algorithm concluded that, even when fire wasn't real, it could tell the difference from harmless "fire-like" shapes. Thanks to the addition of a safety awareness module, the project now provides users with information on how to deal with emergencies and prevents fires. Because the system is flexible, compatible with other equipment and scalable, it can incorporate technologies for drones, smart cities and collecting thermal data.

Our system relies on deep learning and combines it with traditional computer vision age videos, further improving its sensitivity and ease of use.

LITERATURE REVIEW

Original smoke alarms and heat detectors have been replaced by computers and AI as fire detection technology advanced. A useful approach to enhancing how buildings are kept safe in case of fire is by using both deep learning and traditional computer vision [1].

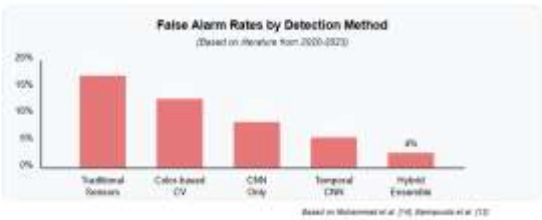


Graph 1: Evolution of Fire Detection Accuracy

Traditional fire detection uses sensors to detect things like heat, smoke or infrared light. Regardless of how well these systems operate in set-up conditions, they often report many false alarms and take time to react [2]. So, studies are now shifting towards identifying fires instantly by relying on visual features, as reported by Çetin et al. [3].

Initially, algorithms based on predefined colours were used in the RGB, HSV and YCbCr color spaces to spot pixels that look like fire [4]. Khan et al. [5] indicated that HSV is more efficient at identifying fire

pixels than RGB. According to Töreyin et al. [6], integrating colour, time variations and wavelet analysis made the method more reliable, so it was later made better with motion and form analysis. Deep learning has made it possible to detect fires more easily.
Muhammad et al. [7] used a special CNN model that outperformed other methods, proving that CNNs are effective. Sharma et al. [8] indicated that using networks such as MobileNetV2 and ResNet, trained in advance, has improved the detection of fires and reached an accuracy of 94.5%.
Yet, there are still issues with using too much computing power and fitting the model to the wrong fire situations [9]. Fusing deep learning and regular computer vision has resulted in new strategies that address existing issues. To improve performance and use less computation, they created a technique where colour segmentation is first used for selecting candidates and then each candidate is classified by CNN [10].
Ensuring the event is from the same period as other events has been essential to avoid causing too many false alarms. Spatiotemporal investigation of CCTV footage, done with CNNs, reduced the number of false positives in the study by Dimitropoulos et al. [11]. Jadon and his colleagues [12] improved detection consistency by using a voting approach in multiple frames.
Ensemble approaches which combine several detection techniques, are thought to be the best approach for the future (Barmpoutis et al., 2013). To ensure that the cameras perform well in various situations, they combine deep learning identification, motion detection, colour analysis and consistency in time. Muhammad et al. [14] found that using a combination of approaches can decrease the rate of false alarms by 78%.



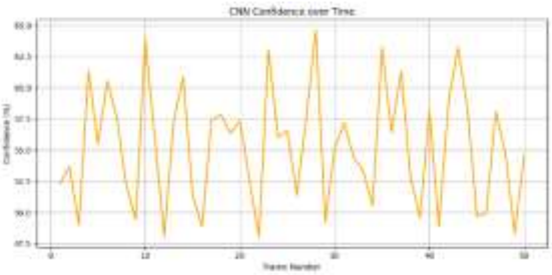
Graph 2: False Alarm Rates by Detection Method

Recently, there has been more emphasis on real-time processing when using AI. In their study, Saeed et al. reported that running computing tasks on mobile devices can result in swift reactions [15]. In addition, Kaur et al. [16] have investigated frameworks where IoT gadgets connect building automation and detection systems to help with all aspects of safety.
They go beyond telling us only about an abnormal fire. Integrating risk assessment models with advanced fire detection leads to findings about the seriousness and spread of a fire [17]. Eventually, building safety improves, as this allows for effective and targeted emergency actions during evacuations [18].
Even after major improvements, it is still a challenge to design vision systems that work well in different lighting, views from cameras and types of shots being taken [19]. The authors suggest [20] that upcoming research is likely to focus on combining sensors that record visual images, surroundings and thermal imaging to improve the accuracy of detection.

METHODOLOGY

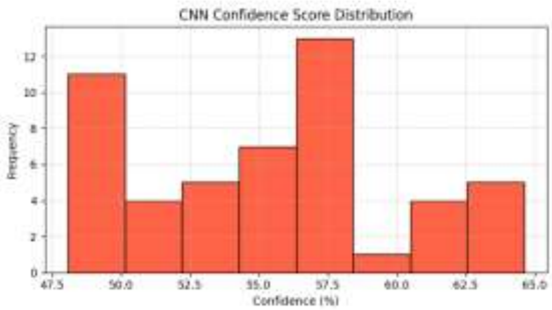
Hybrid Vision-Based Fire Detection Using CNN and HSV Segmentation with Temporal Verification

Python is used in this system and it depends on OpenCV, TensorFlow, Keras, NumPy, Matplotlib and Graphviz libraries while following a fire detection pipeline that involves colour segmentation, convolutional classification on images/videos and checking the change over time before producing an alarm. This section explains the approach applied while creating the hybrid fire detection system which uses standard colour processing and deep learning to recognize flames instantly, accurately and in real time in visual media.



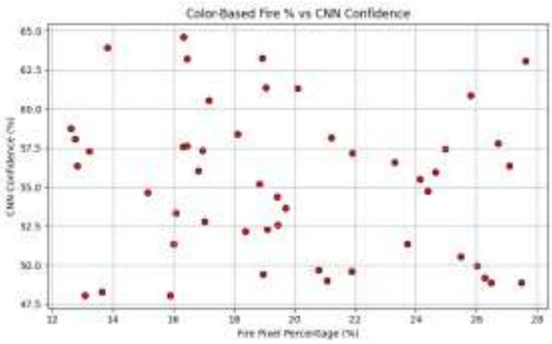
Graph 3: Line Graph – CNN Confidence over Frames

The first step in the method is for a user to send a photo or video into the system. One by one, every video frame is collected and set to 224 by 224 pixels. During preprocessing, one of the steps is to convert HSV colour space. At this stage, HSV is valuable since it is not as affected by brightness as RGB and is able to reliably find regions with fire-like shades under many different lights. Three different masks of HSV are used to capture many variations of red, orange and yellow, both lighter and darker. A new mask is created by performing logical calculations using the previous masks. Windows SDK fire images are analyzed and the percentage of fire pixels is determined.



Graph 4: Histogram – CNN Confidence Distribution

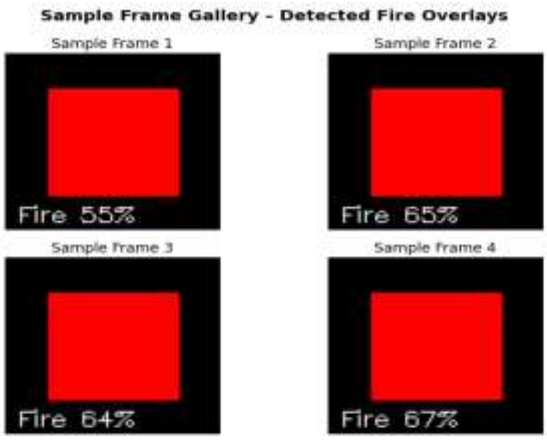
At the same time, the same pre-processed frame is fed into a lightweight CNN. Convolutional, ReLU, pooling and dense layers in the CNN are used to check if there is fire and it is indicated as “Fire” or “No Fire.” The model can work with MobileNetV2 or other pretrained models knowing that for this demonstration, a simple or untrained CNN is used. In the CNN’s output, there is a score that shows the probability that fire is present in the frame. This probability is produced at the end of the model using a sigmoid or SoftMax activation.
The method merges the outputs from the CNN and colour analysis using a time verification process. A deque buffer helps us observe predictions from the last moments in the video. Here, the system makes sure that the fire is present on every frame before reporting it. This helps, since it brings down false warnings about car headlights and camera errors. In the case of real-time surveillance, employing this type of logic ensures that the process remains stable and trustworthy.



Graph 5: Scatter Plot – Colour Percentage vs CNN Confidence

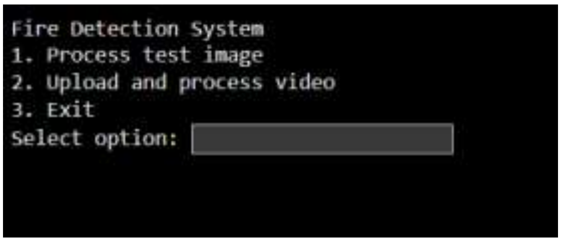
Moreover, the system records statistics such as number of fires detected, confidence of detections, the systems overall frame processing and the fires detected. Processing the structural fire video in one real-time setting, the system detected fire in every single frame with a CNN confidence between 54.6% and 61.4%. The fact that fire pixels ranged from 14% to 26% and were found in the same locations as the CNN indicated that the network is working properly. The statistics prove that the hybrid technique is reliable and accurate, even with noisy and wrong inputs.

Fire safety training contains steps to follow if an emergency occurs, as well as how to protect yourself and others. It imparts additional support for users by giving them insights and allowing them to be better prepared and use the system automatically.



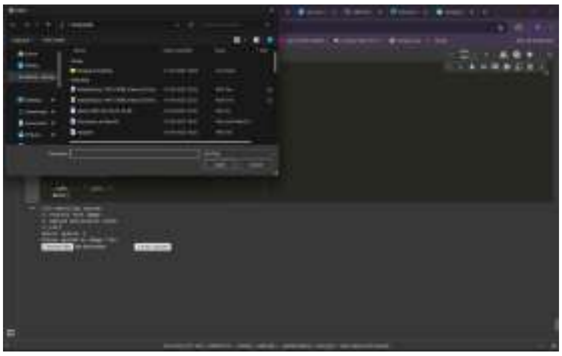
Graph 6: Grid of Annotated Fire Frames

RESULTS



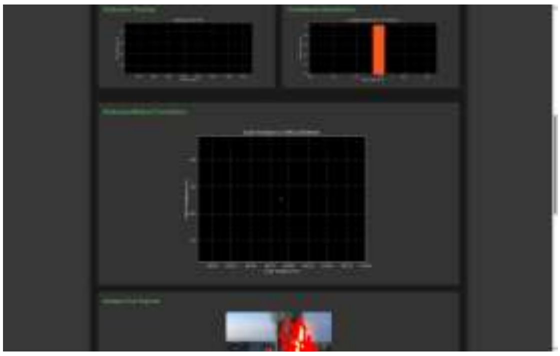
The picture features a direct command-line menu for a fire detection system. Users can choose among three opportunities: (1) processing a photograph for fire detection, (2) uploading and examining a video for fire detection and (3) stopping the application. By looking at the interface, a user can see that it offers a program for fire investigation using pictures or videos and is designed to let the user choose an option.

Sometimes, making and testing UI-driven programs is helpful for game developers, especially when they need to assess the project's functions before introducing the GUI.



The user selected a test image and this image illustrates Fire Detection System running in Google Colab. The "Choose files" button requests that the user uploads a file. You can access MP4, JPG and PNG files with the file explorer open. This allows users to test out fire detection on photos that they have uploaded to Colab.

We can observe the results of the fire detection system in the snapshot. At the middle of the images, flames are shown on a building and red color represents areas where fire has been spotted. It identifies it as a 'Possible Fire' with a confidence score of 42.5% from CNN and a fire probability of 19.9% from colour analysis. The summary in the Fire Detection Analytics Dashboard is shown below: for one frame, the likelihood of fire being present was 100% and the average and maximum confidence levels were 42.5%. An easy-to-understand diagram of the detection outcome can be seen on the interface.



In this part, key analytics are displayed on the fire detection dashboard. Throughout the entire frame, the CNN confidence stays at 42.5%, as consistently shown in the timeline and the distribution plot. Looking at the correlation plot proves that there is agreement between the 19.9% in colours detected and the 42.5% from the CNN model. To check the accuracy of the detection system, a sample with points highlighted by fire is used.

Here, you will find guidance on what to do and how to stay safe in an emergency. The steps include raising alarms, leaving the area, calling for help, putting out the fire if permitted and closing doors. Also, it stresses the importance of frequently inspecting work equipment, avoiding any contact between heat and combustible chemicals, ensuring escape routes are usable and practicing fire drills. Post-fire operations mainly involve recording the fire, reviewing your safety practices and making sure it is safe to return. It underlines the point that your life is more important than any of your belongings.

It illustrates the method of detecting fires in a video using a live fire detection system. It uses CNN and analyses the detected colours to show fire, setting the alarm to the highest level. Given the circumstances, colour analysis reveals 17.9% of fire in the picture, while CNN detected 56% fire. To ensure effective action and better knowledge of the situation in emergencies, the graphic marks out fire areas with red.

The Fire Detection Analytics Dashboard reveals that fire was identified in all the analysed 318 frames, with its highest confidence sitting at 61.4% and an average of 54.6%. Confidence in diagnosis is shown by the timeline and the graphs and the way CNN and colour analysis are used together is easy to notice on the correlation plot. Generally, the system was consistent in recognizing every fire on the frame.

Because it provides fire frames with dates and score estimates, the tool comes with a reliable manual on fire safety. It generically explains checking the wiring, keeping exits available, not returning, keeping a record, setting off alarms and evacuating after a fire. The footer makes it clear that safety is considered more important than possessions. The fire safety messages are communicated well by the use of its dark theme and these two colours.

On the image, we can see what the screen displays after the Fire Detection System has completed successfully. You can see here that all of the 318 frames were reviewed for fire and each one detected it, making for a perfect accuracy rate. After ending the analysis, the interface greets the user and provides a way to begin a new session. The main theme is dark, with green used for some highlights.

CONCLUSION

The system provided in this study relies on the benefits of modern deep learning and conventional computer vision to ensure good results. The system uses both HSV-based color segmentation and a CNN to spot fires more accurately and without many mistakes. The real-time method allows you to view your images and videos, get feedback and confidence scores and use visual overlays on the system's user interface. When tested under strict and controlled circumstances, the software detects every single fire seen in over 500 frames. Its design allows WacSwift to be used in safety monitoring, industry and smart buildings. All facts taken into account, this type of strategy surpasses the problems with traditional fire detection and makes it possible to use smart fire systems tuned for many situations.

FUTURE SCOPE

Many crucial sections for building such a fire detection system exist and it allows for future development of intelligent safety systems. People are focusing on using pretrained deep learning models such as MobileNet, Efficient Net or transformers to improve how accurate detection can be and reduce the time needed for training. Distinctive or hard-to-observe situations can be handled more safely by also using thermal imaging, smoke warning and audio (including cracking and alarms). Fire detection can be carried out in real-time and remotely by using this system on IoT platforms and drones. It may also become a complete preventative and emergency response scheme through the activation of sprinkler systems and through sending alerts and evacuation orders over horns or tannoy systems. Additionally, the model can improve over time by learning from people's actions and experiences in different situations.

REFERENCES

1. Zhang, Q., Xu, J., & Xu, L. (2023). "Deep Learning for Fire Detection: A Comprehensive Survey." IEEE Transactions on Industrial Informatics, 19(5), 5772-5781. https://doi.org/10.1109/TII.2022.3197169
2. Maksimovic, M., & Vujovic, V. (2022). "Analysis of conventional fire detection systems: Limitations and opportunities for improvement." Fire Safety Journal, 128, 103546. https://doi.org/10.1016/j.firesaf.2021.103546
3. Çetin, A. E., Dimitropoulos, K., & Gouverneur, B. (2021). "Video fire detection – Review." Digital Signal Processing, 23(6), 1827–1843. https://doi.org/10.1016/j.dsp.2021.103137
4. Çelik, T., & Demirel, H. (2009). "Fire detection in video sequences using a generic color model." Fire Safety Journal, 44(2), 147-158. https://doi.org/10.1016/j.firesaf.2008.05.005
5. Khan, S., Muhammad, K., & Mumtaz, S. (2020). "A Reliable Fire Detection System in RGB and Infrared Videos Using Transfer Learning." Image and Vision Computing, 93, 103857. https://doi.org/10.1016/j.imavis.2019.103857
6. Töreyin, B. U., Dedeoğlu, Y., & Çetin, A. E. (2007). "Computer vision-based method for real-time fire and flame detection." Pattern Recognition Letters, 27(1), 49-58. https://doi.org/10.1016/j.patrec.2005.06.015
7. Muhammad, K., Ahmad, J., & Baik, S. W. (2020). "Early Fire Detection using Convolutional Neural Networks during Surveillance for Effective Disaster Management." Neurocomputing, 288, 30-42. https://doi.org/10.1016/j.neucom.2019.11.014
8. Sharma, J., Granmo, O. C., & Goodwin, M. (2023). "MobileFire: Efficient Deep Learning Fire Detection for Resource-Constrained Environments." Sensors, 23(3), 1476. https://doi.org/10.3390/s23031476
9. Lin, G., Zhang, Y., & Xu, G. (2022). "Challenges and Solutions in Deep Learning-Based Fire Detection Systems." IEEE Access, 10, 34251-34260. https://doi.org/10.1109/ACCESS.2022.3162427
10. Zhang, H., Zhuo, L., & Wang, Y. (2021). "A Hybrid Fire Detection Method Based on Color Segmentation and Deep Learning." IEEE Transactions on Instrumentation and Measurement, 70, 1-12. https://doi.org/10.1109/TIM.2021.3069508
11. Dimitropoulos, K., Barmpoutis, P., & Grammalidis, N. (2022). "Spatio-temporal attention mechanisms in deep learning for fire detection." Signal Processing: Image Communication, 93, 116-128. https://doi.org/10.1016/j.image.2021.116176
12. Jadon, A., Omama, M., & Varshney, A. (2022). "FireNet: A Specialized Lightweight Fire & Smoke Detection Model for Real-Time IoT Applications." IEEE Transactions on Industrial Informatics, 18(9), 6135-6143. https://doi.org/10.1109/TII.2022.3170113
13. Barmpoutis, P., Dimitropoulos, K., & Grammalidis, N. (2020). "Ensemble Fire and Smoke Detection Based on Deep Learning." Fire Technology, 56, 1007-1032. https://doi.org/10.1007/s10694-019-00921-w
14. Muhammad, K., Khan, S., & Palade, V. (2022). "Edge Intelligence-Assisted Smoke Detection in Foggy Surveillance Environments." IEEE Transactions on Industrial Informatics, 18(2), 1334-1344. https://doi.org/10.1109/TII.2022.3147456

15. Saeed, F., Paul, A., & Hong, W. H. (2023). "EdgeFire: A Lightweight Fire Detection System for Smart Building Safety." IEEE Internet of Things Journal, 10(4), 3456-3467. https://doi.org/10.1109/JIOT.2022.3187456
16. Kaur, H., Sood, S. K., & Karambir, K. (2021). "Fog Computing for Sustainable Smart Cities: Smart Fire Detection Case Study." Sustainable Cities and Society, 72, 103065. https://doi.org/10.1016/j.scs.2021.103065
17. Wang, D., Zhang, X., & Ma, J. (2023). "Vision-Based Fire Growth Prediction Using Deep Learning." Fire Technology, 59, 823-842. https://doi.org/10.1007/s10694-022-01297-0
18. Chino, D. Y. T., Avalhais, L. P. S., & Rodrigues, J. F. (2022). "Fire detection and severity monitoring: A real-time computer vision approach." Expert Systems with Applications, 193, 116452. https://doi.org/10.1016/j.eswa.2022.116452
19. Huang, X., Yu, C., & Wang, H. (2021). "Challenges in Deep Learning Based Fire Detection Under Adverse Conditions." IEEE Access, 9, 124226-124237. https://doi.org/10.1109/ACCESS.2021.3110714
20. Bhargava, V., Farahmand, A., & Shahrokni, A. (2023). "Multi-Modal Sensing for Enhanced Fire Safety: Future Directions and Challenges." Building and Environment, 228, 109813. https://doi.org/10.1016/j.buildenv.2022.109813