Fire Detection in Video Sequence using Deep CNN Based Algorithm and Localization

Mrs.S.kanagamaliga[1], J. Dhivya Princy[2],M. Kiran Patel[3], T.Manipriya.[4]

[1]AP, Department of ECE, Velammal College of Engineering and Technology, Madurai.

[2,3,4]Stuents,Department of ECE, Velammal College of Engineering and Technology, Madurai.

Abstract- *This paper reports about the earlier detection of fire and to monitor the spread of fire. A fire detection system based on light detection rather than smoke detection. Consider an input video and detect fire during surveillance using convolutional neural networks (CNNs). CNN architecture is inspired by the Squeeze Net architecture for surveillance applications. Traditional fire alarm system based on sensors is overcome by using segmentation and classification based on CNN algorithm. Segmentation is performed by using k-means clustering and L*a*b color space. Classification is done using CNN algorithm. After classification a message will be send through message box.*

Keywords: *Fire detection, Convolution Neural Networks(CNN), Color Segmentation, Deep Learning, Image Classification*

I. INTRODUCTION

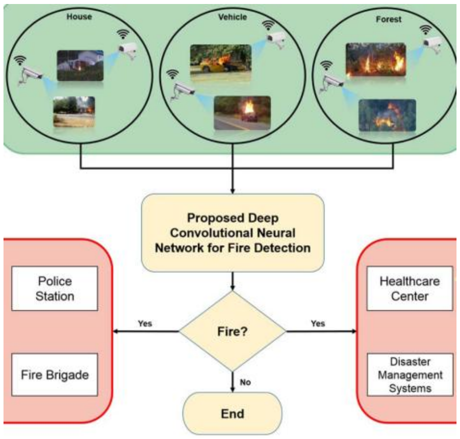
A variety of sensors are used in different applications such as gas leakage, fire alarm, traffic monitoring. Using surveillance systems, various abnormal events are detected. A fire is also an abnormal event that causes damage to the property and human lives. This is caused due to system failure or human error. Traditional fire alarm system is used to detect fire. Fire alarm system based on sensors such as infrared and optical sensors. These sensors require human involvement to find the location of the fire and it does not provide the size and location of the fire. To detect the fire earlier without human involvement and to reduce false alarms and to improve the accuracy of the detection segmentation and classification based on CNN algorithm is used. The implementation is difficult as Alex net architecture requires 238MB which is large in size. To reduce the size Squeeze net architecture is used as it requires only 3MB and thus saving an extra space of 235MB. Thus the minimizing the cost and makes the implementation more feasible**.**

II. LITERATURE SURVEY

Turgay Çelik, Hüseyin (2007) concepts from fuzzy logic are used to replace existing heuristic rules and make the classification more robust in effectively discriminating fire and fire like colored objects. The purpose of Video steganography analysis is to detect the presence of hidden message in cover photographic videos. Supervised learning is an effective and commonly used method to cope with difficulties of unknown video statistics and unknown steganography. Wen-Bing Horng and Jian-Wen Peng (2008) used a fast and practical real-time image-based fire flame detection method based on color analysis. A method of processing halftone Videos that improves the quality of the share Videos and the recovered secret video in an extended visual cryptography scheme for which the size of the share videos and the recovered video is the same as for the original halftone secret video. The resulting scheme maintains the perfect security of the original extended visual cryptography approach. Wenhao Wang, Hong Zhou (2012) used a method to extract flame object precisely; it introduces a kind of the new method of extraction flame object based on the threshold of the area. The end user identifies a Video which is going to act as the carrier of data. The data file is encrypted and authenticated. This message is hidden in the Video. The Video if hacked or interpreted by a third party user will open up in any Video previewer but not displaying the data. This protects the data from being invisible and hence be secure during transmission. The user in the receiving end uses another piece of code to retrieve the data from the Video. Mengxin Li and Weijing Xu (2013) used this approach to detect fire based on video steganography with digital watermarking techniques as an efficient and robust tool for protection. Here considers video as set of frames or Videos and any changes in the output video by hidden data is not visually recognizable. It is analyzed the different techniques for embedding and security. After analyzing these techniques, spatial domain, the least significant bits (LSB) is the best techniques for hiding a secret message or Video into cover media. For protecting the secret message, transform domain techniques DWT and DCT are best. DCT has strong robustness and is widely used in digital video watermarking.

III. EXISTING APPROACH

Fire detection using hand-crafted features is a tedious task, due to the time-consuming method of features engineering. It is particularly challenging to detect a fire at an early stage in scenes with changing lighting conditions, shadows, and fire-like objects; conventional low-level feature based methods generate a high rate of false alarms and have low detection accuracy. To overcome these issues, we investigate deep learning models for possible fire detection at early stages during surveillance. Taking into consideration the accuracy, the embedded processing capabilities of smart cameras, and the number of false alarms, we examine various deep CNNs for the target problem. A systematic diagram of our framework is given in Fig. 1.


House
Vehicle
Forest
Proposed Deep Convolutional Neural Network for Fire Detection
Police Station
Fire Brigade
Yes
Fire?
Yes
No
End
Healthcare Center
Disaster Management Systems

A. Convolutional Neural Network Architecture

CNNs have shown encouraging performance in numerous computer vision problems and applications, such as object detection and localization image segmentation, super-resolution, classification and indexing and retrieval .This widespread success is due to their hierarchical structure, which automatically learns very strong features from raw data. A typical CNN architecture consists of three well-known processing layers.

1) A convolution layer, where various feature maps are produced when different kernels are applied to the input data.

2) A pooling layer, which is used for the selection of maximum activation considering a small neighborhood of feature maps received from the previous convolution layer; the goal of this layer is to achieve translation invariance to some extent and dimensionality reduction.

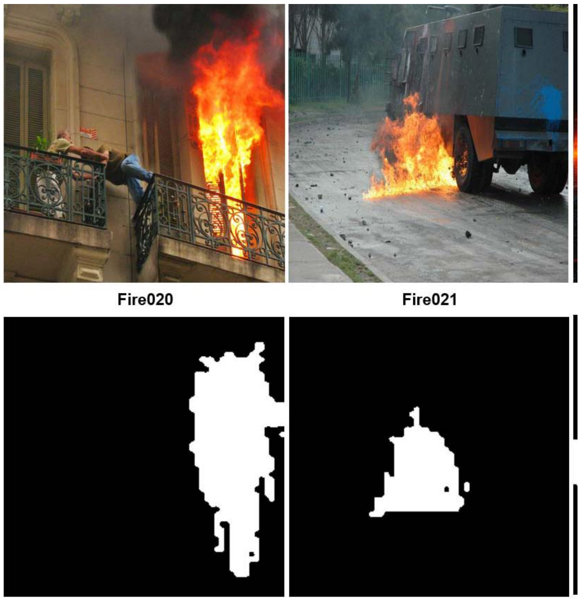
3) A fully connected layer which models high-level information from the input data and constructs its global representation. This layer follows numerous stacks of convolution and pooling layers, thus resulting in a high-level representation of the input data. These layers are arranged in a hierarchical architecture such that the output of one layer acts as the input of the next layer. During the training phase, the weights of all neurons in convolutional kernels and fully connected layers are adjusted and learned. These weights model the representative characteristics of the input training data, and in turn can perform the target classification. We use a model with architecture similar to that of SqueezeNet modified in accordance with our target problem. The original model was trained on the Image Net dataset and is capable of classifying 1000 different objects.

In our case, however, we used this architecture to detect fire and no fire images. This was achieved by reducing the number of neurons in the final layer from 1000 to 2. By keeping the rest of the architecture similar to the original, we aimed to reuse the parameters to solve the fire detection problem more effectively. There are several motivational reasons for this selection, such as a lower communication cost between different servers in the case of distributed training, a higher feasibility of deployment on FPGAs, application-specific integrated circuits, and other hardware architectures with memory constraints and lower bandwidth. The model consists of two regular convolutional Layers, three max pooling layers, one average pooling Layer and eight modules called “fire modules.”

The input of the model is color images with dimensions of 224×224×3 pixels. In the first convolution layer, 64 filters of size 3×3 are applied to the input image, generating 64 feature maps. The maximum activations of these 64 features maps are selected by the first max pooling layer with a stride of two pixels, using a neighborhood of 3×3 pixels. This reduces the size of the feature maps by factor of two, thus retaining the most useful information while discarding the less important details. Next, we use two fire modules of 128 filters, followed by another fire module of 256 filters. Each fire module involves two further convolutions, squeezing, and expansion. Since each module consists of multiple filter resolutions and there is no native support for such convolution layers in the Caffe framework, an expansion layer was introduced, with two separate convolution layers in each fire module. The first convolution layer contains 1×1 filters, while the second layer consists of 3×3 filters. The output of these two layers is concatenated in the channel dimension. Following the three fire modules, there is another max pooling layer which operates in the same way as the first max pooling layer. Following the last fire module (Fire9) of 512 filters, we modify the convolution layer according to the problem of interest by reducing the number of classes to two [$M = 2$ (fire and normal)]. The output of this layer is passed to the average pooling layer, and result of this layer is fed directly into the Softmax classifier to calculate the probabilities of the two target classes.

B. Deep CNN for Fire Detection and Localization

This section explains the process of fire detection and its localization using the proposed deep CNN. Although deep CNN architectures learn very strong features automatically from raw data, some effort is required to train the appropriate model considering the quality and quantity of the available data and the nature of the target problem. We trained various models with different parameter settings, and following the fine-tuning process obtained an optimal model which can detect fire from a large distance and at a small scale, under varying conditions, and in both indoor and outdoor scenarios.



Another motivational factor for the proposed deep CNN was the avoidance of preprocessing and features engineering, which are required by traditional fire detection algorithms. To
test a given image, it is fed forward through the deep CNN, which assigns a label of “fire” or “normal” to the input image. This label is assigned based on probability scores computed by the network. The higher probability score is taken to be the final class label of the input image. A set of sample images with their predicted class labels and probability scores is given in Fig. 2.
To localize a fire in a sample image, we employ the framework given in Fig. 3. First, a prediction is obtained from our deep CNN.
A set of sample fire images with their segmented regions is given in Fig. 4.
The segmented fire is used for two further purposes:
1) Determining the severity level/burning degree of the scene under observation.
2) Finding the zone of influence (ZOI) from the input fire image. The burning degree can be determined from the number of pixels in the segmented fire. The ZOI can be calculated by subtracting the segmented fire regions from the original input image. The resultant ZOI image is then passed from the original SqueezeNet model, which predicts its label from 1000 objects. The object information can be used to determine the situation in the scene, such as a fire in a house, a forest, or a vehicle. This information, along with the severity of the fire, can be reported to the fire brigade to take appropriate action.

C. Fire Localization: Results and Discussion

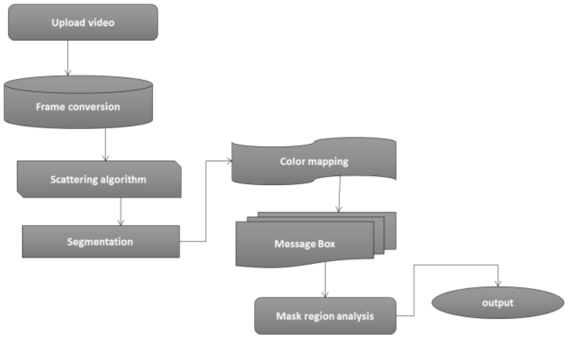
In this section, the performance of our approach is evaluated in terms of fire localization and understanding of the scene under observation. True positive and false positive rates were computed to evaluate the performance of fire localization. The feature maps we used to localize fire were smaller than the ground truth images, and were therefore resized to match the size of the ground truth images. We then computed the number of overlapping fire pixels in the detection maps and ground truth images, and used these as true positives. Similarly, we also determined the number of nonoverlapping fire pixels in the detection maps and interpreted these as false positives.

One further reason for using SqueezeNet was the ability of the model to give larger sizes for the feature maps by using smaller kernels and avoiding pooling layers. This allowed us to perform a more accurate localization when the feature maps were resized to match the ground truth images. These localization results are compared with those of several state-of-the-art methods, such as Chen *et al*, Çelik and Demirel, Chino *et al.* (BoWFire), Rudz *et al*, and Rossi *et a*. We report three different results for our CNNFire based on the threshold *T* of the binarization process. The approach maintains a better balance between the true positive rate and false positive rate, making it more suitable for fire localization in surveillance systems. The results of BoWFire, color classification, Celik and Rudz are almost the same. Rossi gives the worst results in this case, and Chen is better than Rossi. The results from CNNFire are similar to the ground truth.

The performance of all methods for another sample image, with a higher probability of false positives. Although BoWFire has no false positives for this case, it misses some fire regions, as is evident from its result. Color classification and Celik detect the fire regions correctly, but give larger regions as false positives. Chen fails to detect the fire regions of the

ground truth image. Rossi does not detect fire regions at all for this case. The false positive rate of Rudz is similar to our CNNFire, but the fire pixels detected by this approach are scarce. Although our method gives more false positives than the BoWFire method,
it correctly detects the fire regions which are more similar to the ground truth images. In addition to fire detection and localization, our system can determine the severity of the detected fire and the object under observation. For this purpose, we extracted the ZOI from the input image and segmented fire regions. The ZOI image was then fed forward to the SqueezeNet model, which was trained on the Image Net dataset with 1000 classes. The label assigned by the SqueezeNet model to the ZOI image is then combined with the severity of the fire for reporting to the fire brigade.

IV. PROPOSED APPROACH



A. *Input Video:*

Use a Video Reader object to read files containing video data. The object contains information about the video file and enables you to read data from the video. You can create a Video Reader object using the Video Reader function, query information about the video using the object properties, and then read the video using object functions. Timestamp of the video frame to read, specified as a numeric scalar. The timestamp is specified in seconds from the start of the video file. The value of Current time can be between zero and the duration of the video. On some platforms, when you create a Video Reader object, the 'Current Time' property might contain a value close to, but not exactly, zero. This variation in the value of the 'Current Time' property is due to differences in how each platform processes and reads videos.

B. *Frame conversion:*

To select the path and file name to video given format upload the movie player. That file to be loaded after play the video it will be converter Frame row and column wise for the frames. At the same time for all frames will be converted and the number of panels. Then the frames all will be write on particular folder, we need to display our values for the frames.

C. *Preprocessing:*

In computer graphics and digital imaging, video scaling refers to the resizing of a digital Video. In video technology, the magnification of digital material is known as upscaling or resolution enhancement. When scaling a vector graphic Video, the graphic primitives that make up the Video can be scaled using geometric transformations, with no loss of Video quality. When scaling a raster graphics Video, a new Video with a higher or lower number of pixels must be generated. In the case of decreasing the pixel number (scaling down) this usually results in a visible quality loss. From the standpoint of digital signal processing, the scaling of raster graphics is a two-dimensional example of sample rate conversion, the conversion of a discrete signal from a sampling rate (in this case the local sampling rate) to another.

D. *Color Segmentation:*

Segment colors in an automated fashion using the L*a*b* color space and K-means clustering. The L*a*b* color space is derived from the CIE XYZ tristimulus values. The L*a*b* space consists of a luminosity layer 'L*', chromaticity-layer 'a*' indicating where color falls along the red-green axis, and chromaticity-layer 'b*' indicating where the color falls along the blue-yellow axis. All of the color information is in the 'a*' and 'b*' layers. You can measure the difference between two colors using the Euclidean distance metric. Clustering is a way to separate groups of objects. K-means clustering treats each object as having a location in space. It finds partitions such that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. K-means clustering requires that you specify the number of clusters to be partitioned and a distance metric to quantify how close two objects are to each other.

E. *Segmentation:*

Video segmentation often requires that the scheme be time efficient to meet the requirement of real time and format compliance. It is not practical to encrypt the whole compressed video bit stream like what the traditional ciphers do because of the following two constraints, i.e., format compliance and computational cost. Alternatively, only a fraction of video data is encrypted to improve the efficiency while still achieving adequate security. The key issue is then how to select the sensitive data to tracks the objects. According to the analysis given, it is reasonable to encrypt both spatial information (IPM and residual data) and motion information (MVD) during encoding.

F. K-means clustering:

The most common algorithm uses an iterative refinement technique. Due to its ubiquity it is often called the ***k*-means algorithm**; it is also referred to as **Lloyd's algorithm**, particularly in the computer science community. Given an initial set of *k* means $m_1^{(1)},\ldots,m_k^{(1)}$ (see below), the algorithm proceeds by alternating between two steps**. *K-means* clustering** is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. *K-means* clustering aims to partition *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest *m* there are efficient heuristic algorithms that are commonly employed and converge quickly to a local optimum. These are usually similar to the expectation-maximization algorithm for mixtures of Gaussian distributions via an iterative refinement approach employed by both algorithms. Additionally, they both use cluster centers to model the data; however, *k*-means clustering tends to find clusters of comparable spatial extent, while the expectation-maximization mechanism allows clusters to have different shapes. The algorithm has a loose relationship to the *k*-nearest neighbor classifier, a popular machine learning technique for classification that is often confused with *k*-means because of the *k* in the name. One can apply the 1-nearest neighbor classifier on the cluster centers obtained by *k*-means to classify new data into the existing clusters and serving as a prototype of the cluster.

V. CONCLUSION

The embedded processing capabilities of smart cameras have given rise to intelligent CCTV surveillance systems. Various abnormal events such as accidents, medical emergencies, and fires can be detected using these smart cameras. Of these, fire is the most dangerous abnormal event, as failing to control it at an early stage can result in huge disasters, leading to human, ecological and economic losses. Inspired by the great potential of CNNs, we propose a lightweight CNN basedon the Squeeze Net architecture for fire detection in CCTV surveillance networks. Our approach can both localize fire and identify the object under surveillance. Furthermore, our proposed system balances the accuracy of fire detection and the size of the model using fine-tuning and the Squeeze Net architecture, respectively. We conduct experiments using two benchmark datasets and verify the feasibility of the proposed system for deployment in real CCTV networks. In view of the CNN model’s reasonable accuracy for fire detection and localization, its size, and the rate of false alarms, the system can be helpful to disaster management teams in controlling fire disasters in a timely manner, thus avoiding huge losses. This paper mainly focuses on the detection of fire and its localization, with comparatively little emphasis on understanding the objects and scenes under observation. Future studies may focus on making challenging and specific scene understanding datasets for fire detection methods and detailed experiments. Furthermore, reasoning theories and information hiding algorithms can be combined with fire detection systems to intelligently observe and authenticate the video stream and initiate appropriate action, in an autonomous way.

REFERENCES

[1] B. C. Ko, K.-H. Cheong, and J.-Y. Nam, “Fire detection based on vision sensor and support vector machines,” Fire Safety J., vol. 44, no. 3, pp. 322–329, 2009.

[2] I. Mehmood, M. Sajjad, and S. W. Baik, “Mobile-cloud assisted video summarization framework for efficient management of remote sensing data generated by wireless capsule sensors,” Sensors, vol. 14, no. 9, pp. 17112–17145, 2014.

[3] K. Muhammad, M. Sajjad, M. Y. Lee, and S. W. Baik, “Efficient visual attention driven framework for key frames extraction from hysteroscopy videos,” Biomed. Signal Process. Control, vol. 33, pp. 161–168, Mar. 2017.

[4] R. Hamza, K. Muhammad, Z. Lv, and F. Titouna, “Secure video summarization framework for personalized wireless capsule endoscopy,” Pervasive Mobile Comput., vol. 41, pp. 436–450, Oct. 2017.

[5] P. Harris, R. Philip, S. Robinson, and L. Wang, “Monitoring anthropogenic ocean sound from shipping using an acoustic sensor network and a compressive sensing approach,” Sensors, vol. 16, no. 3, p. 415, 2016.

[6] K. Muhammad et al., “Secure surveillance framework for IoT systems using probabilistic image encryption,” IEEE Trans. Ind. Informat., to be published, doi: 10.1109/TII.2018.2791944.

[7] K. Muhammad, J. Ahmad, and S. W. Baik, “Early fire detection using convolutional neural networks during surveillance for effective disaster management,” Neurocomputing, vol. 288, pp. 30–42, May 2018.

[8] R. Chi, Z.-M. Lu, and Q.-G. Ji, “Real-time multi-feature based fire flame detection in video,” IET Image Process., vol. 11, no. 1, pp. 31–37, Jan. 2016.

[9] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, “An early fire-detection method based on image processing,” in Proc. Int. Conf. Image Process. (ICIP), 2004, pp. 1707–1710.

[10] T. Toulouse, L. Rossi, M. Akhloufi, T. Celik, and X. Maldague, “Benchmarking of wildland fire colour segmentation algorithms,” IET Image Process., vol. 9, no. 12, pp. 1064–1072, Dec. 2015.

[11] G. Coatrieux, W. Pan, F. Cuppens, and C. Roux, “Reversible watermarking based on invariant Video classification and dynamic histogram shifting,” IEEE Trans. Inf. Forensics Secur., vol. 8, no. 1, pp. 111–120, 2013.

[12] D. Coltuc, “Improved embedding for prediction-based reversible watermarking,” IEEE Trans. Inf . Forensics Secur ., vol. 6, no. 3, pp. 873–882, 2011.

[13] B. Ou, X. Li, Y. Zhao, R. Ni, and Y. Q. Shi, “Pairwise prediction error expansion for efficient reversible data hiding,” IEEE Trans . Image Process. vol. 22, no. 12, pp. 5010–5021, 2013.

[14] I. Dragoi and D. Coltuc, “Local prediction based difference expansion reversible watermarking,” IEEE Trans. Video Process., vol. 23, no. 4, pp. 1779–1790, 2014.

[15] S. W. Weng and J. S. Pan, “Reversible watermarking based on eight improved prediction modes,” J. Inf. Hiding Multimedia Signal Process., vol. 5, no. 3, pp. 527–533, 2014.

[16] H. Wu, J. Dugelay, and Y. Q. Shi, “Reversible Video data hiding with contrast enhancement,” IEEE Signal Process. Lett ., vol. 22, no. 1, pp. 81–85, 2015.

[17] M. Gao and L. Wang, “Comprehensive evaluation for HE based contrast enhancement,” Adv. Intell. Syst. Applicat, vol. 2, pp. 331–338, 2013.