



NotiPoy: Early Notification of Detected Fire Using Image Processing Applying Convolutional Neural Network

Jennalyn G. Nicolas-Mindoro¹, Mon Arjay F. Malbog², Joshua S. Gulmatico³, Jennifer B. Enriquez⁴

¹Computer Engineering Department, Technological Institute of the Philippines – Manila, jnicolas.cpe@tip.edu.ph

²Computer Engineering Department, Technological Institute of the Philippines – Manila, malbog.monarjay@gmail.com

³Computer Engineering Department, Technological Institute of the Philippines – Manila, enr.gulmatico@gmail.com

⁴Computer Engineering Department, Technological Institute of the Philippines – Manila, witty_jhen@yahoo.com

ABSTRACT

Fire incident is one of the most common undesirable events that cause damages toward houses and human. It can start anytime and anywhere without any prior hint when it would happen. With the cause it brings, the implementation of a device capable of detecting and providing notification is necessary. This study presents the method of early-stage fire-flame detection utilizing image processing integrated with the Android application for emergency notification. The proposed system utilized the image processing applying Convolutional Neural Network (CNN) integrated with mobile app as early notification to the end user. After executing 3999 iterations of 210 images are used for testing to get the Train Accuracy (how accurate the training is) and Cross entropy (how far is the prediction of the actual result), it resulted to 100% trained accuracy and 96% validation. The test evaluation accuracy result was 99% using a cached file of 210, it means that expect the model to perform ~99% accuracy on the new data. From the test that has been done, the proposed system was able to capture the fire flame images and send necessary notification real time. The test results show the captured actual images and the information sent to the end user. The summary test results verified that all gathered data was higher than 90% to be able to send notification. The android app used to show both the pre-investigation information useful for the house owner and BFP personnel.

Key words : Android Application, Convolutional Neural Network, Fire Detection, Emergency Notification, Image Processing

1. INTRODUCTION

The cost of fires in human is a global public health concern. Direct physical harm that causes deaths and injuries such as burns, and psychological and emotional trauma are the two-fold effects of fire to human [1]. It was estimated cost was due to losses from fire accidents is approximately one percent of global Gross Domestic Product per annum, according to

World Fire Statistics. It was recorded that most number of occurrences are in (NCR) National Capital Region from a total of 883 fires and 824 fire-related casualties were reported from 2010 to 2012 in the Philippines [2]. Based on records, a total of 2,316 fire incidents was reported from 2011 to 2014. The Bureau of Fire Protection (BFP) has listed several causes of fire such as defective electrical wiring/association, unattended open fire, ignore cooking/stove, LPG, cigarette butt, light, electrical gadgets, and apparatuses, among others [1]. BFP reported that an average of 9 fire incidents take place per day nationwide and NCR has the most astounding fire occurrence record over the years. In 2016, over 700 fire incidents were recorded in the city resulting in damage worth more than 130 million pesos. These findings bring to fore the importance of raising awareness and providing education related to fire prevention [2].

Civilian fire injury and death occurred every 33.5 minutes and 160 minutes, respectively. Among the fire a death, 78% occurred at home. One of the main reasons is the delayed escape for disabled people as the traditional fire alarming systems need strong fires or proximity, failing to generate an alarm on time for such people. This necessitates the existence of effective fire alarming systems for surveillance [3], and providing respond to these emergencies situations as quickly as possible are the obvious way to minimize this kind of loss [4]. Application of fire detection as a tool had increased due to the frequent occurrence [5]. Fires in the early detection and early warning are two important ways to extinguish the fire promptly and avoid great casualties and property loss. Therefore, the requirement of placing intelligent fire alarm system is important within buildings especially in the buildings where contain many people inside or valuable belongings [6].

According to the study, early detection of fire and an autonomous response is important and helpful to disaster management systems [7]. Since light travels much faster than smoke or heat, visual analysis of the scene offers an effective means to detect fire flame in an early stage, which enables the prompt reaction to fire accidents. Fast and accurate detection of fire flame will ensure a sufficient amount of time to

dispatch fire engines to the fire site [8]. Earlier detection of fire often leads to a smaller size at the initial attack, thus increasing the probability of containment [9]. Due to this need, fire detection has been an interesting topic for researchers using computer vision because of its impact on the community. Intelligent systems methods are increasingly used to complement computer vision techniques. The solutions proposed encompassing the use of expert systems include neural networks [10]. Deep learning methods, especially Convolutional neural networks (CNN) have achieved state-of-the-art performance on the computer vision tasks, including image classification, object detection, pose estimation and semantic segmentation, etc. [11] [12] [13]. The previous study used framework fine-tuned convolutional neural networks for CCTV surveillance cameras to detect fire in varying indoor and outdoor environments [14]. Similar study was cost-effective fire detection CNN architecture for surveillance videos and a study that utilized the application of image-based fire detection using a neural network to estimate the burning degree of fire flames so that users could be informed with a proper warning alarm [15]. Another study focuses more on the color segmentation technique. It was stated that is very important technique that are used for image segmentation and used for semantic labeling that gives good results to differentiate each object.

Considering the application of different computer vision techniques, for the response team to take actions when the fire has been detected, communication between the system and the user is required. Today, information and Communications Technology (ICT) increasingly used by several Governments, and mobile technology offers several opportunities [16]. Related literature shows that solutions performed well with high accuracy of fire detection at early stage in varying indoor and outdoor environment while minimizing the false alarms, however there were no pre-investigative information of the early stage of a fire incident was provided. Previous studies focus more on the methods of detection of fire-flame in early stage and determining the accuracy level in a given time. It does not discuss the possible cause of fire and notify the end user of its status and fire category such as ordinary combustibles, fluid etc. that caused the fire. Even though the concept of the research is not new, this study was made to enhance the early fire detection management system in the country. This study presents the method of early-stage fire-flame detection utilizing image processing integrated with the Android application for emergency notification. This will notify the BFP and the end user specifically the house owner that a fire incident will likely to occur. The system will provide an early recommendation to the end user on how to put out the fire using the emergency notification app. The system will provide a real-time notification once an image of fire flame has been captured. This also provides the possible cause and the information of the fire category, tips and safety pre-caution measure for the end user while waiting for the BFP response. BFP will be informed of the location of the

incident and the possible cause of the fire to be used for re-investigation.

The rest of the paper is structured as follows. Section 2 presents the methodology used. Data gathering and testing are discussed in Section 3. Results and discussion in conducting the testing are discussed in Section 4. And this study concluded with a summary discussed in Section 5.

2. METHODOLOGY

A. Conceptual Framework

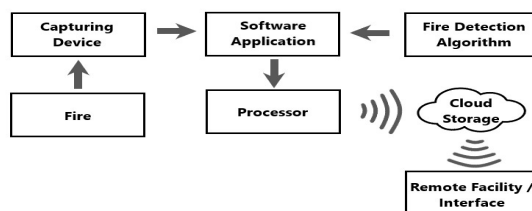


Figure 1: Conceptual Framework of NotiPoy

Figure 1 shows the development of NotiPoy. It was divided into three submodules, capturing device, fire detection using image processing and fire notification. Once the fire flame has been captured, this will be process by the microcontroller for fire detection. It will determine if the captured image is a fire flame or not. A notification will be sent to the end user.

B. Block Diagram

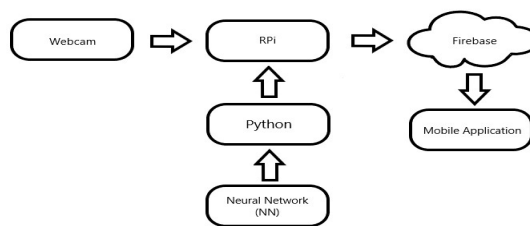


Figure 2: Block Diagram of NotiPoy

NotiPoy uses webcam that serves as a capturing device. The processing of the captured images uses the Raspberry Pi model B for controlling the communication of the input, Python programming and Neural Network algorithm for the training and learning of data. Firebase was used for the android mobile application development.

C. Graphical Representation

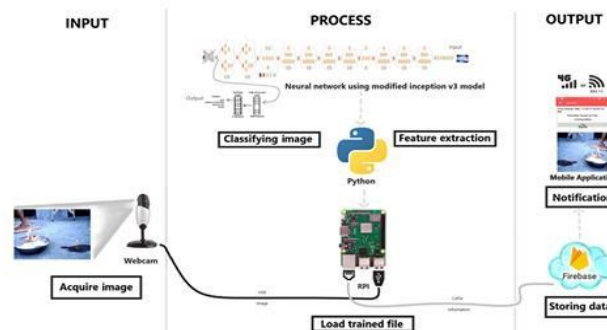


Figure 3: Graphical Representation of NotiPoy

Figure 3 shows the graphical presentation of the system. The captured images were analyzed using convolutional neural network that uses modified inception v3 model for classifying and feature extraction of data. Wired connection was used to send the processed data to firebase platform. Trained data was stored in firebase and notifies the end user.

3. DATA GATHERING AND TESTING

A. Gathering Dataset

There are 1455 images used for dataset with four fire classifications such as ordinary combustibile, fluid, electrical fire, and smoke. The datasets comprise of a positive and negative image. Positive images are the image that has the object itself while the negative images contain the object, background and the things related to that object. The positive image was used to extract the features of flames while the negative image was used to process as a basis for detecting fire at a certain range. The type dataset was mostly positive images with a resolution of 40x40 to 299x299 pixels. All data has been collected and generated in labeled images through scripts.

Table 1: Distribution of Datasets

Category	Training Images	Validation Images	Testing Images
Ordinary Combustibile	342	138	130
Fluid	222	48	60
Smoke	165	50	50
Electrical Fire	152	52	46

Table I shows the number of datasets in different categories. The materials that cause the fire and the actual image of fire is arranged as datasets. Actual images used was ranging from 40 by 40 up to 299 by 299 pixels as resolution.

B. Processing using Modified Inception v3 Model with Applied Transfer Learning - Bottlenecks

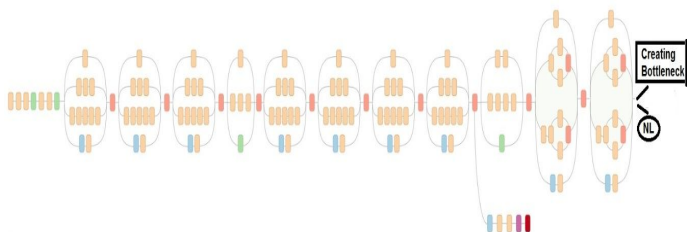
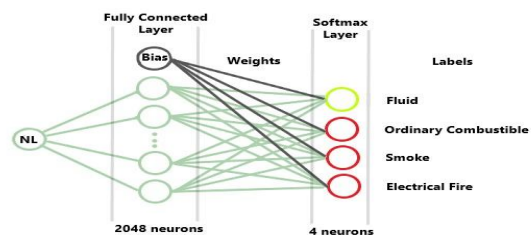


Figure 4: Convolutional Neural Network from Inception v3 Model

Dataset was fed in inception model v3 to analyze all the images in the training set and calculate bottleneck values. The bottleneck is an informal term that is often used that refers to the layer just before the final output layer that does the classification. Retraining this layer outputs values eventually support the inception classify the different categories in the dataset represented in the training data. Simply retraining this layer was enough as inception has already learned from

1,000 different classes as part of ImageNet, which is useful to distinguish between new objects.

In a convolution, there are 3 main parameters that need to be improved to modify the behavior of a convolutional layer. These parameters are filter size, stride and zero padding. The size of the output feature map generated depends on the above 3 important parameters. A larger size kernel can overlook at the features and could skip the essential details in the images whereas a smaller size kernel could provide more information leading to more confusion. Thus, there is a need to determine the most suitable size of the kernel/filter. When the stride is 1, we move the filter one pixel at a time. When we set the stride to 2 or 3 (uncommon), It move the filter 2 or 3 pixels at a time depending on the stride. The value of the stride also controls the size of the output volume generated by the convolutional layer. The zero padding also allows to control the spatial size of the output volume.



C. Training

Figure 5: Neural Network and the newly added layer

Using the bottlenecks from the model to retrain our new layer is a technique called transfer learning. After bottleneck process in the default setup 4000 training steps had been ran, where in each iteration ten images were chosen randomly to feed through bottleneck. The resulting predictions was compared to the actual labels and using back-propagation process the final layer's weights and biases are updated.

Table 2: Training data result using tensor flow

Step	Train Accuracy	Cross entropy	Validation
0	46.0%	1.290170	36.0%
10	94.0%	0.887403	82.0%
20	93.0%	0.666402	94.0%
30	94.0%	0.622212	89.0%
⋮	⋮	⋮	⋮
1000	99.0%	0.073424	98.0%
1010	98.0%	0.068056	97.0%
1020	100.0%	0.036969	98.0%
1030	100.0%	0.052646	99.0%
⋮	⋮	⋮	⋮
3970	100.0%	0.017303	98.0%
3980	100.0%	0.020237	98.0%
3990	100.0%	0.017260	99.0%
3999	100.0%	0.013293	96.0%
Final test validation accuracy = 99.0% (N=286)			

There are 210 images used for testing to get the Train Accuracy (how accurate the training is) and Cross entropy (how far is the prediction of the actual result). There are 286 images used to test the validation testing (only used to evaluate the model's performance). Final test evaluation accuracy is 99% using a cached file of 210; it means that expect the model to perform ~99% accuracy on the new data. The training accuracy gives the percentage of images used in the current training batch that were labeled correctly. The validation accuracy is the precision of a randomly selected group of images from a different set during training. The training accuracy is less reflective of the performance of the classifier because based on images that have already learned from, and hence, the network is at risk of over-fitting. A better measure is the validation accuracy. If there is a significant mismatch between the training accuracy and validation accuracy, then that is indicative that the network is memorizing potentially unhelpful features that do not generalize well.

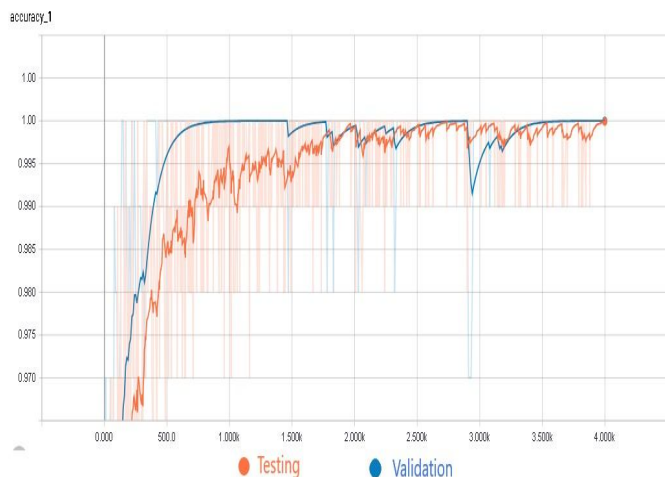


Figure 6: Accuracy between Training and Validation

The figure above shows the summary of how accurate the training is and the evaluation of the model's performance from 0 – 4000 iterations. The training's objective is to make the loss as small as possible. Inception splits the training data into three parts, where 80% used as the training set, 10% used a validation set, and 10% used as a testing set during the training. In that way, over-fitting is avoided, and bottlenecks are fine-tuned. Then the validation set has been passed through for tuning and the test set has sent through for classification.

With this, training of images was conducted in three division trials. On the first trial, 70% of training set, 15% of validating set, 15% of testing set. The result of the total accuracy of training is 80%. Second trial has 80% of training set, 10% of validating set, 10% of testing set. By these percentage, the accuracy of training reaches up to 99%. And for the third trial, 90% of training set, 5% of validating set, 5% of testing set. The total accuracy of training resulted to 95%. The 80-10-10 percentage determines the highest result of the accuracy of training the images.

The Training set has the highest percentage so that the system will be able to learn the images and can easily determine images from validation to testing of images.

D. Testing

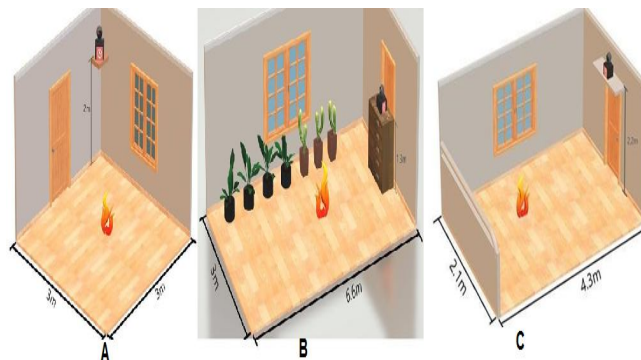


Figure 7: Room Installation Setup

There are three different location and distance was setup for testing. In figure 7-A show the device place to the upper corner of the room with a 2m high from the ground and the surrounding have a 3m sides. Figure 7-B shows that the device to the center part of the room with the 1.3m high from the ground and the sides of the room have a 6.6m and 3m. Figure 7-C it shows that the device place to the upper center part of the room with a 2.2m high from the ground and the size of the room is 2.1m and 4.3m.

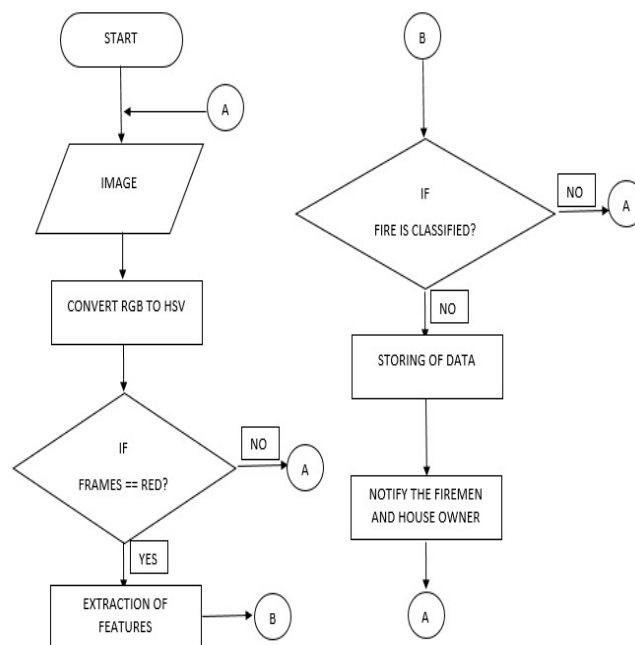


Figure 8: Testing Procedure

The figure shows how the device processed captured images. The RGB image was converted to HSV image value. It will determine if the frames is equal to red and it will load the details and extract the features. Once classify and extracted, this determine the type of fire flame and send the detected fire as notification.

4. RESULTS AND DISCUSSION

A. Combustible Material Fire Detection and Notification

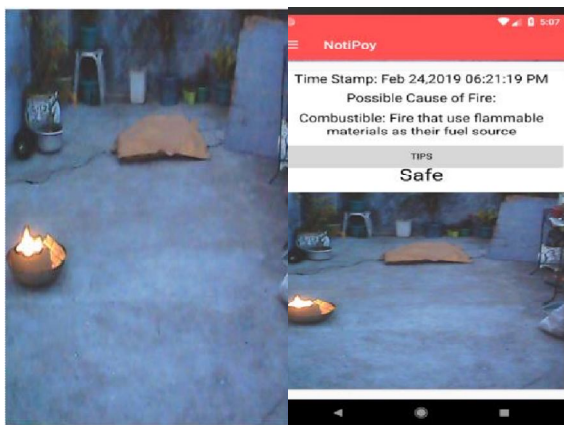


Figure 9: Detection of Fire and Notification of Combustible Material

The device was setup 1.3 m high from the ground at the center room show in figure 7-A. The burning material was placed in different distance to test the accuracy. Fire was detected and send notification. Another test was made using fluid gas. A real time notification was sent to the end user.

B. Fluid Material Fire Detection and Notification

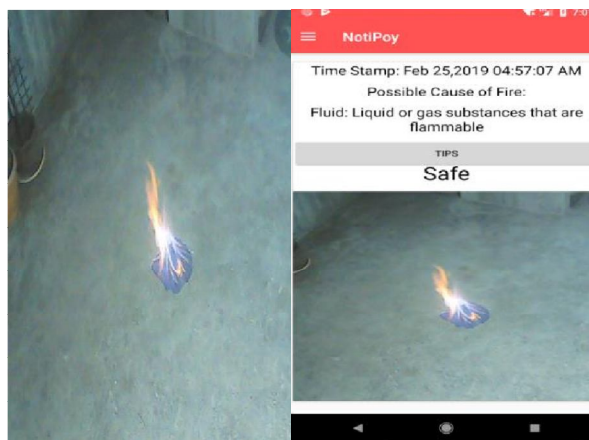


Figure 10: Detection of Fire and Notification of Fluid Material

The device was setup 2m high from the ground at the corner side of the room as shown in figure 7-B. Burning materials are placed in different distances for testing. A fluid gas was used to test if the device detects a fire from fluid material. The system captured 90.589% actual value and identified the type of fire and its possible caused. A real time notification was sent to the end user. Another test using the same material was made in setup C. The device was setup in two different places, the first was 2.2m high from the ground at the corner of the room and the second setup was 2.2 m high from the ground at the center side of the room. A successful notification was made after it detected the fire flame.

C. Summary of Test Results

Table 3: Summary of Testing and Results

# of Testing	Gathered Value	Fire Category	System Response
1	97.102%	Ordinary Combustible	Sent Notification
2	97.545%	Ordinary Combustible	Sent Notification
3	97.079%	Fluid	Sent Notification
4	92.726%	Fluid	Sent Notification
5	99.314%	Fluid	Sent Notification
6	98.852 %	Ordinary Combustible	Sent Notification
7	97.556 %	Ordinary Combustible	Sent Notification
8	98.142 %	Ordinary Combustible	Sent Notification
9	94.445 %	Fluid	Sent Notification
10	97.775 %	Ordinary Combustible	Sent Notification
11	90.120 %	Ordinary Combustible	Sent Notification
12	99.859 %	Fluid	Sent Notification
13	96.946 %	Fluid	Sent Notification
14	97.731 %	Fluid	Sent Notification
15	94.275 %	Ordinary Combustible	Sent Notification

The predicted result must be ninety percent higher for notification and sending data to cloud server. When it reaches ninety percent higher, there will be an interval before it will process another image. In the actual simulation, interval has been set to thirty seconds, the captured image looks unchanged because of the interval. While the interval works the video, simulation stops and continues after the interval. The summary of test results shows the actual gathered values with its corresponding category and system responses. The summary test results verified that all gathered data was higher than 90% to be able to send notification.

5. CONCLUSION

After executing 3999 iterations of 210 images are used for testing to get the Train Accuracy and Cross entropy resulted to 100% trained accuracy and 96% validation. The test evaluation accuracy result was 99% using a cached file of 210, it means that expect the model to perform ~99% accuracy on the new data. From the test that has been done, the proposed system was able to capture the fire flame images and send necessary notification. The test results show the captured actual images and the information sent to the end user. The summary test results verified that all gathered data was higher than 90% to be able to send notification. The android app used to show both the pre-investigation information useful for the house owner and BFP personnel.

ACKNOWLEDGEMENT

The authors would like to thank the Tracer Volunteer Fire Brigade in Pilar, Tondo, Manila for the time given for problem validation and data gathering.

REFERENCES

- 1 F. F. Balahadia and A. O. Trillanes, **Temporal Analysis and Geo-mapping of Fire Incidents in the City of Manila**, no. December, 2015.
<https://doi.org/10.1109/HNICEM.2015.7393211>
- 2 F. F. Balahadia, **Improving Fire Services using Spatio-Temporal Analysis: Fire Incidents in Manila**, no. January 2018, 2017.
- 3 K. Muhammad, S. Member, J. Ahmad, and S. Member, **Convolutional Neural Networks Based Fire Detection in Surveillance Videos**, *IEEE Access*, vol. 6, pp. 18174–18183, 2018.
<https://doi.org/10.1109/ACCESS.2018.2812835>
- 4 M. S. Reddy and K. R. Rao, **Fire Accident Detection and Prevention monitoring System using Wireless Sensor Network enabled Android Application**, vol. 9, no. May, 2016.
<https://doi.org/10.17485/ijst/2016/v9i17/93097>
- 5 K. Poobalan and S. Liew, **Fire detection algorithm using image processing techniques**, no. December, 2015.
- 6 Y. A. Shokova, **Design of an Arduino-based home fire alarm system with GSM module** Design of an Arduino-based home fire alarm system with GSM module, 2018.
- 7 K. Muhammad, J. Ahmad, and S. W. Baik, **Neurocomputing Early fire detection using convolutional neural networks during surveillance for effective disaster management**, *Neurocomputing*, vol. 288, pp. 30–42, 2018.
- 8 S. G. Kong, D. Jin, S. Li, and H. Kim, **Fast fire flame detection in surveillance video using logistic regression and temporal smoothing**, *Fire Saf. J.*, vol. 79, pp. 37–43, 2016.
<https://doi.org/10.1016/j.firesaf.2015.11.015>
- 9 A. Koltunov, S. L. Ustin, B. Quayle, B. Schwind, V. G. Ambrosia, and W. Li, **Remote Sensing of Environment The development and first validation of the GOES Early Fire Detection (GOES-EFD) algorithm**, *Remote Sens. Environ.*, vol. 184, pp. 436–453, 2016.
- 10 M. João, A. Moutinho, and M. Almeida, **Classification of potential fire outbreaks: A fuzzy modeling approach based on thermal images**, *Expert Syst. Appl.*, vol. 129, pp. 216–232, 2019.
<https://doi.org/10.1016/j.eswa.2019.03.030>
- 11 Q. Zhang *et al.*, **ScienceDirect Wildland Smoke Detection Based on Protection Faster R-CNN using Wildland Forest Smoke Detection Based on Faster using (on Fire the Development of Smoke Synthetic Images Wildland Forest Fire Smoke Detection Based on Faster using Wildland For**, *Procedia Eng.*, vol. 211, pp. 441–446, 2017.
- 12 S. Ibrahim, M. Hazrin, C. Rozan, and N. Sabri, **International Journal of Advanced Trends in Computer Science and Engineering", Comparative Analysis of Support Vector Machine (SVM) and Convolutional Neural Network (CNN) for White Blood Cells ' Classification**, vol. 8, no. 1, pp. 394–399, 2019.
<https://doi.org/10.30534/ijatcse/2019/6981.32019>
- 13 M. R. Devi, **International Journal of Advanced Trends in Computer Science and Engineering An Efficient Technique to Classify Human Activity using Convolutional**, vol. 8, no. 1, pp. 59–67, 2019.
<https://doi.org/10.30534/ijatcse/2019/1381.32019>
- 14 K. Muhammad, J. Ahmad, and S. W. Baik, **Early Fire Detection using Convolutional Neural Networks during Surveillance for Effective Disaster Management**, *Neurocomputing*, 2017.
- 15 W. Horng, **Image-Based Fire Detection Using Neural Networks**, no. January 2006, 2015.
<https://doi.org/10.2991/jcis.2006.301>
- 16 B. S. Fabito, F. F. Balahadia, and J. D. N. Cabatlo, **AppLERT: A Mobile Application for Incident and Disaster Notification for Metro Manila**, no. 1, pp. 288–292, 2016.