ISSN 2278-3091
Volume 9, No.1.3, 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse2991.32020.pdf
https://doi.org/10.30534/ijatcse/2020/2991.32020

MobileNet SSDv2 Inference Approach of Smoke Hazard Detection and Alert System: A Smoke-Induced Simulated Home-Environment

Alvin Sarraga Alon[1], Jesusimo L. Dioses, Jr.[2], Renante A. Diamante[3], Rhowel M. Dellosa[4], Francis Jesmar P. Montalbo[5]
[1]Technological Institute of the Philippines, Manila Philippines, aalon.cpe@tip.edu.ph
[2] Isabela State University, Echague Isabela Philippines, jdiosesjr@gmail.com
[3]Iloilo State College of Fisheries, Iloilo Philippines, rdiamantetip@gmail.com
[4]Technological Institute of the Philippines, Quezon City Philippines, rdellosa.cpe@tip.edu.ph
[5]Batangas State University, Batangas City Philippines, francismontalbo@yahoo.com

ABSTRACT

A smoke detector is an instrument that detects smoke usually as a fire warning. The devices available are placed in the ceiling and they take too long to respond to smoke. These smoke alarms are helpful when the smoke is big enough to reach the ceiling and alarm that there is a fire in the area. It takes a big fire and a lot of smoke before it can be detected and before the alarm goes on. In this study, the detection of smoke will be done by using an object-detection algorithm. It detects smoke early even if there is only a small amount to be detected. These detected smokes are then analyzed and will then inform the user about the detected smoke. This study can help a lot in fire prevention because of the detection of smoke inside the house and can prevent fire as early as the smoke has been recognized by the camera. The system produced an overall 89% testing accuracy.

Key words : Deep Learning, Machine Vision, Smoke Detection, Smoke Alarm.

1. INTRODUCTION

Smoke detectors are especially helpful in such situations where the fire is expected to produce a substantial amount of smoke until temperature adjustments are necessary to trigger a warm detection process and a fire is identified [1]. Smoke detectors use a photoelectric pillar between components and sources of light [2]. A alarm is issued on the off chance that smoke clouds are at the bar. Additionally, there are refractive-type models that calculate the light changes that exist inside the instrument as particles of smoke reach it [3].

The fire and smoke checking frameworks are helpful in various industries, for example, the military, the government managed savings and practical [4]. The ongoing techniques for fire and smoke location are utilized just movement and shading qualities subsequently many wrong cautions are going on and this is decline the exhibition of the frameworks. This exploration displays another technique for fire and smoke recognition through picture preparation [5].

The available existing devices nowadays are all placed in the ceiling and they take too long to respond to fire and smoke. It takes a big fire and a lot of smoke before it can be detected and before the alarm goes on [6]. When the fire is big enough, it is harder to stop it when it is detected very late. Commonly, there are lots of fire cases in the Philippines, most especially in Manila which is the Capital of the Philippines because of overpopulation and a lot of informal settlers where the houses are built from flammable materials [7]. Also, there are lots of dormitories and condominiums that have fire alarms and smoke detectors but it takes too long to be detected because the alarms are placed at the ceiling which causes the fire to increase until it was detected, and it makes the fire later to be prevented [8]. It can cause a big fire most especially when there is no one at home and the fire alarm goes on later that the fire has already spread.

The main purpose of a smoke detector is to identify the fire is present. It depends on the sort of detector it is, how it detects the fire. A photoelectric model has a light source which enters the sensor chamber [9]. The light is placed away from the sensor at an angle. When smoke enters, light reflects on the sensor, which activates the alarm. These devices are better identified when smoldering fires emerge with smoke before finally bursting into flames. There is another smoke detection that doesn't use photoelectric sensing such as [10]-[14]. Some others used machine vision based deep learning for general object detection such as [15]-[17].

With this project, it is easier to detect smoke and it can prevent the fire earlier than expected with the existing equipment in the market. This device doesn't need to reach the sensor because it will just detect the smoke in real-time and even if it is just minimal, if the smoke is thick, it will already be detected as fire-causing smoke and the device will turn on its alarm.

The objective of this study is to create a smoke alarm detector that can sense smoke even the thinnest ones. It is to create a system that will notify the user if smoke has been detected inside his/her house.

This study's scope is to detect smoke inside the house that will inform the user immediately if smoke has been detected. The study will cover a limited area only or indoors and narrow areas like school hallways, office rooms, and school classrooms. The possible clients of this study, generally, are the public because it is not limited to specific people with the profession but it can be used by everyone that needs it inside their homes and offices. The study is limited to detecting smoke only whether it is smoke-causing cooking, vaping, smoking or fire.

2. METHODOLOGY

The study developed a system that detects the presence of smoke automatically and tells the user about the detected smoke by using the object-detection algorithm in raspberry pi. Figure 1 shows the methodology of the study.

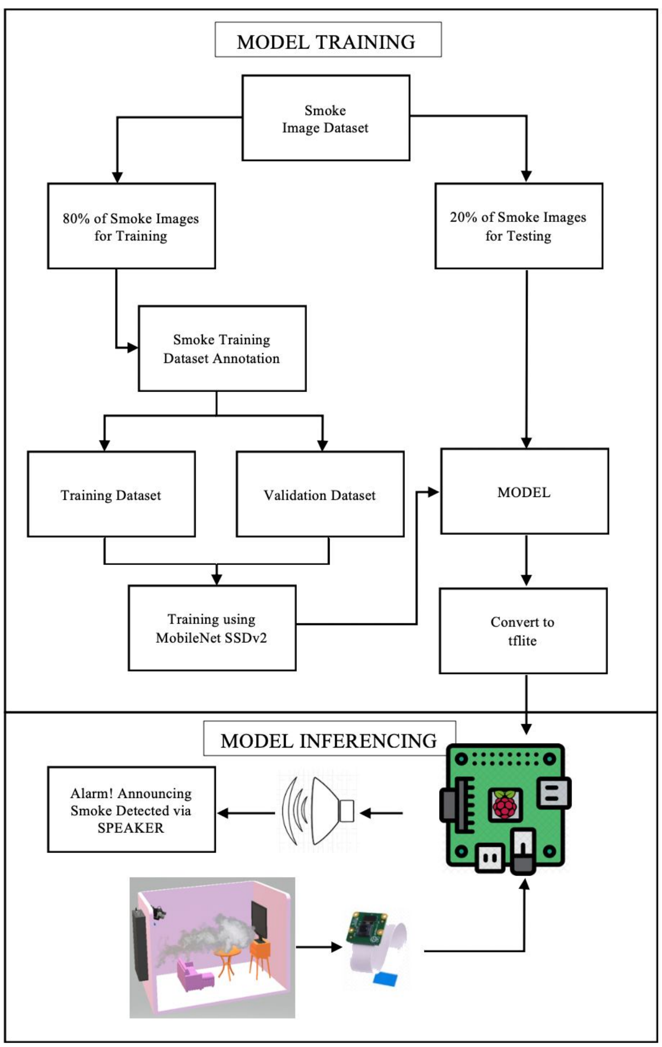

Figure 1: Methodology of Smoke Detection System

2.1 Smoke Dataset

The study gathered images containing smokes, vapes and converted them into frames. After gathering images, the images gathered are resized into 300x300 pixels. The dataset shown in Figure 2 gathered one by one is indifferent backgrounds. The study split the data for training and testing, 80% for training which includes 2364 images and 20% for validation which includes 595 images.

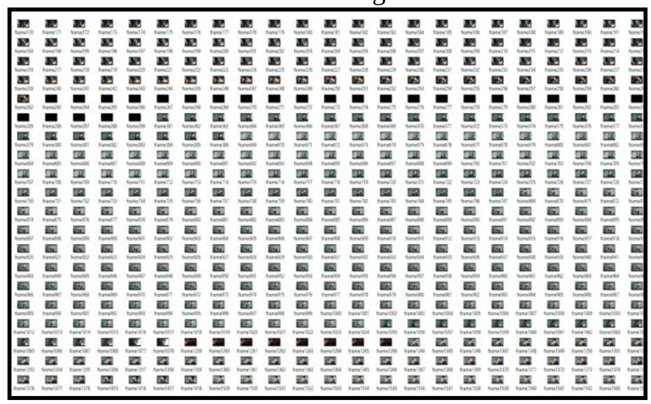

Figure 2: Smoke Dataset

2.2 Individual Smoke Annotation for Training

For annotating, the study used labelImg as shown in Figure 3. LabelImg is a resource for graphical labels of images. It is written in Python, and its graphical GUI uses Qt. Annotations are stored in PASCAL VOC format, the format ImageNet uses as XML data. In addition, it also supports YOLO format.

Figure 3: Annotation using LabelImg

2.3 Splitting Dataset

A. Training Dataset

The images were divided into 80:20 giving the training data set 80 percent of the total images as shown in Figure 4. The images are trained as 1 as smoke and 0 as not smoke.

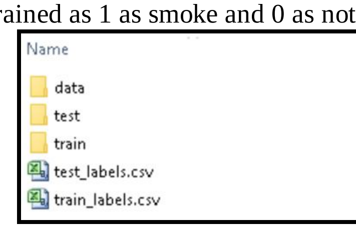

Figure 4: Splitting Dataset

B. Testing Dataset

The images were divided into 80:20 giving the training data set 20 percent of the total images. The images are trained as 1 as smoke and 0 as not smoke.

2.4 Training

The study used the ssd_mobilenet_v2_quantized_coco model for training. It is the most compatible model for raspberry pi. The process was completed once the loss is below 2.0 as shown in Figure 5.

```
INFO:tensorflow:global step 1: loss = 36.0484 (40.203 sec/step)
INFO:tensorflow:global step 2: loss = 28.4704 (1.179 sec/step)
INFO:tensorflow:global step 3: loss = 26.5873 (1.166 sec/step)
INFO:tensorflow:global step 4: loss = 23.8073 (1.173 sec/step)
INFO:tensorflow:global step 5: loss = 23.1436 (2.968 sec/step)
INFO:tensorflow:global step 6: loss = 21.2458 (1.183 sec/step)
INFO:tensorflow:global step 7: loss = 20.0173 (1.173 sec/step)
INFO:tensorflow:global step 8: loss = 18.3241 (1.192 sec/step)
INFO:tensorflow:global step 9: loss = 17.8512 (3.174 sec/step)
INFO:tensorflow:global step 10: loss = 16.4809 (1.196 sec/step)
INFO:tensorflow:global step 11: loss = 15.2043 (1.181 sec/step)
INFO:tensorflow:global step 12: loss = 13.9645 (4.041 sec/step)
INFO:tensorflow:global step 13: loss = 13.9639 (1.181 sec/step)
INFO:tensorflow:global step 14: loss = 12.9485 (1.172 sec/step)
INFO:tensorflow:global step 15: loss = 11.7583 (1.167 sec/step)
INFO:tensorflow:global step 16: loss = 11.5966 (1.178 sec/step)
INFO:tensorflow:global step 17: loss = 10.6319 (1.210 sec/step)
INFO:tensorflow:global step 18: loss = 11.1104 (1.187 sec/step)
INFO:tensorflow:global step 19: loss = 9.9175 (1.180 sec/step)
INFO:tensorflow:global step 20: loss = 10.6300 (1.197 sec/step)
```

Figure 5: Training Process

2.5 Model

This provides a common framework for the different programming models as shown in Fig. 6. After having a test, the model must be always implemented for a particular instruction set providing the same programming model for the smoke.

Name	Date modified
model.ckpt-19320.meta	7 Feb 2020 1:00 AM
model.ckpt-19451.data-00000-of-00001	7 Feb 2020 1:09 AM
model.ckpt-19451.index	7 Feb 2020 1:10 AM
model.ckpt-19451.meta	7 Feb 2020 1:10 AM
model.ckpt-19582.data-00000-of-00001	7 Feb 2020 1:19 AM
model.ckpt-19582.index	7 Feb 2020 1:20 AM
model.ckpt-19582.meta	7 Feb 2020 1:20 AM
model.ckpt-19711.data-00000-of-00001	7 Feb 2020 1:29 AM
model.ckpt-19711.index	7 Feb 2020 1:30 AM
model.ckpt-19711.meta	7 Feb 2020 1:30 AM
model.ckpt-19842.data-00000-of-00001	7 Feb 2020 1:39 AM
model.ckpt-19842.index	7 Feb 2020 1:40 AM
model.ckpt-19842.meta	7 Feb 2020 1:40 AM
model.ckpt-19973.data-00000-of-00001	7 Feb 2020 1:49 AM
model.ckpt-19973.index	7 Feb 2020 1:50 AM
model.ckpt-19973.meta	7 Feb 2020 1:50 AM
model.ckpt-20103.data-00000-of-00001	7 Feb 2020 1:59 AM
model.ckpt-20103.index	7 Feb 2020 2:00 AM
model.ckpt-20103.meta	7 Feb 2020 2:00 AM
model.ckpt-20234.data-00000-of-00001	7 Feb 2020 2:10 AM
model.ckpt-20234.index	7 Feb 2020 2:10 AM
model.ckpt-20234.meta	7 Feb 2020 2:10 AM
pipeline.config	6 Feb 2020 12:18 A...
ssd_mobilenet_v2_quantized_300x300_co...	29 Jan 2020 1:12 AM

Figure 6: Model

2.6 Converting TFlite

Fig. 7 below shows that the TensorFlow in Windows must be converted into TensorFlow Lite or tflite for the model to be run in the Raspberry Pi. The label map as shown in Fig. 8 must be configured accordingly.

Name	Date modified	Type	Size
detect.tflite	20 Jan 2020 8:07 PM	TFLITE File	4,601 KB
labelmap	20 Jan 2020 8:06 PM	Text Document	1 KB
tflite_graph.pb	10 Feb 2020 8:56 PM	PB File	18,633 KB
tflite_graph.pbtxt	10 Feb 2020 8:56 PM	PBTXT File	52,056 KB
tflite_graph	10 Feb 2020 8:59 PM	Compressed (zipp...	40,690 KB

Figure 7: TFLite Model

labelmap.pbtxt - Notepad

File Edit Format View

```
item {
  id: 1
  name: "smoke"
}
```

Figure 8: Label Map

2.7 Detected Smoke

After training, the trained model can now be use and test using the pi-camera or webcam as shown in Fig. 9. It should detect smoke and inform the user by announcing the "smoke detected" sound for every ten counts of detection via speaker or buzzer as shown in Fig. 10.

Figure 9: Detected Smoke

Python 3.6.10 Shell

File Edit Shell Debug Options Window Help

```
1
2
3
4
5
6
7
8
9
10
```

Figure 10: Announcing "smoke detected" for every ten counts of detection

3. HARDWARE DESIGN

The study created a 3D design prototype of the simulated home-environment as shown in Fig. 11. During the process of creating the design, the design took considerations to the size of the case so that the camera would fit and would cover the entire prototype to be detected.

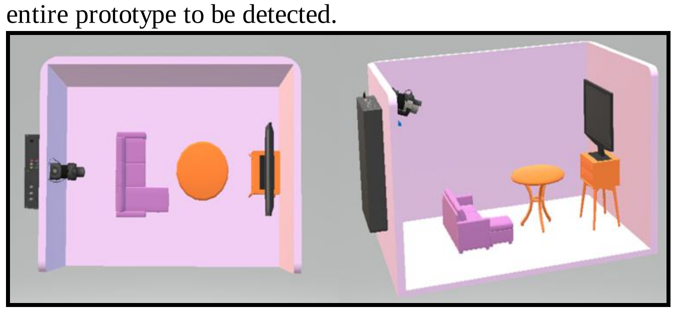

Figure 11: 3D Design Prototype

Figure 12 shows that the actual prototype used acrylic glass for the case and foam for the seats and Sintra board for the table, TV and the table. The size of the case is 15 by 10 by 7 inches.

Figure 12: Actual Prototype

A. Raspberry Pi 4B

A Raspberry Pi 4B works like a computer; by copying the model script and other files needed, the study executed the python script and can now detect the smoke.

B. Raspberry Pi-Camera

A Pi-Camera is installed at the corner ceiling of the room where it can be monitored. It is connected to the Raspberry Pi 4B via ribbon where the smoke detection is done. To work properly, it needs to be positioned properly.

C. Live Feed

After running the python script, a live feed will appear and ready to detect smoke whenever possible.

4. RESULTS AND DISCUSSIONS

A. Training Results

The total loss value given is a sum of the classification loss and the localization loss as shown in Fig. 13. The optimization algorithms are trying to reduce these loss values until the researchers are satisfied with the results and considered the model 'trained'. Generally, the lower the score the better the model.

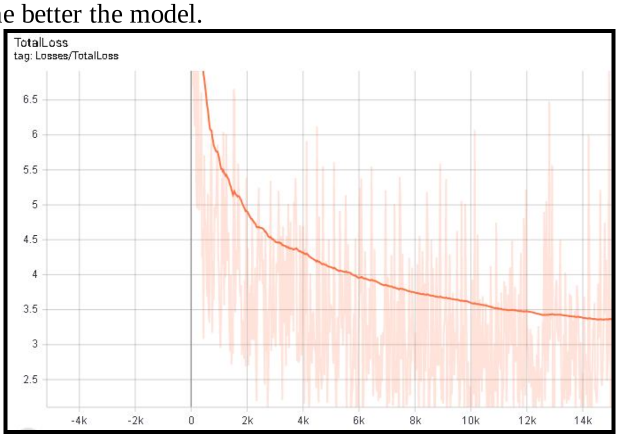

Figure 13: Total Loss

The X-axis accounts for the number of phases while the Y-axis reflects the overall loss rate. The study used a loss function to assess how much the expected values deviate from the observed training data values. Table 1 shows the loss value at each step in the training.

Table 1: Total Loss

Wall Time	Step	Loss Value
1.58E+09	14419	3.598059
1.58E+09	14445	2.372559
1.58E+09	14471	2.30938
1.58E+09	14498	2.165246
1.58E+09	14523	2.459556

B. Testing Results

The system detects the smoke inside or outside the case, as long as the smoke is inside the range of the camera. Through this, the device automatically detects the smoke and it will inform the user of the detected smoke. It will trigger the alarm "smoke detected" continuously as long as it detects smoke. Fig. 14 shows the testing results using the model.

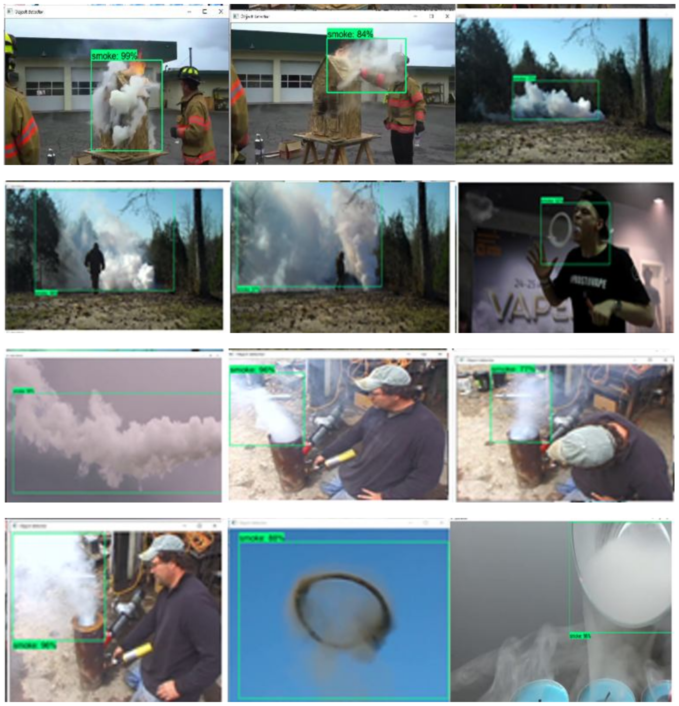

Figure 14: Testing Results

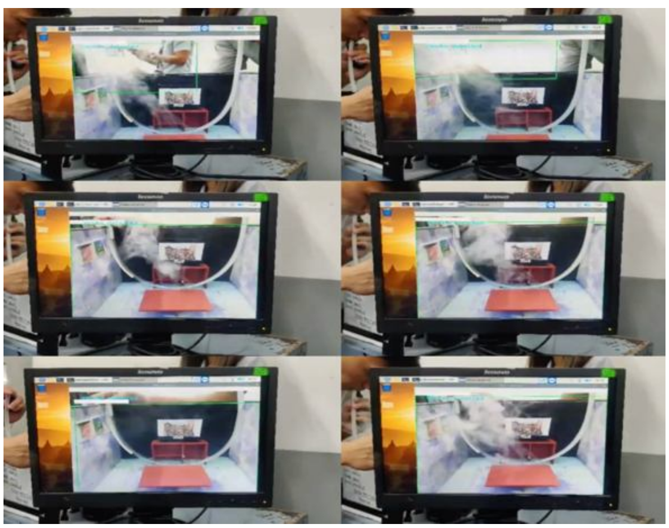

Figure 15: Testing Results using the Smoke-Induced Simulated Home-Environment

Table 2: Testing Results

Test #	Expected Detection	Actual Output (% Detection)
1	Smoke	95%
2	Smoke	99%
3	Smoke	84%
4	Smoke	86%
5	Smoke	71%
6	Smoke	88%
7	Smoke	69%
8	Smoke	76%
9	Smoke	99%
10	Smoke	84%
11	Smoke	77%
12	Smoke	97%
13	Smoke	92%
14	Smoke	95%
15	Smoke	98%
16	Smoke	96%
17	Smoke	97%
18	Smoke	88%
19	Smoke	98%
20	Smoke	81%
21	Smoke	92%
22	Smoke	77%
23	Smoke	96%
24	Smoke	92%
25	Smoke	96%
26	Smoke	99%
27	Smoke	78%
28	Smoke	88%
29	Smoke	98%
30	Smoke	98%
	Mean Average	**89%**

Table 2 shows the testing results and Fig. 15 shows the actual testing of the system using the simulated-home environment. The study was able to get an 89% overall success rate in detecting the smoke.

5. CONCLUSION AND FUTURE WORKS

In this study, the system has presented an alternative way for people to prevent fire by detecting early smoke that can cause a big fire after a long time. The existing devices and equipment that can prevent fire are lacking as they detect fire only when the smoke is very thick and if it reaches the ceiling where the smoke detectors are placed but with this, the detection of smoke doesn't need to be thick and high and fire can be prevented early by informing the user that smoke has been detected. This study proved an 89% overall testing accuracy.

In this study, the detection of smoke covers a minimal range of coverage making all of the smoke outside the area (like those in wide-area open parking lots) hard to detect. This issue can be solved by the help of the following modifications: First, changing the model where the dataset images are far from the camera or adding more dataset images of smoke that are long-ranged, secondly, by using other algorithms in detecting smoke that is capable of detecting smoke far from the camera.

ACKNOWLEDGEMENT

The author would like to extend the sincere thanks to his Microprocessor's Class of 2nd SEM SY19-20. Additionally, to MR-SUAVE Lab and Microprocessors of the Technological Institute of the Philippines.

REFERENCES

1. I. Sutton, **Emergency Management**, *Process Risk and Reliability Management*, pp. 519-537, 2015.
2. N. Chen, T. Zhang and Z. Mei, **Experimental Study on the Response Characteristics of Photoelectric Smoke Detectors Below Grid Ceilings of Different Hollowing Rate**, *Procedia Engineering*, vol. 45, pp. 602-609, 2012. https://doi.org/10.1016/j.proeng.2012.08.210
3. K. Moinuddin, D. Bruck and L. Shi, **An experimental study on timely activation of smoke alarms and their effective notification in typical residential buildings**, *Fire Safety Journal*, vol. 93, pp. 1-11, 2017.
4. Y. He, J. Wang, Z. Wu, L. Hu, Y. Xiong and W. Fan, **Smoke venting and fire safety in an industrial warehouse**, *Fire Safety Journal*, vol. 37, no. 2, pp. 191-215, 2002. https://doi.org/10.1016/S0379-7112(01)00045-5
5. S. Griffiths, P. Chappell, J. Entwistle, F. Kelly and M. Deary, **A study of particulate emissions during 23 major industrial fires: Implications for human health**, *Environment International*, vol. 112, pp. 310-323, 2018.
6. O. Willstrand, J. Brandt and R. Svensson, **Detection of fires in the toilet compartment and driver sleeping compartment of buses and coaches—Installation considerations based on full scale tests**, *Case Studies in Fire Safety*, vol. 5, pp. 1-10, 2016. https://doi.org/10.1016/j.csfs.2015.11.002
7. A. Surjan, Atta-ur-Rahman, G. Parvin and R. Shaw, **Urban Regulation and Enforcement**, *Urban Disasters and Resilience in Asia*, pp. 69-78, 2016.
8. L. Qiang, **Estimation of Fire Detection Time**, *Procedia Engineering*, vol. 11, pp. 233-241, 2011. https://doi.org/10.1016/j.proeng.2011.04.652
9. S. Kubba, **Impact of Energy and Atmosphere**, *LEED v4 Practices, Certification, and Accreditation Handbook*, pp. 409-518, 2016.
10. Li Jinghong, Zou Xiaohui and Wang Lu, **The design and implementation of fire smoke detection system based on FPGA**, *2012 24th Chinese Control and Decision Conference (CCDC)*, 2012. https://doi.org/10.1109/CCDC.2012.6244626
11. R. Kaabi, S. Frizzi, M. Bouchouicha, F. Fnaiech and E. Moreau, **Video smoke detection review: State of the art of smoke detection in visible and IR range**, *2017 International Conference on Smart, Monitored and Controlled Cities (SM2C)*, 2017.
12. A. Filonenko, D. Hernandez, Wahyono and K. Jo, **Smoke detection for surveillance cameras based on color, motion, and shape**, *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, 2016.
13. Y. Deldjoo, F. Nazary and A. Fotouhi, **A novel fuzzy-based smoke detection system using dynamic and static smoke features**, *2015 23rd Iranian Conference on Electrical Engineering*, 2015.
14. S. Vijayalakshmi and S. Muruganand, **Smoke detection in video images using background subtraction method for early fire alarm system**, *2017 2nd International Conference on Communication and Electronics Systems (ICCES)*, 2017. https://doi.org/10.1109/CESYS.2017.8321258
15. S. Tirpude, **Abnormal X-Ray Detection System using Convolution Neural Network**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 1, pp. 828-832, 2020. https://doi.org/10.30534/ijatcse/2020/119912020
16. A. Alsufyani, **Detection of single-trial EEG of the neural correlates of familiar faces recognition using machine-learning algorithms**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 6, pp. 2855-2860, 2019. https://doi.org/10.30534/ijatcse/2019/28862019
17. L. S.K., **Automated Leaf Disease Detection in Corn Species through Image Analysis**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 6, pp. 2893-2899, 2019. https://doi.org/10.30534/ijatcse/2019/34862019
18. A. Alon, "**A YOLOv3 Inference Approach for Student Attendance Face Recognition System**", *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 2, pp. 384-390, 2020. https://doi.org/10.30534/ijeter/2020/24822020