Comparative study of teachable machine for forest fire and smoke detection by drone

Mounir Grari[1], Mimoun Yandouzi[2], Berrahal Mohammed[3], Mohammed Boukabous[1], Idriss Idrissi[1]
[1]Mathematics, Signal and Image Processing, and Computing Research Laboratory (MATSI), Ecole Supérieure de Technologie Oujda (ESTO), Mohammed First University, Oujda, Morocco
[2]LSI Research Lab., École Nationale des Sciences Appliquées d'Oujda (ENSAO), Mohammed First University, Oujda, Morocco
[3]Modeling and Combinatorics Laboratory, Polydisciplinary Faculty of Safi (PFS), Cadi Ayyad University, Safi, Morocco

Article Info

ABSTRACT

Forests play a vital role in maintaining ecological equilibrium and serving as vital habitats for wildlife. They regulate global climate, safeguard soil and water resources, and provide crucial ecosystem services such as air and water purification, essential for human well-being and sustainable development. Forest fires wreak havoc on ecosystems and wildlife, emitting harmful pollutants, disrupting communities, and increasing the risk of erosion and landslides. Detecting forest fires through satellite imaging, aerial reconnaissance, and ground-based sensors is pivotal for early detection and containment, safeguarding human lives, wildlife, and preserving natural resources for future generations. Utilizing drones and deep learning (DL) algorithms can significantly enhance early fire detection and minimize their devastating impact. In this paper, we examine teachable machine, a Google tool for creating DL models. We compare the top model generated by teachable machine for fire and smoke detection to models obtained through transfer learning from established DL models in image recognition and computer vision (CV), such as VGG16, VGG19, MobileNet, MobileNetv2, and MobileNetv3. The results underscore the significance of employing the teachable machine model in specific fire and smoke detection scenarios.

Corresponding Author:

Mounir Grari
Mathematics, Signal and Image Processing, and Computing Research Laboratory (MATSI)
Ecole Supérieure de Technologie Oujda (ESTO), Mohammed First University
Oujda 60000, Morocco
Email: m.grari@ump.ac.ma

1. INTRODUCTION

Forests play a crucial role in regulating the climate by absorbing carbon dioxide from the atmosphere and producing oxygen, as well as providing habitats for a wide variety of plant and animal species. They are also important sources of forest products such as wood, leaves, fruits, and seeds, which are used in a variety of consumer products such as paper, furniture, and medicines, as well as ecosystem services such as flood control and soil erosion protection. Forest fires have become an increasing phenomenon due to climate change, deforestation, and human expansion into forest areas, resulting in significant economic losses as well as significant damage to the environment and biodiversity. They can also have serious consequences for human health due to the release of fine particles and air pollutants, increasing the risk of respiratory and cardiac diseases.

Early detection of wildfires is crucial to limiting the spread of fires and protecting surrounding populations and property, allowing for rapid response by emergency responders. Early wildfire detection

systems, such as smoke sensors and surveillance cameras, can detect fires in their early stages, increasing the chances of controlling them before they get out of control [1]. The use of drones in the environmental field has evolved significantly in recent years, with increasing adoption in the detection and monitoring of forest fires, allowing for more efficient and accurate data collection and response efforts. Drones equipped with different kinds of cameras have proven to be highly effective in detecting and mapping forest fires, providing real-time data and allowing for quick and efficient deployment of firefighting resources. Deep learning (DL) algorithms can be used to process images captured by drones, such as image classification, object detection and image segmentation [2].

In our research paper, we explore the use of the DL solution developed by Google that allows users to create image or sound recognition models using their own training data. We generate a variety of models by modifying the hyperparameters. The best resulting model is then compared with models derived by transfer learning from DL models that have confirmed their effectiveness in image recognition and computer vision (CV) in general, such as VGG16, VGG19, MobileNet, MobileNetv2, and MobileNetv3 [3].

The remainder of the paper is structured as follows: the second section delves into the techniques utilized, the third section provides a concise summary of the related literature, the fourth section introduces the proposed methodologies, the fifth section scrutinizes and deliberates upon the study's findings, and the conclusion serves as the final segment.

2. BACKGROUNDS
2.1. Computer vision
CV is a domain within artificial intelligence that empowers machines to interpret and interact with their surroundings using image and video analysis. It involves techniques such as image analysis, signal processing, pattern recognition, and motion recognition to comprehend received images. CV has many applications, including facial recognition, speech recognition, video surveillance, and robotics. It is also utilized in industrial vision systems to inspect products and improve quality. Navigation systems for drones and autonomous vehicles use it to detect and avoid obstacles [4]. DL has recently been successfully applied to CV, allowing machines to better understand complex data structures and patterns. CV includes classification, segmentation, and object detection.

− Classification aims to classify images or videos by assigning a label or category based on visual characteristics. This can be applied to identify objects, people, animals, scenes, or actions. Techniques include image recognition, pattern recognition, motion recognition, and speech recognition. These methods employ various algorithms such as neural networks, pattern detection, and motion tracking to extract essential features and facilitate recognition.
− Segmentation is used to isolate different elements in an image or video. This aids in image recognition, object detection, and video segmentation. Common techniques include region-based, color-based, depth-based, and video-based segmentation. Region-based segmentation separates an image into regions based on visual characteristics, while color segmentation separates objects by color. Depth segmentation separates objects by depth, and video segmentation uses motion tracking to separate moving objects from static ones.
− Object detection involves identifying objects in images or videos by locating their boundaries and assigning them a label or category. Common techniques for object detection include region detection, feature detection, neural network detection, and motion detection. Combined approaches that use multiple techniques can enhance the accuracy and robustness of object detection.

2.2. Deep learning
DL is a branch of machine learning that empowers computers to uncover patterns and insights within data. It's a highly efficient method for crafting advanced models with the ability to learn and identify objects, textures, and patterns within datasets. It involves training artificial neural networks with multiple hidden layers to learn intricate patterns and representations from data, distinguishing it from traditional neural networks. DL has revolutionized CV, allowing machines to excel in tasks like object recognition, image segmentation, and scene reconstruction [5]. Its potential is still being explored, and it continues to push the boundaries of what machines can achieve in fields such as natural language processing, speech recognition, and autonomous driving.

2.2.1. Convolutional neural networks
A convolutional neural networks (CNN) is a deep neural network for processing images and videos. It uses convolution and pooling layers to extract visual features and reduce image size. CNNs detect features like edges and textures, and select important features for classification. They can be trained on large datasets

to improve accuracy and are used in areas such as speech recognition, image recognition, object detection, machine translation, and speech understanding [6].

a. MobileNet, MobileNetv2, and MobileNetv3

MobileNet is a family of CNN models created by Google in 2017 for use in low-power computing devices. Its design uses "filter factorization" and smaller convolution layers to decrease computation and memory usage. MobileNetV2 and MobileNetV3 are enhanced versions that improve accuracy and reduce memory consumption. MobileNetV2 uses "inverted residuals" to reduce calculations, and MobileNetV3 uses "efficient convolution." Both also use smaller layers to reduce memory usage [7].

b. VGG16 and VGG19

VGG16 and VGG19 are CNN designs created by the DL visualization team at the University of Oxford. They won the 2014 ImageNet contest in the "object localization" category. VGG16 has 16 convolution layers, while VGG19 has 19, with each group of layers having a higher depth than the previous one. The final layers of both models have fully connected layers used for object classification, with VGG16 having 3 and VGG19 having 4 fully connected layers, each with 4096 neurons. The convolution layers use 3x3 size filters and include batch normalization and dropout layers, followed by max-pooling layers that reduce image dimensions. This approach captures features of different sizes in the image [8].

c. Teachable machine

Teachable machine is an online tool developed by Google to create image and sound recognition models using machine learning. It is designed to be used by people without programming or artificial intelligence knowledge. It works by using a three-step learning system [9]:

− Registration: the user records images or sounds in different categories to train the model.
− Learning: the model uses this data to learn to recognize the different categories.
− Testing: the user can then test the model by providing it with new images or sounds to see how it performs.

The teachable machine is a free online tool that creates models for image and sound recognition. It's versatile, integrating easily into projects like mobile apps, games, robots, and art installations.

2.2.2. Transfer learning

Transfer learning is a method in machine learning where a pre-trained model's weights and parameters are reused to train a new model on data that may differ from the initial training data [10]. This approach aims to enhance the performance of the new model by leveraging the knowledge gained from the previous training. There are several types of transfer learning, including:

− Fine-tuning transfer learning, which involves taking a pre-trained model and adapting it to the specific data by training new weights for certain layers of the model.
− Feature extraction transfer learning, which consists in using the pre-trained weights of a model to extract relevant features from the training data and to use them to train a new model on similar data.
− Multi-task learning, which consists in using the knowledge acquired during the training of a model on a task to improve the performance of another model on a similar or related task.

Transfer learning is particularly useful when training data is limited or expensive to obtain, or when training and test data are different.

2.3. Drones/unmanned aerial vehicles

Drones or unmanned aerial vehicles (UAV) have diverse applications such as wildlife monitoring, flood zone mapping, industrial emissions monitoring, power grid inspection, and forest fire monitoring [11]. They offer benefits like quick coverage of large areas, access to remote locations, low altitude and night flight capability, thermal imaging, and real-time information for effective response [2]. Drones complement sensor-based architectures by swiftly confirming potential fires detected through telemetry [12]. Additionally, drones provide a cost-effective and eco-friendly alternative to traditional surveillance methods, reducing the need for human intervention and minimizing environmental impact.

3.   RELATED WORK

Aslan *et al.* [13] suggested using deep convolutional generative adversarial neural networks (DC-GANs) for smoke detection. The training framework is designed to provide a robust representation of both smoke and non-smoke sequences. We attain this by regularly training a DC-GAN with authentic images and noise vectors, while also separately training the discriminator solely with smoke images, without the generator's participation. The proposed method demonstrates high accuracy in identifying smoke images in real-time, with a TNR of 99.45% and a TPR of 86.23%, resulting in very few false positive results.

Wang *et al.* [14] developed a system for identifying forest fire images through a combination of conventional image processing techniques and CNN. The system incorporates an adaptive pooling method for automated fire recognition. Through this approach, the distinctive characteristics of fire flames can be extracted and learned beforehand. Results from experiments have demonstrated that this adaptive pooling approach, using CNN, outperforms traditional methods and has a recognition accuracy of up to 90.7%.

Chen *et al.* [15] introduced a new method for detecting forest fires using UAVs that is based on CNN. This approach aims to detect potential fires at their early stages. The proposed fire detection system's effectiveness was assessed through tests involving simulated flames in a controlled indoor setting, and the outcomes confirmed its efficacy.

In their study, Yin *et al.* [16] utilized a recurrent neural network (RNN) architecture to capture data related to the smoke region and movement context. Results from various publicly available benchmarks showed that their method achieved superior performance compared to other methods.

Alves *et al.* [17] proposed an automated system for forest fire detection which was able to classify 695 photos taken in daylight settings with an accuracy of 94.1% and 187 images taken during nighttime with an accuracy of 94.8%. Another DL system for forest fire detection was suggested by Kansal *et al.* [18], which achieved a detection accuracy of 90% when tested on various datasets.

4. PROPOSED METHOD
4.1. Proposed architecture

Our solution suggests employing drones and DL algorithms to detect wildfires in their early stages. By flying over forests, the drones can locate hot spots that have the potential to start a wildfire. The DL algorithm can then analyze the data collected by the drones to confirm whether or not a hot spot is an actual wildfire. If a wildfire is detected, the algorithm can alert the authorities by using a server cloud (Figure 1). The utilization of drones and DL algorithms in detecting forest wildfires presents several benefits compared to traditional methods. To start with, drones can swiftly cover extensive areas, surpassing the speed of ground crews. Furthermore, the precision in identifying wildfires sees a remarkable boost when employing DL algorithms as opposed to human observers. Finally, the integration of drones and DL algorithms can bolster the safety of firefighters and other first responders by detecting wildfires before they evolve into perilous situations.
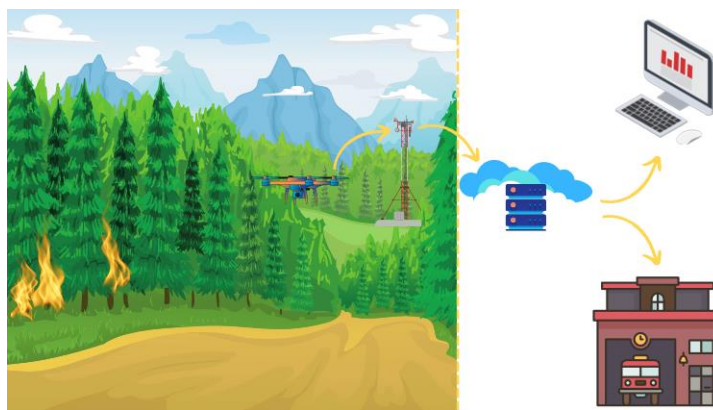

Figure 1. Proposed architecture

In this article we compare the model generated for the detection of fire and smoke by the teachable machine solution to the most common models used in the world of CV and which have demonstrated their effectiveness in previous works.

4.2. Dataset

Obtaining datasets that are both large and diverse is a frequent obstacle in DL [19]. Images serve as a prime illustration of this kind of dataset. Image data comprises information that can be employed to train deep neural networks for various tasks, including classification, recognition, and object detection [3].

Our model training dataset is comprised of numerous images of forest fires with and without smoke, captured in various locations globally, as well as images of non-fire forest landscapes. The dataset was built

by combining various smaller datasets from search engines and Kaggle [20]. After cleaning up some corrupted images, the final dataset consists of 5902 images with labels "No Fire No Smoke" (2252 images), "Fire No Smoke" (1136 images), "Smoke No Fire Smoke" (758 images), and "Smoke and Fire" (1756 images) (see Figure 2).

Figure 2. The four database labels

Subsequently, we implemented data augmentation on the dataset, resulting in a substantial enhancement of the training dataset and ultimately, the accuracy of the models trained [21]. The techniques utilized in data augmentation included rotation, horizontal and vertical mirroring, zoom crop, and random adjustments to the brightness, contrast, hue, saturation, and value of the images. This technique generated similar yet slightly altered data, which positively impacted the learning ability of the neural networks and improved their accuracy levels [22].

MobileNet and VGG models and their variants will be used by the transfer learning technique. The resulting models will be constructed by initially training them on a substantial dataset, like the ImageNet dataset. These models will then be used as "base models" for another model trained on our dataset [23].

4.3. Building the models

To implement the teachable machine solution, we will adhere to the traditional steps. We will choose a suitable image recognition model, and use our labeled and preprocessed dataset to train it. 80% of our images will be used for training and the remaining 20% will be utilized for testing. We experimented with around 20 different models, not by selecting a base model, but by modifying the hyperparameters-these are parameters set prior to training, as opposed to those learned during the training process. The hyperparameters we adjusted include the number of hidden layers, the number of neurons per layer, the learning rate, the activation function, batch size, and regularization. The results we will showcase for teachable machine are the most favorable outcomes obtained from all the varied models.

For the MobileNet, VGG, and their derivatives, we will implement the transfer learning technique of fine-tuning. This involves adapting a pre-existing model to a new task by using a limited amount of data that is specific to the task. In our scenario, the task-specific data will be the small database we have collected. The pre-trained model is usually one that has been trained on a vast dataset for a similar but different task. The concept behind fine-tuning is that the model has already acquired valuable features and representations from its original task, which can be utilized for the new task, thus decreasing the need for large amounts of data and computational resources during training. Fine-tuning can be accomplished by training the final layers of the pre-trained model using only the task-specific data or by training the entire model using a combination of task-specific data and pre-trained data (Figure 3).
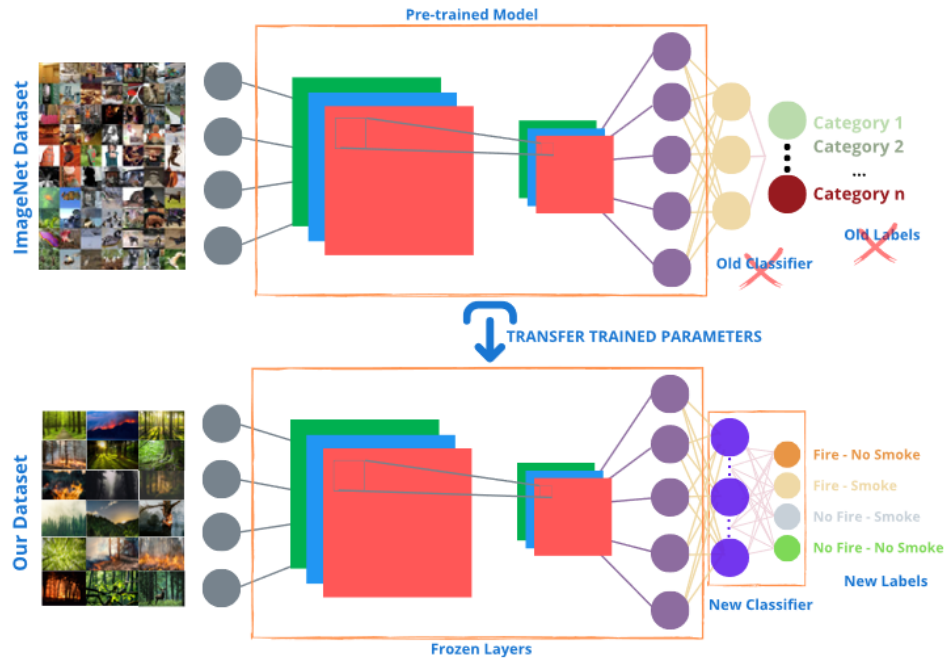
Figure 3. Transfer learning technique-fine-tuning

5.   RESULTS AND DISCUSSION
5.1. Hardware characteristics
For training and evaluating our DL models, we utilized TensorFlow on a computing system with high performance, featuring the following hardware specs:
− Dual Intel Gold 6148 CPUs with 20 cores each, running at 2.4 GHz
− A pair of NVIDIA Tesla V100 GPUs, each equipped with 32 gigabytes of memory
Our experiments utilized TensorFlow version 2.11.0, originally created by the Google Brain team in 2015, TensorFlow offers an extensive set of functionalities for data analysis and machine learning, encompassing DL, numerical computations, linear algebra, and graph processing [24].

5.2. Evaluation metrics
Having an appropriate evaluation metric is crucial for identifying the optimal model during the training process [25]. The assessment of DL models requires the use of specific metrics, such as accuracy, precision, recall, F1 score, and loss, four measures are utilized to compute these metrics [26]:
− True positive (TP): the count of correctly classified records in the positive class.
− True negative (TN): the count of correctly classified records in the negative class.
− False positive (FP): the count of incorrectly classified records in the positive class.
− False negative (FN): the count of incorrectly classified records in the negative class.

5.2.1. Accuracy
Accuracy quantifies the ratio of accurately classified items within a classification task. This is a commonly employed assessment metric, computed by dividing the count of accurate predictions by the total count of predictions generated [2]. The accuracy is defined as (1):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

5.2.2. Precision
Precision evaluates the accuracy of a model in recognizing true positive results. This means that the model correctly classifies positive cases [27]. A model exhibiting a strong precision score will effectively detect the majority of true positives, while a model with low precision will incorrectly classify numerous positive instances as negative. Precision is formally defined as (2):

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

5.2.3. Recall

The term "recall" refers to the proportion of positive cases that were correctly identified by a prediction model. It is also referred to as the "true positive rate" or "sensitivity". A model with a high recall is one that is effective in detecting the majority of positive instances [27]. A model that has poor recall is not useful because it tends to categorize the majority of actual positive cases as negative. The recall metric is defined as (3):

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Generally, recall should be combined with other performance measures, such as precision and accuracy, to obtain a comprehensive understanding of a model's effectiveness.

5.2.4. F1 score

The F1 score assesses a model's accuracy by factoring in both precision and recall. It is computed as the harmonic mean of precision and recall, considering both the count of correctly predicted positives in comparison to the total count of both true positives and false positives (precision), and the count of correctly predicted positives in comparison to the total count of both true positives and false negatives (recall) [27]. The formula for determining the F1 score is (4):

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

5.2.5. Loss

The loss metric is a way of determining the deviation of the algorithm from the expected outcome. The higher the loss, the poorer the algorithm is performing. Conversely, a low loss indicates better performance [28]. A popular loss metric for classification issues is the cross-entropy loss, which is defined as (5):

$$Loss = -\sum_i y_i * \log(p_i) \tag{5}$$

where y_i is the true label and p_i is the predicted probability of the true label. The cross-entropy loss evaluates the classifier's performance by imposing a penalty on incorrect predictions. The classifier is considered to be making more incorrect predictions if the cross-entropy loss is higher.

5.2.6. The number of parameters

In the realm of DL, the term "parameters" pertains to the variables employed in shaping the structure of a neural network. These variables may include weights, biases, activation function type, or network size, and they determine the model's capacity. Models with more parameters can capture more complex patterns compared to those with fewer parameters [29]. Additionally, the quantity of parameters impacts the memory prerequisites for model storage. A model boasting an extensive array of parameters necessitates a greater amount of memory as compared to a model featuring a reduced parameter count [30].

5.3. Evaluating the results

In Table 1, the results of the implemented models are summarized. All models were able to classify images with varying degrees of success. The VGG models performed the best, particularly VGG16 which achieved an accuracy of 99.61%, a precision of 99.42%, a recall of 99.69%, a best F1-score of 99.55%, and a loss of 0.49%. In our case study, VGG16 outperformed VGG19 despite having fewer layers. The MobileNet models perform less well than VGG but with quite small numbers of parameters especially MobileNetV3 which has less than 3.2 million parameters. The teachable machine model achieved average results, with an accuracy of 97.93%, a precision of 98.13%, a recall of 97.29%, a F1-score of 97.71%, and a loss of 2.49%. This model was lighter compared to the VGG models, but slightly heavier than the MobileNet models, with almost 6 million parameters. The training phase for this model took longer, around 40 epochs, as opposed to the other models which converged before 18 epochs. This was because the other models were pre-trained (see Figure 4).

The teachable machine model can be used for image classification on board the UAV or from a terrestrial server, as it is effective at solving image classification problems at an acceptable speed, with good performance, given few constraints on processing time and hardware resources. It remains to be proven whether it can effectively address highly specialized or resource-intensive applications, as it may be necessary to employ more advanced tools and models for such tasks.

Table 1. Obtained results for the implemented models

DL algorithm	Number of parameters	Loss (%)	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
VGG16	16 813 923	0.49	99.61	99.42	99.69	99.55
VGG19	23 103 797	0.48	99.54	99.53	99.51	99.52
Teachable machine	6 152 385	2.49	97.93	98.13	97.29	97.71
MobileNet	4 352 879	5.24	97.23	97.33	98.03	97.68
MobileNetV2	3 514 705	9.53	95.35	95.74	96.57	96.15
MobileNetV3	3 104 277	10.17	95.07	95.63	95.14	95.38
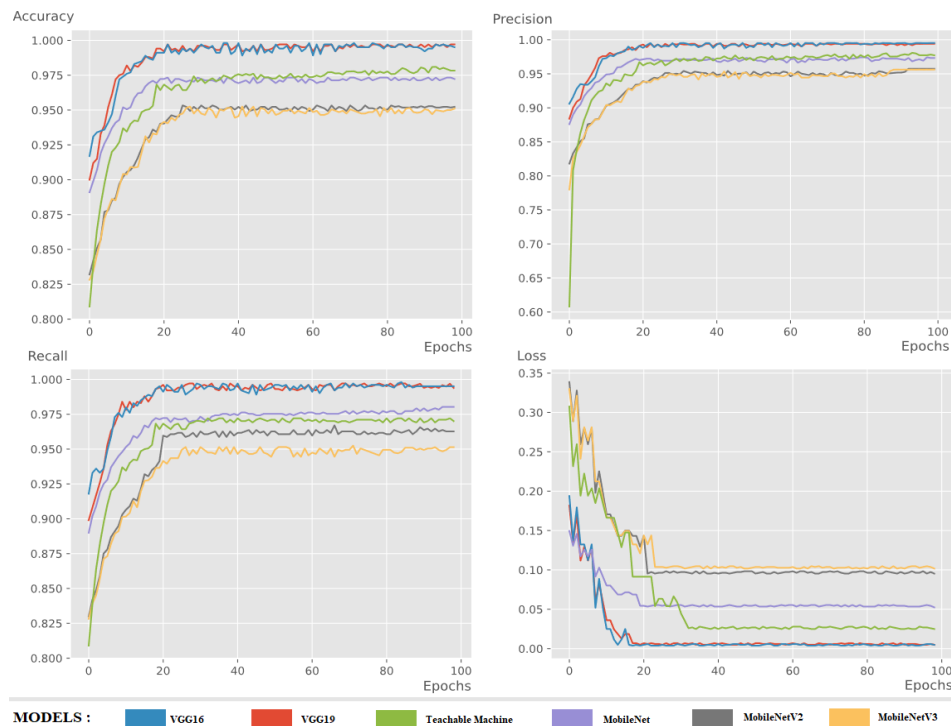
Figure 4. Achieved accuracy, precision, recall, and loss for the implemented models

6. CONCLUSION

The revolution of CV caused by DL has dramatically changed the way we interact with technology and process information, transforming it from a purely passive task to a much more intelligent and active process that can be customized for specific applications and uses, leading to a vast improvement in the accuracy and efficiency of object detection, image classification, and video analysis, providing new and exciting opportunities for innovation and growth in a wide range of industries. The emergence of solutions like Google's teachable machine, which allow individuals with minimal programming knowledge to generate DL models, is a significant step towards democratizing artificial intelligence. In this paper, we made a comparison between the results of the best model generated via teachable machine, for the detection of forest fires by drones' images, with models having demonstrated their efficiency in image classification and CV in general like MobileNet and VGG. The performance of the teachable machine model is average, boasting a 97.93% accuracy, a 97.71% F1 score, and a loss of 2.49%. Although VGG-based models are highly efficient, they are too resource-intensive and better suited for ground-based drone image processing. For in-drone processing, MobileNet-based models are more appropriate due to their lighter weight. In future work, we plan to explore other CV techniques such as segmentation and object detection to more precisely identify fire locations, sizes, and post-fire effects.

REFERENCES

[1] A. A. A. Alkhatib, "A Review on Forest Fire Detection Techniques," *International Journal of Distributed Sensor Networks*, vol. 2014, pp. 1-12, 2014, doi: 10.1155/2014/597368.

[2] M. Grari *et al.*, "Using Iot and Ml for Forest Fire Detection, Monitoring, and Prediction: a Literature Review," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 19, pp. 5445–5461, 2022.

[3] M. Yandouzi *et al.*, "Review on Forest Fires Detection and Prediction Using Deep Learning and Drones," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 12, pp. 4565–4576, 2022.

[4] M. Yandouzi *et al.*, "Investigation of Combining Deep Learning Object Recognition with Drones for Forest Fire Detection and Monitoring," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 3, pp. 377–384, 2023, doi: 10.14569/IJACSA.2023.0140342.

[5] Z. H. Zhan, J. Y. Li, and J. Zhang, "Evolutionary deep learning: A survey," *Neurocomputing*, vol. 483, pp. 42–58, 2022, doi: 10.1016/j.neucom.2022.01.099.

[6] F. H. K. Zaman, J. Johari, and A. I. M. Yassin, "Learning Face Similarities for Face Verification using Hybrid Convolutional Neural Networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 16, no. 3, pp. 1333–1342, 2019, doi: 10.11591/ijeecs.v16.i3.pp1333-1342.

[7] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv*, 2017, doi: 10.48550/arXiv.1704.04861.

[8] S. Tammina, "Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images," *International Journal of Scientific and Research Publications*, vol. 9, no. 10, pp. 143-150, 2019, doi: 10.29322/ijsrp.9.10.2019.p9420.

[9] J. J. N. Wong and N. Fadzly, "Development of species recognition models using Google teachable machine on shorebirds and waterbirds," *Journal of Taibah University for Science*, vol. 16, no. 1, pp. 1096–1111, 2022, doi: 10.1080/16583655.2022.2143627.

[10] K. Weiss, T. M. Khoshgoftaar, and D. D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, p. 9, 2016, doi: 10.1186/s40537-016-0043-6.

[11] Y. Diez, S. Kentsch, M. Fukuda, M. L. L. Caceres, K. Moritake, and M. Cabezas, "Deep learning in forestry using uav-acquired rgb data: A practical review," *Remote Sensing*, vol. 13, no. 14, p. 2837, 2021, doi: 10.3390/rs13142837.

[12] M. Yandouzi *et al.*, "A Lightweight Deep Learning Model for Forest Fires Detection and Monitoring," in *Lecture Notes in Electrical Engineering*, pp. 697–705, 2023, doi: 10.1007/978-981-19-6223-3_71.

[13] S. Aslan, U. Gudukbay, B. U. Toreyin, and A. E. Cetin, "Early Wildfire Smoke Detection Based on Motion-based Geometric Image Transformation and Deep Convolutional Generative Adversarial Networks," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing-Proceedings*, IEEE, pp. 8315–8319, 2019, doi: 10.1109/ICASSP.2019.8683629.

[14] Y. Wang, L. Dang, and J. Ren, "Forest fire image recognition based on convolutional neural network," *Journal of Algorithms and Computational Technology*, vol. 13, p. 174830261988768, 2019, doi: 10.1177/1748302619887689.

[15] Y. Chen, Y. Zhang, J. Xin, Y. Yi, D. Liu, and H. Liu, "A UAV-based Forest Fire Detection Algorithm Using Convolutional Neural Network," in *Chinese Control Conference*, IEEE, pp. 10305–10310, 2018, doi: 10.23919/ChiCC.2018.8484035.

[16] M. Yin, C. Lang, Z. Li, S. Feng, and T. Wang, "Recurrent convolutional network for video-based smoke detection," *Multimedia Tools and Applications*, vol. 78, no. 1, pp. 237–256, 2019, doi: 10.1007/s11042-017-5561-5.

[17] J. Alves, C. Soares, J. M. Torres, P. Sobral, and R. S. Moreira, "Automatic forest fire detection based on a machine learning and image analysis pipeline," in *Advances in Intelligent Systems and Computing*, pp. 240–251, 2019, doi: 10.1007/978-3-030-16184-2_24.

[18] A. Kansal, Y. Singh, N. Kumar, and V. Mohindru, "Detection of forest fires using machine learning technique: A perspective," in *Proceedings of 2015 3rd International Conference on Image Information Processing, ICIIP 2015*, IEEE, pp. 241–245, 2016, doi: 10.1109/ICIIP.2015.7414773.

[19] H. M. Nasir, N. M. A. Brahin, M. M. M. Aminuddin, M. S. Mispan, and M. F. Zulkifli, "Android based application for visually impaired using deep learning approach," *International Journal of Artificial Intelligence*, vol. 10, no. 4, pp. 879–888, 2021, doi: 10.11591/ijai.v10.i4.pp879-888.

[20] "Forest Fire | Kaggle," 2023, Avaiable online: https://www.kaggle.com/datasets/kutaykutlu/forest-fire (accessed: Apr. 08, 2023).

[21] A. F. Y. Althabhawee and B. K. O. C. Alwawi, "Fingerprint recognition based on collected images using deep learning technology," *International Journal of Artificial Intelligence*, vol. 11, no. 1, pp. 81–88, 2022, doi: 10.11591/ijai.v11.i1.pp81-88.

[22] J. Y. W. Jien, A. Baharum, S. H. A. Wahab, N. Saad, M. Omar, and N. A. M. Noor, "Age-based facial recognition using convoluted neural network deep learning algorithm," *International Journal of Artificial Intelligence*, vol. 9, no. 3, pp. 424–428, 2020, doi: 10.11591/ijai.v9.i3.pp424-428.

[23] M. Yandouzi *et al.*, "Forest Fires Detection using Deep Transfer Learning," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 8, pp. 268–275, 2022, doi: 10.14569/IJACSA.2022.0130832.

[24] "Module: tfm.nlp | TensorFlow v2.11.0," 2023, Available online: https://www.tensorflow.org/api_docs/python/tfm/nlp (accessed: Apr. 08, 2023).

[25] H. M and S. M.N, "A Review on Evaluation Metrics for Data Classification Evaluations," *International Journal of Data Mining and Knowledge Management Process*, vol. 5, no. 2, pp. 01–11, 2015, doi: 10.5121/ijdkp.2015.5201.

[26] M. Aditya, "Metrics to Evaluate your Machine Learning Algorithm | by Aditya Mishra | Towards Data Science," *Towards Data Science*, pp. 1–8, 2018, [Online]. Available: https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234 (accessed: Jun. 06, 2023).

[27] F. Cakir, K. He, X. Xia, B. Kulis, and S. Sclaroff, "Deep metric learning to rank," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 1861–1870, 2019, doi: 10.1109/CVPR.2019.00196.

[28] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," *Advances in Neural Information Processing Systems*, vol. 2018, pp. 8778–8788, 2018.

[29] M. Geiger *et al.*, "Scaling description of generalization with number of parameters in deep learning," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2020, no. 2, p. 023401, 2020, doi: 10.1088/1742-5468/ab633c.

[30] J. E. Moody, "The Effective Number of Parameters: An Analysis of Generalization and Regularization in Nonlinear Learning Systems," *Advances in Neural Information Processing Systems 4 (NIPS 1991)*, 1991.

BIOGRAPHIES OF AUTHORS

Mounir Grari has a Ph.D. in Computer Engineering at Mohammed First University in Oujda, Morocco, where he is conducting research on the use of the internet of things and machine learning in the detection and monitoring of forest fires. He holds an Engineering degree in Computer Science from EMI, University Mohammed 5 in Rabat, Morocco (2002). Furthermore, he is certified in artificial intelligence, 3D modeling, and programming. Additionally, he has served as a reviewer for a number of international conferences and journals, and is currently employed at Mohammed First University as Secretary General of the College of Technology. He can be contacted at email: m.grari@ump.ac.ma.

Mimoun Yandouzi Ph.D. in Computer Science at Mohammed First University in Oujda, Morocco, where he is conducting research on the use of computer vision and deep learning techniques for the analysis of drone data, particularly in the case of forest fire detection. He holds a degree in Computer Engineering from the School of Mineral Industry in Rabat, Morocco (2001). Furthermore, he holds several certifications in artificial intelligence, computer vision, cloud computing, big data, and data mining. He also acted as a reviewer for several international conferences. He is currently employed at Mohammed First University as a professor at the ENSA Engineering School. He can be contacted at email: m.yandouzi@ump.ac.ma.

Berrahal Mohammed is a Professor at the Polydisciplinary Faculty of Safi (PFS), Cadi Ayyad University, in Safi, Morocco. His research focuses on the applications of deep learning in security and law enforcement. He obtained his M.Sc. in internet of things from the National School of Computer Science and Systems Analysis (ENSIAS), Mohammed V University in Rabat, Morocco, in 2018, and his B.Sc. in Computer Engineering from the School of Technology (ESTO), Mohammed I University in Oujda, Morocco, in 2016. Berrahal Mohammed is also recognized for his expertise in artificial intelligence, 3D modeling, and programming, holding certifications in these areas. His contributions extend to serving as a reviewer for various international conferences and journals. He can be contacted at email: m.berrahal@uca.ac.ma.

Mohammed Boukabous Ph.D. in Computer Science at Mohammed First University in Oujda, Morocco, where he is conducting research in security intelligence using deep learning algorithms in exchanged messages. He holds a M.Sc degree in internet of things from Sidi Mohamed Ben Abdellah University in Fez, Morocco (2019), as well as a B.Sc. degree in Computer Engineering from Mohammed First University (2016). Furthermore, he holds several certifications in natural language processing, artificial intelligence, security intelligence, big data, and cybersecurity. Additionally, he served as a reviewer for various international conferences. He is currently employed at Mohammed First University as an administrative. He can be contacted at email: m.boukabous@ump.ac.ma.

Idriss Idrissi Ph.D. in Computer Engineering at Mohammed First University in Oujda, Morocco, where he is researching internet of things security using deep learning. He has an M.Sc degree in internet of things from Sidi Mohamed Ben Abdellah University in Fez, Morocco (2019), a B.Sc degree in Computer Engineering from Mohammed First University (2016). Additionally, he holds several certifications in networking, artificial intelligence, cybersecurity, and programming. Also, he was a reviewer for various international conferences and journals. And is currently employed at Mohammed First University as a professor at the College of Technology. He can be contacted at email: idrissi@ump.ac.ma.