

Early wildfire detection using machine learning model deployed in the fog/edge layers of IoT

Mounir Grari, Idriss Idrissi, Mohammed Boukabous, Omar Moussaoui, Mostafa Azizi,
Mimoun Moussaoui

Mathematics, Signal and Image Processing, and Computing Research Laboratory (MATSI), Higher School of Technology (ESTO),
Mohammed First University, Oujda, Morocco

Article Info

Article history:

Received Nov 19, 2021

Revised May 17, 2022

Accepted Jun 11, 2022

Keywords:

Edge computing
Ensemble learning
Fog computing
Forest fire
Internet of things
Machine learning
Wildfire

ABSTRACT

The impact of wildfires, even following the fire's extinguishment, continues to affect harmfully public health and prosperity. Wildfires are becoming increasingly frequent and severe, and make the world's biodiversity in a growing serious danger. The fires are responsible for negative economic consequences for individuals, corporations, and authorities. Researchers are developing new approaches for detecting and monitoring wildfires, that make use of advances in computer vision, machine learning, and remote sensing technologies. IoT sensors help to improve the efficiency of detecting active forest fires. In this paper, we propose a novel approach for predicting wildfires, based on machine learning. It uses a regression model that we train over NASA's fire information for resource management system (FIRMS) dataset to predict fire radiant power in megawatts. The analysis of the obtained simulation results (more than 99% in the R2 metric) shows that the ensemble learning model is an effective method for predicting wildfires using an IoT device equipped with several sensors that could potentially collect the same data as the FIRMS dataset, such as smart cameras or drones.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Mounir Grari

Mathematics, Signal and Image Processing, and Computing Research Laboratory (MATSI)

Higher School of Technology (ESTO), Mohammed First University

Oujda, Morocco

Email: m.grari@ump.ac.ma

1. INTRODUCTION

Forests are natural guardians of the earth's ecological equilibrium. Unfortunately, forest fires are frequently discovered only after having spread over a broad region, making control and extinguishment more difficult, even impossible in some cases. Forest fires generate 30% of the carbon dioxide (CO₂) in the atmosphere, resulting in catastrophic losses and irreversible damage to the ecosystem [1]. Wildfires are unplanned, unwanted, and uncontrolled. These fires start with a few flammable vegetation in rural areas (such as forests) and grow around speedily with winds and hot temperatures. Most of them are usually consequences of bad human behaviors, meanwhile, the cause of other wildfires remains unknown [2]. Wildfires have a huge impact on many fields; they can disrupt transportation, communication, power and gas services, and water supply. They can deteriorate the quality of the air and property, destroy crops and resources, burn animals and people. Between 1998 and 2017, wildfires represented 3.5% of world disasters [3] and caused nearly 2400 deaths worldwide [2]. Recently in the summer of 2021, hot temperatures in the entire Mediterranean basin led to extreme weather conditions, causing wilderness fires in Turkey, Italy, France, Greece, Morocco, and Algeria [4]. These countries have suffered from the worst wildfires in decades, with hundreds of dead people and heavy economic losses. Climate change appears to be unfolding

considerably quicker than expected, according to a scathing assessment [5]. Authorities employ a variety of detection and monitoring techniques, such as observers in the shape of patrols or monitoring towers. These last primitive techniques are used mainly in most countries [1].

Algorithms of machine learning can assist computers in chess and surgery, as well as in more smarter applications [6]. Nowadays we are experiencing a continuous technological growth and we can make predictions over the coming days, looking at how computers have progressed over the past days or years [7]. The way how computer tools and techniques have been democratized is one of the major elements of this revolution [8]. Data scientists have created powerful edge-cutting computing models with the smooth use of modern technologies [9].

New methods for detecting and monitoring forest fires can be built on the basis of those available in computer vision, machine learning, and remote sensing technology. Connected sensors have made it possible to spot active forest fires more efficiently [1], [10]–[14].

In this paper, we apply the fire information for resource management system (FIRMS) datasets (moderate resolution imaging spectroradiometer (MODIS) and visible infrared imaging radiometer suite (VIIRS)) [15], [16] to build an ML-based model that can predict wildfires. We benchmark these datasets on various regression algorithms to fit the deployment of this model in an IoT device. We focus here on those IoT devices equipped with sensors that may potentially gather the same features as in the FIRMS datasets.

The remainder of this paper is organized as follows. The second section presents a background and related works. The third section illustrates the research method used for building our model with the aforementioned datasets. Before concluding, the obtained results are discussed in the fourth section.

2. THE COMPREHENSIVE THEORETICAL BASIS

2.1. Internet of things

The internet of things (IoT) is used as a term to describe devices that communicate with each other. Devices from basic sensors to smartphones and wearables are all part of IoT. We may collect data, analyze that data, and take action to assist with a specific activity or learn from a process using these linked devices and automated systems. Data and networks are the core of IoT since it enables devices with internet connections to interact with each other. To build a more interconnected environment, IoT allows devices to interact with each other over a wide range of networks [17].

2.2. Machine learning

Machine learning (ML) is a subfield of artificial intelligence (AI) that relies on using data and algorithms to mimic the way people learn and improve accuracy over time [18]. ML is a key element of the rapidly expanding area of data science. Algorithms are trained to produce models using statistical approaches, revealing significant insights into data mining initiatives [19]. Following that, these insights drive decision-making within applications, with the goal of influencing important growth indicators [20].

2.3. Regression analysis

Regression is a supervised ML approach that helps in the discovery of variable correlations and allows us to forecast a continuous output variable using one or more predictor variables. Prediction, forecasting, time series modeling, and establishing the causal-effect link between variables are all common applications. ML regression algorithms have the ability to produce adaptable, robust connections, and they can be used quickly once they have been trained. They may be better candidates for operational applications [21].

2.4. Ensemble learning

Ensemble learning is a broad meta-approach of ML that combines predictions from several models to improve predictive performance. Although there appears to be no limit to the number of ensembles that create predictive modeling issues. The area of ensemble learning is dominated by three approaches: bagging, boosting, and stacking [22].

- Bagging [22] is the process of fitting several decision trees to various samples of the same dataset and then averaging the results. This technique is used in numerous prominent ensemble algorithms, including random forest and extra trees.
- Boosting [22] is the process of sequentially adding ensemble algorithms that correct prior model predictions and produce a weighted average of the predictions. This technique is used in several prominent ensemble algorithms, including AdaBoost and gradient boosting machines.
- Stacking [22] is the process of fitting many types of models to the same data and then using another model to learn how to integrate the predictions in the best way possible.

In our work, we used particularly some of the most known ensemble learning regression algorithms.

2.4.1. Random forest regressor

As an interpreted algorithm, the decision tree may not be able to learn all of the features from only one tree. so, we use another algorithm, random forest, which simultaneously combines several decision trees quality features to make decisions. It is a forest of randomly generated decision trees [23]. Overfitting is a significant drawback of the decision tree method. The random forest regression might be used instead of the decision tree regression to reduce this drawback. Furthermore, the random forest approach outperforms alternative regression models in terms of speed and robustness [24].

2.4.2. Gradient boosting regressor and histgradient boosting regressor

Gradient boosting regressor (GBR) is a technique that merges poor learners and weak predictive models to produce an ensemble model [25]. Algorithms that use gradient boosting can be utilized to train both regression and classification models. Continuous value is predicted in the model using the method 'GBR'. GBR creates an additive mode through the use of a multitude of fixed-size decision trees as weak learners or weak prediction models. The option of n estimators determines the number of decision trees utilized during boosting phases. GB differs in the way decision stumps (one node & two leaves) are employed in AdaBoost, whereas decision trees of fixed size are utilized for gradient boosting [25]. When the sample size is more than tens of thousands, these histogram-based estimators could be much quicker than gradient boosting classifier (GBC) and GBR. By decreasing (binding) the continuous input variables to a few hundred distinct values, the training of trees introduced to the ensemble may be substantially accommodated. Gradient boosters that use that method and customize the algorithm for training around the input variables under this transformation are known as Histogramic gradient booster sets [26].

2.4.3. Light gradient boosting machine

Light gradient boosting machine (LightGBM) is an open-source framework for gradient-boosted machines developed originally by Microsoft [27]. It is used by default for training a gradient boosted decision tree (GBDT), but as well it endorses random forests. Dropouts meet multiple additive regression trees (DART), and Microsoft's gradient-based one-side sampling (GOSS). LightGBM employs a tree-based learning method. When compared to other algorithms, LightGBM generates trees in a leaf-wise manner while other algorithms grow trees level-wise. The leaf with the greatest delta loss will be selected for growth. The leaf-wise method reduces loss more than the tree level-wise strategy when the same leaf is grown over and over again [28]. LightGBM trains really faster compared to other gradient boosting implementations.

2.4.4. Extreme gradient boosting

Extreme gradient boosting (XGBoost) is a distributed gradient boosting library that is aimed to be very efficient, adaptable, and portable. The gradient boosting framework is used to build the ML algorithms [28]. It has recently become a popular algorithm for winning teams in ML competitions due to its success in solving problems quickly and accurately using XGBoost's parallel tree boosting (GBDT, GBM) method. Compared to other gradient boosting implementations, XGBoost performs exceptionally well [29].

2.4.5. AdaBoost regressor

AdaBoost (AB), an abbreviation for adaptive boosting, is a meta-algorithm developed for ML [30]. It can be used to increase performance in combination with many different learning methods. Other techniques of learning ("weak learners") are combined into a weighted sum in the final output of the boosted classifier. In the sense that future weak learners are adjusted for examples that have been misclassified by prior classifiers. In some cases, it may be more vulnerable than other learning algorithms to overfitting. Each learner can be poor yet the final model can converge with a powerful one as long as their performance is somewhat better than random deviations [31].

An AdaBoost regressor (ABR) is a meta-estimator that initially fits in a regressor to the original dataset. It is then fitted to the same dataset with extra copies of the regressor but adjusts the instance weight to the existing prediction. There is a greater focus on tough instances in successive regressors [32].

2.4.6. Bagging regressor

The bagging regressors (BR) are ensemble meta-estimators that fit base regressors to randomized subsets of the original dataset and then combine their individualized predictions (either by voting or average) to produce a final prediction [33]. This algorithm is based on several publications in the literature. Pasting is a technique that involves generating random subsets of the dataset as randomized subsets of the samples. When samples were taken with replacement, the procedure is known as bagging. Randomized subspaces is a technique in which random subsets of the dataset are generated for randomized subsets of the attributes. Finally, "Random patches" are a technique for generating base estimators using sample and feature subsets [33].

2.4.7. Extra trees regressor

Extra trees (ET), or extremely randomized trees, is an ensemble ML technique. It is a decision tree ensemble that is linked to other decision tree ensembles techniques such as bootstrap aggregation (bagging) and random forest. The ET technique uses the training dataset to generate a huge number of unpruned decision trees. In the case of regression, predictions are produced by averaging the prediction of the decision trees, whereas, in the case of classification, majority voting is used [34].

Unlike bagging and RF, which generate each decision tree using a bootstrap sample of the training dataset, “extra trees” fits each decision tree to the completely training dataset. ET, like random forest, samples features at every split point of a decision tree at random. The ET technique picks a split point at random, unlike random forest, which employs a greedy approach to identify the best split point [34]. An extra trees regressor (ETR) is a meta-estimator that fits a variety of randomized controlled decision trees (for example extra-trees) on numerous datasets and utilizes averaging to enhance prediction accuracy and control overfitting [35].

3. RELATED WORKS

Multilayer perceptron (MLP) and k-nearest neighbor (kNN) algorithms have been used by Kumar and Kumar [36] to generate a fire detection and classification models, which have been tested with data gathered by LANCE FIRMS, a NASA-operated Earth Science Data and Information System Project (ESDIS). With a 99.96% accuracy rate, the MLP algorithm outperformed the kNN algorithm in terms of accuracy in their proposed approach.

Kaur *et al.* [37] proposed a fog-cloud computing IoT framework supported by energy-efficient IoT for early wildfire prediction. Using the Jaccard similarity analysis, they were able to identify duplicate data gathered from IoT devices in real-time and evaluate it at the fog computing layer, resulting in the vulnerability index score. The ANN model is then supplemented with a self-organized mapping method to effectively visualize the geographical region's wildfire susceptibility based on Wildfire Leading Parameters. Performance estimation results have been compared with various state-of-the-art methods using diverse datasets, with an accuracy of 95.32%.

Sun *et al.* [38] presented an architecture for an unmanned aerial vehicle (UAV)-enabled system comprised of several industrial internet of things (IIoTs), in which data gathered by IIoT sensors may be transmitted directly to UAVs for processing, where IIoT sensors have been used to keep track of various forest fire indices, taking priority restrictions into account may help ensure that forest fire monitoring responds quickly. According to this research, the most effective way to allocate UAV resources is to use an algorithm that uses learning-based cooperative particle swarm optimization (LCPSO) and Markov random fields (MRF). Decomposed decision variables in the MRF network structure deconstruct the solution space of UAV resource allocation into sub-solution spaces, and LCPSO cooperatively searches in many sub-solution spaces for the optimum resource allocation strategy. With the use of three simulation tests based on two different datasets, the validity of LCPSO has been shown, and this is evident in forest fire monitoring's fastest reaction time when compared with other techniques.

In order to identify, disseminate, and monitor active fire locations (AFL) for agricultural operations, Sharma *et al.* [39] suggests a multi-model IoT and deep learning-inspired system. The suggested system's IoT module uses a combination of IoT sensors and deep learning detectors to identify anomalies. Fuzzy logic is utilized to combine many senses and locate AFL in real-time. Using a new self-created dataset, the deep learning detector trains on IP camera-based MobilenetV2 architecture for precise and long-distance detections. A software module for tracking and monitoring different AFL was included in the proposed architecture. With the software, users can extract fire locations automatically from remote sensing sites, assign active fire locations to various stakeholders, extract farmers' names who are involved in the fire, send a notification to government agencies automatically, and allow citizens to participate centrally. With up to 100 percent recall, precision and an F1 score of 1, the findings of the suggested framework are very promising.

Jia *et al.* [40] presented a surface energy balance (SEB) method to estimate cloudy-sky land surface temperature (LST) from polar-orbiting satellite observations. The hypothetical clear-sky LST for those cloudy pixels was reconstructed using the simultaneous retrieval algorithm and an reanalysis 5th generation (ERA5) reanalysis model. The cloudy-sky LST was estimated by superimposing cloud effects on the reconstructed clear-sky LST using SEB theory. The overall RMSE of the estimated cloudy-sky LST from VIIRS data was 3.54 K with a bias of 0.36 K and R2 of 0.94 (N=2411), which was slightly lower than the accuracy of the high-quality clear-sky LST retrieval results, but better than the likely cloud-contaminated retrieval.

4. MATERIAL AND METHOD

4.1. Fire information for resource management system datasets

The FIRMS publishes the near real-time (NRT) active and live-fire data from the MODIS aboard the Terra and Aqua satellites, as well as the VIIRS aboard the NOAA 20 and S-NPP satellites, within three hours of monitoring [15]. FIRMS utilizes MODIS and VIIRS tools to detect active and thermal abnormalities in almost real-time, employing email warnings, ready-to-analyze data, online maps, and web services, to convey this info to decision-makers.

4.1.1. MODIS-derived global fire products

They are digital maps derived from Terra and Aqua MODIS data, particularly for use in emissions modeling. The algorithms were created to deliver a comprehensive worldwide solution that would perform effectively over a wide variety of fire situations and scene diversity. The objective was to increase product correctness while reducing commission and omission mistakes. One product describes actively burning fire sites at satellite overpass time, while the other displays the burnt area, also known as fire-affected areas [41].

4.1.2. VIIRS 375 m active fire product

The active fire product VIIRS 375 m (VNP14IMGTDL NRT) is the most recent addition to FIRMS. It transmits data from the VIIRS sensor onboard the suomi-national polar-orbiting partnership (Suomi-NPP) and NOAA-20 satellites, which are jointly operated by NASA and NOAA. The 375 m data complements MODIS fire detections; both exhibit high agreement in hotspot detection, but the 375 m data's enhanced spatial resolution allows for a faster reaction over minor flames and better mapping of broad fire perimeters. Nighttime performance has also improved with the 375 m data. As a result, these data are ideally suited to be used in firefighting operations [42].

As they cross the globe, these satellites capture a “snapshot” of occurrences. The center of flamed pixels with one or more flames or other thermal anomalies is represented by each hotspot / active fire sensing (such as volcanoes). The pixel is around 1km for the MODIS while the pixel is about 375 m for VIIRS. The central point of the pixel is the "location" (not necessarily the coordinates of the actual fire). The actual size of the pixel varies depending on the scan and the track. The fire often exists below the pixel size. The precise fire size cannot be determined; however, we know that there is at least one fire in the marked pixel. Many of the current flames are seen in one line. This is usually a firefront (see Figure 1) [16].

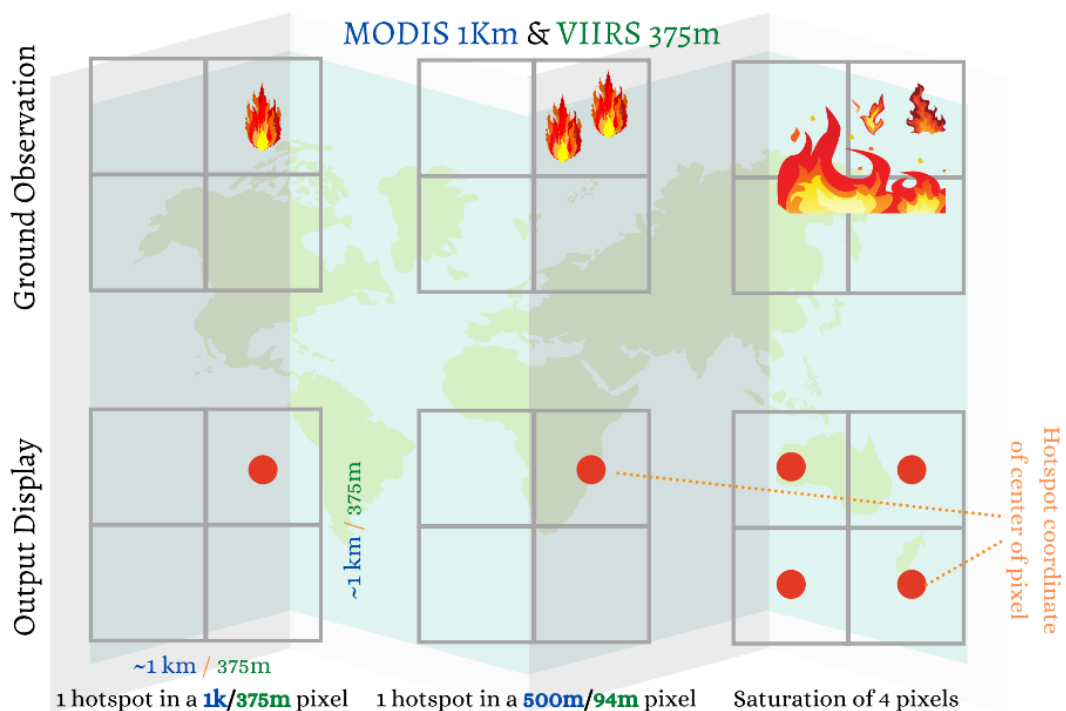


Figure 1. Hotspot/active fire detection collection method

4.1.3. Dataset attribute fields

Each feature, or column, represents a quantifiable piece of data that we have analyzed, such as latitude, longitude, brightness, and so on. Features are also known as "variables" or "attributes". Table 1 lists the FIRMS dataset features.

Table 1. FIRMS dataset attribute fields

| Attribute | Short description |
|------------|--|
| Latitude | Latitude |
| Longitude | Longitude |
| Brightness | The temperature of the brightness 21 (Kelvin) |
| Scan | Pixel size for the Along Scan |
| Track | Pixel size tracking |
| Acq_Date | Date of Acquisition |
| Acq_Time | Time of Acquisition |
| Satellite | A = Aqua and T = Terra |
| Confidence | 0-100% - It makes estimations ranging from 0 to 100 percent and assigns them to one of three fire classifications (low-confidence fire, nominal-confidence fire, or high-confidence fire). |
| Version | Version (Collection and source) |
| Bright_T31 | The temperature of the brightness 31 (Kelvin) |
| Type | Inferred hot spot type 0 = assumed vegetation fire 1 = active volcano 2 = other static land sources 3 = offshore |
| DayNight | Day or Night; D= Daytime fire, N= Nighttime fire |
| FRP | Fire Radiative Power in megawatts (MW) |

4.2. Proposed method

Our approach uses the FIRMS datasets to train a ML model that can eventually forecast the fire radiative power in megawatts. After generating this model based on the best-achieved ML algorithm (the base work in this paper), we intend to deploy it in an IoT device equipped with different sensors that may theoretically gather the same characteristics as the FIRMS dataset; such as a camera (see Figure 2).

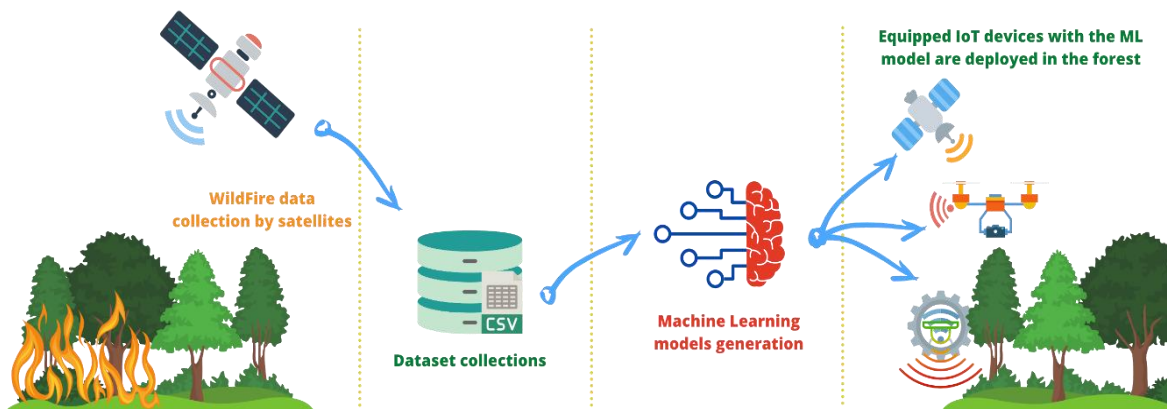


Figure 2. Proposed model methods building

These IoT devices (ideally a camera or an IoT device equipped with a camera such as a drone, or buying and using satellite imagery [43]) can either make this prediction locally (meaning in the edge layer [44]) or can pass the data to the gateway, where a more powerful IoT device (in the fog layer) can make this prediction. This IoT should be an AI-enabled circuit (which is found in the market at low-cost) or can be on a device with higher processing power, preferably linked to a lightweight neural network hardware accelerator (like the Intel Neural Compute Stick 2, Google Coral edge TPU, or Nvidia jetson nano) [45]. Afterward, if a fire is predicted either on the edge or the fog layers, a notification is transmitted to the fire department to take suitable arrangements for the predicted fire. Then, if a fire detection is forecasted (either on the layers of edge or fog, depending on the deployment model), a notification is sent to the fire department to handle the anticipated wildfire (see Figure 3).

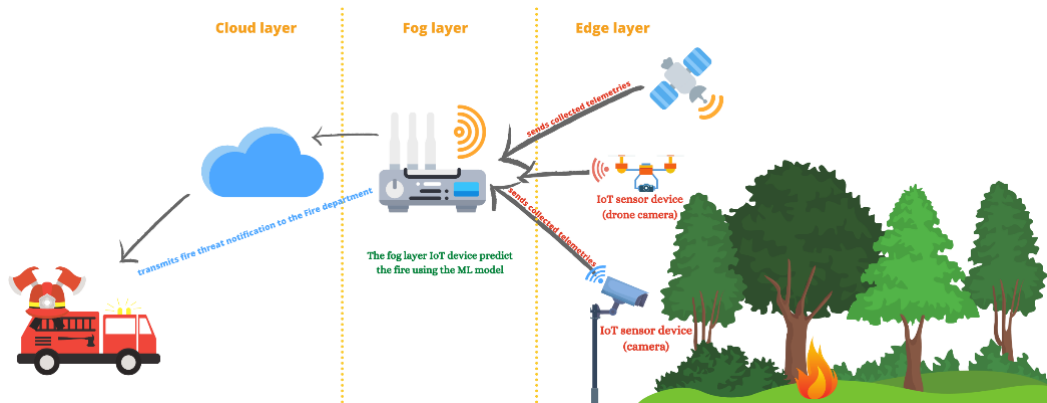


Figure 3. Deployment of our proposal

4.2.1. First step: importing and concatenating the datasets

Both MODIS and VIIRS datasets are available in CSV files. Each file is a plain text file that contains tabular data from a given period and location, we used multiples periods for this experiment. A data record is represented by each line in the file. We read the CSV files as a dataframe and concatenate them into a single dataframe using the Pandas library.

4.2.2. Second step: data preprocessing

We rarely obtain data that is homogeneous. When data is missing, it must be managed carefully, so that the ML model performance is not harmed. Then, we encode the categorical data; any variable that is not quantitative, such as “Satellite” and “DayNight”, is categorical. We cannot employ values like “Day” and “Night” or “Terra” and “Aqua” in the model mathematical equations, therefore we have to encode these variables into “0” and “1” integers.

4.2.3. Third step: splitting the dataset into the training dataset and validation dataset

In this step, we split the dataset into two sub-datasets, one for training the model and the other for assessing its performance, referred to as the training dataset “X” (80%) and the validation dataset “Y” (20%). We split the dataset using the K-Folds cross-validator, which offers train/test indices to split the data into train/test datasets, and into k consecutive folds.

4.2.4. Fourth step: model training

In this phase, we used multiple ML algorithms for regression analysis. Passing the training dataset to each algorithm and evaluating it on each fold of the splitted data to get the best results. The building of our proposed model is presented in Figure 4 which demonstrates the different steps.

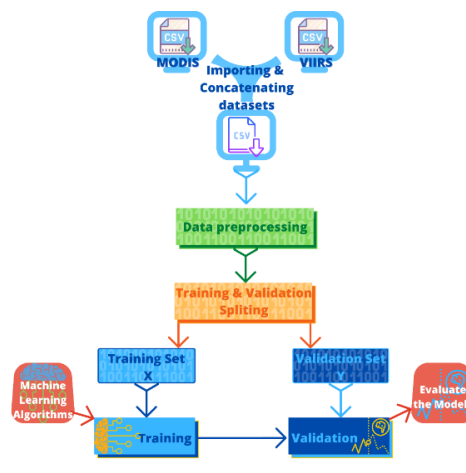


Figure 4. Building steps of our models

5. RESULTS AND DISCUSSION

5.1. Hardware characteristics

Our results were achieved on a Debian LXC container deployed on the laboratory server with the following hardware characteristics: i) CPU(s): 2× AMD Opteron(tm) Processor 6344 (24 cores) and ii) RAM: 64 GB. In our experiments, we worked with Scikit-learn [46], which is a free and open-source ML library that allows both supervised and unsupervised learning. Scikit-learn offers a wide range of modeling and data processing capabilities, as well as the ability to choose and evaluate different models.

5.2. Metrics of performances

5.2.1. Mean squared error

The mean squared error (MSE) or mean squared deviation (MSD) measures the average squared error, which is the difference between the estimated and actual values. It is a risk equation that represents the predicted value of the squared error loss. It is never negative, therefore numbers of MSE near zero are preferable. The MSE is the second moment of error (around the origin) and so contains both the estimator's variance and bias [47]. The MSE equation is shown in (1).

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2 \quad (1)$$

Where y is the predicted value and \tilde{y}_i is the real value.

5.2.2. Mean absolute error

The mean absolute error (MAE) estimates the average error size without taking into account the direction of the abnormalities. All individual variations in the sample set have the same weight, therefore the average of the absolute errors between predictions and taking into account the effects are calculated [48], [49]. The MAE equation is shown in (2).

$$MSE = \frac{1}{n} \sum_{i=1}^n |y_i - \tilde{y}_i| \quad (2)$$

Where y is the predicted value and \tilde{y}_i is the real value.

5.2.3. R2 score

The coefficient of determination, commonly known as the R2 score, is a metric used to assess the effectiveness of a linear regression model. It is the degree of variation in the output-dependent characteristic that can be predicted based on the input independent variable. It is used to determine how effectively the model reproduces observed results, based on the ratio of total deviation of results represented by the model [50]. It can range from 0 to 100 percent. If it is 100 percent, the two studied variables are completely correlated, meaning they have no variance. A low number indicates a low amount of correlation, implying that a regression model is not always valid [47]. The R-2 equation is shown in (3).

$$R2 = 1 - \frac{SE}{SEt} \quad (3)$$

Where SE is the sum of squares of the residual errors (see (1)) and SEt is the total sum of the errors.

5.3. Evaluation of our models

According to Table 2, extra trees then gradient boosting, and random forest are the best ensemble approaches in our study when compared to the other ML algorithms, having higher R2 values. Indeed, from the analysis of our findings, we note that Extra Trees Regressor achieves the highest results, up to 100% in R2 score, and the lowest value on both the MAE and MSE metrics. Table 2 shows that the other Bagging algorithms and Gradient Boosting Regressor have obtained more than 99% in R2 score and less than 3 in MAE metric, proving that the selection of these regressors is optimal for forecasting wildfires. However, for the Boosting algorithms such as AdaBoost Regressor, HistGradient Boosting Regressor, XGBoost, and LightGBM, we obtained less competitive results. XGBoost and the HistGradient Boosting Regressor got decent results around 97% in the R2 score and 3 in the MAE metric, but LightGBM and AdaBoost regressor were the worst in all the results, their best one was around 80% in the R2 score and 30 in the MAE metric despite all efforts to regularize the hyperparameters for this approach. The comparison of the six regressors, based on the R2 score is presented in Figure 5 using the box and whisker plot to illustrate the mean value of prediction. This figure shows the superiority of both Bagging algorithms and the Gradient Boosting algorithm over the other boosting ones, especially the ETR.

Table 2. Metrics results for each studied algorithm

| Machine learning model | R2 score | MAE | MSE |
|--|----------|-----------|-----------|
| Extra trees regressor (ETR) | 100.00% | 2.59 e-07 | 1.27 e-08 |
| Gradient boosting regressor (GBR) | 99.96% | 2.469 | 16.98 |
| Random forest regressor (RFR) | 99.67% | 3.118 | 153.02 |
| Bagging regressor (BR) | 99.54% | 1.708 | 209.44 |
| Extreme gradient boosting (XGBoost) | 97.61% | 3.962 | 1,162.64 |
| HistGradient boosting regressor (HGBR) | 96.67% | 3.69 | 1,470.27 |
| Light gradient boosting machine (LightGBM) | 83.58% | 31.00 | 7,689.01 |
| AdaBoost regressor (ABR) | 76.28% | 59.16 | 9,723.87 |

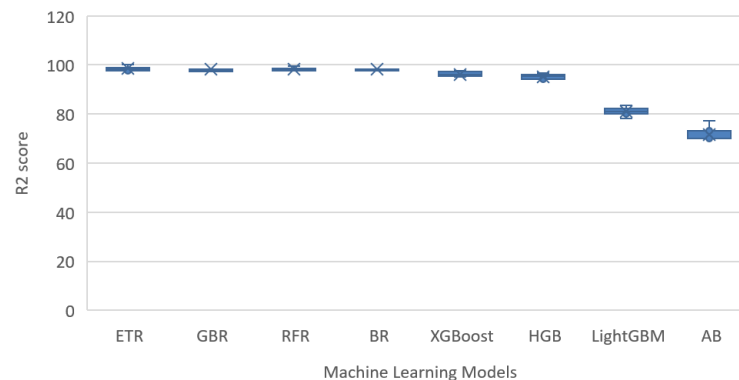


Figure 5. Ensemble algorithms comparison by the R2 score

6. CONCLUSION

Forests contribute strongly to the ecological balance of our planet. However, the existence of these vital natural barriers is seriously threatened. Wildfires often occur in large areas, making their management and extinction almost impossible. These disasters strike without warning, are unwanted, unpredictable, and are caused either by humans, climate change, or even lightning. There is a high wildfires threat of interrupting transportation, communications, power, gas, water, or other services. Air, crops, resources, animals, and humans may also be harmed.

From the research that has been carried out, we conclude that our proposed method can be an effective way to forecast wildfires. Indeed, using the ETR, we built an ML model trained on the FIRMS datasets, and deploy it in an IoT device equipped with sensors that collect the same features as the datasets (cameras, drones, IR cameras, brightness sensors, buying and using satellite imagery). Our obtained simulation results are very promising, which leads us next to apply our proposal in a real context. As future works, we are going to put our prototype into practice over a real manmade fire to validate and improve our proposed approach, as well as develop a hybrid method that uses multiple collaborative techniques for wildfire detection and prevention.

ACKNOWLEDGEMENTS

This work is supported by the Mohammed First University under the PARA1 Program (Low-cost, real-time Forest Fire Detection System based on Wireless Sensor Networks - SDF-RCSF).

REFERENCES




- [1] A. A. A. Alkhatib, "A review on forest fire detection techniques," *International Journal of Distributed Sensor Networks*, vol. 2014, pp. 1–12, Mar. 2014, doi: 10.1155/2014/597368.
- [2] WHO, "Wildfires," *World Health Organization*, 2017. <https://www.who.int/health-topics/wildfires> (accessed Oct. 03, 2020).
- [3] UNISDR and CRED, "Economic losses, poverty and disasters 1998-2017," *United Nations Office for Disaster Risk Reduction (UNISDR) - Centre for Research on the Epidemiology of Disasters (CRED)*, 2017. .
- [4] D. Coffey, "From wildfires to floods, the Mediterranean bears the brunt of catastrophic climate change," *RFI*. <https://www.rfi.fr/en/international/20210813-from-wildfires-to-floods-the-mediterranean-bears-the-brunt-of-catastrophic-climate-change> (accessed Sep. 18, 2021).
- [5] W. W. F., "Fires, forest and the future: a crisis raging out of control?," *World Wide Fund for Nature*, 2020. https://wwf.panda.org/discover/our_focus/forests_practice/forest_publications_news_and_reports/fires_forests/ (accessed Sep. 18, 2021).

- [6] M. Berrahal and M. Azizi, "Improvement of facial attributes' estimation using transfer learning," in *2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, Mar. 2022, pp. 1–7, doi: 10.1109/IRASET52964.2022.9737845.
- [7] M. Berrahal and M. Azizi, "Optimal text-to-image synthesis model for generating portrait images using generative adversarial network techniques," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 25, no. 2, pp. 972–979, Feb. 2022, doi: 10.11591/ijeecs.v25.i2.pp972-979.
- [8] M. Boukabous and M. Azizi, "Multimodal sentiment analysis using audio and text for crime detection," in *2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, Mar. 2022, pp. 1–5, doi: 10.1109/IRASET52964.2022.9738175.
- [9] M. Boukabous and M. Azizi, "Crime prediction using a hybrid sentiment analysis approach based on the bidirectional encoder representations from transformers," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 2, pp. 1131–1139, 2022, doi: 10.11591/ijeecs.v25.i2.pp1131-1139.
- [10] A. Gaur *et al.*, "Fire sensing technologies: a review," *IEEE Sensors Journal*, vol. 19, no. 9, pp. 3191–3202, May 2019, doi: 10.1109/JSEN.2019.2894665.
- [11] A. Gaur, A. Singh, A. Kumar, A. Kumar, and K. Kapoor, "Video flame and smoke based fire detection algorithms: a literature review," *Fire Technology*, vol. 56, no. 5, pp. 1943–1980, Sep. 2020, doi: 10.1007/s10694-020-00986-y.
- [12] F. Bu and M. S. Gharajeh, "Intelligent and vision-based fire detection systems: a survey," *Image and Vision Computing*, vol. 91, Nov. 2019, doi: 10.1016/j.imavis.2019.08.007.
- [13] C. Yuan, Y. Zhang, and Z. Liu, "A survey on technologies for automatic forest fire monitoring, detection, and fighting using unmanned aerial vehicles and remote sensing techniques," *Canadian Journal of Forest Research*, vol. 45, no. 7, pp. 783–792, Jul. 2015, doi: 10.1139/cjfr-2014-0347.
- [14] R. Allison, J. Johnston, G. Craig, and S. Jennings, "Airborne optical and thermal remote sensing for wildfire detection and monitoring," *Sensors*, vol. 16, no. 8, pp. 1–29, Aug. 2016, doi: 10.3390/s16081310.
- [15] NASA, "NASA | LANCE | FIRMS," NASA. <https://firms.modaps.eosdis.nasa.gov/> (accessed Jul. 30, 2021).
- [16] National Aeronautics and Space Administration (NASA), "Firms FAQ | Earthdata," *FIRMS FAQ*, 2018. <https://earthdata.nasa.gov/faq/firms-faq#ed-fire-detection> (accessed Jul. 30, 2021).
- [17] I. Idrissi, M. Boukabous, M. Azizi, O. Moussaoui, and H. El Fadili, "Toward a deep learning-based intrusion detection system for IoT against botnet attacks," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, pp. 110–120, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp110-120.
- [18] I. Idrissi, M. Azizi, and O. Moussaoui, "Accelerating the update of a DL-based IDS for IoT using deep transfer learning," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 23, no. 2, pp. 1059–1067, Aug. 2021, doi: 10.11591/ijeecs.v23.i2.pp1059-1067.
- [19] A. Panesar, "What is machine learning?," in *Machine Learning and AI for Healthcare*, Berkeley, CA: Apress, 2019, pp. 75–118.
- [20] A. Kherraki and R. El Ouazzani, "Deep convolutional neural networks architecture for an efficient emergency vehicle classification in real-time traffic monitoring," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 11, no. 1, p. 110120, Mar. 2022, doi: 10.11591/ijai.v11.i1.pp110-120.
- [21] J. Verrelst *et al.*, "Machine learning regression algorithms for biophysical parameter retrieval: opportunities for sentinel-2 and -3," *Remote Sensing of Environment*, vol. 118, pp. 127–139, Mar. 2012, doi: 10.1016/j.rse.2011.11.002.
- [22] Z.-H. Zhou, "Ensemble learning," in *Machine Learning*, Singapore: Springer Singapore, 2021, pp. 181–210.
- [23] E. Frontoni, M. Paolanti, and R. Pietrini, "People counting in crowded environment and re-identification," in *Advances in Computer Vision and Pattern Recognition*, Springer London, 2019, pp. 397–425.
- [24] P. Kapoor and R. Rani, "Efficient decision tree algorithm using j48 and reduced error pruning," *International Journal of Engineering Research and General Science*, vol. 3, no. 3, pp. 1613–1621, 2015.
- [25] P. Prettenhofer and G. Louppe, "Gradient boosted regression trees in scikit-learn," Feb. 2014, [Online]. Available: <https://orbi.uliege.be/handle/2268/163521>.
- [26] A. Guryanov, "Histogram-based algorithm for building gradient boosting ensembles of piecewise linear decision trees," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11832, Springer, Cham, 2019, pp. 39–50.
- [27] G. Ke *et al.*, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, 2017, vol. 2017-Decem, pp. 3147–3155.
- [28] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.
- [29] Github, "xgboost/demo at master · dmlc/xgboost," *github*. <https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions> (accessed Nov. 04, 2021).
- [30] F. Wang, Z. Li, F. He, R. Wang, W. Yu, and F. Nie, "feature learning viewpoint of adaboost and a new algorithm," *IEEE Access*, vol. 7, pp. 149890–149899, Apr. 2019, doi: 10.1109/ACCESS.2019.2947359.
- [31] R. M. Freund, P. Grigas, and R. Mazumder, "AdaBoost and forward stagewise regression are first-order convex optimization method," *arXiv*, Jul. 2013.
- [32] H. Drucker, "Improving regressors using boosting techniques," in *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*, 1997, pp. 107–115.
- [33] Scikit-learn, "sklearn.ensemble.BaggingRegressor — scikit-learn 0.24.2 documentation," *scikit-learn*. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingRegressor.html> (accessed Aug. 02, 2021).
- [34] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, Apr. 2006, doi: 10.1007/s10994-006-6226-1.
- [35] S. M. Kasongo and Y. Sun, "A deep learning method with wrapper based feature extraction for wireless intrusion detection system," *Computers & Security*, vol. 92, p. 101752, May 2020, doi: 10.1016/j.cose.2020.101752.
- [36] N. Kumar and A. Kumar, "Australian Bushfire Detection Using Machine Learning and Neural Networks," in *2020 7th International Conference on Smart Structures and Systems (ICSSS)*, Jul. 2020, pp. 1–7, doi: 10.1109/ICSSS49621.2020.9202238.
- [37] H. Kaur, S. K. Sood, and M. Bhatia, "Cloud-assisted green IoT-enabled comprehensive framework for wildfire monitoring," *Cluster Computing*, vol. 23, no. 2, pp. 1149–1162, Jun. 2020, doi: 10.1007/s10586-019-02981-7.
- [38] L. Sun, L. Wan, and X. Wang, "Learning-based resource allocation strategy for industrial IoT in UAV-enabled MEC systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 5031–5040, Jul. 2021, doi: 10.1109/TII.2020.3024170.
- [39] A. Sharma *et al.*, "IoT and deep learning-inspired multi-model framework for monitoring Active Fire Locations in Agricultural Activities," *Computers & Electrical Engineering*, vol. 93, Jul. 2021, doi: 10.1016/j.compeleceng.2021.107216.




- [40] A. Jia, H. Ma, S. Liang, and D. Wang, "Cloudy-sky land surface temperature from VIIRS and MODIS satellite data using a surface energy balance-based method," *Remote Sensing of Environment*, vol. 263, Sep. 2021, doi: 10.1016/j.rse.2021.112566.
- [41] C. O. Justice *et al.*, "MODIS-derived global fire products," in *Remote Sensing and Digital Image Processing*, vol. 11, Springer International Publishing, 2011, pp. 661–679.
- [42] W. Schroeder, P. Oliva, L. Giglio, and I. A. Csiszar, "The new VIIRS 375 m active fire detection data product: Algorithm description and initial assessment," *Remote Sensing of Environment*, vol. 143, pp. 85–96, Mar. 2014, doi: 10.1016/j.rse.2013.12.008.
- [43] EOS, "High-resolution satellite imagery available on LandViewer," *earth observing system*. <https://eos.com/products/high-resolution-images/> (accessed Sep. 23, 2021).
- [44] I. Idrissi, M. Azizi, and O. Moussaoui, "A stratified IoT deep learning based intrusion detection system," in *2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, Mar. 2022, pp. 1–8, doi: 10.1109/IRASET52964.2022.9738045.
- [45] I. Idrissi, M. Azizi, and O. Moussaoui, "A lightweight optimized deep learning-based host-intrusion detection system deployed on the edge for IoT," *International Journal of Computing and Digital Systems*, vol. 11, no. 1, pp. 209–216, Jan. 2022, doi: 10.12785/ijcds/110117.
- [46] "scikit-learn: machine learning in Python — scikit-learn 0.24.2 documentation," *scikit-learn*, 2021. <https://scikit-learn.org/stable/> (accessed Jul. 06, 2021).
- [47] GeekforGeeks, "Python | Mean squared error," *GeekforGeeks*, 2019. <https://www.geeksforgeeks.org/python-mean-squared-error/> (accessed Aug. 04, 2021).
- [48] C. Willmott and K. Matsuura, "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance," *Climate Research*, vol. 30, no. 1, pp. 79–82, Dec. 2005, doi: 10.3354/cr030079.
- [49] Y. Hammoudi, I. Idrissi, M. Boukabous, Y. Zerguit, and H. Bouali, "Review on maintenance of photovoltaic systems based on deep learning and internet of things," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 26, no. 2, pp. 1060–1072, May 2022, doi: 10.11591/ijeecs.v26.i2.pp1060-1072.
- [50] A. C. Cameron and F. A. G. Windmeijer, "An R-squared measure of goodness of fit for some common nonlinear regression models," *Journal of Econometrics*, vol. 77, no. 2, pp. 329–342, Apr. 1997, doi: 10.1016/S0304-4076(96)01818-0.

BIOGRAPHIES OF AUTHORS






Mounir Grari    is a Ph.D. candidate in Computer Engineering at Mohammed First University in Oujda, Morocco, where he is researching internet of things security using Machine Learning. Received a State Engineer degree in computer network from the Engineering School EMI of Rabat, Morocco in 2003, He participated in several scientific & organizing committees of national and international conferences. Additionally, he holds several certifications in networking, artificial intelligence, cybersecurity, and programming. And is currently employed at Mohammed First University. He can be contacted at email: m.grari@ump.ac.ma..






Idriss Idrissi    is a Ph.D. candidate 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. He can be contacted at email: idrissi@ump.ac.ma.






Mohammed Boukabous    is a Ph.D. candidate in Computer Engineering at Mohammed First University in Oujda, Morocco, where he is conducting research in security intelligence using deep learning algorithms in exchanged messages. He holds an 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.






Prof. Dr. Omar Moussaoui    is an Associate Professor at the Higher School of Technology (ESTO) of Mohammed First University, Oujda – Morocco. He has been a member of the Computer Science Department of ESTO since 2013. He is currently director of the MATSI research laboratory. Omar completed his Ph.D. in computer science at the University of Cergy-Pontoise France in 2006. His research interests lie in the fields of IoT, wireless networks and security. He has actively collaborated with researchers in several other computer science disciplines. He participated in several scientific & organizing committees of national and international conferences. He served as reviewer for numerous international journals. He has more than 20 publications in international journals and conferences and he has co-authored 2 book chapters. Omar is an instructor for CISCO Networking Academy on CCNA Routing & Switching and CCNA Security. He can be contacted at email: o.moussaoui@ump.ac.ma.



Prof. Dr. Mostafa Azizi    received a State Engineer degree in Automation and Industrial Computing from the Engineering School EMI of Rabat, Morocco in 1993, then a Master degree in Automation and Industrial Computing from the Faculty of Sciences of Oujda, Morocco in 1995, and a Ph.D. degree in Computer Science from the University of Montreal, Canada in 2001. He earned also tens of online certifications in Programming, Networking, AI, Computer Security. He is currently a Professor at the ESTO, University Mohammed First of Oujda. His research interests include Security and Networking, AI, Software Engineering, IoT, and Embedded Systems. His research findings with his team are published in over 100 peer-reviewed communications and papers. He also served as PC member and reviewer in several international conferences and journals. He can be contacted at email: azizi.mos@ump.ac.ma



Prof. Dr. Mimoun Moussaoui    received his doctorate in Numerical Analysis in 1984 from University Paris XI (Orsay, France). He obtained his Ph. D in Nonlinear Analysis in 1991 from the Free University (Université Libre) of Brussels Belgium. He is currently professor at the University Mohamed First of Oujda (Morocco). He teaches Mathematics for economics and scientists, Numerical analysis and linear programming. He supervised several theses in applied mathematics and computing. He was the previous director of Mathematics, signal and image processing and computing research Laboratory (MATSI). He can be contacted at email: m.moussaoui@ump.ac.ma.