New GIS Approach using Machine Learning Algorithm for early floods Detection

SATAURI Imane¹, Ohamouddou Mohamed ², HAJJI Said³, ZOUTEN Mohammed⁴, CHAAOUAN Jamal⁵, Setti Larbi⁶

¹ Laboratory of Computer Science, Signals, Automation and Cognitivism, FSDM, National School of Commerce and Management, USMBA, FEZ, MOROCCO

² Laboratoiry Ingeneering, Systems and Applications (LISA) ENSA Fez, Faculty Polydisciplinary of TAZA, Departement of Geography, Sidi Mohammed Ben Abdellah University

³,⁵ Laboratory EH3D, Department of Geography Faculty polydisciplinary TAZA, University Sidi Mohamed ben Abdelah of FEZ, Morocco

⁴ LaboratorY sciences and advanced technologies Faculty polydisciplinary of Larache, University Abdelmalek ESSADI TETOUAN, Morocco

Abstract. Floods are one of the most devastating natural hazards. Early intervention plays an important role in saving resources and many lives during the floods. Various technologies are used for rapid response, such as the use of satellite images in geographical information systems. Aerial satellite images of areas for candidate flooding, we have proposed a method of classification of aerial images to identify areas likely to be affected by flooding using various classifiers such as SVM, fine tree, KNN and neural networks. Their performance is compared, and it is observed that the SVM classifier exceeds the remaining classifiers with almost 93.1% due to its simplicity, although neural networks minimize the amount of training to a greater extent. This classification of images is then used to identify the areas likely to be affected by the floods in order to identify and manage the extent of the floods. An electronic complement is also proposed to assist the intervention teams in real time for better results.

I. Introduction

Currently, thousands of artificial satellites are in orbit around the earth. The large number of these satellites, called remote sensing satellites, are used for scientific studies from space. Remote sensing is particularly effective for natural disaster risks study on the scale of a large site or region. Application which aims to analyze and optimize real satellite images in order to have a map of critical limits and to warn in the event of overflows to anticipate the phenomenon is essential. We focus our work on analyze and filter images captured by NASA (LANDSAT program). Our contribution is an implementation of a model for processing these images using CNNs with a real time video classifier. We also offer practical solutions for better assistance to intervention teams using on-board programs in order to improve prevention.

II. Method

Earth resource satellites are procurators of meteorological satellites. These are extensions of the initial Mercury, Gemin ect. Their first missions involved the experimentation of photographic and electronic sensors which provided the first generation of spatial images of the earth, thus stimulating multidisciplinary research and investigation. Beyond the Earth and its environment, the Moon was the next target of Soviet space science. As far back as 1954, when he wrote Report on an Artificial Satellite of the Earth, Mikhail Tikhonrov outlined how a rocket could send a satellite not just into Earth orbit, but on its way to the Moon [10]. In 1972, the USA launched the first civilian land remote sensing satellite, the Earth Resources Technology Satellite, later renamed Landsat 1. The satellite was a highly
innovative apparatus that, as the first of its kind, offered a wide range of applications for environmental monitoring and data gathering. However, in order for users to apply Landsat images to terrestrial problems, openly available and affordable data had to be accessible to all potential users. By 1978, the Carter and later Reagan administrations viewed Landsat as an operational program rather than an experimental project. Thus, they began the process of commercializing Landsat, and met budgetary and programmatic issues that nearly caused the satellite program to cease operations. This chapter argues that the availability of Landsat data fluctuated from 1972 until roughly 1978, when data became available to many users at affordable pricing, but by 1984 Landsat became a commercial entity and began to stifle innovations in Landsat data application [3]. Landsat 8 was launched on February 11, 2013 from Vandenberg Air Force Base in California, on an Atlas rocket. The payload of the Landsat 8 satellite consists of two scientific instruments: the operational field imager (OLI) and the thermal infrared sensor (TIRS). Landsat 8 was developed in collaboration with NASA and the US Geological Survey (USGS). NASA led the design and construction and launch and calibration phases in orbit, during which the satellite was called the Data Continuity Mission [1].

In recent years, with global warming, industrialization, and human interventions, the frequency and severity of forest fires have been increasing significantly in many parts of the world [6, 9, 24]. Flood areas susceptibility is often affected by many factors and has typical nonlinear and complex characteristics; therefore, it is still a difficult task to develop flood or forest fire prediction models with satisfactory accuracy [21]. Various approaches have been developed for modeling floods susceptibility, ranging from physics-based methods to statistical and machine learning (ML) methods [7], [12],[15],[18],[27]. Compared to traditional qualitative and statistical analysis methods, ML approaches have shown the ability to provide better results for the spatial prediction of candidates’ areas [3]. In the last decade, various ML algorithms—such as artificial neural networks[1], [7], [25] random forests (RF) [2], [10],[22],[23], support vector machine (SVM) [20], multilayer perceptron neural network (MLP) [25] [29], kernel logistic regression (KLR) [4],[13] naive Bayes [8],[14], gradient boosted decision trees [41], and particle swarm optimized neural fuzzy [27]—have been successfully developed and widely applied for producing wildfire susceptibility maps. Comparative studies of multiple ML algorithms have also been employed [5],[16],[21],[23]. Therefore, advanced ML approaches are very promising for forest fire spatial prediction. However, the ML approaches mentioned above are pixel-based classifiers with shallow architectures, which do not make use of the spatial patterns that are implicit in images [55]. In addition, these classifiers directly classify the input data without feature extraction, and representative features cannot be mined from the input data to improve classification accuracy. Deep learning (DL) methods [11],[17] have recently received more attention and achieved remarkable success. Deep learning algorithms attempt to discover multiple representation levels [26] and have been broadly applied in areas such as object recognition and detection, speech recognition, and natural language processing [17]. The convolutional neural network (CNN) which has been recognized as one of the most successful and widely used DL algorithms, has produced significant improvements in the latest studies in areas such as disaster damage detection [20],[30] remotely sensed image classification [19],[55] and landslide susceptibility mapping [31]. However, none of these studies evaluated the effectiveness of CNN in the prediction of forest fire susceptibility. The first law of geography [28] emphasizes that near things are more closely related than distant things. Whether a pixel is an ignition point should not only consider the situation of the pixel itself, but also consider other pixels within a certain range around the pixel. While pixel-based classifiers may overlook certain information in spatial patterns, the contextual-based CNN explores the complex spatial patterns that are implicit in images [32]. The performance of the proposed model tested by using Mathworks and ToolsBox, which is an environment for building and evaluating machine-learning algorithms was compared with benchmark methods using several statistical measures. The satellite imagery used in this project contained 280,000 image tiles provided by Esri’s (the market leader in geoinformation systems) World Imagery. The images cover the bounding-box area of Larache in Morocco. For each image tile, the corresponding geographic metadata was tracked. For the training data set 500 images and 200 for the validation dataset were selected, including 200 wind turbines of different types and in different landcover situations. For both sets, wind turbine polygons have been created.
The image shown in figure 2 captured in the day that interests us is noisy, that is to say it is not usable for our study, they present erroneous frames for the network within the 700 satellite images. The application of unsupervised clustering, namely a K-Means colour clustering, helped with pattern recognition and extracting the polygon shape. Another time-consuming challenge was to detect the false positives of the network, i.e. recognized image segments that were falsely identified as lakes or rivers. Further training epochs were needed to train the neural net to differentiate between those similar looking objects.

It therefore requires filtering before use. We present some methods used in filtering digital images. A brief overview of one-dimensional filtering is given, then linear and non-linear techniques are discussed in the context of 2D images. We end with an opening on the variational methods widely used today for deconvolution and restoration of images.
For the realization of the system we opted for an Arduino programmable interface card based on microcontroller and a GSM SIM 900 module and a water sensor. The card allows, from events detected by sensors, to program and control actuators. Their technical characteristics are:

**Arduino board**
- ATmega328P microcontroller
- Operating voltage 5V
- Input voltage (recommended) 7-12V
- Input voltage (limit) 6-20V
- Pins 14 digital I / O (6 of which provide PWM output)
- Digital PWM I / O Pins 6
- Analog input pins 6
- DC Current by I O Pin 20 mA
- DC current for 3.3V Pin 50 mA
- 32 KB flash memory (ATmega328P)
- of which 0.5 KB used by bootloader
- SRAM 2 KB (ATmega328P)
- EEPROM 1 KB (ATmega328P) Clock speed 16 MHz
- Length 68.6 mm - Width 53.4 mm - Weight 25 g

**GSM SIM 900 module**

**Water sensors**
A level sensor is an electronic device which gives a sign to Arduino that it detects water. (Figure) it has three pins:
- 5V
- GND
- S: for an analog arduino output.

For the Arduino board power supply it can be with a USB port or by external power supply. The power source is automatically selected. External power can come from either an AC-DC adapter or a battery. The adapter can be connected by plugging a 2.1mm plug into the card's power socket or from a battery connected in the GND pin (or pin) and V-in (external power supply). The processor can operate on external power from 6 to 20 volts. However, if the voltage is less than 7V, the 5V pin can supply less than five volts and the processor can become unstable.

The GMS / GPRS module from SeedStudio is an Arduino compatible interface card. It allows you to send and receive SMS, data and voice switching from the mobile network. The module is based on the SIM900 circuit from the SIMCOM company. It is controlled via AT commands from an Arduino board. The module is delivered with a remote patch antenna. A connector on the back of the board is provided for receiving a SIM card as well as a Lithium CR1220 battery. Communication between the module and an Arduino board is carried out by the asynchronous serial link: UART or a software serial link. System is like:
III. Results and discussion

To train our we used two classes called "Flood" and "NoFlood". The number of images we used for each class in the training phase is: 1790 images and 142 images used in the test phase is:

Out:

Found 3580 images belonging to 2 classes.
Found 142 images belonging to 2 classes.

And the number of parameters trained are: 1058961 permit 1 059121, that is to say that the number of parameters not trained therefore is: 160.

Out:

<table>
<thead>
<tr>
<th>Layer (type)</th>
<th>Output Shape</th>
<th>Param #</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv2d_1 (Conv2D)</td>
<td>(None, 64, 64, 32)</td>
<td>896</td>
</tr>
<tr>
<td>max_pooling2d_1 (MaxPooling2)</td>
<td>(None, 64, 64, 32)</td>
<td>0</td>
</tr>
<tr>
<td>batch_normalization_1 (Batch)</td>
<td>(None, 64, 64, 32)</td>
<td>128</td>
</tr>
<tr>
<td>dropout_1 (Dropout)</td>
<td>(None, 64, 64, 32)</td>
<td>0</td>
</tr>
<tr>
<td>conv2d_2 (Conv2D)</td>
<td>(None, 64, 64, 16)</td>
<td>4624</td>
</tr>
<tr>
<td>max_pooling2d_2 (MaxPooling2)</td>
<td>(None, 64, 64, 16)</td>
<td>0</td>
</tr>
<tr>
<td>batch_normalization_2 (Batch)</td>
<td>(None, 64, 64, 16)</td>
<td>64</td>
</tr>
<tr>
<td>dropout_2 (Dropout)</td>
<td>(None, 64, 64, 16)</td>
<td>0</td>
</tr>
<tr>
<td>conv2d_3 (Conv2D)</td>
<td>(None, 32, 32, 32)</td>
<td>4640</td>
</tr>
<tr>
<td>max_pooling2d_3 (MaxPooling2)</td>
<td>(None, 32, 32, 32)</td>
<td>0</td>
</tr>
<tr>
<td>batch_normalization_3 (Batch)</td>
<td>(None, 32, 32, 32)</td>
<td>128</td>
</tr>
<tr>
<td>flatten_1 (Flatten)</td>
<td>(None, 32768)</td>
<td>0</td>
</tr>
<tr>
<td>dense_1 (Dense)</td>
<td>(None, 32)</td>
<td>1048608</td>
</tr>
<tr>
<td>dense_2 (Dense)</td>
<td>(None, 1)</td>
<td>33</td>
</tr>
</tbody>
</table>

Total params: 1,059,121
Trainable params: 1,058,961
Non-trainable params: 160

Fig. 4. Parameters trained and parameters not trained by the system

In the evaluation section, we will compare and select the best model in terms of accuracy. Various results of the predictive models are given below:

In order to better situate the predictive Machine Learning models, we start by the confusion matrix which help us calculate the accuracy of the model. The formula to calculate is given below: = (True positive + True Negative) / (True positive + True Negative + False positive + False Negative). Confusion Matrix is a table shows actual vs predicted values. It is one of the easiest ways to find accuracy and also it helps to avoid over fitting. The figure below presents the confusion matrix values for each ML model. The confusion matrix here shows RF model produces 100 % of positive predictive value where the rate of both small (Class=0) and larger (Class=1) fire prediction is 100 % while the
false discovery rate – error type – is 0%. For SVM and KNN, the rate of error that they produce respectively 35%, 45% for the small fire and 29%, 45% for the large fire. In consequence, the performance classification rate of the two models SVM and KNN decrease. The prediction accuracy of Random Forest is interesting. This is one of the best algorithms, which can provide better accuracy when the data is uneven. Hence, it reduces the noise in the data. The overall accuracy of RF is 100%. This shows that RF has the best prediction results comparing with SVM and KNN which they respectively get 67.7% and 54.9% (uci). The Receiver Operating Curve (ROC curve) will summarize the performance of the model by assessing the trade-off between sensitivity and specificity. We must always think of $p > 0.5$ when we draw the ROC, because we are concerned about the success rate. The Area Under the Curve (AUC), usually called the precision index or concordance index, is a metric of the excellent performance of the ROC curve.

Table 1. Accuracy for RF, SVM and KNN method

<table>
<thead>
<tr>
<th>Method</th>
<th>RF</th>
<th>SVM</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>1</td>
<td>0.74</td>
<td>0.5</td>
</tr>
</tbody>
</table>

We implemented around three different Machine learning algorithms for the forest fire data of Larache city in Morocco. From the table 1, we could find the Random forest provides a performance of classification around 100 % while SVM model provides 74% and KNN provides 58% as very low results in terms of classification performance.

With a percentage of 100% our program arrives to know the images which contain the flood among the images which does not. Our system can detect analyzing images if image contain flood or not like presented in next figures:

Fig. 5. Detection of flood

First an image pre-processing was performed to normalize satellite images for differing brightness, saturation and contrast levels. Then the training and validation data had to be generated by visually locating wind parks and turbines in ArcGIS Pro (Esri’s Professional GIS-Tool). The located floods were then marked and converted into georeferenced polygons. After matching the resulting polygons with the corresponding image tile, they were converted into image masks. The mask classifies if an image pixel belongs to a flood or not. This serves as the desired classification scheme for the developed artificial neural network. The deep learning framework used, is based on a U-Net architecture, which has been proven to perform very well for segmentation tasks with a low amount of training data. The segmentation...
performance was tracked using the Jaccard-Index, which is an intersection over union measure. The training was calibrated to achieve the maximum accuracy in the validation set in order to prevent model overfitting. The final layer of the neural net outputs an image mask with a pixelwise prediction of the likelihood of a pixel to belong to a flood zona.

IV. Conclusion

This work allowed us to assess and spatialize the risk of flooding. Remote sensing by satellite and analysis by computer systems are the two pillars of our work. We focused on the study of satellite images of the Larache area in Morocco. The images used are remotely detected by Landsat 8 and Landsat 7 satellites, their processing and analysis, thus their scientific effects will allow the forecast of future floods. In the Hardware Implementation part, we exposed our model which aims to predict floods using the method of convolutional neural networks. The CNN method as well as the video classification in real time are therefore an effective tool for better prediction of floods.

References


