An Approach to Rain Detection using Sobel Image Pre-processing and Convolutional Neuronal Networks

Godoy-Rosario, José A.[1[0000-0003-0859-5590]], Ravelo-García, Antonio G.[1[0000-0002-8512-965X]], Quintana-Morales, Pedro J.[1[0000-0001-8462-8855]] and Navarro-Mesa, Juan L.[1[0000-0003-3860-3424]]

[1] Institute for Technological Development and Innovation in Communications-IDeTIC, University of Las Palmas de Gran Canaria (ULPGC), Las Palmas de Gran Canaria 25017, Spain

{jose.godoy,antonio.ravelo}@ulpgc.es

Abstract. Rain fall detection has been an important factor under study in a multitude of applications: estimation of floods in order to minimize damage before an environmental risk situation, rain removal from images, agriculture field, etc. Actually, there are numerous methods implemented in order to try to solve this issue. For example, some of them are based on the traditional weather station or in the use of radar technology. In this work, we propose an approach to rain detection using image processing techniques and Convolutional Neuronal Networks (CNN). In order to improve the results of classification, images in rain and no rain conditions are pre-processed using the Sobel algorithm to detect edges. The architecture that defines the CNN is LeNet and it is carried out with three convolutional layers, three pooling layers and a soft max layer. With the proposed method, it is possible to detect the presence of rain in certain region of the image with an accuracy of 89%. The purpose of the proposed system is just to complete with a different added value, other traditional methods for detection of rain.

Keywords: Rain fall detection, Sobel image processing, Convolutional Neuronal Network (CNN).

1 Introduction

The determination of weather conditions has supposed an important research factor during years. Sectors which are interested in this issue are numerous: farmers, automotive industry, emergency situations in order to forecast actions before water floods, video processing, etc. In addition, the concept of climate change has acquired a growing interest with the consequent concern of people, governments and in the economic sector [1]. A result of this climatic change is the increase in the risk of storms which could have terrible effects on urban and rural areas. The research work we present here is intended to be part of a monitoring system in order to detect meteorological phenomena.

Traditionally, methods based on weather station are used to detect rain precipitation, however, it is interesting to provide other meteorological measurement sources in order to complete the information provided by traditional weather stations that not always are automatized, thus avoiding real-time rain monitoring.

The presence of rain over images or videos has been widely studied in order to detect the effect of this and to remove this information. In [2], a study to detect and remove rain from images and videos was carried out. Furthermore, they developed a model which describes rain intensities due to rain streaks and a dynamic model that captures the spatial-temporal properties of rain. A similar work is presented in [3] where a system based on computer is showed to detect the presence of rain or snow. However, that work was focused in the detection of the presence of rain or snow streaks. The method consisted on the separation of the foreground from the background in images. It is in the foreground plane where it was possible to detect rain or snow conditions. Then, they applied a Histogram of Orientations of rain or snow Streaks (HOS) to detect the presence of these phenomena. In [4], an image-based disdrometer is developed with the objective of detecting rainfalls due to the movement of raindrops. Other authors have been working in the detection of raindrop using slow motion cameras. Thus, in [5] a method for counting the number of drops and measure their size in a video frame is presented. In [6], a method for real-time raindrops detection in a car windscreen is proposed using cellular neural networks and Support Vector Machines. However, the method presented in [6] is quite different to our proposal. We are centered in rainfall detection in certain distant regions in an image, so the raindrop estimation is not really useful for our objective. Although in the state of the art few works related to rain detection from images are described, there are not enough material to make a comparison analysis in similar conditions.

Other interesting works are related to the possibility of detecting fog through image processing techniques as is the case of [7] where a real-time fog detection system for vehicles is proposed for driving applications. This system is based on the estimation of the visibility distance and blurring due to the fog. In this case, the Sobel algorithm was used as sunny/foggy detector. This method consists on detecting edges on the images. When an image is in presence of fog, it appears a reduction of high frequency components, that can be measured using edge detection methods as Sobel. A similar work to [7] it is presented in [8]. However, in this case, they focused on analyzing the properties of local objects of the roads: lane markings, traffic signs, back lights of vehicles, etc.

Our work is based on a similar principle as studied in [7], however, we used a preprocessing based on Sobel algorithm to make an emphasis on the high frequency components of the image and, then, we train a Convolutional Neuronal Networks (CNN) to classify between rain and no rain images. The use of Neuronal Networks for rain detection has been approached using other sources of information: received signal strength [9] and the carrier-to-noise power ratio (C/N) [10].

An alternative for identification of adverse weather phenomena is the use of neuronal networks tools. These systems are able to learn in a wide range of situations, or meteorological event classes, from a set of multivariate data or images, in order to identify characteristic patterns with complex network models. In recent years, the introduction of Deep Neuronal Networks (DNN), particularly Convolutional Neuronal Networks, which allow automatic learning of feature from images, has obtained great acceptance. The convolutional layers allow the convolution of the input image with a filter, whose weight can be initialized randomly. These weights will be those that are update during the learning process. More recent models of neuronal networks are being development with Convolutional Neuronal Networks architectures which use a variable and numerous sets of filters and convolutional layers that are able to deepen in the learning of very complex models.

The purpose of this paper is to propose an approach to provide general aspects for the detection of distant rain falls using images. For this purpose, we build a CNN based on LeNet architecture. This architecture is able to recognize patterns in an image characterized by high variability and with robustness to distortions [11-12]. It has been applied in a great number of applications like robot vision, image classification, handwriting recognition, etc.

The network will be trained to the identification of rain. The images will be taken from fixed cameras and they will be pre-processed with an algorithm for edge detection to remark the variability of the phenomena. Further experiments about the application of a transformation matrix based on the Discrete Cosine Transform (DCT) have been conducted to study the possibility to improve the classification scores.

This paper is organized as follow: first, in section 2, a brief explaining about the methods used in this work, we introduced the image weather dataset, image pre-processing techniques and the classification system used in this work. In section 3, we explain the performed experiments and obtained results. Finally, some future works and discussions about the study realized are summarized in section 4.

2 Methods

In this section the methods and materials needed to carry out this research are presented. Firstly, it is detailed the dataset used to train and test the CNN. Secondly, it is presented the methods to determinate a Region of Interest (ROI) and the application of the Sobel algorithm for detecting edges in the ROI. Thirdly, the details of the CNN architecture are presented as the basis of the classification system.

2.1 Dataset

In this section, we explain the dataset used in this work. The dataset is divided in two different sections. First of all, we analyze the images which constitutes the dataset. Secondly, we introduce the weather data used to classify the images in rain or no rain conditions.

Dataset of images

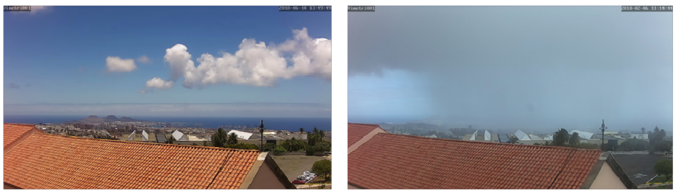
The dataset of images is composed by 2.098 images. These images are distributed as follow:

- 552 images with rainy conditions.

- 1.544 images where no raining conditions are presented in the image.

The images were taken by a video camera [13] located in Campus of Tafira at University of Las Palmas de Gran Canaria, Spain (see point 1 in Fig.1). The images were obtained in different time intervals from January to December of 2018. All images were captured in JPG format with a resolution of 1920x1080 pixels. An example of no rain conditions and rain conditions images are shown in Fig. 2.

Fig. 1. Location Map of the camera and weather stations in the city of Las Palmas de Gran Canaria



(a) Example of no rain image (a) Example of rain image

Fig. 2. Examples of images of rain and no rain conditions used in the image dataset

Weather dataset

The method of classification used in this work consisted of crossing data of AEMET in order to know when precipitation occurred in the weather stations located in "Zurbarán, Tafira" (see point 2 in Fig.1) and "Plaza de la Feria" (see point 3 in Fig.1), both located in the city of Las Palmas de Gran Canaria, because they are the stations which are closed to the position that the camera was filming. Thus, differences between rain or no rain images is possible to obtain. In table 1, it is shown the geographical coordinates of each weather station and camera.

Table 1. Location table of camera and weather stations

Id	Type	Location	Latitude	Longitude	Altitude
1	Camera	University Campus of Tafira	28.07125	-15.45342	317 m
2	Weather Station	Zurbarán, Tafira	28.07829	-15.45304	265 m
3	Weather Station	Plaza de la Feria	28,11317	-15.42116	6m

2.2 Preprocessing

In order to extract relevant information of the image, a Sobel algorithm is applied over a region of interest to detect edges and thus the high frequency components of the image than can be correlated with the presence of rain.

Determination of regions of interest

A ROI is extracted from the images where we want to detect the presence or absence of rain conditions. We have decided to choose a ROI characterized by heterogeneous information due to the presence of buildings. This ROI presents acute high frequency components so we hypothesize that the application of Sobel algorithm can detect with good performance the edges of the image. The selected ROI is specified by coordinates x and y of the pixels that define a rectangle portion of the image. The ROI selected is defined by the points specified in Table 2.

Table 2. Pixel coordinates used to extract the ROI in images

Parameter	Value
Origin (x)	726
Origin (y)	675
Destination (x)	844
Destination (y)	724

This ROI was strategically selected due to that part of the image is static, in other words, not others factors such as vehicles movement, bird flies, etc. modifies the characteristics of the image in this zone. The unique factors that can produced changes in the image are weather conditions: rain, airborne dust or fog. Thus, it is possible to apply images techniques in order to detect changes in the ROI image which corresponds with alterations due to weather phenomena. In the topic of this paper, the most relevant factor that can affect the image is the rain. The ROI that is considered in this approach is marked by a green box and shown in Fig.3.

(a) Studied ROI for a rain conditions image

(b) Studied ROI for a no-rain conditions image

Fig. 3. Selected ROIs for the rain / no rain study

Sobel edge detection algorithm

The main purpose of Sobel algorithm is to detect edges in an image, thus it is a very suitable option to the needs presented in this work. An edge can be defined as a sector of an image where the variability between two consecutive pixels is maximum. According to [14], most edge detection system consider that an edge appears when there is a discontinuity in the intensity function or a very steep intensity gradient in the image. The use of Sobel algorithm as edge detector has two main advantages [15]: First, Sobel introduces an average factor so it has a smoothing effect of the image noise. The other advantage is that Sobel is the result of differentiating two rows or two columns so the edge detected is clearer than the borders detected by other methods.

Sobel is a kind of orthogonal gradient operator. Gradient corresponds to the first derivative and gradient operator corresponds to the derivative factor. For a continuous function or image g(x,y), in a certain position, its gradient can be represented as a vector as follow [15]:

$$\nabla g(x,y) = [G_x G_y]^T = \left[\frac{\partial g}{\partial x}\frac{\partial g}{\partial y}\right] \tag{1}$$

The magnitude and direction angle of the vector can be calculated as follows:

$$mag(\nabla g) = \left|\nabla g_{(2)}\right| = \sqrt{\mathrm{G}_x^2 \mathrm{G}_y^2} \tag{2}$$

$$\emptyset(x,y) = arctang\left({G_x}/{G_y}\right) \tag{3}$$

The partial derivatives of mathematical expressions should be calculated for each image pixel. G_x and G_y are defined by two matrixes which they are showed in (4) and (5). Each image pixel is convoluted with these two matrixes. The first matrix offers the maximum response in the vertical edge and the second matrix gives the maximum response to the level edge. The maximum value of these two convolutions is used as the output bit of the point, and the result is an image of edge amplitude.

$$G_x = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \tag{4}$$

$$G_y = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \tag{5}$$

For a better application of Sobel algorithm, first it is necessary to change de map color of the image from RGB to grayscale.

Classification System

The method used to classify a ROI consists on the use of a CNN which is able to label the images in rain or no rain conditions. A CNN can be defined as a set of convolutional layers which conform a network totally connected. The convolutional layers perform as a wide and varied set of convolutional filters which try to detect patterns in the image. Each convolutional layer has associated an activation layer and a pooling layer which introduces non-linearity and it reduces the characteristics of each layer which enhance the learning and the abstraction of the CNN. The input of the network is composed by images and the output corresponds with a vector of 2 units which are related to the probability that an input image belongs to rain or no rain type.

Filters used in each layer are fixed sized (I x J x C), where I is the image width, J is image height and C corresponds with the number of channels. Each layer can have a different number of filters and different sizes. Each convolutional filter works with the image regardless of the layer which it is processing. The number of resultant characteristic maps of the convolutions corresponds with the number of filters which are working in the layer. Once, the images pass the activation layer, a pooling layer serves to the next convolutional layer or to the end of the layer. All filters will be adjusted as adaptive form along the training process. These filters will extract the most outstanding features, not only of the original image, also the features of the characteristics of the successive convolutional layers. The pooling layer reduces the number of characteristics taking the maximum or other representative value of a kernel of size A x B (where A is the width and B is the height of the filter mask) which segment the image by steps. The kernel and the steps could be different from one layer to the next. As we are advancing through the convolutional layers, the characteristics will be easier to detect. This factor indicates an increase in the abstraction level of the extracted features. Usually, as the number of characteristics extracted by pooling layer is reduced, the number of characteristics extracted by the filters increase in each convolutional layer. Finally, the last layer feeds the completely connected layer which ends to extract the more relevant characteristics to identify the evaluated event and it is possible to provide a probability measurement in the last layer.

The training of the CNN will adjust the parameters of the filters and the weights completely connected, spreading the error from the input to the output of the CNN. The system will minimize a loss function based on crossed entropy.

The designed CNN is compound by 3 convolutional layers, the first has 20 filters, the second layer has 40 filters and the third layer has 60 filters. Each filter has a size of 5x5x3 and with adjustable parameters. Each convolutional layer has an activation layer type ReLu that implements the next function:

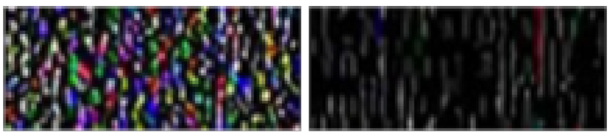
$$f(x) = \max(x, 0) \tag{6}$$

Where x is the neuron input. Then of each activation layer there is a pooling layer of size 2x2 and a step of 2 pixels. Finally, a completely connected network of 2 layers is performed, which corresponds with the classes we want to identify, rain vs no rain.

3 Experiments and Results

In order to train and test the CNN and measure the performance of the system a set of tests were performed. First, it is necessary to select those images used to test or train the CNN. The classification of these images was realized according to data of the Spanish Meteorology Agency (AEMET).

Once the images were classified in rain or no rain conditions, the next step is to select the number of images used to train the CNN. Typically, the number of images of each type should be balanced. Thus, we use the same number of images of rain and no rain to train: 280 images. This number of images corresponds with, approximately 50% of rain images of our dataset due to the little number of samples of rain condition images that we dispose. The rest of images will be used to test the network. Subsequently, the pre-processing explained in section 2 is applied to all the images (training and testing). First of all, the same ROI is extracted to each image and later the Sobel algorithm to detect edges is also applied. As a result, an example for rain and no rain images after applying this process is shown in Fig. 4. With these resultant images, the training of the CNN is performed. The result of this training is a model which was used to test the CNN.



(a) (b)

Fig. 4. Comparison between no rain (a) or rain (b) images after Sobel algorithm was applied to the selected image ROIs.

With the resultant images from applying the pre-processing techniques explained in section 2 the training of the network is performed with the objective of create a model of CNN.

We train three different models of CNN: the first CNN was trained with images not previously filtered, the second of them was trained with the images where a transformation matrix based on the Discrete Cosine Transform (DCT) was applied, and finally we train the CNN with the images filtered by Sobel algorithm. Some interesting parameters for the evaluation of the CNN are collected in table 3. The result of the training was a generated model for each one.

As this work represents an approach, and the size of the database has a limited number of examples, the design of the experiments considers aspects related to generalization. Thus, we repeat the test 20 times and the number of epochs is 30. In Table 4, it is shown a comparison of percentage of success (accuracy) and standard deviation of each model of neuronal network used in each test.

Table 3. Interesting parameters for CNN training.

Parameter	Value
Epochs	30
Initial Learning Rate (LR)	$1e^{-5}$
Batch Size (BS)	20

Table 4. Comparison between accuracy and standard deviation of each CNN model

Model	Accuracy	Standard deviation
Original ROI images	84.47 %	1.74
Applying Sobel filter	91.97 %	1.79
Applying DCT	79.89 %	1.29

As can be seen in Table 4, the percentage of mean accuracy increases approximately in 8 % in the detection of rain/no rain images when the CNN is trained with images filtered with Sobel. On the other hand, the application of DCT worsen the accuracy results of the proposed system in this approach. As it can be seen in Table 4, the results in each iteration are quite stable as standard deviation indicates.

Furthermore, to continue analyzing the performance of the proposed system, a set of tests was realized with the goal to determine accuracy, sensitivity and specificity of the use of CNN trained with images in which the Sobel algorithm was applied. This test consists in making an analysis of true positives, false positives, true negatives and false negatives to the results obtained from the test of the CNN. In this work, a true positive is considered as the determination of rain realized by the CNN of an image with raining conditions. Consequently, a true negative can be considered as the performance of the CNN to classify a no rain image as no rain. Thus, we analyzed a total of 400 images in where rain conditions occur and other 400 images with no rain conditions. The images were randomly selected from the total of images that make up our dataset.

In Fig. 5, each column corresponds with the real value that the CNN should predict and each row indicates the value offered by CNN as output. The figure specifies the values of truth positives (rain asserts), truth negatives (no rain asserts), false positives (rain fails), false negative (no rain fails).

Values obtained by CNN	Reference values: Rain	Reference values: No rain
Rain	385	73
No rain	15	327

Fig. 5. CNN results for rain and no rain images.

According to data of Fig. 5, it is possible to determinate the accuracy, sensitivity and specificity of the CNN for raining detection in images making use of the following expressions:

$$Accuracy = \frac{T_p+T_n}{T_p+T_N+F_p+F_n} \tag{7}$$

$$Sensitivity = \frac{T_p}{T_p+F_p} \tag{8}$$

$$Specificity = \frac{T_n}{T_n+F_p} \tag{9}$$

Where T_p is the true positive values, T_n is the true negative, F_p is the false positive and F_n is the false negative results of the CNN output. If data shown in Fig. 5 is applied in (7), (8) and (9) it is possible to calculate the performance of the proposed system:

$$\text{Accuracy} = 0.89\ (89\ \%) \tag{10}$$

$$\text{Sensitivity} = 0.84\ (84\ \%) \tag{11}$$

$$\text{Specificity} = 0.817\ (81.7\%) \tag{12}$$

4 Discussion and Future Work

The research work we present here is a contribution to the design of a monitoring system to detect meteorological phenomena in risk situations. Particularly, we make an approach to provide video systems a complementary value in weather observation services.

The determination of a ROI is not a trivial task. As mentioned previously, the ROI should be selected in such a way that a meteorological phenomenon produce changes in the image. However, other weather conditions similar to rain could produce misleading results, for example, fog or airborne dust. Thus, the results of detection through the proposed method can be improved correlating other types of data as, for example, weather stations measures. Furthermore, other methods to detect edges can be used [16], some based on gradient operators as: Canny operator [17], Prewitt operator [18] and Roberts operator [19]. Other methods can be based on the use of Laplacian operators: Marr-Hildreth [20] or Zero Cross [21]. As a last limitation we should expose the lack of rain images used in the dataset that could reduce the performance of the model as a classification system.

In any case, we can conclude that it is possible to detect meteorological phenomena in the images making use of difference techniques. In this case, the suggested system is composed by pre-processing tasks of the images in order to extract the characteristic parameters. The Sobel algorithm has shown a good performance due to the capacity the algorithm has to detect edges in rain conditions images. The main advantage of performing a pre-processing and training based on ROI is that we can segment the images in different zones. The use of a CNN has reached good performance as the results indicate. In this work, we obtained an accuracy of about 89% and a sensitivity of 84%. These values give us an idea of the reliability of the implemented system for detecting rain conditions in the image.

As part of our future work, we are also working on the generation of a database with more examples. This objective can be achieved with longer recordings and a higher number of cameras. Thus, it will be possible to test the proposed system in other heterogeneous scenarios.

Additionally to detection, rainfall quantification can also be proposed using the powerful of CNN and other machine learning techniques. Another open issues are: 1) the selection of several ROIs and the integration of the detection over them in order to increase the quality of our detector; and 2) analyze and compare different techniques of detecting edges and use different neuronal network architectures.

5 Acknowledgments

The authors acknowledge the involvement of the Spanish Meteorological Agency (AEMET) [22] for the support in the acquisition of meteorological data. The research work presented in this paper has been financed by the Programa de Cooperación Territorial. INTERREG V A España-Portugal. MAC 2014-2020, Eje 3 [23]. Project VIMETRI-MAC "Sistema de vigilancia meteorológica para el seguimiento de riesgos medioambientales/Meteorological monitoring system for tracking environmental risks", código MAC/3.5b/065 [24].

References

1. United Nations Website, http://www.un.org/en/sections/issues-depth/climate-change/index.html, last accessed 2019/03/04
2. Garg, K., Nayar, S.K.: Vision and Rain. International Journal of Computer Vision, Vol. 75, Issue 1, pp. 3-27 (2007).
3. Bossu, J., Hautiere, N., Tarel, J.P.: Rain or Snow Detection in Image Sequences Through Use of a Histogram of Orientation of Streaks. International Journal of Computer Vision, Vol. 93, Issue 3, pp. 348-367 (2011).
4. Sawant, S., Ghonge, P.A.: Estimation of Rain Drop Analysis Using Image Processing. In: International Journal of Science and Research (2013).

5. Al Machot, F., Ali, M., Haj Mosa, A., Schwarzlmüller, C., Gutmann, M., Kyamakya, K.: Real-time Raindrop Detection Based on Cellular Neuronal Networks for ADAS. Journal of Real-Time Image Processing, pp. 1-13 (2016)
6. Chen, C.Y., Hsieh, C.W., Chi, P.W., Lin, C.F., Weng, C.J., Hwang, C.H.: High-Speed Image Velocimetry System for Rainfall Measurement. IEEE Access, Vol. 6, pp. 20929-20936 (2018).
7. Bronte, S., Bergasa, L.M., Alcantarilla, P.F.: Fog Detection System Based on Computer Vision Techniques. In: 12th International IEEE Conference on Intelligent Transportation Systems (2009).
8. Pavlic, M., Belzner, H., Rigoll, G., Ilic, S.: Image based fog detection in vehicles. In: IEEE Intelligent Vehicles Symposium (2012).
9. Beritelli, F., Capizze, G., Lo Sciuto, G., Napoli, C., and Scaglione, F.: Rainfall Estimation Based on the Intensity of Received Signal in a LET/4G Mobile Terminal by Using a Probabilistic Neuronal Network. IEEE Acess, Vol. 6, pp. 30865-30873 (2018).
10. Gharanjik, A., Bhavami Shankar, M.R., Zimmer, F. and Ottersten, B.: Centralized Rainfall Estimation Using Carrier to Noise of Satellite Communication Links. IEEE Journal on Selected Areas in Communications, Vol. 36, No. 5 (2018)
11. Joshi, A. M., Thakar, K.: A Survey on Digit Recognition Using Deep Learning. International Journal of Novel Research and Development, Vol. 3, Issue 4 (2018).
12. Dung, P.V.: Multiple Convolution Neural Networks for an Online Handwriting Recognition System. In: 6th International Conference on Advances in System Simulation – SIMUL (2014).
13. Oneway devices Homepage, http://www.onewaydevices.com/product/owipcam25.php, last accessed 2019/02/27.
14. Vicent, O.R, Folorunso, O.: A Descriptive Algorithm for Sobel Image Edge Detection. Proceedings of Informing Science & IT Education Conference (2009).
15. Gao, W., Yan, L., Zhan, X., Liu, H.: An improved Sobel Edge Detection. 3rd International Conference on Computer Science and Information Technology (2010).
16. Anphy, J., Deepa, K., Naiji, J., Silpa George, E., Anjitha, V.: Performance Study of Edge Operators. In: International Conference on Embedded Systems (2014).
17. Qiang, S., Guoying, L., Jingqi, M., Hongmei, Z.: An Edged-detection method based on adaptive canny algorithm and iterative segmentation threshold. In: 2nd International Conference on Control Science and System Engineering (ICCSSE) (2016).
18. Dong, W., Shinsheng, Z.: Color Image Recognition Method Based on Prewitt Operator. In: International Conference on Computer Science and Software Engineering (2008).
19. Rosenfeld, A.: The Max Roberts Operator is a Hueckel-Type Edge Detector. IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume: PAMI-3, Issue 1, pp. 101-103 (1981).
20. Öztürk, C. N., Albayrak, S.: Edge detection on MR images with Marr-Hildreth method extended to third dimension. In: 23nd Signal Processing and Communications Applications Conference (2015)
21. Haralick, R.M.: Zero Crossing of Directional Derivative Edge Operator. Proc. SPIE 0336, Robot Vision, (1982)
22. AEMET Homepage, http://www.aemet.es/en/portada, last accessed 2019/02/27.
23. Interreg-Mac Homepage, approved projects, https://www.mac-interreg.org, last accessed 2019/02/06.
24. ViMetRiMAC Homepage, http://http://www.vimetrimac.ulpgc.es/, last accessed 2019/02/06.