# Real Time Weed Detection using a Boosted Cascade of Simple Features

**Adil Tannouche, Khalid Sbai, Miloud Rahmoune, Rachid Agounoune, Abdelhai Rahmani, Abdelali Rahmani**
Laboratoire d'Etude des Matériaux Avancés et Applications, FS-EST, Moulay Ismail University,
BP 11201, Zitoune, Meknes, Morocco

**Article Info**

***Article history:***

Received Mar 7, 2016
Revised Aug 14, 2016
Accepted Aug 28, 2016

***Keyword:***

AdaBoost algorithm
Artificial vision
Haar-like features
Precision agriculture
Weed detection

## ABSTRACT

Weed detection is a crucial issue in precision agriculture. In computer vision, variety of techniques are developed to detect, identify and locate weeds in different cultures. In this article, we present a real-time new weed detection method, through an embedded monocular vision. Our approach is based on the use of a cascade of discriminative classifiers formed by the Haar-like features. The quality of the results determines the validity of our approach, and opens the way to new horizons in weed detection.

***Corresponding Author:***

Adil Tannouche,
Department of Electrical and Computer Engineering,
LEM2A, High School of Technology,
Meknes, +212 65 09 60 207, Morocco.
Email: tannouche@gmail.com

## 1. INTRODUCTION

Weed control is a key element in precision agriculture. Chemical weed control is widely practiced to reduce infection rates and improve crops. In this regard, recent computer vision research has spawned several effective techniques for the detection and / or classification of weeds. In [1] the authors have developed a rapid shape descriptor for the selection of weeds. In [2] the authors elaborated a multilayer perceptron-based technology to real time detection of weeds in growing onions. In [3] a hybrid of support vector machine (SVM) with Radial basis function (RBF) are used to identification of four major types of crop pest which occurs on teff, wheat, sorghum, barley and maize. In [4] the authors proposes a novel angular texture pattern (ATP) extraction method for crop and weed discrimination to overcome the defectsof background complexity and variation of illumination in the agricultural scene. In [5] the authors introduce a new weed detection system using fractal-based low cost commodity hardware Raspberry Pi.

In this article, we present a new method for detecting adventitious board by monocular vision. The spatial information contained in the images from the field is described by the rectangular filters called Haar-like features [6]-7]. Then, this discriminative descriptor utilizes a concept of cascade similar to that proposed in [8-9] to find weeds in cultivated land. The originality of our approach lies in the study and evaluation of this technique of artificial vision, which is widely known and recognized in object recognition in the area of weed detection.

## 2. MATERIAL AND METHOD

The vision system presented here is intended to eventually be embedded on agricultural machinery to optimize its operation in the field (Figure 1). This concept is used in differentworks [2], [10] and [11], its interest lies in its easy way of implementation and low cost. This entire work is based on a detailed analysis of images from the field, with artificial vision tools to extract specific information to the decision on the location of weeds.

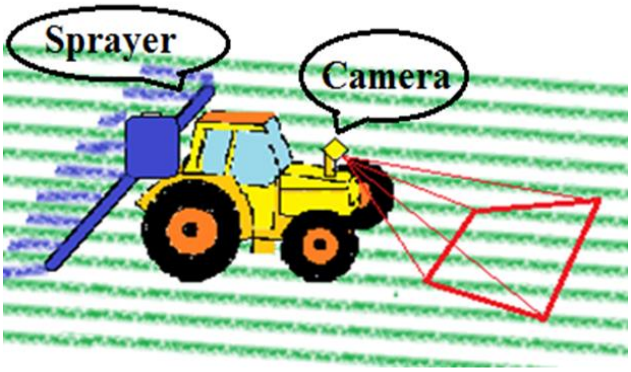


Figure 1. Experimental setup

Real-time weeds detection is required for this type of application, in such a way as to leave a response time to the sprayer control. However, exhaustive search of weeds on the complete image is prohibitive for real-time application. To solve this problem, in the literature, most object detection methods are divided into two stages according to attentional processes [12]:

a. Development of hypotheses: the system provides a very simple and quick way of potential positions of objects to be detected in order to restrict the search field.
b. Verifying hypotheses: The previously developed hypotheses are validated by using more complex algorithms.

The approaches used to validate these assumptions are based on the use of primitives or the appearance of objects in question. The primitive methods define a template for each particular class. Thus, the validation is performed by correlation between the model and the input image. Secondly, methods based on appearance learn the characteristics relating to each class from a set of images. In fact, a vector of local or global descriptors describes each image used in learning. Then, learning a classifier (Neural Networks, Support Vector Machine (SVM), Bayesian Act, etc.) generally permits the estimation of a boundary decision for the assignment of a new object to a class or another. This commonly known discriminative cascade of weak classifiers technique has been successfully used in the literature for the real-time detection of vehicles [13-14] and pedestrians [15-16]. In addition to these classification methods, the authors in [8] propose an original concept attentional applied to the face detection. Its principle is based on the use of increasing complexity classifiers cascade. Thus, each stage of the cascade can increasingly restrict, the research area by rejecting an important set of areas not containing faces. This method uses the so-called Haar-like features also called rectangular filters [6-7] and the learning algorithm AdaBoost [17]. This allows them to select on each stage a limited number of descriptors. The operation of the set, in real time, is ensured by the combined use of rapid calculation Haar-like features, based on the concept of the integral image, and the discriminative cascade. In this article we consider the rectangular filters (Haar-like features). These filters initially introduced by [6-7] to detect pedestrians and vehicles are based on a decomposition of the image with Haar wavelets. Then, their use has improved so much in different studies [8-9], it is no longer strictly correspond to the wavelet theory. They are then appointed Haar-like features. Currently, these filters are commonly and widely used in pattern recognition for detection or object recognition. Our approach proposes to integrate discriminative classifier level descriptors Haar. So we studied the behavior of detectors, similar to the one proposed in [8] and constructed with Haar descriptors associated with discriminant classifiers [14]. The choice of descriptors can be discussed but the main objective of this study is to show the feasibility of our theory. In the remainder of this article, we will develop our idea and present more precisely how we defined descriptors and the associated classifier used in AdaBoost algorithm. Then we will explain our implementation with or without a cascade, then present several results on the basis of embedded images. These results will allow us to analyze the realized system and conclude our approach.

### 2.1. Descriptor

Positive examples (weeds) or negative (crop and other), are distributed in a N-dimensional space which depends on the parameters selected to represent the information. In the original space (defined by the image of gray levels), these examples can be mixed. Thus, the distinction is made by selecting the space of representation and the right classifier.

In this work, Haar-like features are used to define a discriminating model that separates between regions of weeds grown in two classes with a boundary (hyperplane). Test examples are classified according to their position in space with respect to the separating hyperplane.This space can extract high-level information (outline, texture, etc) from image of gray levels. In this section, we describe the space of parameters selected to carry out the detection of weeds.

Haar-like features or rectangular filters, provide information about the distribution of gray levels between two neighboring regions in the image. Figure 2 shows all Haar-like filters used in our work. The chosen filters are those two and three rectangles. To get the value (output) of a filter applied to a region of the image, the sum of the pixels in the white rectangle is subtracted from the sum of the pixels in the blue rectangle (multiplied by a coefficient, in the case of the filter three ridges).

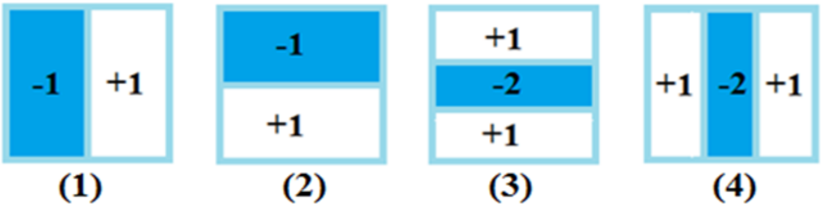


Figure 2. Haar-Like Filter Assembly Used

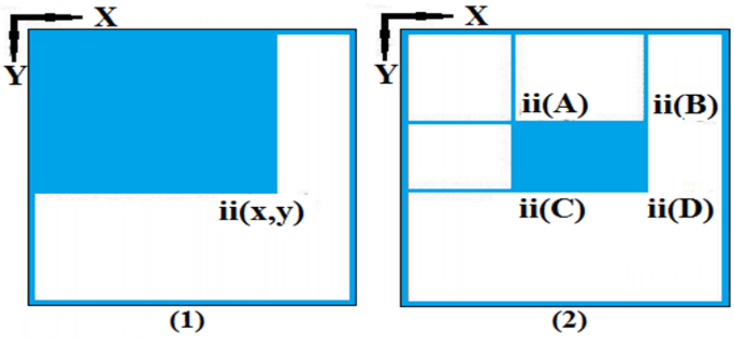


Figure 3. Concept of Integral Image: Image integral (1), Calculates any Square Area from Four References A, B, C and D (2)

In [8], the authors introduced the concept of the integral image. This is an intermediary representation of the input image that allows us to reduce the computation time associated with the application of these filters (especially when they overlap). The value of the integral image, named ii (x, y) at the position (x, y) is the sum of all the pixel values of the original image, above and to the left of (x, y) (Figure 3 (1)), calculation is performed according to this formula:

$$ii(x, y) = \sum_{x' \leq x \quad y' \leq y} i(x', y') \qquad (1)$$

with i (x, y) represents the luminance of the pixel in the original image at the position (x, y). Thus, the sum of the pixels of any rectangular area of an image can be evaluated from four points of references in the corresponding integral image. As an example, in Figure 3(2), where the sum of the values of the blue area zone (A, B, C, D) is given by the following formula:

$$\mathrm{Zone}(A,B,C,D) = \mathrm{ii}(D) - \mathrm{ii}(B) - \mathrm{ii}(C) + \mathrm{ii}(A) \tag{2}$$

Therefore, and for a two-rectangle filter, the difference between the two adjacent rectangles is obtained via six points of references in the integral image. In the case of a three-rectangle filter, eight points of references are used.

Figure 4 illustrates an example of filtering a picture of our database with a vertical rectangular filter (2×2) pixels wide. The figure shows that the selected filter enhances the vertical edges (bright pixels) of the image (Figure 4 (2)). We note that doubling the size of the filters, we filter the details of the original figure, while retaining the main outlines.

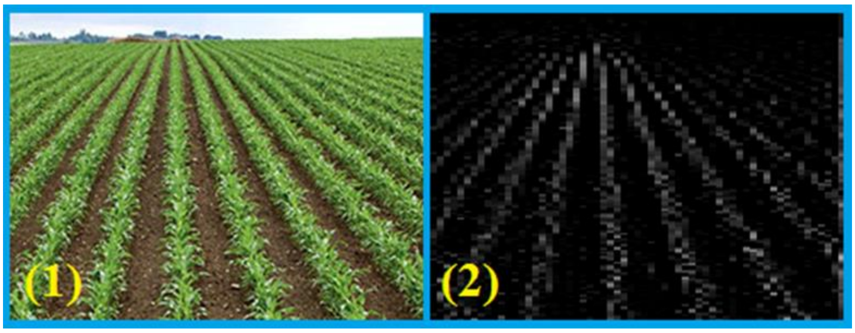


Figure 4. Application of vertical Haar-like features filters (2) on an original image (1)

Each descriptor (j) is defined as a function $f_j$ ($x_i$, $y_i$, $E_j$, $C_j$), where ($x_i$, $y_i$) the position in the thumbnail, $C_j$ is the type Haar-like features (Figure 2) and $E_j$ is the scale (we use 5 in total: 1×2, 2×4, 4×8, 8×16 and 16×32). Filters roam the thumbnails in steps of one pixel. For a thumbnail image of (32x32) pixels size, the Haar space parameters is thus of 8151 dimension.

## 2.2. Classifier

The dimensional size of the parameter spaces is much greater than the number of pixels of the input image. Using all this together to achieve the classification is an inadequate choice from the standpoint of the execution time and robustness, since some of these parameters do not contain relevant information (noise).

The doping algorithm known Adaboost [17] allowed the improved performance of several classification and detection systems. It gives specific hypotheses combining several weak classification functions with moderate accuracy in general. This is an iterative algorithm that searches in the descriptor vector space, the most discriminating weak classification functions to combine them into a strong classification function according to this formula:

$$G = \begin{cases} 1 & \sum_{t=1}^{T} \propto_t g_t \geq \frac{1}{2}\sum_{t=1}^{T} \propto_t = S \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

where G and g are the strong and weak classification functions respectively. α is a weighting factor for each weak classifier g, S being the overall classification of the function threshold G. Different variations of the doping algorithm were developed : Discrete AdaBoost [8] Real AdaBoost [18] Gentle AdaBoost, etc. a recent performance of these alternatives analysis is presented in [19]. We use here the first defined by the following pseudo algorithm:

1. Discrete AdaBoost algorithm

```
1. Let N examples

   $(x_1, y_1) \dots (x_n, y_n)$  With  $x \in \Re$  and  $y \in \{1,0\}$

2. Initialize

   $w_i = \frac{1}{N}$  With  $i = \{1, \dots, N\}$
```

3. For

$$t = \{1, \dots, T\}$$

For each descriptor (j), train a classifier ($G_i$). The error is given by:

$$\varepsilon_j = \sum_{i=1} w_i \left|g_j(x_i) - y_i\right|$$

Choose ($G_t$) classifier having the error ($\varepsilon_t$) the smaller;
Update the weights:

$$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i} \text{ With } \beta_t = \frac{e_t}{1 - e_t} \text{ and } e_i = \begin{cases} 0 & g_t(x_i)=y_i \\ 1 & \text{Otherwise} \end{cases}$$

4. Output:

$$G = \sum_{t=1}^{T} \propto_t g_t \geq \frac{1}{2}\sum_{t=1}^{T} \propto_t \text{ With } \propto_t = \log\left(\frac{1}{\beta_t}\right)$$

To use it, we must now define the weak classification functions for the type of descriptors chosen. We define low classification function associated with the descriptor (j), as a binary response given by the following equation:

$$G_{Haar} = \begin{cases} 1 & p_j f_j < p_j \theta_j \\ 0 & \text{Otherwise} \end{cases} \quad (4)$$

where ($f_j$) gives the value of the descriptor, ($\theta_j$) is the threshold and ($p_j$) is the parity. For each descriptor (j), AdaBoost determines the threshold optimal ($\theta_j$) minimizing the number of misclassified examples of the learning base (positive and negative).

### 2.3. Implementation

In this section, we describe the content of the image databases used for training and testing. Then we analyze in more details the learning of the selected detector. In use, the research area is restricted in the image using a prior knowledge of the agricultural scene. The image is traversed by a sliding window evaluated by the detector to the working resolution set to (32×32) pixels.

#### 2.3.1. Data base

The database used is designed from real images captured manually and collected on the net or from simulation agricultural scenes [20]. It consists of over 500 thumbnails of different crops (corn, wheat, beans, potato, bean, etc.), healthy and infected by weeds. These thumbnails taken at different times of day are randomly transformed by rotation (π), by translation and scale factor to increase the size of the database. Then, this database was sorted and labeled by hand in both positive and negative classes respectively in the presence or absence of weeds in thumbnails (Figure 5).

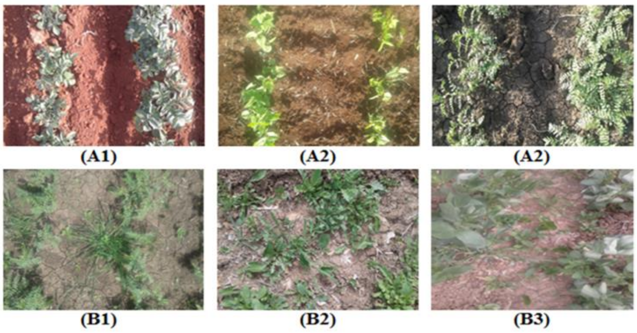


Figure 5. Example of positive thumbnails (Bx) and negative (Ax) used in our study

Since the smallest defined learning scale is a thumbnail resolution of (32×32) pixels, we consider this size as the minimum size detection of objects in the image. This size corresponds to a unitary sub-area of (1 $m^2$) in the scene viewed through a standard RGB camera (640×480) resolution, onboard the carrier vehicle (Figure 1).

### 2.3.2. Detector

A simple detector comprises a strong classifier, built with a T number of descriptors (weak classifiers). Four detectors were created for different types of Haar-like features considered (Figure 2).

To assess the performance of the learning method, we will perform a cross-validation procedure. We train the detectors on different learning samples: the distribution of positive and negative samples on learning and validation data is done randomly. The rate of correct detections (True Positives (TP)) is defined as the average of the TP of the four detectors. Similarly, we use the average of false alarms (False Positive (FP)).

Then, we implemented a cascading architecture [8] consisting of the succession of strong classifiers. Each strong classifier is driven by using the AdaBoost algorithm. Instead of stopping the iterative loop of AdaBoost for maximum number of descriptors T, we set two performance thresholds : a minimum rate of correct detections $TP_{min}$ and a maximum acceptable rate of false alarms $FP_{max}$.

The negative base used to drive the strong classifier on actual stage is formed of negative examples that were misclassified (i.e. considered as weeds) in previous stages.

Thus, we have defined four criteria to stop the learning of the cascade:

a. The algorithm achieved the maximum number of iterations for the learning of the strong classifier without reaching the maximum false alarm rate or the minimum rate of correct detections. We state in our results that there was non-convergence of AdaBoost in the last stage (noted No.Conv).
b. The cascade reaches a false positive rate (FP) overall below a given rate, defined by $FP_{max}$= 0.05 (noted $FP_{max}$).
c. It was not possible to find a sufficient number of negative samples (noted N.Sample).
d. We arrived to a fixed number of maximum stages of 20 (noted $E_{Max}$).

Three different versions of the cascade are obtained by varying the amount of negative samples used for the learning: 100, 200 and 300.

Eventually, to avoid the non-convergence of a cascade (which happens frequently with AdaBoost), we will modify the learning criterion of the classification function. if a fixed limitation attached to each stage on the selected number of descriptors is achieved without reaching a convergence (at $VP_{min}$ and $FP_{max}$ level), then AdaBoost iteration is stopped and the classification function is retained in the state, then we move on to learning the function of the next stage. This approach does not betray the fundamental concept of the cascade, since its purpose is retained, always acting to eliminate a substantial portion of negative samples (not containing weed) while keeping the positive samples (containing weeds). To set the number of descriptors to each stage of the cascade, we can use an increasing function at our convenience. Here we have chosen to follow an exponential law.

## 2.4. Real-Time Process

We have adapted the proposed method for real-time application. The system consists of an onboard monocular camera on the tractor moving in the direction of sowing lines (Figure 1). We wish to detect weeds located in front of the vehicle to actuate the spraying of the herbicide. First, we define a region of interest. Then we use "Sliding Window" to cover all this workspace (see Figure 6): with an assumption of flat mode and a prior knowledge of seedling lines geometry. In line with this, the static nature of the agricultural scene (agricultural scene is almost identical in precision agriculture), the hypotheses generated in 3D space are projected directly into the 2D image. At a region of interest (3×2 $m^2$) area at the front of the tractor embedding the camera, a sub area of (1 $m^2$) represents the size of the sliding window. We then obtain a set from 6 to 10 sub-window candidates. All these sub-windows are adjusted in size (32×32) pixels. We then apply the detector, a response is generated for each thumbnail. If the presence of the weed is detected on one of them, the area is declared contaminated. The process stops for the image in process and goes to the next image. The implementation was carried out with the software C# and Open CV.

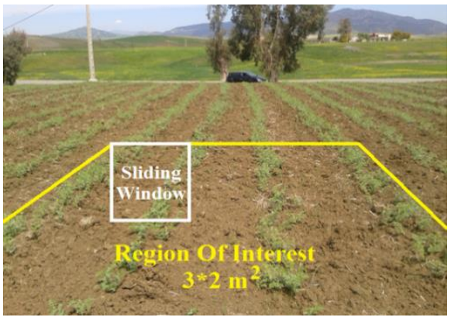


Figure 6. Region of Interest and sliding window

## 3. RESULTS AND DISCUSSIONS

In this section, we analyze the results for the descriptor based on Haar-like featuresdescribed above. For gray levels images with a resolution of (640×480) pixels, the indices of performance considered are:

a. Rate of True Positives (TP) is the percentage of thumbnails representatives weed correctly identified on all the images of the test set

b. Rate of False Positive (FP) is calculated from the average of false alarms per image (calculated on all images of the test basis).

c. Average processing time thumbnail of (32×32) pixels is evaluated on a regular laptop (Core i7 @ 2.66 GHz and 6GB RAM).

We varied the value of the maximum number of descriptors (T = 50, 100 and 150 descriptors). Figure 7 shows the ROC curves for the simple detector (Figure 2).

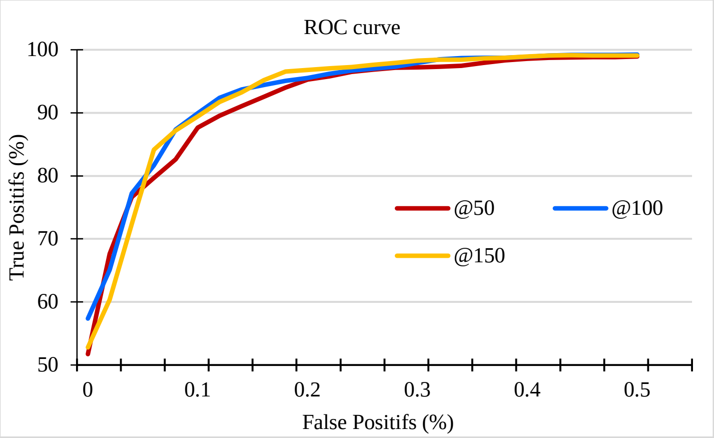


Figure 7. ROC Curve for T = {50, 100, 150}

Table 1. Results Obtained for the Single Detector for the Good Detection Rate 95%

| Nb. Descriptor(T) | TP(%) | FP(%) | Time($10^{-3}$ s) |
|---|---|---|---|
| 50 | 95.2 | 0.16 | 18 |
| 100 | 95.6 | 0.11 | 40 |
| 150 | 95.2 | 0.09 | 64 |

Table 1 shows the results for the single detector with a threshold of correct detections based on positive validation set at 95%.We also observe that the computation time per image naturally increases with the amount of descriptors. Using a large number of descriptors refines the separation boundary, but becomes unrealized as far as an application in real time is concerned. Hence cascading architectural interest.

Table 2 indicates the cascade detector architecture and performance. The performance parameter values for the function Gi are the minimum rate of correct detections $TP_{min}=0.99$, and the maximum acceptable rate of false alarms $FP_{max} = 0.0005$. We observe the existence of significant disparity in the amount of descriptor stages by the cascade. We also note that all learning is stopped because of the non-convergence of the AdaBoost algorithm.

Table 2. Results for the Detector in Cascade

| Nb. Base Negatif | Nb. Stage | Nb. Descriptor | TP (%) | FP (%) | Time ($10^{-3}$ s) | Stopping Criterion |
|---|---|---|---|---|---|---|
| 100 | 12 | 400 | 95.4 | 0.008 | About 8 | No Conv. |
| 200 | 11 | 479 | 96.4 | 0.007 | About 7.5 | No Conv. |
| 300 | 10 | 272 | 97.7 | 0.001 | About 4 | No Conv. |

When we increase the number of negative thumbnails in learning, the process will converge more for a number of lower stages. This is easily explained: a larger negative base allows the generation of a strong enough border to eliminate, from the first stages of the cascade, a large number of false detections. Very quickly, leaving only the most difficult cases to reject, hence the non-convergence of the algorithm. Moreover, if we look at the number of descriptors selected in each stage (Figure 8), we find that the Haar detector requires more descriptors to estimate correctly better border between classes.

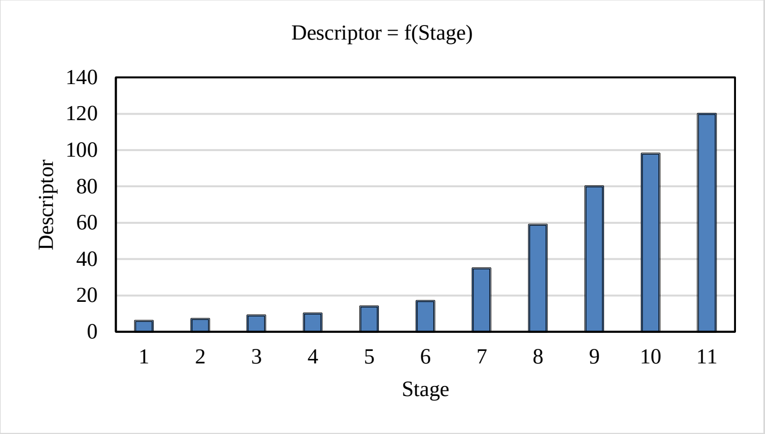


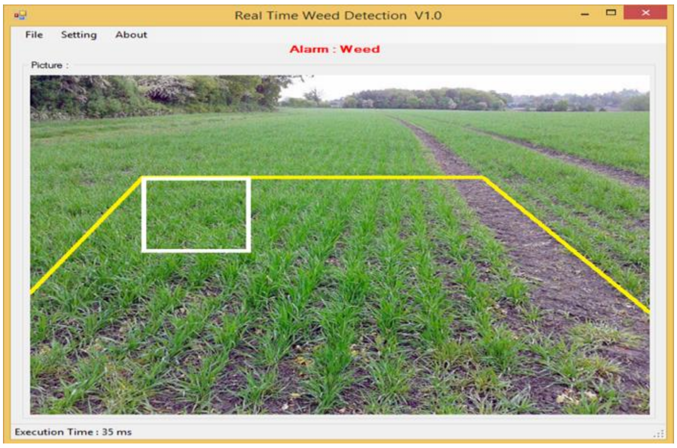
Figure 8. The Amount of the Descriptors Included in each Stage by the Detector for 100 Negative Thumbnails

Table 3 shows the performance achieved by our method by applying the limiting rule of the number of descriptors per stage in the controlled cascade. This technique allowed us to increase the total number of descriptors and to achieve a rate of good detections of 99% and a false alarm rate of 0.0005% by avoiding the non-convergence of the AdaBoost algorithm. The running time is acceptable for an application in real time through the principle of attentional cascade. Indeed, the average processing time for an image is in the order of 0.2 second. Thus, theoretically, the tractor can reach a maximum speed about (10 m /s) or (36 km / h). Figure 9 shows an example of weed detection in one frame.

Table 3. Results for the Detector in Controlled Cascade

| Nb. Base Negatif | Nb. Descriptor | TP (%) | FP (%) | Time ($10^{-3}$ s) |
|---|---|---|---|---|
| 100 | 866 | 99.2 | 0.0005 | About 15.5 |
| 200 | 812 | 99 | < 0.0005 | About 15 |
| 300 | 802 | 99 | < 0.0005 | About 15 |

This Figure shows the sliding window (White Square) focused on a populated area of weeds in the region of interest (Yellow Square). The running time of this sliding window is (35.10$^{-3}$ s) as shown in the Figure 9.



Alarm : Weed

Execution Time : 35 ms

Figure 9. Screenshot of Our Application

## 4. CONCLUSION

We have proposed a new approach by machine learning to detect weeds. The originality of our approach lies in the study and evaluation of artificial vision technology, which is widely known and recognized in object recognition, in weed detection. The method uses Haar-like features to describe the agricultural scene. The description vector created feeds the AdaBoost algorithm. Then, a relevant discriminative cascade of weak classifiers allows the detection of weeds in real-time. The results prove the validity of our approach and show remarkable performance. The correct and false detection rates reach the value of 99% and 0.0005%, respectively, with an execution rate of 5 fps. Research studies are currently underway to test the performance of our method in the field. Other approaches with more complex features (Histogram of Oriented Gradients (HOG) [21], Gabor filter [22]) considered individually or in combinations, in discriminative or generative mode [14] are also under study by our research group.

## REFERENCES

[1] Tannouche A, Sbai K, Rahmoune M, Zoubir A, Agounoune R, Saadani R, Rahmani A. A Fast and Efficient Shape Descriptor for an Advanced Weed Type Classification Approach. *International Journal of Electrical and Computer Engineering (IJECE)*. 2016; 6(3): 1168-1175.

[2] Tannouche A, Sbai K, Ounejjar Y, Rahmani A. A real time efficient management of onions weeds based on a multilayer perceptron neural networks technique. *International Journal of Farming and Allied Sciences (IJFAS)*. 2015; 4(2): 161-166.

[3] Alemayehu DM, Mengistu AD, Mengistu SG. Computer vision for Ethiopian agricultural crop pest identification. *Indonesian Journal of Electrical Engineering and Computer Science*. 2016; 3(1).

[4] Murugan D, Prema P. A Novel Angular Texture Pattern (ATP) Extraction Method for Crop and Weed Discrimination Using Curvelet Transformation. *ELCVIA: electronic letters on computer vision and image analysis*. 2016; 15(1): 27-59.

[5] Suriansyah MI, Sukoco H, Solahudin M. Weed Detection Using Fractal-Based Low Cost Commodity ardware Raspberry Pi. *Indonesian Journal of Electrical Engineering and Computer Science*. 2016; 2(2), 426-430.

[6] Oren M, Papageorgiou C, Sinha P, Osuna E, Poggio T. *Pedestrian detection using wavelet templates*. Computer Vision and Pattern Recognition, 1997. Proceedings. 1997 IEEE Computer Society Conference on (pp. 193-199). IEEE.

[7] Papageorgiou C, Poggio. A trainable system for object detection. *International Journal of Computer Vision*. 2000; 38(1): 15-33.

[8] Viola P, Jones M. *Rapid object detection using a boosted cascade of simple features*. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol. 1, pp. I-511). IEEE.

[9] Lienhart R, Kuranov A, PisarevskyV. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *Joint Pattern Recognition Symposium* (2003; pp. 297-304). Springer Berlin Heidelberg.

[10] Herrera PJ, Dorado J, Ribeiro Á. A Novel Approach for Weed Type Classification Based on Shape Descriptors and a Fuzzy Decision-Making Method. *Sensors*. 2014; 14(8):15304-15324.

[11] Gée C, Bossu J, Jones G, TruchetetF.Crop/weed discrimination in perspective agronomic images. *Computers and Electronics in Agriculture*. 2008; 60(1): 49-59.

[12] Sun Z, Bebis G, Miller R. On-road vehicle detection: A review.*IEEE transactions on pattern analysis and machine intelligence*. 2006; 28(5): 694-711.

[13] Adhikari SP, Yoo HJ, Kim H. Boosting-based on-road obstacle sensing using discriminative weak classifiers. *Sensors*. 2001; 11(4): 4372-4384.

[14] Negri P, Clady X, Hanif SM, Prevost L. A cascade of boosted generative and discriminative classifiers for vehicle detection. *EURASIP Journal on Advances in Signal Processing*. 2008; 136.

[15] Siala M, Khlifa N, Bremond F, Hamrouni K. *People detection in complex scene using a cascade of Boosted classifiers based on Haar-like-features*. In Intelligent Vehicles Symposium, 2009 IEEE (pp. 83-87). IEEE.

[16] Pavani SK, Delgado D, Frangi, A. F.Haar-like features with optimally weighted rectangles for rapid object detection. *Pattern Recognition*. 2010; 43(1): 160-172.

[17] Freund Y, Schapire RE. Experiments with a new boosting algorithm. In *Icml* (1996; 96: 148-156).

[18] Friedman J, Hastie T, Tibshirani R. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*. 2000; 28(2): 337-407.

[19] Wu S, Nagahashi H. Analysis of generalization ability for different adaboostvariants based on classification and regression trees. *Journal of Electrical and Computer Engineering*. 2015, 8.

[20] Jones G, GéeC, Truchetet F. Modelling agronomic images for weed detection and comparison of crop/weed discrimination algorithm performance. *Precision Agriculture*. 2009; 10(1): 1-15.

[21] Dalal N, Triggs B. *Histograms of oriented gradients for human detection*. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). 2005; 1: 886-893. IEEE.

[22] Sun Z, Bebis G, Miller R. On-road vehicle detection using evolutionary Gabor filter optimization. *IEEE Transactions on Intelligent Transportation Systems*. 2005; 6(2): 125-137.

## BIOGRAPHIES OF AUTHORS

**Adil Tannouche** is currently a PhD student at the Laboratoired' Etude des Mateériaux Avanceés et Applications, Moulay Ismail University, Faculty of Sciences in Meknes, Morocco. His research interests are focused in machine vision, artificial intelligence and theirs application in agriculture.

**Khalid Sbai** is a full professor since 2001 in Electronics. He received his M.sc. Degree in Electronics from Valencienne University (France) in 1996 and his Habilitation in Physics from Moulay Ismail University in 2008. His research interests include Structural studies, vibrational and electronic properties of carbon nanotubes

**Miloud Rahmoune** is a full professor at Moulay Ismail University. He received his Msc. Degree in applied mechanics from Universite´ Montpellier 2 (France) and his Ph.D. degrees in Mechatronics from Universit Montpellier 2 (France) and Universite´ Hassan II Mohammedia, in 1993 and 1996 respectively. His research interests include structural Dynamics, active control, and smart materials.

**Rachid Agounoun** is a senior lecturer at University Moulay Ismail, Morocco. He received the MSc. Degree (Magister) in mechanics and energetic system from the Université de Lorraine, Nancy, France and the Ph.D degree in science for engineers from the Universite´ de Lorraine, Nancy, France. He is an active researcher at Thermal & Material Research Unit (advanced materials and energy system). His area of research includes Thermal Comfort, Building Thermal Simulation, renewable energy and Porous Media.

**Abdelhai Rahmani** is a full professor at University Moulay Ismail, Morocco. He received the Ph.D degree in Materials science from the Université Moulayismail, Maroc. His area of research includes Computational physics and nanomaterials.

**Abdelali Rahmani** is a full professor at University Moulay Ismail, Morocco. He received the MSc. Degree in theoretical physics from the Universite´ Montpellier 2, France and the Ph.D degree in Materials science from the theUniversité Montpellier 2, France. He is the Director of Laboratory of Studies of Avanced Materials and Applications). His area of research includes Computational physics and nanomaterials.