**Isolated Handwritten Roman Numerals Recognition using Dynamic Programming, Naïve Bayes and Support Vectors Machines**

R. Salouan  
Department of Mathamatic and Informatic Polydisciplinary Faculty  
Sultan Moulay Slimane University  
Beni Mellal, Morocco

S. Safi  
Department of Mathamatic and Informatic Polydisciplinary Faculty  
Sultan Moulay Slimane University  
Beni Mellal, Morocco

B. Bouikhalene  
Department of Mathamatic and Informatic Polydisciplinary Faculty  
Sultan Moulay Slimane University  
Beni Mellal, Morocco

**ABSTRACT**

Optical character recognition is undoubtedly considered as a one of the most active and dynamic fields of pattern recognition and artificial intelligence; it really provides in fact a solution for recognizing large volume of patterns automatically. The purpose of the present study is to compare in one hand between the performances of three novel hybrid methods used in OCR for extracting efficiently the features from characters which are the structural method called zoning combined in first time with Krawtchouk, then in second time with pseudo-Zernike invariant moments then finally combined with invariant analytical Fourier-Mellin transform in third time, and between the precision of three classifiers which the first one is a statistical that is the support vectors machine, the second is a probabilistic that is the naïve Bayes while the third forms part from optimization that is the dynamic programming on the other hand. For this purpose, we have preprocessed each numeral image by the median filter, the thresholding, the centering and the edge detection techniques. Moreover, the experiments that we have applied provided us convincing and satisfactory results.

**General Terms**


**Keywords**

Isolated handwritten Roman numerals, the median filter, the thresholding, the centering and the edge detection techniques, the zoning method, the Krawtchouk invariant moment, the pseudo-Zernike invariant moment, the invariant analytical Fourier-Mellin transform, the support vectors machine, the naïve Bayes, the dynamic programming.

**1. INTRODUCTION**

Character Recognition (CR) systems offer really potential advantages by providing an interface which makes the interaction between man and machine easier. Some of the powerful application fields of CR there is Optical Character Recognition (OCR) that plays an important role in many different domains as archiving documents and automatic verification of bank checks, etc. on the other hand, several studies carried for character recognition by using the zoning method [1-3] or the invariant moments [4-8] and the support vectors machines [9-12] or the naïve Bayes [13-16] or the dynamic programming [17-20].

In fact, each optical character recognition system is composed from three principal phases which are the preprocessing which serves to clean the character image in order to enhance its quality, in this context we have interested in this work to the median filter, the thresholding, the centering and the edge detection techniques. The second phase is features extraction used to extract some efficient features called also primitives from the numeral image that is in principle presented in form of a matrix in order to convert this last to a vector which will allow thereafter its recognition enough easy, in this sense and for realizing this phase, we exploited the zoning method combined with in firstly with the Krawtchouk Invariant Moment (KIM) [21] then secondly with Pseudo Zernike Invariant Moment (PZIM) [22], thirdly with the Invariant Analytical Fourier-Mellin Transform (IATFM) [23]. The last phase is the recognition, in this framework we have opted three classifiers which are the support vector machines, the naïve Bayes and the dynamic programming. Anyway, this paper is organized as follows: First of all the proposed system is given. In second section the pre-processing process is presented. The features extraction phase is described in third section. The fourth section explains the recognition phase. The experimental results are given in fifth section. Finally, this work is ended by a conclusion. Hence, our recognition system is presented as follow:

<table>
<thead>
<tr>
<th>Image numeral</th>
<th>Pre-processing</th>
<th>Features extraction</th>
<th>Recognition</th>
<th>Output numeral</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Median filter, Thresholding, Centering, Edge detection)</td>
<td>(Zoning + KIM, Zoning + PZIM, Zoning + IATFM)</td>
<td>(Dynamic programing, Naïve Bayes classifier, Support vectors machines)</td>
<td></td>
</tr>
</tbody>
</table>

**Fig 1:** The proposed recognition system.

**2. PRE-PROCESSING**

Pre-processing for image is focused on noise removal and details-enhancement ant to reduce any redundant or useless
information’s which enables producing a much cleaned version of the character image so that it can be used efficiently in the features extraction phase. In this context, we have pre-processed in this study each numeral image by the median filter for filtering the image then by the thresholding used in order to render each numeral image containing only the black (value 0) and the white (value 1) colours according a preset threshold, then by the centering employed to localized the numeral justly in the center of image, finally by the edge detection exploited for finding the edge of numeral.

3. FEATURES EXTRACTION

Features Extraction is the technique by which certain efficient features or primitives from an image are extracted, detected and represented for further processing. In fact, the term ‘feature’ refers to similar characteristics. Therefore, the main goal of a features extraction phase is to accurately retrieve these features. In this sense many methods can be used to compute the features. In this work, we use the zoning method combined with the Krawtchouk then secondly with pseudo Zernike invariant moments, thirdly with the invariant analytical Fourier-Mellin transform.

3.1 The zoning method

In this work we have used the zoning method that can be explained as follow:

Given a black image that contains an numeral written in white, the zoning method consists to divide this image to a several zones then calculating in each of them the number of white pixels, all these numbers are stocked in a vector, that is to say image is converted to a vector has a number of components equal to that of zones.

3.2 The moments of images

In the past decades, various moment and transform functions are the descriptors which are much successfully exploited in pattern recognition field due to their abilities to extract the features of images in an efficient manner. In this sense we have used three powerful invariant descriptors.

3.3 The Krawtchouk invariant moment

3.3.1 The Krawtchouk moment

3.3.1.1 The Krawtchouk polynomial

By definition, the Krawtchouk polynomial of order n is given by:

$$K_n(x; p, N) = \sum_{k=0}^{N} a_{k,n;p} x^k = \frac{\Gamma(a+k)}{\Gamma(a)} e_{p}^{-x} \sum_{k=0}^{N} (a)_k (p)_k \frac{x^k}{k!}$$

(1)

Where $(a)_k = a(a+1)...(a+k-1)$ and $(a)_0 = 1$.

The $\Gamma$ function is defined by:

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt$$

(4)

And:

$$\forall n \in N, \Gamma(n+1) = n!$$

(5)

The set of $(N+1)$ Krawtchouk polynomial $\{K_n(x; p, N)\}$ forms a complete set of discrete basis functions with the weight function:

$$w(x; p, N) = \binom{N}{x} p^x (1-p)^{N-x}$$

(6)

And satisfies the orthogonally condition:

$$\sum_{x=0}^{N} w(x; p, N) K_n(x; p, N) K_m(x; p, N) = \delta_{mn}$$

(7)

Where $\rho(n; p, N)$ is the squared norm defined by:

$$\rho(n; p, N) = (-1)^n \frac{(1-p)^n}{p} \frac{n!}{(-N)!}$$

(8)

And $\delta_{nm}$ is the Kronecker symbol defined by:

$$\delta_{nm} = \begin{cases} 1 & \text{if } n=m \\ 0 & \text{else} \end{cases}$$

(9)
3.3.1.2 The Krawtchouk moment
The Krawtchouk moment have the interesting property of being able to efficiently extract local features of an image this moment of order \((n+m)\) of an image \(f(x,y)\) is given by:

\[
Q_{nm} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} K_n(x; p_1, N - 1) K_m(y; p_2, M - 1) f(x, y)
\]  
(10)

The NxM is the number of pixels of an image \(f(x,y)\). The set of weighted Krawtchouk polynomial \(K_n(x; p_1, N)\) is:

\[
K_n(x; p, N) = K_n(x; p, N) \left( \frac{w(x; p_1, N)}{\rho(x; p, N)} \right)
\]  
(11)

3.3.2 The Krawtchouk invariant moment
The geometric moment of an image \(f(x,y)\) is given by:

\[
M_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} x^p y^q f(x, y)
\]  
(12)

The standard set of the geometric invariant moments that’s independent to rotation, scaling and translation is:

\[
V_{nm} = M_{\infty \infty}^{N-1 \sum_{x=0}^{M-1} (x-y)^n \cos \theta + (y-x) \sin \theta}^n f(x, y)
\]

The Krawtchouk moment invariant is:

\[
\tilde{\Omega}_{nm} = \Omega_{nm} \sum_{i=0}^{n} \sum_{j=0}^{m} \tilde{a}_{i,n,p} a_{j,m,p} V_{ij}
\]  
(14)

\[
\Omega_{nm} = [\rho(n; p_1, N -1), \rho(m; p_2, M -1)]^{-1/2}
\]

\[
V_{ij} = \sum_{p=0}^{i} \sum_{q=0}^{j} \binom{i}{p} \binom{j}{q} \frac{N^2}{2} \left( \frac{N}{2} \right)^{i-p-j} V_{pq}
\]  
(15)

\[
\binom{i}{p} = \frac{x!}{y!(x-y)!}
\]  
(17)

The coefficients \(a_{i,n,p}\) are determined in equation (1).

3.4 The pseudo-Zernike invariant moment
3.4.1 The pseudo-Zernike moment
For an image \(f(x,y)\) the Zernike moment of order \(n\) and repetition \(m\) is given by:

\[
A_{nm} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) V^*(x, y)
\]  
(18)

\[
V^*(x, y) = R_{nm} (x, y) e^{j \arctan (y/x)}
\]  
(19)

\[
R_{nm} (x, y) = \frac{k N+1}{2} \sum_{s=0}^{n} \frac{(-1)^s (x^2 + y^2)^s}{s! (n-s)! (n+1-s)!} (2n+1-s)!
\]  
(20)

\[
n \geq |m|, n \geq 0, j = \sqrt{-1}
\]

\[
x^2 + y^2 \leq 1
\]

The symbol \(*)\ denotes the complex conjugate operator and the NxM is the number of pixels of an image \(f(x,y)\).

3.4.2 The pseudo-Zernike invariants moment
The Zernike moment is invariant under rotation but sensitive to translation and scale. So normalization must be done of these moments.

\[
f(x, y) = f(\tilde{x} + \frac{x}{a}, \tilde{y} + \frac{y}{a})
\]  
(21)

Where \((\tilde{x}, \tilde{y})\) is the center of pattern function \(f(x,y)\) and \(a = (\beta M_{00})^{1/2}\)

\(\beta\) is a predetermined value for the number of object points in pattern.

3.5 The analytic Fourier-Mellin transform
3.5.1 The analytic Fourier-Mellin transform
The standard analytical Fourier-Mellin transform (AFMT) of an function \(f(r, \theta)\) in polar coordinates is given by:

\[
M_{fa}(k, v) = \frac{1}{2\pi} \int_{0}^{2\pi} f(r, \theta) r^{-\sigma} e^{-jvk \theta} d\theta \frac{dr}{r}
\]  
(22)

\[\forall (k, v) \in \mathbb{C} \setminus \mathbb{R}, j = \sqrt{-1}, \sigma > 0\]

3.5.2 The invariants analytical Fourier-Mellin transform
The invariant analytical Fourier-Mellin transform (IFMT) to translation, rotation and scale of an function \(f(r, \theta)\) is defined by:

\[
I_{fa}(k, v) = M_{fa}(0, 0, 0) \sigma^* e^{j k \arg (M_{fa}(1, 1, 0))} M_{fa}(k, v)
\]  
(23)

The AFMT can be written in a Cartesian coordinates as following:

\[
M_{fa}(k, v) = \frac{1}{2\pi} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)(x+fy)^{-\sigma} (x^2 + y^2)^{-1/2}
\]  
(24)

The NxM is the number of pixels of an image \(f(x,y)\).

4. RECOGNITION
4.1 Dynamic programming
The dynamic programming is a method which forms part of the problems linked to optimization and is frequently employed in order to find the shortest path (optimal path) from one point to another. In fact, the application of dynamic programming is based on the following phases:

**Phase 1.** Calculate the distance matrix \(d\) between the vector of test \(X_{test}\) of each one of the numeral vector \(X\) of learning base. This matrix of distance \(d\) is given by the following formula:

\[
d(i, j) = \left| X_{test}(i) - X(j) \right|
\]  
(25)

Where \(i, j = 1, 2, \ldots, n\)

Where \(n\) is the number of components of vectors \(X\).
\[ S(i, j) = d(i, j) + \min \left\{ S(i-1, j), S(i, j-1) \right\} \]  

(26)

Where:\n\[ S(i, j)\] is the cumulative distance along the optimal path between initial point (1,1) and point (i, j).

**Phase 2.** Calculate the optimal path from initial point (1,1) to a point \((i, j)\) by the recursive formula:

\[ P(X_{\text{test}} / C_i) = \prod_{s=1}^{k} P(x_{\text{test},s} / C_i) \]

(32)

Where \(E\) called evidence that is a scaling factor that depends only on \(X_{\text{test}}\).

Finally the class \(X_{\text{test}}\) of is given by:

\[ \text{Class}(X_{\text{test}}) = \arg \max_{i=1,2, \ldots, N} (P(X_{\text{test}} / C_i)) \]  

(33)

### 4.2 Naïve Bayes classifier

The naïve Bayes classifier is a probabilistic classifier based on Bayes theorem:

\[ P(X / Y) = \frac{P(X \cap Y)}{P(Y)} \]  

(28)

Where \(X\) and \(Y\) are two random variables and \(P(X)\) is the probability of \(C\). In fact, naïve name means a hypothesis says that all the attributes of the random variable are independent between them.

Then first being a set of classes \(C_i, i = 1,2, \ldots, N\) each of them contains a set of vectors \(X_{C_i,j}, j = 1,2, \ldots, M\) With \(X_{C_i,j} = (x_{C_i,1}, x_{C_i,2}, \ldots, x_{C_i,k})\)

The mathematical es esperance of each attribute \(X_{C_i,s}\) of each class \(C_i\) is given by:

\[ \mu_{x_{C_i,s}} = \frac{1}{M} \sum_{j=1}^{M} x_{C_i,j} \]  

(29)

\[ s = 1, 2, \ldots, k, \ i = 1, 2, \ldots, N \]  

(30)

When its variance is given by:

\[ \sigma_{x_{C_i,s}}^2 = \frac{1}{(M-1)} \sum_{j=1}^{M} (x_{C_i,j} - \mu_{x_{C_i,s}})^2 \]  

(31)

Or use later as the parameters of the probability of an unknown vector or vector of test \(X_{\text{test}} = (x_{\text{test},1}, x_{\text{test},2}, \ldots, x_{\text{test},k})\) knowing that a class \(C_i, i = 1, 2, \ldots, N\) is given by:

\[ f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b \]  

(34)

**4.3 Support vectors machine**

The SVM[24] is a powerful statistic tool used in many scientific fields as data mining applications such as text categorization, handwritten character recognition, image classification and bioinformatics. Support vector machines (SVM) are one of the most important, and powerful statistic tools used efficiently in many scientific fields as a pattern recognition, because it support high dimensional data and at the same time, providing good generalization properties. For a two-class classification problem, assume that we have a series of input vectors \(x \in \mathbb{R}^n\) with corresponding labels \(y \in \{-1, 1\}\) for \(i = 1, 2, \ldots, N\) where +1 and −1 indicate the two classes. The idea of SVM is to map the input vectors \(x, \in \mathbb{R}^n\) into a high dimensional feature space \(\Phi(x) \in \mathbb{H}\), and it constructs an optimal separating hyperplane \(H\) that will maximizes the marginal distance between the hyperplane and the nearest data points of each class in the space \(H\).
The dual variables \( \alpha_i \) intervening in the Lagrangian is called Lagrange multipliers.

To maximize \( D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \) \hspace{1cm} (35)

Subject to \( \sum_{i=1}^{n} \alpha_i y_i = 0 \), \( 0 \leq \alpha_i \leq C \) \hspace{1cm} \forall i = 1, 2, ..., n \hspace{1cm} (36)

The parameter \( C \) which appears here is a positive constant fixed in advance; it’s called the constant of penalty.

Some examples of kernel functions:

Table 1: Example of different kernel functions used in nonlinear SVM

<table>
<thead>
<tr>
<th>Kernel type</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>( (x^T y + b)^{\alpha} )</td>
</tr>
<tr>
<td>Gaussian radial basis function (GRBF)</td>
<td>( \frac{1}{\gamma} \sum_{i} y_i e^{-\frac{x_i^T x}{2\gamma^2}} )</td>
</tr>
</tbody>
</table>

The method described above is designed for a problem of two classes only; many studies treat a generalization of the SVM to N classes. Among these studies, we have used in this work the strategy of one against all that is based to use N decision functions allowing to make a discrimination of a class bearing a label equal to 1 and containing a one vector against all other vectors included in a other class opposite that is labelled by the value -1. In the classification phase, we calculate the value image of an unknown vector \( X \) (test numeral) by all N decision functions that are obtained in the learning phase. The recognition will be attributed to the numeral that the decision function separates its class to another class containing the rest of numerals which gives the biggest value.

5. EXPERIMENTS AND RESULTS

Firstly, we present an image of some isolated handwritten Roman numerals.

![Fig 5: Some isolated handwritten Roman numerals](image)

The desired goal is to compare between the performances in terms of recognition rate (precision) and recognition time (rapidity) of:

- The three hybrid methods of features extraction that are Zoning + KIM (Z+ KIM), Zoning + PZIM(Z+ PZIM), Zoning + IAFMT (Z+ IAFMT).
- Between three different classifiers which are dynamic programming(PD), Naïve Bayes(NB), Support Vectors Machine (SVM).

Therefore in order to achieve these comparisons, we have used the following data’s:

- Each numeral image has a size equal to 30x30 pixels.
- The number of all images of learning and of test that we have used is equal to 3000 images.

The number of zones whose each image numeral is divided is equal to 9 zones for having consequently that each numeral is converted to a vector has 9 components noted \( X_{\text{zoning}} \).

- The number of calculated values obtained by each descriptor is equal to 9 values to get therefore that each numeral is converted to a vector has 9 components noted \( X_{\text{descriptor}} \), therefore finally each numeral image is converted to a vector has 18 components obtained by the following relationship:

\[
X = X_{\text{zoning}} \cup X_{\text{descriptor}}
\]

- The parameters KIM are equal to \( p=0.85, q=0.55 \).
- The parameter of IAFMT is equal to \( \sigma = 1 \).
- The kernel function chosen in the SVM is the GRBF with a standard deviation equal to \( \gamma = 0.9 \).

Therefore, we grouped the values that we obtained of the recognition rate \( \tau_x \) of each numeral (given in %) and of the global rate \( \tau_g \) (given in %) also the global recognition time \( t_g \) (given in second) i.e. of all numerals for each hybrid method of features extraction and each classifier in the following table:
Table 2: The recognition rates and time for each hybrid method and for each classifier

<table>
<thead>
<tr>
<th>N</th>
<th>(\tau_n) (DP)</th>
<th>(\tau_n) (NB)</th>
<th>(\tau_n) (SVM)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Z+KIM</td>
<td>Z+PZIM</td>
<td>Z+IAFMT</td>
</tr>
<tr>
<td>I</td>
<td>82.45</td>
<td>81.10</td>
<td>79.75</td>
</tr>
<tr>
<td>II</td>
<td>80.15</td>
<td>78.34</td>
<td>77.25</td>
</tr>
<tr>
<td>III</td>
<td>78.55</td>
<td>76.00</td>
<td>74.20</td>
</tr>
<tr>
<td>IV</td>
<td>77.14</td>
<td>75.42</td>
<td>73.77</td>
</tr>
<tr>
<td>V</td>
<td>82.37</td>
<td>80.14</td>
<td>78.67</td>
</tr>
<tr>
<td>VI</td>
<td>81.25</td>
<td>79.65</td>
<td>77.25</td>
</tr>
<tr>
<td>VII</td>
<td>83.47</td>
<td>81.12</td>
<td>78.14</td>
</tr>
<tr>
<td>VIII</td>
<td>84.25</td>
<td>82.00</td>
<td>80.16</td>
</tr>
<tr>
<td>IX</td>
<td>84.17</td>
<td>81.25</td>
<td>80.34</td>
</tr>
<tr>
<td>X</td>
<td>83.67</td>
<td>81.34</td>
<td>79.55</td>
</tr>
<tr>
<td>(\tau_g)</td>
<td>81.75</td>
<td>79.64</td>
<td>77.91</td>
</tr>
<tr>
<td>(t_g)</td>
<td>25.55</td>
<td>17.34</td>
<td>12.67</td>
</tr>
</tbody>
</table>

The associated graphical representation to recognition rate \(\tau_n\) is:

Fig. 6: The recognition rate of each numeral for each features extraction hybrid method and for each classifier.

**Interpretation:**

Considering all the results that we obtained, we can to conclude that:

- The numerals the most correctly recognized are: I, II, VIII, IX.
- The numerals the less correctly recognized are: IV, X.
- The most precise hybrid method of features extraction is zoning combined with Krawtchouk invariant moment followed then with pseudo-Zernike invariant moment then with invariant analytical Fourier-Mellin transform.
- The most precise classifier is the support vectors machine then the naïve Bayes then dynamic programming.

Moreover the associated graphical representation to time of execution is presented as follow:
Taking into account all the results that we obtained, we can to conclude that the most precise recognition system is that which contains a zoning method combined with Krawtchouk invariant moment as a features extraction and the support vectors machine as a classifier.

The global time of execution is presented graphically as follow:

Fig. 7: The global recognition rate for each features extraction hybrid method and for each classifier.

Interpretation:

Taking into account all the results that we obtained, we can to conclude that the most precise recognition system is that which contains a zoning method combined with Krawtchouk invariant moment as a features extraction and the support vectors machine as a classifier.

The global time of execution is presented graphically as follow:

Fig. 7: The global time of execution for each features extraction hybrid method and for each classifier

Interpretation:

Having regard the results given in graphical representation above the most fast hybrid method is the zoning method combined with invariant analytical Fourier-Mellin transform then with pseudo-Zernike invariant moment then Krawtchouk invariant moment in one hand and the fatest classifier is dynamic programming followed by naive Bayes then finally the support vectors machine on the other hand. The most precise recognition system is that includes zoning method combined with Krawtchouk invariant moment and support vectors machine classifier but it is the slowest system, for fixing this idea, we note:

We note the difference of precision between \( h_{m1}, c_{l1} \) and \( h_{m2}, c_{l2} \)

\[
\Delta P = T_{s,h_{m1},c_{l1}} - T_{s,h_{m2},c_{l2}}
\]

(38)

- If \( \Delta P > 0 \) we will have a gain of precision, in this case \( \Delta P \) is called the rate of growth of precision.
- If \( \Delta P < 0 \) we will have a losing of precision, in this case \( \Delta P \) is called the rate of decay of precision.

In a like manner, we note the difference of rapidity between \( h_{m1}, c_{l1} \) and \( h_{m2}, c_{l2} \)

\[
\Delta R = T_{s,h_{m1},c_{l1}} - T_{s,h_{m2},c_{l2}}
\]

(39)

- If \( \Delta R > 0 \) we will have a advancement of rapidity. in this case \( \Delta R \) is called the rate of decay of rapidity.
- If \( \Delta R < 0 \) we will have a delay of rapidity. in this case \( \Delta R \) is called the rate of growth of rapidity.

Where \( h_{m} \) means to hybrid method and \( c_{l} \) to classifier.

Hence, the following table presents different values of \( \Delta P \) (gain of precision) and \( \Delta R \) (lateness of time of execution)
In this paper, we have presented both comparisons which the first one is between the performances in terms of precision and rapidity of three hybrid methods of features extraction which are the structural method zoning combined with three statistical methods that are Krawtchouk, then with pseudo Zernike invariant moments, then with analytical invariant Fourier-Mellin transform. While the second comparison is carried out between three classifiers that are the dynamic programming, the naive Bayes and the support vectors machines. For this purpose we have used in order to preprocess each numeral image the median filter, the thresholding, the centering and the edge detection techniques. We have concluded that the most precise but in the same time the most slow is the hybrid method zoning combined with Krawtchouk, invariant moment and the support vectors machine classifier.

Moreover, we will hope later on to introduce more other hybrid methods of features extraction and other classifiers in order to compare between their performances.

7. ACKNOWLEDGEMENT
In conclusion we are very grateful to our professors Mister Said Saﬁ and Mister Belaid Bouikhalene for their encouragement, their cooperation, their advices and their guidance in the realization of this work. Many thanks again to them.

8. REFERENCES


