A Generic Description of the Concept Lattices' Classifier: Application to Symbol Recognition

Stéphanie Guillas, Karell Bertet, and Jean-Marc Ogier

L3I, Université de La Rochelle, av M. Crépeau, 17042 La Rochelle Cedex 1, France
{sguillas, kbertet, jmogier}@univ-lr.fr

Abstract. In this paper, we present the problem of noisy images recognition and in particular the stage of primitives selection in a classification process. We suppose that segmentation and statistical features extraction on documentary images are realized. We describe precisely the use of concept lattice and compare it with a decision tree in a recognition process. From the experimental results, it appears that concept lattice is more adapted to the context of noisy images.

1 Introduction

The work presented in this paper tackles the problem of the automatic re-engineering of documents, and proposes a first theoretical approach concerning the use of concept lattices for automatic recognition of graphic objects, under the multi-scale and multi-orientation constraints.

In the field of invariant pattern recognition, there is a consensus about the fact that each stage of the recognition process is important [1, 2]. Furthermore, the review of the literature highlights several difficulties that existing techniques try to tackle more or less partially.

The first difficulty is the adaptation to the notion of context, aimed at trying to find some adequate recognition scenarios to a particular problem, if possible by integrating the capacity of evolution of the system. Another difficulty is related to the problem of combination of recognition schemas, by integrating structural and statistical description of the shapes, without any previous distortion of the recognition schema. At last, the problem concerning the selection of relevant primitives, adapted to a particular context, in adequation with evolutive systems stays an open problem, and is not made explicit. Literature is rich in terms of classification strategy. A lot of references indicate these problems, depending on different techniques : statistical [3] and/or structural approaches [4], parametric and non parametric approaches [5], connexionnist [6], training problems, primitives selection or fusion of classifiers problems [7], . . .

In this paper, we propose a first contribution concerning symbols recognition based on concept lattices. Indeed, lattices seem to bring some interesting answers to the previously discussed difficulties, thanks to their natural ability to integrate statistical and structural description, and to their capacity to validate some relevant primitives in regard with a particular context. Moreover, the concept

W. Liu and J. Lladós (Eds.): GREC 2005, LNCS 3926, pp. 47–60, 2006.
© Springer-Verlag Berlin Heidelberg 2006

lattice presents the advantage of a good readability and thus to be easy to understand. Shape recognition is classically realized in two stages : *learning* on symbols images which is the subject of the next part and *classification* of damaged symbols images presented in section 3.

2 Learning

The learning stage consists in organizing the information extracted from a set of objects by a concept lattice. In our case objects are images of symbols. These symbols are described by same-size numerical signature computed thanks to image processing techniques. Previously, it is necessary to have normalized data so that their representation is equivalent. More precisely, the learning stage can be described by:

Name: Learning
In: a *set of objects* O where each object $p \in O$ is a symbol described by a normalized signature $p = (p_1, \dots p_n)$ and a label of class $c(p)$.
Out: a *concept lattice* $(\beta(C), \leq)$ described by a set of concepts $\beta(C)$ and a relation $\leq$ between its concepts.

The learning involves two stages as shown in Figure 1:

- the *discretization* of signatures: the data are assigned to disjoined intervals. It is possible to find again the initial data by the union of these intervals. Discretization is essential to build the concept lattice. It is parameterized by a *cutting criterion* necessary to the construction of the intervals.
- the *building of concept lattice* from discretized data. This stage does not need any parameter.

2.1 Discretization

The discretization stage consists in organizing the numerical data of the different objects in discrete intervals to obtain a specifical characterization of each class of objects.

Name: Discretization
In: a *set of objects* O where each object (symbol) $p \in O$ is described by a signature which is a *numerical vector* $p = (p_1, \dots p_n)$ where each value is normalized and a *label of class* $c(p)$.
Out:
- the *intervals* organized in sets of intervals $I = I_1 \times I_2 \times \dots \times I_m$ where the intervals of each set I_i are disjoined, and cover the values p_i of the whole objects $p \in O$.
- a *membership relation* R which is defined for each object $p \in O$ and each interval $x \in I$ by: $pRx \Leftrightarrow$ it exists $i = 1 \dots m$ such as $p_i \in x \in I_i$

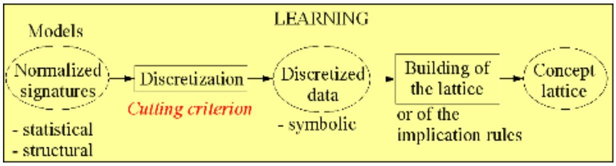

Fig. 1. Schematic description of learning

Description. The discretization is realized on the signatures organized in a table of data (fig. 1). At the beginning, we build for each feature i an interval $x \in I_i$ by gathering the whole values p_i taken by the symbols $p \in O$. Thus, we can initialize the membership relation R which is deduced. After this initialization stage, each set I_i contains one interval, and each symbol $p \in O$ is in relation with each interval $x \in I_i$. Then, we have to select an interval x to cut, and a cutting point in this interval x. To do that, we introduce the following notations:

- For a symbol $p \in O$, we define the set I_p of the intervals in membership relation with p: $I_p = \{x \in I \text{ such as } pRx\}$.
- For an interval $x \in I$, we define the set V_x of the numerical values of the symbols in which it is in relation and sorted by ascending order : $V_x = (p_i \text{ such as } p_i \in x)$ sorted by ascending order so : $V_x = v_1 \leq v_2 \leq \ldots \leq v_n$

Thus we have to select an interval $x \in I$ among the whole set of intervals, and a value $v_j \in V_x$ among the wholes values, and then to cut the interval x in two intervals x' and x'' with $V'_x = v_1 \leq \ldots \leq v_j$ and $V_{x''} = v_{j+1} \leq \ldots \leq v_n$. Each symbol will have a membership relation with one of these two created intervals, that enables to differentiate the two subsets of formed symbols. We repeat this process of cutting the intervals until we can distinguish each class. The selection of interval to cut depends on a *cutting criterion* that have to be defined.

When each class can be characterized by an own set of intervals, we obtain a discretized table involving the whole symbols $p \in O$ and the whole intervals $I = I_1 \times I_2 \times \ldots \times I_m$ where I_i is the set of intervals obtained for each feature $i = 1 \ldots m$. Notice that if a feature k has never been selected to be discretized, it contains only one interval ($|I_k| = 1$) which is in relation with the whole symbols. This feature is not discriminative, and thus can be removed from the discretized table. From this table, it is possible to deduce the membership relation R, and consequently, for an symbol $p = (p_1, p_2, \ldots, p_m) \in O$ where p_i is the value for the feature $i = 1 \ldots m$, to know the set I_p of intervals associated to p.

Example 1. *Table 1 (left) shows normalized data of 10 symbols distributed in 4 classes. The signature characterizing each symbol is composed of 3 features (a, b and c). After discretization by the entropy criterion, we obtain Table 1 (right). Each feature has been selected and cut one time, they consequently are kept.*

Table 1. Signatures of the 10 symbols (left) and discretized table with the entropy criterion (right)

Class	Ident.	Signature		
		a [0-20]	b [0-20]	c [0-20]
1	1	1	4	15
	2	0	0	18
2	3	1	12	13
	4	0	16	15
	5	3	12	11
3	6	8	16	15
	7	6	20	20
	8	15	12	15
4	9	18	4	0
	10	20	12	2

Class	Ident.	Intervals					
		a1 [0-3]	a2 [6-20]	b1 [0-4]	b2 [12-20]	c1 [0-2]	c2 [11-20]
1	1	X		X			X
	2	X		X			X
2	3	X			X		X
	4	X			X		X
	5	X			X		X
3	6		X		X		X
	7		X		X		X
	8		X		X		X
4	9		X	X		X	
	10		X		X	X	

Cutting Criterion. A large number of criteria allows to select the interval in order to divide and to determine the cutting point, and the choice of this parameter is decisive in the learning process. It is necessary to search an interval $x \in I$, with the values $V_x = (v_1 \dots v_n)$ sorted by ascending order, that *maximizes* a criterion, for a given value v_j. The interval will be cut between the values v_j and v_{j+1}. We can define a lot of cutting criteria depending or not on the data. Among these criteria, we mention the maximal distance, the entropy and the Hotelling's coefficient:

Maximal distance: $distance(v_j) = v_{j-1} - v_j$

Entropy: $gain_E(v_j) = E(V_x) - (\frac{j}{n}E(v_1 \dots v_j) + \frac{n-j}{n}E(v_{j+1} \dots v_n))$
with $E(V) = -\sum_{k=1}^{|c(V)|} \frac{n_k}{n} log_2(\frac{n_k}{n})$ the measure of entropy of an interval with n values where n_k is the number of symbols of the class k of the interval.

Hotelling's coefficient:
$gain_H(v_j) = H(V_x) - (\frac{j}{n}H(v_1 \dots v_j) + \frac{n-j}{n}H(v_{j+1} \dots v_n))$
with $H(V) = \frac{VarB(V)}{VarW(V)}$ the Hotelling's measure of an interval V of n values, with n_k the number of symbols of the class k, g_k the gravity center of the class k, g the gravity center of V, v_{k_i} the i-th element of the class k, $VarB(V) = \frac{1}{n}\sum_{k=1}^{|c(V)|} n_k(g_k - g)^2$ the measure of between class variance and $VarW(V) = \frac{1}{n}\sum_{k=1}^{|c(V)|} n_k(\sum_{i=1}^{n_k}(v_{k_i} - g_k)^2)$ the measure of within class variance.

Maximal distance method consists in searching the primitive which has the maximal gap between two consecutive values when values are in ascending order. Entropy function is a measure characterizing the degree of mixture of the classes. Hotelling's coefficient takes in consideration the maximization of the distance between classes and the minimization of the dispersion of each class.

Extensions. Note the possibility to integrate symbolic data with the numeric one. This integration consists in computing an extension of the membership relation R which can be done after discretization, to add symbolic data to the building of the lattice. This extension can also be done during the initialization of this relation

R, before the discretization. Thus it is useful to refine the cutting criterion. But it is only interesting when the cutting criterion takes into consideration the indication of class of the symbols, as the entropy criterion.

For some criteria, as the maximal distance one, we can cut the intervals after the stage that enables the characterization of each class. Indeed, instead of stopping when the table is discretized (for a number of discretization stages equals to t_1), we pursue the discretization n times to obtain more intervals for characterizing each class. The discretized table to t_n has got more intervals than the one to t_1, but it enables to refine the description of each class.

2.2 Construction of the Concept Lattice

After the discretization stage comes the building of the concept lattice. This stage is totally determined by the obtained membership relation R. There is no criterion or parameter to be considered for the construction of this graph because it represents the whole possible combinations of relation R between objects and intervals.

Name: Building of the concept lattice
In: a *membership relation* R between a set of objects O and a set of intervals I.
Out: a *concept lattice* $(\beta(C), \leq)$ described by a set of concepts $\beta(C)$ and a relation $\leq$ between its concepts.

Description. Concept lattice has first been studied from a theoretical point of view [8] before being developed in [9] to represent data in *formal concept analysis.* A *c*oncept lattice is defined from data organized by a discretized table. More formally, a *concept lattice* is defined as a set of *concepts* ordered by inclusion.

We associate to a set of symbols $A \subseteq O$, the set $f(A)$ of intervals in relation with the symbols of A: $f(A) = \bigcap_{p \in A} I_p = \{x \in I \mid pRx \ \forall \ p \in A\}$ Dually, we associate to a set of intervals $B \subseteq M$, a set $g(B)$ of symbols in relation with the intervals of B: $g(B) = \{p \in O \mid pRx \ \forall \ x \in B\}$ These two functions f and g defined between symbols and intervals establish a *connection of Galois.* Moreover, $g \circ f$ and $f \circ g$ verify the properties of a closure operator. We note $\varphi = g \circ f$ the closure on the set I.

A *formal concept* is a pair symbols-intervals in relation according to R. More formally, a formal concept is a pair (A, B) with $A \subseteq O$, $B \subseteq I$, $f(A) = B$ and $g(B) = A$. The *concept lattice* associated to the relation R is a pair $(\beta(C), \leq)$ where:

- $\beta(C)$ is the set of the whole concepts of C.
- $\leq$ is an order relation on $\beta(C)$ defined for two concepts of $\beta(C)$, (A_1, B_1) and (A_2, B_2) by: $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow \left\| \begin{array}{l} A_2 \subseteq A_1 \\ \text{(equivalent to } B_1 \subseteq B_2) \end{array} \right.$

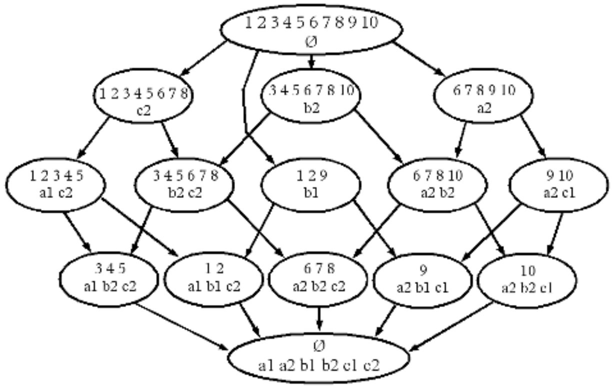

Fig. 2. Concept lattice

The relation $\leq$ is an order relation [1], thus it can be associated to a cover relation noted $\prec$. $(\beta(C), \prec)$ is then the Hasse diagram [2] of the concept lattice $(\beta(C), \leq)$.

The minimal concept to the sense of the relation R contains the whole symbols O. It is the concept $(O, f(O))$. The set $f(O)$, generally empty, corresponds to the intervals shared by the whole symbols. Dually, the maximal concept is $(g(I), I)$.

The representation of the concept lattice of the relation R is uniquely defined and the concepts corresponding to the relations symbols-intervals are ordered by inclusion. There are a lot of algorithms to generate the concept lattice : Bordat [10], Ganter [9], Godin et al. [11] and Nourine et Raynaud [12] which has the best theoretic complexity (quadratic complexity by elements of the produced lattice). To build the lattice, we have to set up the list of its whole concepts. The search of the concepts consists in finding in the discretized table the maximal rectangles, meaning the biggest sets of symbols and intervals in relation. After the generation of the whole concepts, it only remains to order them by inclusion (Figure 2).

Extension. The main limit of the use of concept lattice is its cost in time and space. Indeed, the size of the concept lattice is bounded by $2^{|S|}$ in the worst case, and by $|S|$ in the best case. Consequently the complexity is exponential in time and space in the worst case. It is very difficult to use studies of average complexity because the size of the lattice depends on the data. However notice that its size stays reasonable in practice as stated by the large number of experimentations which have been done.

To limit this exponential complexity, notice the possibility to generate only a *representation* of the lattice. An effective representation is defined by the fact that it is smaller, easily understandable, and that it determines the concept

[1] An order relation is a reflexive, symmetric and transitive relation.

[2] Representation of an order relation without its relations of reflexivity and transitivity.

lattice via efficient algorithms of generation. There are a large number of representations of a lattice proposed in the literature which verify these criteria (representation of a lattice by an order in lattice theory, by a table in formal concept analysis, by a conjunctive normal form in logic, by functional dependencies in databases). We mention the representation by a *system of implicational rules* [13, 14, 15] that we can find in data analysis. Such a representation enables, beyond its property of digest representation of the lattice, to avoid the complete generation of the lattice due to its possibility to use a *on line generation* of the only concepts which are necessary during the recognition stage. It also enables a description of the links between the features on another form, and highlights the links between features of type "The whole symbols which have the features x and y also have the feature z", that is formalized by the implicational rule $\{x, y\} \to z$ (or simply $x\, y \to z$).

3 Classification

After the learning stage and the generation of the concept lattice, it is the classification stage. The principle is to determine the class of new representations of the symbols, that is to say, to recognize the class of the symbols which can be more or less noised as shown in Figure 3.

Name: Classification
In:
- the signature $s = (s_1 \ldots s_n)$ of the symbol s to class
- the concept lattice $(\beta(C), \leq)$ comes from the learning stage

Out: a label of class $c(O)$ for s

3.1 Navigation Principle

The concept lattice can be used as a research area in which we can move depending on the validated features. The first step is the *minimal concept* $(O, f(O))$ meaning that each classes of the symbols are candidate to be recognized and any interval is validated. Then the progression to a next step in the concept

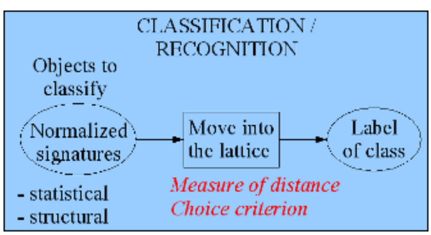

Fig. 3. Schematic description of the classification

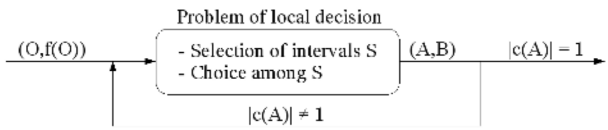

Fig. 4. Progression in the concept lattice

lattice corresponds to the validation of new intervals and consequently to the reduction of the set of symbols. The final step is the *final concept* where the remaining symbols, which are in relation with the whole intervals validated during the progression in the lattice, have the same label. Formally, from the minimal concept $(O, f(O))$, a local decision step is iterated until reaching a final concept (A, B) where $|c(A)| = 1$ (Fig. 4). In each local decision step, we progress in the graph from a *current concept* to one of its successors and a new set of intervals is validated bigger and bigger. It is necessary to define a criterion for the selection of the intervals in a local level. In practice, the progression is done in the Hasse diagram which is the transitive reduction of the concept lattice.

Description of an Elementary Stage of Classification. An elementary stage of classification consists in choosing some intervals in a subset S of intervals selected from the lattice. More precisely, S is deduced from the successors of the current concept in the lattice. Let (A, B) be the current concept, and $(A_1, B_1), \ldots,$ (A_n, B_n) be the n successors of (A, B) in the lattice. Then S is a family of intervals: $S = \bigcup_{i=1}^{n} B_i \backslash B = \{X_1, \ldots, X_n\}$

such that the following properties are satisfied:

- $X_i \bigcap X_j = \emptyset$, $\forall i, j \leq n, i \neq j$
- $|X_i \bigcap I_j| \leq 1$, $\forall i \leq n, \forall j \leq m$, meaning that X_i does not contain 2 intervals from a same feature j.

The computation of the family S of selected intervals is completely defined from the concept lattice. When S is computed, a subset X_i has to be chosen from S, thus the following choice problem: *Choosing* X_i *from* $S = \{X_1, \ldots, X_n\}$.

This choice is a main step of any elementary stage of classification, and therefore of the navigation principle in the lattice whose intend is to provide a class for the symbol s. However, this choice depends on data (and not only on the structure of the lattice as for the computation of S). More precisely, it depends on a *choice criterion* using a *distance measure* between s and an interval x.

Choice Criterion. Any choice from S is described using a *distance measure* according to data, and more precisely a distance between the i^{th} value s_i of the symbol s to be classified, and an interval $x \in I$. We abuse notation and denote such a distance $d(s, x)$ instead of $d(s_i, x)$, thus an extension to a set $X \subseteq I$ of intervals: $d(s, X) = \frac{1}{|X|} \sum_{x \in X} d(s, x)$

We can define many choice criteria depending on data, and sometimes equivalent. Let us propose some simple choice criteria, all of them use the distance d:

1. Choosing i such that $d(s, X_i)$ is minimal.
2. Choosing i such that $|X_i \bigcap I_k| = |\{x \in X_i \bigcap I_k\}|$ is maximal, where I_k is the set of the k first intervals of S sorted according to the distance $d(s, x)$.
3. Choosing i such that $|\{x \in X_i$ such that $d(s, x) < d_c\}|$ is maximal, with d_c a constant.

The second choice uses the principle of the k nearest neighbors [16]. Notice that the third choice is a particular and simplest case of the second choice.

Extensions. It is possible to evaluate the decision risk in each elementary stage of classification (i.e. the confidence degree in a decision) during the navigation in the lattice. For example, a confidence degree for the second choice criterion could be the rate $\frac{X_i \bigcap I_k}{k}$. Such a confidence degree represents another indicator useful for the decision-making. It can be used :

- to try another way of navigation in the lattice from a former explored concept which has given the second best result with the considered choice criterion.
- to compute a more complete signature of the symbol and to make again the whole process.

When we need to search a more accurate signature of the symbol, with new features, and to make again the whole process (discretization and construction of the lattice), it is possible to proceed in an incremental way, meaning without reconstructing the whole lattice, but by a simple addition of the new data.

4 Experimentation

Context. We would like to highlight the links between the decision tree and the concept lattice since both integrate the primitive selection and the classification stages at a time. The decision tree, as the concept lattice, requires a discretization stage and the use of a selection criterion for the feature. Thus it is possible to build the decision tree with the same discretized data. However its construction requires the use of a selection criterion of the feature to be tested at each node of the tree. As a matter of fact, it is possible to obtain a large number of trees with the same data by using different selection criteria. Thus the representation with a decision tree is not only as the one obtained for the lattice.

Figure 5 presents the decision tree (in bold) and the concept lattice associated to the data of the example. The decision tree is built according to a criterion of selection based on a measure of entropy. Notice that the size of the tree is more condensed than the one of the lattice. Indeed, its size is polynomial in the size of the data whereas the size of the concept lattice is exponential in the worst case. Moreover, it is important to notice that each node of the decision tree also is a node in the concept lattice, whatever the selection criterion used for the construction of the tree. Finally, the organization of the structure of the tree forms itself in the lattice, where the tree (in bold) is included in the lattice.

Using a decision tree or a concept lattice, the elementary stage of classification remains the same. Using a tree, the selected intervals S are also defined from

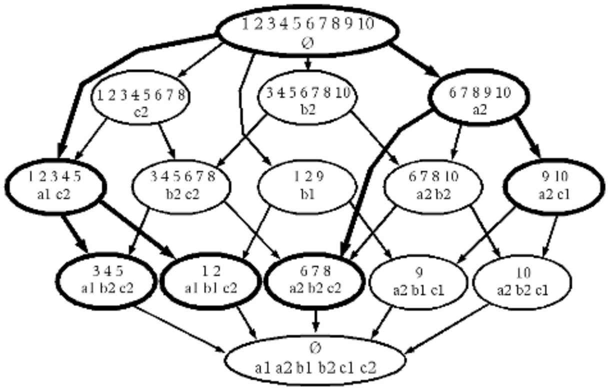

Fig. 5. Inclusion of the decision tree (in bold) in the concept lattice

the successors of a node, where each subset X_i of S is of cardinality 1, and the intervals of all the X_i's correspond exactly to intervals of a same feature j: $S = \{\{x\} : x \in I_j\}$. Therefore the navigation principle is the same with a decision tree and with a concept lattice, depending on a choice criterion defined according to a distance measure. We use the following cutting criteria, distance measure and choice criterion:

Cutting: *maximal distance, entropy or Hotelling's coefficient.*

Distance: $d(s,x) = \frac{\sqrt{(x_m - s)^2}}{\sqrt{(x_m - x_{min})^2}}$ where x_m is the middle of the interval x and x_{min} is the inferior boundary of the interval x. Note that we could replace x_{min} by x_{max} the superior boundary of the interval x and the formula will be the same. This distance is inferior or equal to 1 if the value of the symbol s is in the interval x, and superior to 1 if the value is out of the interval.

Choice: *Choosing i such that* $|\{x \in X_i \text{ such that } d(s,x) < 1\}|$ *is maximal. Then in case of multiple choice, choosing i such that* $|\{x \in X_i \text{ such that } d(s,x) < 1,1\}|$ *is maximal. Then in case of multiple choice, choosing i such that* $d(s, X_i)$ *is minimal.*

We compare the recognition rate using these two structures according to: the *signatures* and the *cutting criteria*. This experimentation has been performed with the intention to compare decision tree and concept lattice and not to obtain the best results in terms of recognition. We used symbols of GREC 2003 [17] and characterize them, by several signatures. For each model of symbol, we had 90 symbols noised by the Kanungo et al.'s method [18].

Evaluation of Signatures with the Hotelling's Coefficient Cutting Criterion. We first compare the 6 following signatures : 33 invariants of Fourier-Mellin [19], 50 invariants of Radon transform [20], 24 invariants of Zernike [21] and combination of these 3 signatures : 83 invariants of Fourier-Mellin and Radon combined, 57 invariants of Fourier-Mellin and Zernike combined and 74 invari-

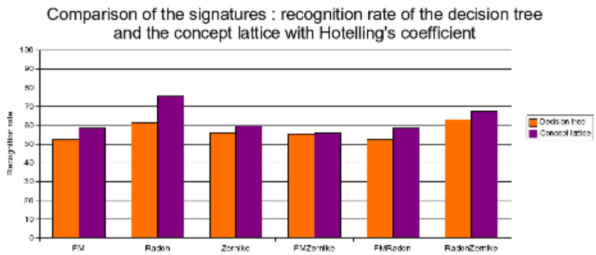

Fig. 6. Evaluation of the recognition with the Hotelling's coefficient

ants of Radon and Zernike combined. We compare these 6 signatures by comparing for each of them the size of the structure and the recognition rate. This experimentation has been realized using the Hotelling cutting criterion.Thus we made a comparison of the size of the structures obtained on the data according to the 6 different signatures. First, with the Hotelling's coefficient as cutting criterion, the number of discretization stages, the number of intervals and the size of the decision tree are almost the same with the whole signatures. However, the size of the concept lattice fluctuates and is smaller for the signature of Radon. Second, the size of the decision tree is really smaller to the one of the concept lattice. Figure 6 shows the recognition rate of the decision tree and the concept lattice according to each signature. The recognition rate is always better for the concept lattice than for the decision tree. Moreover, the Radon signature obtains the best rate of recognition for the concept lattice.

Evaluation of the Cutting Criteria with the Radon Signature. We made a comparison of the size of the structures obtained on this example of data according to the cutting criterion of the discretization stage, i.e. maximal distance, entropy and Hotelling's coefficient. We verify that the entropy and Hotelling's coefficient criteria need a lower number of discretization stages than the maximal distance criterion, because they consider the labels of class of the symbols. The concept lattice size is also smaller with the entropy and Hotelling's coefficient criteria, but it is not the case of the decision tree which has about the same number of nodes with both criteria. The comparative results of the lattice and the tree are shown in Figure 7. This comparative study of efficiency enables the constatation that with the both cutting criteria, the concept lattice improves the recognition results of the noised symbols in comparison with the decision tree. We can add that the best results are obtained with the maximal distance as cutting criterion but it is also the criterion which gives the biggest concept lattice. The Hotelling's coefficient criterion gives almost as good results as the maximal distance one but the size of the concept lattice is really smaller. So, this cutting criterion is the best compromise.

Comparison with Bayesian Classifier and k-NN Classifier. Figure 8 presents the recognition rates of 4 classification methods (Bayesian classifier, k-NN classifier with k=1 and k=3, decision tree and concept lattice) obtained on

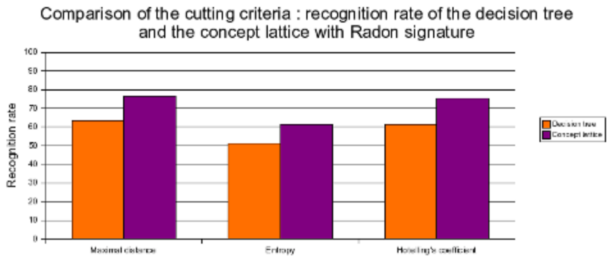

Fig. 7. Evaluation of the recognition with the Radon signature

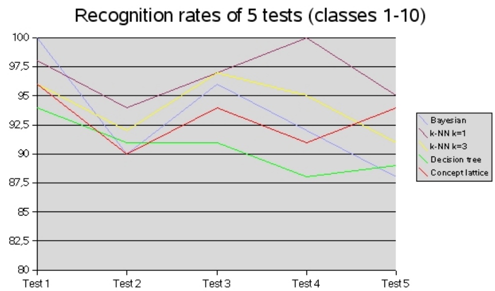

Fig. 8. Recognition rates of the methods obtained for 5 tests

Table 2. Mean recognition rates obtained with 5 tests of the 4 methods (left) and theoretical complexity of the 4 methods (right) with n the size of the learning set, w the number of classes, i the number of values of the signature selected by the cutting criterion, i' the size of the signature where $i \ll i'$, a the number of nodes in the tree and c the number of concepts in the concept lattice where $w \leq a \leq c \leq 2^w$

Mean of recognition rates (%)	*Classes 1-10*	*Classes 11-20*
Bayesian	93,2	94,6
k-NN k=1	96,8	98,4
k-NN k=3	94,2	98,4
Decision tree	90,6	89,8
Concept lattice	93	89,8

Theoretical complexity	*Learning*	*Classification*
Bayesian	O(wi^2)	O(wi^2)
k-NN		O(ni)
Decision tree	O(ni' + a)	O(wi)
Concept lattice	O(ni' + c)	O(wi)

two sets of noised symbols of 10 classes (namely classes 1-10 and classes 11-20), with 5 different learning sets computed from each set of data (namely Test 1 to Test 5). Each learning set is composed of 5 symbols per class : 1 model symbol; and 4 noised symbols randomly extracted from the set of noised symbols from which they are removed. Each symbol is described by its Radon signature, restricted to the features selected by the cutting criterion of Hotelling. These selected features are used in entry of each classifier. Notice that recognition rates really depend on the learning set whatever the method. Table 2 (left) shows the mean results of these 5 tests for classes 1-10. The k-NN classifier gives best results, then the Bayesian classifier, the concept lattice and the decision tree. Table 2 (right) presents a comparison of these methods in terms of complexity of the learning and the classification steps. Notice that best results are obtained by Bayesian and k-NN classifiers directly on numerical data (i.e. without discretization stage). The constraint of these methods are to make an hypothesis on the type of distribution of the data (gaussian, uniform...) for the bayesian classifier and to stock the whole data for the k-NN classifier. Concept lattice and decision tree give lower rates and need discretized data. However, their assets are a good readability, the taking into account of the linked between features in the construction of the graphs and the fact that they don't need hypothesis on the data.

5 Conclusion

The aim of this work is not to reach the best classification results for the moment, but to harmonize a quite un-explored strategy, based on concept lattices, with the well known decision tree method. The size of the decision tree, which is smaller than the concept lattice's one, permits to optimize the processing, but may produce some classification errors, due to the noise. The lattice approach proposes a higher number of classification sequences, and appears to be more adapted to the context of noisy images, to the detriment of a higher dimension. Its other advantage is its good readability. As for the decision tree, a non specialist can easily understand the principle of the progression in the graph by validating intervals. Our future experiments will refer to a comparative study concerning order structures and concept lattices for primitives selection, in the context of an increasing noise, to tackle robustness and scalability problems. Also, our current works deal with the use of concept lattices for a statistical-structural description of the data. Finally, it seems interesting to reduce the construction of the lattice cost, especially through the use of a non exponential but a canonical representation of a lattice, by using a rules system [13, 15], that would permit to generate the lattice on-line, that is to say, to generate the selection stages, if required.

References

1. S. Adam and al., "Multi-scaled and multi oriented character recognition : An original strategy," *ICDAR'99*, pp. 45–48, September 1999.
2. K. Tombre and B. Lamiroy, "Graphics recognition - from re-engineering to retrieval," *Proceedings of 7th ICDAR, Edinburgh (Scotland, UK)*, pp. 148–155, August 2003.
3. S. Canu and A. Smola, "Kernel methods and the exponential family," *National ICT*, 2005.
4. H. Bunke, "Recent developments in graph matching," *15th International Conference on Pattern Recognition*, vol. 2, pp. 117–124, 2000.

5. E. Lefevre, O. Colot, P. Vannoorenberghe, and D. De Brucq, "Contribution des mesures d'information la modlisation crdibiliste de connaissances," *Traitement du Signal*, vol. 17(2), 2000.
6. M. Milgram, *Reconnaissance des formes : mthodes numriques et connexionnistes.* Colin, A., 1993.
7. V. Gunes, M. Menard, and P. Loonis, "A multiple classifier system using ambiguity rejection for clustering-classification cooperation," *IJUFKS, World Scientific*, 2000.
8. G. Birkhoff, *Lattice theory*, vol. 25. American Mathematical Society, 3rd ed., 1967.
9. B. Ganter and R. Wille, *Formal concept analysis, Mathematical foundations.* Springer Verlag, Berlin, 1999.
10. J. Bordat, "Calcul pratique du treillis de Galois d'une correspondance," *Math. Sci. Hum.*, vol. 96, pp. 31–47, 1986.
11. R. Godin and H. Mili, "Building and maintening analysis-level class hierarchies using Galois lattices," *OOPSLA*, pp. 394–410, 1993.
12. L. Nourine and O. Raynaud, "A fast algorithm for building lattices," in *Third International Conference on Orders, Algorithms and Applications*, (Montpellier, France), august 1999.
13. K. Bertet and M. Nebut, "Efficient algorithms on the family associated to an implicationnal system," *DMTCS*, vol. 6, no. 2, pp. 315–338, 2004.
14. R. Taouil and Y. Bastide, "Computing proper implications," *Proceedings of ICCS-2001 Internationnal Workshop on Concept Lattices-Based Theory, Methods and tools for Knowledge Discovery in Databases*, pp. 290–303, 2001.
15. S. Obiedkov and V. Duquenne, "Incremental construction of the canonical implication basis," in *Fourth International Conference Journée de l'Informatique messine*, pp. 15–23, 2003. submitted to Discrete Applied Mathematics.
16. T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.
17. GREC, "www.cvc.uab.es/grec2003/symreccontest/index.htm."
18. T. Kanungo and al., "Document degradation models: parameter estimation and model validation," in *IAPR Workshop on machine vision applications, Kawasaki (Japan)*, pp. 552–557, 1994.
19. S. Derrode, M. Daoudi, and F. Ghorbel, "Invariant content-based image retrieval using a complete set of fourier-mellin descriptors," *Int. Conf. on Multimedia Computing and Systems (ICMCS'99)*, pp. 877–881, june 1999.
20. S. Tabbone and L. Wendling, "Recherche d'images par le contenu l'aide de la transforme de radon," *Technique et Science Informatiques*, 2003.
21. M. Teague, "Image analysis via the general theory of moments," *Journal of Optical Society of America (JOSA)*, vol. 70, pp. 920–930, 2003.