Fuzzy Intervals For Designing Structural Signature: An Application to Graphic Symbol Recognition

Muhammad Muzzamil Luqman[1,2], Mathieu Delalandre[1], Thierry Brouard[1], Jean-Yves Ramel[1] and Josep Lladós[2]

[1] Laboratoire d'Informatique, Université François Rabelais de Tours - France
muhammadmuzzamil.luqman@etu.univ-tours.fr
{mathieu.delalandre,brouard,ramel}@univ-tours.fr
[2] Computer Vision Center, Universitat Autònoma de Barcelona - Spain
josep@cvc.uab.es

Abstract. The motivation behind our work is to present a new methodology for symbol recognition. The proposed method employs a structural approach for representing visual associations in symbols and a statistical classifier for recognition. We vectorize a graphic symbol, encode its topological and geometrical information by an attributed relational graph and compute a signature from this structural graph. We have addressed the sensitivity of structural representations to noise, by using data adapted fuzzy intervals. The joint probability distribution of signatures is encoded by a Bayesian network, which serves as a mechanism for pruning irrelevant features and choosing a subset of interesting features from *structural signatures of underlying symbol set*. The Bayesian network is deployed in a supervised learning scenario for recognizing query symbols. The method has been evaluated for robustness against degradations & deformations on pre-segmented 2D linear architectural & electronic symbols from GREC databases, and for its recognition abilities on symbols with context noise *i.e. cropped symbols*.

Key words: symbol recognition, overlapping fuzzy interval, structural signature, Bayesian network.

1 Introduction

Graphics recognition deals with graphic entities in document images and is a subfield of document image analysis. These graphic entities could correspond to symbols, mathematical formulas, musical scores, silhouettes, logos etc., depending on the application domain. Llados & Sanchez [1] have very correctly pointed out that the documents from electronics, engineering, music, architecture and various other fields use domain-dependent graphic notations which are based on particular alphabets of symbols. These industries have a rich heritage of hand-drawn documents and because of high demands of application domains, overtime symbol recognition is becoming core goal of automatic image analysis and understanding systems. Hand-drawn based user interfaces, backward conversion from raster images to CAD, content based retrieval from graphic document databases and browsing of graphic documents are some of the typical applications of symbol recognition. Detailed discussion on the application domains of symbol recognition has been provided by Chhabra [2] and Llados et al. [3].

The research surveys by Chhabra [2], Llados et al. [3], Cordella & Vento [4] and Tombre et al. [5] provide a detailed and state of the art historical review of work done in the field of symbol recognition over last two decades. Graphic symbol recognition is generally approached by syntactic, structural, statistical or hybrid methods of pattern recognition. Syntactic approaches involve the use of grammars and syntactical parsing [6] and are usually considered as a special case of structural approaches [3]. Structural and statistical approaches are normally differentiated by the data structures that they employ for pattern representation. Structural approaches use symbolic data structures such as strings, trees and graphs, whereas, the statistical approaches are characterized by the use of feature vectors for representing patterns [6].

Cordella & Vento [4] have provided a detailed listing of methods employing different structural, statistical or hybrid approaches for graphic symbol recognition. The symbolic data structures are very powerful in their representational capabilities. However, the structural approaches lack in the availability of efficient tools for matching and comparison [6]. On the other hand, the use of feature vectors by many statistical approaches limits their representational capabilities but the availability of a much richer repository of mathematical tools in statistical domain [6] and the associated computational advantages, makes them a more favorable choice in certain cases. The use of light weight feature vectors and computationally powerful statistical classifiers allows to design fast and efficient systems which are sufficiently scalable and domain independent.

Several research works have been undertaken to combine structural and statistical approaches, with the aim to utilize their strengths and avoid the weaknesses. Delalandre et al. [7] employ a statistical technique for extracting the components *(that compose the symbol)* and the loops formed by these components. Afterwards, they construct graphs from these loops and deploy an inexact graph matching algorithm for recognition. Hse et al. [8] have used Zernike descriptor with various statistical classifiers for sketched symbol recognition. Barrat et al. [9] have used various shape descriptors with naive Bayes classifier for symbol recognition. Among the hybrid approaches, specially over last decade, the vectorial signatures *(also referred as structural signatures)* (2.1) have gained considerable attention [10]. The vectorial signatures encode the geometric and topologic relations between elementary vectorial primitives. Many recognition and spotting systems have been developed around these signatures [11,12,13,14,15,16].

The rest of paper is organized as follows: §2 gives a general description of our method, §3 is devoted to detailed description of each part of the proposed system, experimental results are presented in §4 and the paper is concluded with some future directions of work in §5.

2 The proposed method and related works

In this section we outline our proposed method and highlight its placement with past works *(that employ a similar methodology)*. The method is a hybrid of structural and statistical pattern recognition approaches; we exploit representational power of structural approaches and employ computational efficiency of statistical classifiers.

2.1 Vectorial signatures *(or structural signatures)*

Ventura & Schettini [17] were the first to introduce the concept of vectorial signatures for symbol recognition, back in 1994. They extract thin and thick elementary structures *(in terms of geometric constraints between line primitives)* from symbol, describe them by local features and create a signature for symbol. Finally, they deploy a hypothesis-and-test paradigm for detecting occurrences of symbols in line drawings. A recent overview of past works using structural signatures is given by Rusinol [10]. These works include [11,12,13,14,15]. Coustaty et al. [15] have applied structural signature for symbol recognition. They extract segments from the symbol in image by Hough transform, describe their spatial organization by a topological graph and compute a structural signature. They have deployed a Galois Lattice as classifier and have shown the robustness of their method against high levels of degradation. Dosch et al. [14] *(originally proposed for symbol spotting)* and its improvement by Rusinol et al. [13] *(for both recognition and spotting)*, work on a vectorial representation. They extract spatial relationships between pairs of segments and hierarchically organize them into basic shapes. Their signature is comprised of the cardinalities of occurrences of spatial relations between segments in a shape. Zhang et al. [12] work on a vectorial representation, and use circle and arcs as well, in addition to line primitives. They define a structural signature in terms of relations between these primitives and employ a brute force comparison for recognizing a query signature. Qureshi et al. [11] vectorize a graphic symbol, construct its Attributed Relational Graph (ARG) and compute a structural signature for it (the G-Signature). For classification of query symbol they use nearest neighbors rule with Euclidean distance as measure of dissimilarity. Their G-Signature is discriminant in case of hand-drawn deformations and has been shown invariant of rotation and scaling.

These works show the invariance of structural features to transformations and illustrates their representational capabilities. However, we argue that the sensitivity of structural signatures to noise (degradations & deformations) limits these systems to be used for real-life applications, and to scale to large number of symbol models. In this work, we propose to take structural signatures to the domain of fuzzy sets, to enable them to cope with uncertainties, and extend our previous work [16], which in fact takes forward the work of [11]. We have selected Bayesian networks *for dealing with uncertainty in symbol signatures during learning and recognition phases*, and propose to use *(overlapping)* fuzzy intervals instead of rigid boundaries [16] for features in signature. Our motivation behind these choices are the previous works involving Bayesian framework [18] and fuzzy sets [19], that have shown the significance of these methodologies in improving robustness against uncertainties in data. We have increased the scalability capabilities of structural signatures by employing uncertainty-management during signature design, learning and classification phases. The signature is given in Fig.2 and it is discussed in §3.2.

2.2 Bayesian networks

Bayesian networks are probabilistic graphical models and are represented by their structure and parameters. Structure is given by a directed acyclic graph and it encodes the

dependency relationships between domain variables whereas parameters of the network are conditional probability distributions associated with *(each of)* its nodes. A Bayesian network, like other probabilistic graphical models, encodes the joint probability distribution of a set of random variables, and could be used to answer all possible inference queries on these variables. A humble introduction to Bayesian networks is in [20,21].

Bayesian networks have already been applied successfully to a large number of problems in machine learning and pattern recognition and are well known for their power and potential of making valid predictions under uncertain situations. But in our knowledge there are only a few methods which use Bayesian networks for graphic symbol recognition. Recently Barrat et al. [9] have used the naive Bayes classifier in a *'pure'* statistical manner for graphic symbol recognition. Their system uses three shape descriptors: Generic Fourier Descriptor, Zernike descriptor & R-Signature 1D, and applies dimensionality reduction for extracting the most relevant and discriminating features to formulate a feature vector. This reduces the length of their feature vector and eventually the number of variables *(nodes)* in Bayesian network. The naive Bayes classifier is a powerful Bayesian classifier but it assumes a strong independence relationship among attributes given the class variable. We believe that the power of Bayesian networks is not fully explored; as instead of using predefined dependency relationships, if we find dependencies between all variable pairs from underlying data we can obtain a more powerful Bayesian network classifier. This will also help to ignore irrelevant variables and exploit the variables that are interesting for discriminating symbols in underlying symbol set (§3.3 and §3.4).

2.3 Originality of our approach

Our method is an original adaptation of Bayesian network learning for the problem of graphic symbol recognition. *For symbol representation*, we use a structural signature. The signature is computed from the ARG of symbol and is composed of geometric & topologic characteristics of the structure of symbol. We use *(overlapping)* fuzzy intervals for computing noise sensitive features in signature. This increases the ability of our signature to resist against irregularities [19] that may be introduced in the shape of symbol by deformations & degradations. *For symbol recognition*, we employ a Bayesian network. This network is learned from underlying training data by using the quite recently proposed genetic algorithms by Delaplace et al. [22]. A query symbol is classified by using Bayesian probabilistic inference *(on encoded joint probability distribution)*. We have selected the features in signature very carefully to best suit them to linear graphic symbols and to restrict their number to minimum; as Bayesian network algorithms are known to perform better for a smaller number of nodes. Our structural signature makes the proposed system robust & independent of application domains and it could be used for all types of 2D linear graphic symbols. Also, relatively basic computations are involved for recognizing a query symbol which enables our system to respond in real time and it could be used, for instance, as a preprocessing step of a traditional symbol recognition method or for indexation & browsing of graphic documents.

3 Detailed description

In this section we describe the representation, description, learning and classification phases of our system. These phases have been outlined by Cordella & Vento [4] in their research survey on symbol recognition. The authors have remarked that *almost* all graphics recognition systems could be looked upon as operating in representation, description and classification phases.

3.1 Representation phase

This important & basic phase of our system concerns the formation of an Attributed Relational Graph (ARG) data structure, *as proposed by Qureshi et al. [11]*, and is summarized in Fig.1. The topological and geometric details about structure of symbol are extracted and are represented by an ARG. In first step, the symbol is vectorized and is represented by a set of primitives *(labels 1, 2, 3, 4 in Fig.1)*. In next step, these primitives become nodes and topological relations between them become arcs in ARG. Nodes have *'relative length' (normalized between 0 and 1)* and *'primitive-type' (Vector for filled regions of shape and Quadrilateral for thin regions)* as attributes; whereas arcs of the graph have *'connection-type' (L, X, T, P, S)* and *'relative angle' (normalized between 0° and 90°)* as attributes.

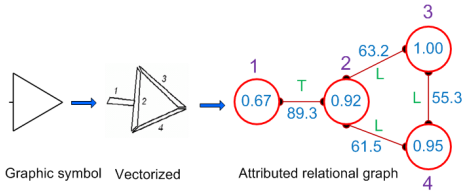


Fig. 1. The representation phase.

3.2 Description phase *(fuzziness of signature)*

This phase concerns the extraction of features and computation of structural signature, from ARG of an underlying symbol. In order to increase the robustness of our signature and to enable it to resist the irregularities & uncertainties introduced in shape of symbol as result of noise *(degradations & deformations)*, we introduce *(overlapping)* fuzzy intervals to our previous work [16]. Fig.2 presents our proposed structural signature for a symbol. Our motivation behind choosing structural features is to exploit their ability to identify symbols in context [13].

Group-1 & Group-2 *(of features in structural signature)* encode the size of symbol and arrangement of its primitive components, respectively. These features discriminate between symbols of different sizes and also between symbols of same size but with a different arrangement of primitives. Group-3 encodes the density of connections for nodes. This group discriminates between symbols that have similar number of primitives with a similar arrangement but different density of connections at nodes. Group-4 & Group-5 exploits the attributes of primitives and encodes details of length & angle

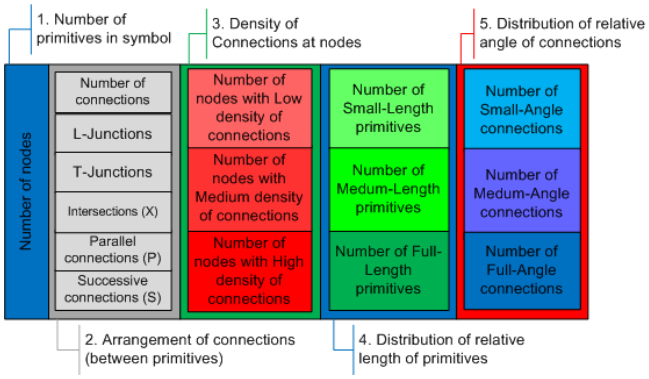


Fig. 2. Structural signature for graphic symbol.

attributes. These groups complement the criteria (of Groups-1, Group-2 & Group-3) for outlining boundaries, between symbol classes, in feature space.

The computation of features in Group-1 & Group-2 is straightforward and is achieved by counting the relevant information in ARG of graphic symbol. For features in Group-3, we first compute a list of connection-density counts of all nodes of all ARG *of symbols in underlying symbol set*. And then use this list of connection-density counts for finding connection-density intervals for computing feature in Group-3 of structural signature. We use a histogram based binning technique from [23] for this purpose. The technique is originally proposed for discretization of continuous data and is based on use of Akaike Information Criterion (AIC) [24]. It starts with an initial m-bin histogram of data and finds optimal number of bins for underlying data. Two adjacent bins are merged by using an AIC-based cost function as criterion; until the difference between AIC-before-merge and AIC-after-merge becomes negative. We arrange these bins in overlapping fashion (fuzzy approach) and use them as intervals for computing number of nodes lying in different connection-density intervals. This gives us a distribution of nodes in structural ARG with low, medium and high density of connections, which we use as features of our signature.

Group-4 (and Group-5) is computed by dividing relative length (and relative angle) in three overlapping intervals, as shown in Fig.3 (and Fig.4). The overlapping intervals (fuzzy approach) handle the irregularities caused by distortions and degradations, and ensure that these irregularities do not affect the signature.

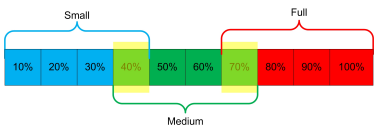


Fig. 3. Intervals for computing number of small, medium and full length primitives.

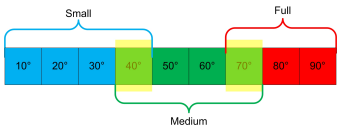


Fig. 4. Intervals for computing number of small, medium and full angle connections.

3.3 Learning phase

After representing the symbols in learning set by ARG and describing them by structural signatures, we proceed to learning of a Bayesian network. The signatures are first discretized [23]. We discretize each feature variable (of signature) separately and independently of others. The class labels are chosen intelligently in order to avoid the need of any discretization for them. The discretization of *'number of nodes'* and *'number of arcs'* achieves a comparison of similarity of symbols (instead of strict comparison of exact feature values). This discretization step also ensures that the features in signature of query symbol will look for symbols whose number of nodes and arcs lie in same intervals as that of the query symbol.

The Bayesian network is learned in two steps. First we learn the structure of the network by genetic algorithms proposed by Delaplace et al. [22]. These are evolutionary algorithms, but in our case they have provided stable results *(for a given dataset multiple invocations always returned identical network structures)*. Each feature in signature becomes a node of network. The goal of structure learning stage is to find the best network structure from underlying data which contains all possible dependency relationships between all variable pairs. The structure of the learned network depicts the dependency relationships between different features in signature. Fig.5 shows one of the learned structures from our experiments. The second step is learning of parameters of network; which are conditional probability distributions $\Pr(node_i|parents_i)$ associated to nodes of the network and which quantify the dependency relationships between nodes. The network parameters are obtained by maximum likelihood estimation (MLE); which is a robust parameter estimation technique and assigns the most likely parameter values to best describe a given distribution of data. We avoid null probabilities by using Dirichlet priors with MLE. The learned Bayesian network encodes joint probability distribution of the symbol signatures.

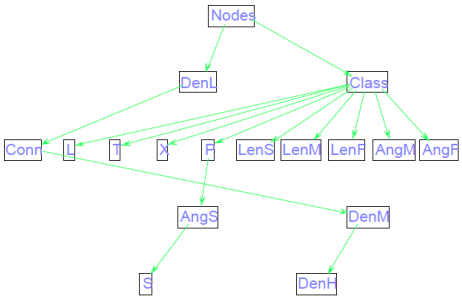


Fig. 5. A Bayesian network structure after learning; each node corresponds to a feature variable.

The conditional independence property of Bayesian networks helps us to ignore irrelevant features *in structural signature* for an underlying symbol set. This property states that a node is conditionally independent of its non-descendants given its immediate parents [20]. Conditional independence of a node in Bayesian network is fully exploited during probabilistic inference (see §3.4) and thus helps us to ignore irrele-

vant features for an underlying symbol set while computing posterior probabilities for different symbol classes (see §3.4).

3.4 Classification phase *(graphic symbol recognition)*

For recognizing a query symbol we use Bayesian probabilistic inference on the encoded joint probability distribution. This is achieved by using junction tree inference engine which is the most popular exact inference engine for Bayesian probabilistic inference and is implemented in [23]. The inference engine propagates the evidence *(signature of query symbol)* in network and computes posterior probability for each symbol class. Equation 1 gives Bayes rule for our system. It states that posterior probability or probability of a symbol class c_i given a query signature 'evidence e' is computed from likelihood (probability of e given c_i), prior probability of c_i and marginal likelihood (prior probability of e). The marginal likelihood *(Equation 3)* is to normalize the posterior probability; it ensures that the probabilities fall between 0 and 1.

$$\Pr(c_i|e) = \frac{\Pr(e, c_i)}{\Pr(e)} = \frac{\Pr(e|c_i) \times \Pr(c_i)}{\Pr(e)} \tag{1}$$

where,

$$e = f1, f2, f3, ..., f16 \tag{2}$$

$$\Pr(e) = \sum_{i=1}^{k} \Pr(e, c_i) = \sum_{i=1}^{k} \Pr(e|c_i) \times \Pr(c_i) \tag{3}$$

The posterior probabilities are computed for all ' k' symbol classes and the query symbol is then assigned to class which maximizes the posterior probability i.e. which has highest posterior probability for the given query symbol.

4 Experimentation

The organization of four international symbol recognition contests over last decade [25,26,27,28], has provided our community an important test bed for evaluation of methods over a standard dataset. These contests were organized to evaluate and test the symbol recognition methods for their scalability and robustness against binary degradation and vectorial deformations. The contests were run on pre-segmented linear symbols from architectural and electronic drawings, as these symbols are representative of a wide range of shapes [26]. GREC2005 [27] & GREC2007 [28] databases are composed of the same set of models, whereas GREC2003 [26] database is a subset of GREC2005.

4.1 Symbols with vectorial and binary noise

We experimented with synthetically generated 2D symbols of models collected from database of GREC2005 [27,31]. In order to get a true picture of the performance of our

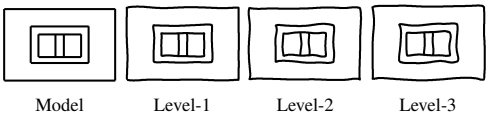


Fig. 6. Model symbol with deformations; used for simulating hand-drawn symbols and applied using an application from project Epeires [27].

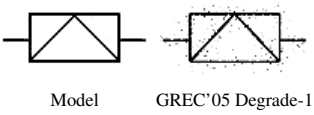


Fig. 7. Model symbol with degraded example; used to simulate photocopying / printing / scanning and applied using ImageMagick [29] & QGar package [30].

proposed method on this database, we have experimented with 20, 50, 75, 100, 125 & 150 symbol classes. We generated our own learning & test sets *(based on deformations & degradations of GREC2005)* for our experiments. For each class the perfect symbol *(the model)* along with its 36 rotated and 12 scaled examples was used for learning; as the features have already been shown invariant to scaling & rotation [11,16] and because of the fact that generally Bayesian network learning algorithms perform better on datasets with large number of examples. The system has been tested for its scalability on clean symbols (rotated & scaled), various levels of vectorial deformations and for binary degradations of GREC symbol recognition contest (Fig.6 and Fig.7). Each test dataset was composed of 10 query symbols for each class.

Table 1. Results of symbol recognition experiments.

Number of classes (models)		**20**	**50**	**75**	**100**	**125**	**150**
Clean symbols (rotated & scaled)		100%	100%	100%	100%	100%	99%
Hand-drawn deformation	Level-1	99%	96%	93%	92%	90%	89%
	Level-2	98%	95%	92%	90%	89%	87%
	Level-3	95%	77%	73%	70%	69%	67%
Binary degrade		98%	96%	93%	92%	89%	89%

Table 1 summarizes the experimental results. A 100% recognition rate for clean symbols illustrates the invariance of our method to rotation & scaling. Our method outperforms all GREC participants *(available results from GREC2003 [26] and GREC2005 [27] competetions)* in scalability tests and is comparable to contest participants for low levels of deformation & degradations. The recognition rates decrease with level of deformation and drop drastically for high binary degradations. This is an expected behaviour and is a result of the irregularities produced in symbol signature; which is a direct outcome of the noise sensitivity of vectorization step, as also pointed out by [3]. We used only clean symbols for learning and (thus) the recognition rates truely illustrate the robustness of our system against vectorial and binary noise. Fig.8 compares our results with [11] *(The system proposed in [11] presents recognition rates only for 20 models).*

4.2 Symbols with contextual noise

A second set of experimentation was performed on a synthetically generated corpus, of symbols cropped from complete documents [32]. These experiments focused on evaluating the robustness of the proposed system against context noise i.e. the structural

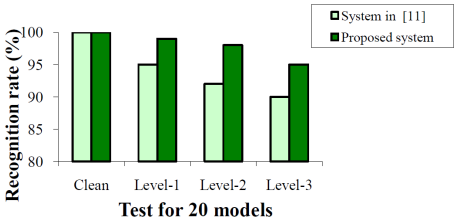


Fig. 8. Comparison of recognition rates.

noise introduced in symbols when they are cropped from documents. We believe that this type of noise gets very important when we are dealing with symbols in context in complete documents and to the best of our knowledge; no results have yet been published for this type of noise. We have performed these experiments on two subsets of symbols: consisting of 16 models from floor plans and 21 models from electronic diagrams. The models are derived from GREC2005 database [27,31] and are given in Fig.9 and Fig.10. For each class the perfect symbol (model), along with its 36 rotated and 12 scaled examples was used for learning. The examples of models, for learning, were generated using ImageMagick [29] and the test sets were generated synthetically [32] with different levels of context-noise (Fig.11) in order to simulate the cropping of symbols from documents. Test symbols were randomly rotated & scaled and multiple query symbols were included for each class. The test datasets are available at [33].

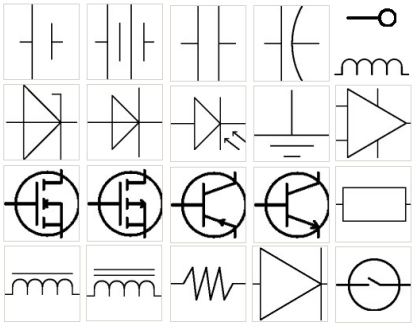


Fig. 9. Model symbols from electronic drawings.

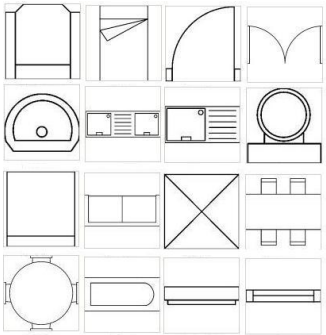


Fig. 10. Model symbols from floor plans.

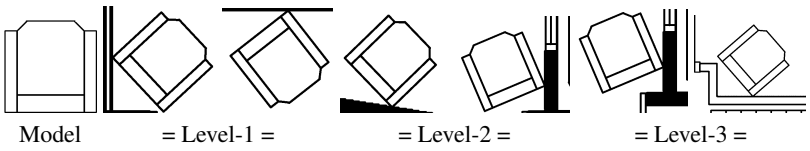


Fig. 11. An arm chair with different levels of context noise.

Table 2 summarizes the results of experiments for context noise. We have not used any sophisticated de-noising or pretreatment and our method derives its ability to resist against context noise, directly from underlying vectorization technique, the fuzzy

approach used for computing structural signature and the capabilities of Bayesian networks to cope with uncertainties. The models for electronic diagrams contain symbols consisting of complex arrangement of lines & arcs, which affects the features in structural signature as the employed vectorization technique is not able to cope with arcs & circles; as is depicted by the recognition rates for these symbols. But keeping in view the fact that we have used only clean symbols for learning and noisy symbols for testing, we believe that the results show the ability of our signature to exploit the sufficient structural details of symbols and it could be used to discriminate and recognize symbols with context noise.

Table 2. Results of symbol recognition experiments for context noise.

	Noise	**Model symbol (classes)**	**Query symbol (each class)**	Recognition rate (match with topmost result)	Recognition rate (a match in top-3 results)
Floor plans	Level-1	16	100	84%	95 %
	Level-2	16	100	79%	90 %
	Level-3	16	100	76%	87 %
Average recog. rate				80%	91%
Electronic diagrams	Level-1	21	100	69%	89%
	Level-2	21	100	66%	88%
	Level-3	21	100	61%	85%
Average recog. rate				65%	87%

5 Conclusion

Structural methods are the strongest methods for graphics representation and statistical classifiers provide efficient recognition techniques. By designing a mechanism to convert a structural representation to feature vector, the whole range of statistical tools *(classifiers)* are opened for that structural representation. First, we have presented an *overlapping* fuzzy interval based methodology to convert an ARG based representation of graphic symbol to a feature vector. Our signature exploits the structural details of symbols. And second, an original adaptation of Bayesian network learning for the problem of graphic symbol recognition, has been presented. We represent symbols by signatures and encode their joint probability distribution by a Bayesian network. We then use Bayesian probabilistic inference on this network to classify query symbols. Experimental results show an improvement in recognition rates and scalability of the old system.

Our system does not use any sophisticated de-noising or pretreatment and it drives its power to resist against deformations and degradations, directly from representation, description, learning and classification phases. We have addressed the issue of sensitivity of structural representations to noise and deformations; by introducing *overlapping* fuzzy intervals for computing structural signature. The features in signature are affected by the small quadrilaterals that are produced during vectorization (in case of noisy symbols), which produce irregularities in signature. The use of fuzzy approach for computing structural signature and probabilistic inference of Bayesian networks gives our system a certain level of resistance against these irregularities.

We believe that the recognition rates will be improved for real learning sets which include deformed and degraded examples as well. The system is extensible to new models. The signature

is invariant to rotation & scaling and robust against deformations & degradations. It is adapted to underlying symbol set and has a resistance against context noise. The proposed system has the capability to generate its learning set from models and could be used for 2D linear symbols from a wide range of application domains. The use of lightweight signature and statistical classifier makes our method efficient *(could be used for real time queries)* and scalable to a large number of symbol classes. In future we plan to use this method, as quick graphic symbol discrimination technique, for designing a system for symbol spotting and indexation of line drawing documents.

References

1. Llados, J., Sanchez, G.: Symbol recognition using graphs. In: ICIP. (2003) 49–52
2. Chhabra, A.K.: Graphic symbol recognition: An overview. LNCS **1389** (1998) 68–79
3. Llados, J., Valveny, E., Sanchez, G., Marti, E.: Symbol recognition: Current advances and perspectives. LNCS **2390** (2002) 104–128
4. Cordella, L.P., Vento, M.: Symbol recognition in documents: a collection of techniques? IJDAR **3**(2) (2000) 73–88
5. Tombre, K., Tabbone, S., Dosch, P.: Musings on symbol recognition. In: GREC. Volume 3926 of LNCS. (2006) 23–34
6. Bunke, H., Gunter, S., Jiang, X.: Towards bridging the gap between statistical and structural pattern recognition: Two new concepts in graph matching. In: ICAPR. Volume 2013 of LNCS. (2001) 1–11
7. Delalandre, M., Trupin, É., Ogier, J.M.: Symbols recognition system for graphic documents combining global structural approaches and using a XML representation of data. In Fred, A.L.N., Caelli, T., Duin, R.P.W., Campilho, A.C., de Ridder, D., eds.: SSPR/SPR. Volume 3138 of LNCS. (2004) 425–433
8. Hse, H., Newton, A.R.: Sketched symbol recognition using zernike moments. In: ICPR. (2004) 367–370
9. Barrat, S., Tabbone, S., Nourrissier, P.: A bayesian classifier for symbol recognition. In: GREC. Volume 7. (October 24 2007)
10. Rusinol, M.: Geometric and Structural-based Symbol Spotting. Application to Focused Retrieval in Graphic Document Collections. PhD thesis, Universitat Autonoma de Barcelona (2009)
11. Qureshi, R.J., Ramel, J.Y., Cardot, H., Mukherji, P.: Combination of symbolic and statistical features for symbols recognition. In: ICSCN. (2007) 477–482
12. Zhang, W., Liu, W.: A new vectorial signature for quick symbol indexing, filtering and recognition. In: ICDAR. Volume 9. (2007) 536–540
13. Rusiñol, M., Lladós, J.: Symbol spotting in technical drawings using vectorial signatures. In: GREC. Volume 3926 of LNCS. (2006) 35–46
14. Dosch, P., Llados, J.: Vectorial signatures for symbol discrimination. In: GREC. Volume 3088 of LNCS. (2003) 154–165
15. Coustaty, M., Guillas, S., Visani, M., Bertet, K., Ogier, J.M.: On the joint use of a structural signature and a galois lattice classifier for symbol recognition. In: GREC. Volume 5046 of LNCS. (2007) 61–70
16. Luqman, M.M., Brouard, T., Ramel, J.Y.: Graphic symbol recognition using graph based signature and bayesian network classifier. In: ICDAR. Volume 10. (2009) 1325–1329
17. Ventura, A., Schettini, R.: Graphic symbol recognition using a signature technique. In: ICPR. Volume 2. (1994) 533–535
18. Valveny, E., Marti, E.: Deformable template matching within a bayesian framework for hand-written graphic symbol recognition. In: GREC. Volume 1941 of LNCS. (1999) 193–208
19. Mitra, S., Pal, S.K.: Fuzzy sets in pattern recognition and machine intelligence. FSS **156**(3) (2005) 381–386
20. Charniak, E.: Bayesian networks without tears. AI Magazine **12**(4) (1991) 50–63
21. Heckerman, D.: A tutorial on learning with bayesian networks. In: Innovations in Bayesian Networks. Volume 156 of Studies in Computational Intelligence. (2008) 33–82 Bayesian Network.
22. Delaplace, A., Brouard, T., Cardot, H.: Two evolutionary methods for learning bayesian network structures. In: CIS. Volume 4456 of LNCS. (2006) 288–297
23. Leray, P., François, O.: BNT structure learning package : Documentation and experiments. Technical report, Laboratoire PSI - INSA Rouen- FRE CNRS 2645 (November 2004)
24. Colot, O., Olivier, C., C., P., A., E.M.: Information criteria and abrupt changes in probability laws. In: Signal Processing VII : Theories and Applications. (1994) 1855–1858
25. Aksoy, S., Ye, M., Schauf, M., Song, M., Wang, Y., Haralick, R.M., Parker, J.R., Pivovarov, J., Royko, D., Sun, C., Farneback, G.: Algorithm performance contest. In: ICPR. Volume IV. (2000) 870–876
26. Valveny, E., Dosch, P.: Symbol recognition contest: A synthesis. In: GREC. Volume 3088 of LNCS. (2003) 368–386
27. Dosch, P., Valveny, E.: Report on the second symbol recognition contest. In: GREC. Volume 3926 of LNCS. (2006) 381–397
28. Valveny, E., Dosch, P., Fornes, A., Escalera, S.: Report on the third contest on symbol recognition. In: GREC. Volume 5046 of LNCS. (2007) 321–328
29. http://www.imagemagick.org/: (As on 20 March 2010)
30. http://www.qgar.org/: (As on 20 March 2010)
31. http://symbcontestgrec05.loria.fr/symboldescription.php: (As on 20 March 2010)
32. Delalandre, M., Pridmore, T., Valveny, E., Locteau, H., Trupin, E.: Building synthetic graphical documents for performance evaluation. In: GREC. Volume 5046 of LNCS. (2007) 288–298
33. http://mathieu.delalandre.free.fr/projects/sesyd/queries.html: (As on 20 March 2010)