A New Strategy for Applying Grammatical Inference to Image Classification Problems

Hemerson Pistori
INOVISAO Computer Vision Lab.
Dom Bosco Catholic University, UCDB
Campo Grande, MS, Brazil
pistori@ucdb.br

Andrew Calway and Peter Flach
Computer Science Dept.
University of Bristol
Bristol, UK
andrew@compsci.bristol.ac.uk,Peter.Flach@bristol.ac.uk

***Abstract*—This paper presents a new strategy to represent an image as a string so that standard grammar induction techniques can be used in computer vision problems. Two sets of experiments using an artificial and a real dataset have been conducted in order to explore the new strategy parameters and to have a first glimpse on its comparative performance against some standard machine learning techniques. The results are encouraging and the proposal opens new paths of exploration for syntactical pattern recognition.**

***Keywords*-Syntactical Pattern Recognition; Grammar Inference; Image Processing**

I. Introduction

Constructing a formal language representation from a finite set of exemplar strings is a classical problem in computer science. Usually, the language is an infinite set of strings and the algorithms that induce, for instance, grammars and automata, from the finite subset of strings, can be viewed as machine learning algorithms. Grammar learning or grammatical inference is commonly used in fields like bioinformatics [1] and machine translation [2] as the problems are naturally related to string processing in these fields. The use of techniques derived from the formal language disciplines for visual pattern recognition is not new and was much explored in the area called syntactical or structural pattern recognition [3]. Recently, the idea of image grammars and image parsers regained attention with a series of works by Zhu at al [4], [5], [6]. A main issue in exploring formal languages techniques and grammar inference in computer vision is how to represent the images. The proposals vary from a standard textual string to strategies that try to explicitly preserve spatial information, like matrix grammars, graph grammars [7] and attribute graph grammars [5].

As the image representation strategy moves further away from the standard textual string, in order to preserve spatial information explicitly, it becomes more difficult to take advantage of the advances in formal languages processing. In this paper, a return to a pure string representation for the image, exploring keypoints detectors and visual words, is investigated as an alternative to simplify the re-use of string processing algorithms. The main idea of the proposal is to impose an order to the n keypoints detected using any keypoint detector and produce a string of size n. The alphabet for the string symbols is defined using k-means, as in the Bag of Visual Words or simply Bag of Words technique [8]. Experiments using a synthetic image dataset were conducted in order to analyse the impact of the alphabet size and the keypoints ordering strategy on the classification performance. A comparison of the proposed technique against K-Nearest Neighbourhood (KNN), Support Vector Machines (SVM) and Decision Trees (C4.5) based machine learning techniques using the 15-scenes benchmark described in [9] is also reported. For these experiments, the grammar inference algorithm K-Testable [10] and the keypoints detector Speed-Up Robust Features (SURF) [11] have been used, but the technique can be applied using any string grammar inference and keypoints detector algorithms available.

In the next section a review on the related works is presented. Section III reviews some concepts used in this paper. The proposed method is detailed in Section IV, followed by the experimental setup description in Section V. Results and analysis are presented in Section VI, followed by the conclusions and future works in Section VII.

II. Related Work

The first attempts to use linear string languages to describe images can be traced back to the beginning of the sixties [12], [13]. A formal descriptive mechanism , called Picture Description Language, PDL, has been presented in [14] and was aimed to be a language of discourse about pictures both as an analysis (computer vision) and as a generator (computer graphics) tool. In PDL, the terminal symbols were called primitive patterns or picture primitives, and their semantics were dependent on the application. In general, picture primitive symbols should be associated to image patches that are more conveniently recognized as unit than in terms of their parts. Each primitive has one and just one tail and one head that are used to connect it to other primitives. This property ensured the linearity of the language. An especial primitive was used to represent the "blank spaces" that could happen among disconnected pictorial elements in the same image. Our approach is similar to PDL in the way that it also entails linear strings to describe images but using a different strategy to define and combine picture primitive symbols.

In order to overcome the string language limitation to express more naturally the two dimensional relations inherited in image representation, several extensions to the standard grammars have been proposed [15], the most populars being related to the use of some category of arrays [16] or graphs [17], [6] in place of strings. The substitution of strings by multi-dimensional constructs introduces the important problem of embedding, as unlike with strings, there is no unique obvious way to select what and how parts of a graph can be changed to generate the next graph in a parse derivation. Several different strategies to represent the graphs, to restrict the graph categories to be used and to define the embedding transform have been proposed [18]. In [19] the use of pattern grammars expressed using predicate logic is suggested to improve accuracy in object detection. The proposed method uses standard techniques to achieve a first rough estimation of object parts and the predicate logic reasoning to improve the estimation. As a by-product, the system can produce proofs explaining why the object could or could not be identified.

The And-Or-Graph representation proposed in [6] is based on a digraph with edges representing different kinds of relations among scenes, objects and image patches associated with the graph vertices. Image primitives in this approach can be of three kinds: (1) textons (blobs, bars, junctions, etc) , (2) texture areas and (3) flat areas. In [20] a mechanism called visual grammar is applied to the problem of scene recognition. Most of the proposal is related to the application of statistical techniques to derive region features and spatial relations from many different image pixel attributes. The grammatical component of the proposal seems to be related only to the use of a hierarchical structure to represent the image through attributed relational graphs (a graph with labelled vertices and edges).

Wang et al. [21] propose an extension of the Stochastic Context Free Grammars (SCFG), used in string processing, to multidimensional domains by introducing the spatial random trees (SRTs). An SRT is a stochastic hidden tree whose leafs are associated with rectangular areas of the image. The rectangles (tiles) can be of different sizes an form a complete tiling of the image.

III. Background

In this section the major concepts on which the investigation has been based are reviewed in order to keep the paper as self-contained as possible.

A. Speed-Up Robust Features - SURF

The Speed-Up Robust Features or SURF algorithm is a strategy to both detect and describe interest points from an image. Interest points are detected using a very basic but fast approximation of the Hessian matrix (Gaussian Second Order Partial Derivatives), based on box filters. Implicit image pyramids are used to search for interest points in different scales and points with small Hessian matrix determinants are suppressed using a non-maximum suppression strategy in a $3 \times 3 \times 3$ neighbourhood. Figure 1 shows several examples of interest points detected (circled) in some scene images.

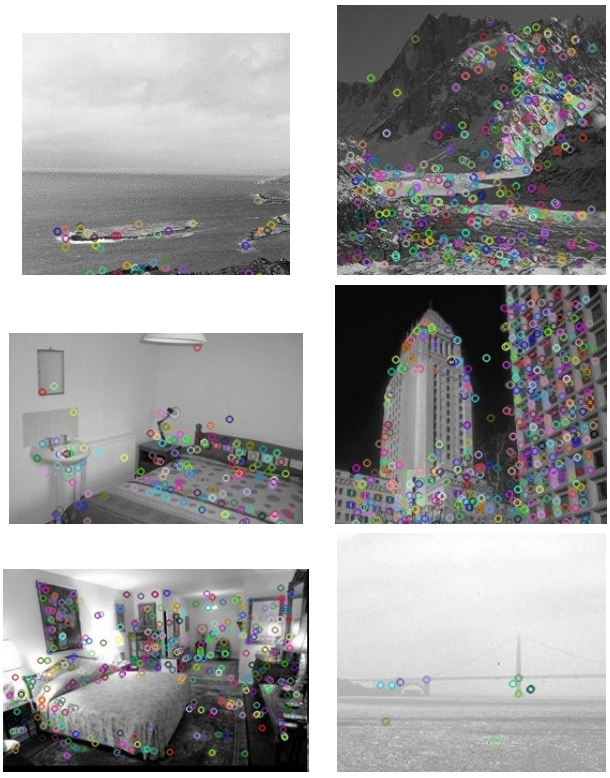

Fig. 1. SURF interest points extracted from 6 images of the 15 scenes dataset described in [9]

For each interest point detected, SURF generates a feature vector containing 64 values. SURF descriptors are based on similar properties as the Scale-Invariant Feature Transform (SIFT) technique [22] but even more simplified. First, the overall orientation (regarding gradient information) of a small circular region around the interest point is calculated. A rectangular region aligned using this orientation is construct around the interest point. This rectangular region is split regularly into 4×4 square sub-regions and sampled Haar wavelet responses in vertical and horizontal direction (regarding the rectangle orientation), weighted with a Gaussian centred at the keypoint, are extracted for each sub-region. These wavelet responses and their absolute values are summed up for each sub-region. And so, for each of the 16 sub-regions, 4 values are calculated, resulting in the 64d feature vector. Integral images [23] are used whenever possible in SURF algorithm in order to speed-up filters response calculations.

B. Bag of Words - BOW

Both SIFT and SURF algorithms describe an image using a variable size set of interest points, with their respective keypoint descriptors (a 64 valued feature vector in the case of SURF). In order to use a standard machine learning that needs a fixed size vector as input, the Bag of Words (BOW) strategy can be used. In BOW, the interest points extracted from a set of training images are clustered, using k-means and the

resulting k cluster centers, also called keypoints (in analogy with the keywords in string processing) form a dictionary of size k. Each interest point can now be represented by the dictionary entry corresponding to its nearest cluster center (nearest keypoint). A k-bins histogram of keypoints is used to represent an image. Each bin of this histogram is also called a visual word. The size of the dictionary is a key feature in this strategy an it is application dependent. The strategy can be straightforwardly adapted to any interest point detector, but was initially formulated using SIFT (SURF was introduced after BOW).

C. K-Testable

A K-Testable language is a subclass of the regular languages for which parsing of any string can be done using a fixed memory of size k. The K-Testable grammar inference algorithm is able to infer K-Testable languages in polynomial time. In essence, this algorithm finds prefixes, substrings and suffixes that occur in the training data [24]. In this paper, the implementation of K-Testable available with GI Toolbox for Matlab has been used [10]

IV. Proposed Approach

The approach proposed in this paper consists of a supervised learning strategy that combines interest points detection and grammar inference. At first, a dictionary is constructed, from training images, using the same strategy used in BOW. This dictionary is mapped to an alphabet using one symbol (E.g.: ASCII character) for each visual word (keypoint cluster centers) of the dictionary. For each image, both in training and in testing phase, keypoints are detected and mapped to a visual word. The keypoints are traversed using a predefined order and a string is created by concatenating the symbols (characters) that correspond to each visual word. The keypoints can be traversed in several different ways. In the experiments described in this paper the following orders were tested: (1) the order returned by the SURF implementation in OpenCV which traverses the image several times, in reading order, for each different scale (image pyramid) used by SURF and described in [11]; (2) a random order; (3) a radial order that starts from the keypoints central point and proceeds by choosing the nearest point using the Euclidean distance from the center; (4) a reading order that traverses the image from left to right and from top to bottom; (5) an approximate reading order that before the traversal, quantize or smooth the keypoints position using a quantization parameter.

During training (learning), for each class, a grammar (or an alternative representation for a grammar, like an automaton) is induced from the set of strings corresponding to the training images of this class. In the case of K-Testable, which is used in the experiments of this paper, a definite finite state automaton (DFA) is used to represent the grammar, and so, for each class, an automaton is induced. As some symbols from the dictionary may not be available in certain set of images used to train a class, the DFA may not have a transition related to that symbol. In this case, during execution of the automaton, the symbol is ignored and the automaton proceeds reading the next symbols without changing the current state. An error recovery strategy is used so that the automaton will always read all the symbols of the input string. The number of error corrections used during the string processing is used for classification during testing.

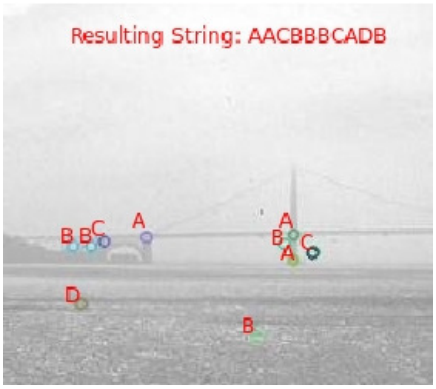

Fig. 2. Illustration of a string extracted from an image using the proposed approach

During testing, for each image a string is generated using the same strategy as during training. This string is parsed by each class corresponding parser and is assigned to the class whose parser produces the smallest error counter when processing the string. In the case of a tie, one class is returned arbitrarily. In K-Testable, this parsing corresponds to running the automaton on the string, with error recovery. Obviously, other grammar induction strategies using different kinds of representation (E.g: push-down automata, context free grammars, etc) could be used in place of K-Testable.

Figure 2 illustrates the derivation of a string from an image using the proposed approach. In this image, 10 keypoints have been detected and a dictionary $\Sigma = \{A, B, C, D\}$ of size 4 was used. The resulting string, using the reading order would be $AACBBBCADB$.

V. Experimental Setup

Two experiments have been performed. The first used a synthetic set of images involving spatial relations between triangles and squares. The dataset is composed of 6 classes, each containing 8 exemplar images. The 48 images from this dataset are shown in Figure 3. The proposed approach has been experimented using 153 different configurations. The dictionary size has been varied from 2 to 50, with an increment of 3. Seven different keypoints orderings have been tested: (1) SURF order (PYRAMID); (2) Random Order (RAND); (3) Radial Order (RADIAL), (4) Reading Order (READ) and Quantized Reading Order using the quantization factor (5) 5 x 5 (READ 5x5), (6) 10 x 10 (READ 10x10) and (7) 20 x 20 (READ 20x20). A quantization factor of N x N means that both the X and the Y coordinates of each keypoint are integer divided by N before applying the reading order. A 75% random split sampling strategy, with 8 repetitions, has being

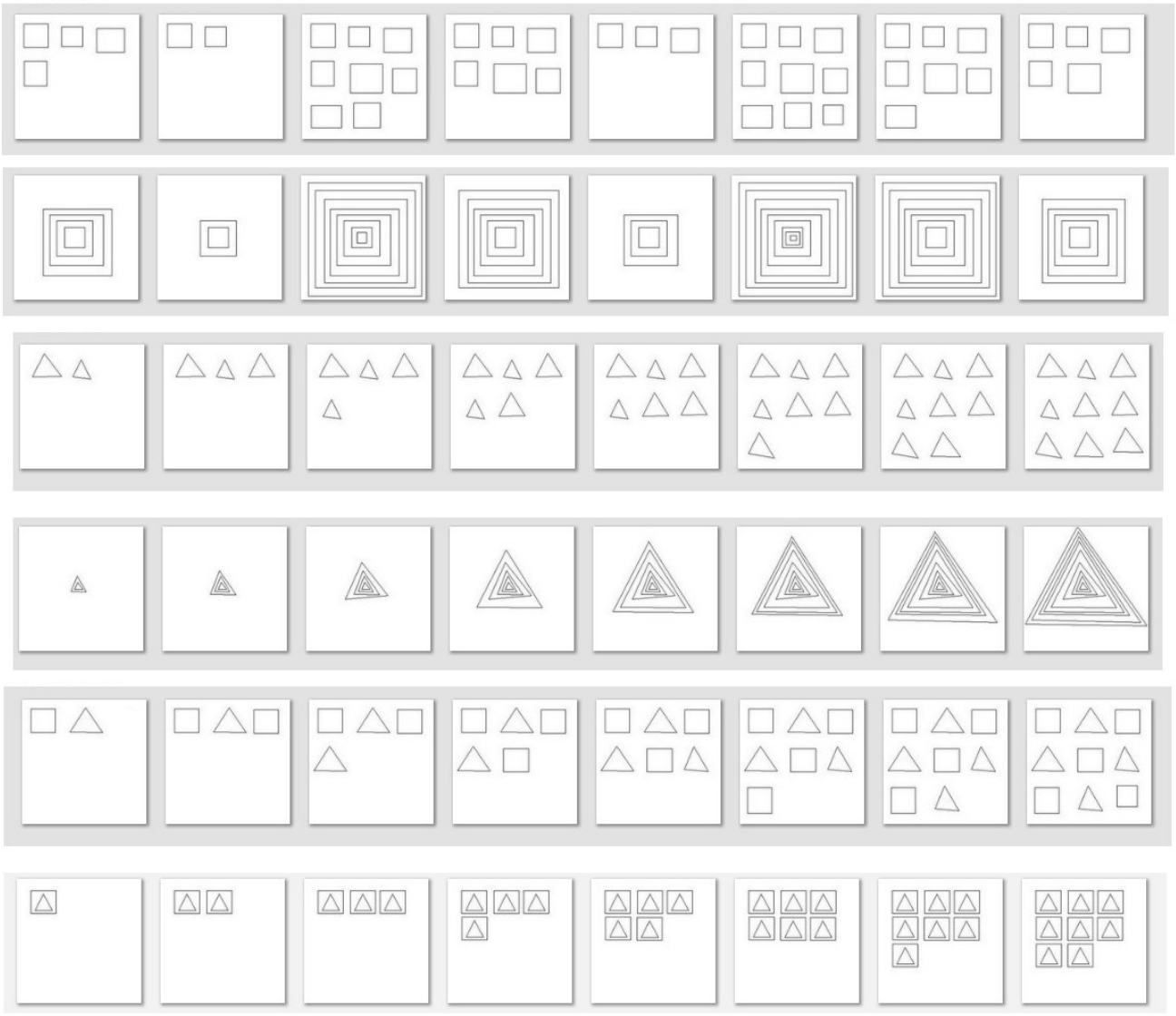

Fig. 3. The 48 images from the geom dataset. Each row corresponds to one of the 6 classes

used for choosing training and testing sets.

In the second experiment, the 15 scenes dataset [9] has been used[1]. Some samples of this dataset are presented in Figure 4. It comprises a set of 4485 gray-scale images from 15 different categories of scenes classified by humans. Table I summarizes the categories represented in the 15 scenes dataset. The average image size is 300×250.

The method has been compared to Support Vector Machines using the WEKA 3.6.1 SMO implementation with the default parameters, to KNN with K=10 and to C4.5, also from WEKA. The Bag of Words approach, with a dictionary of size 50 and SURF features, has been adopted to generate the input vector for these three algorithms. The K parameter for KNN has been chosen experimentally from the best F-Measure when K varies from 1 to 40 (using the same dataset but with another randomly chosen training and test partition). The F-Measure score for each classifier applied to each different class has been

[1]The 15 scenes dataset has been downloaded in March, 2012, from $www-cvr.ai.uiuc.edu/ponce\_grp/data/$

TABLE I
NUMBER OF TOTAL SAMPLES AVAILABLE FOR EACH SCENE CATEGORY

Class	Samples	Class	Samples
bedroom	216	inside city	308
suburb	241	mountain	374
industrial	311	open country	410
kitchen	210	street	292
living room	289	tall building	356
coast	360	office	215
forest	328	store	315
highway	260		

TABLE II
F-MEASURES USING K-TESTABLE GRAMMAR INDUCTION STRATEGY AND DIFFERENT DICTIONARY SIZES (LINES) AND KEYPOINTS ORDERS (COLUMNS)

Dic. Size	PYRAMID	RAND	RADIAL	READ	READ 5x5	READ 10x10	READ 20x20
2	20.00%	13.00%	25.00%	22.00%	21.00%	23.00%	**48.00%**
5	**94.00%**	71.00%	67.00%	91.00%	77.00%	80.00%	83.00%
8	89.00%	69.00%	70.00%	90.00%	92.00%	**94.00%**	88.00%
11	**96.00%**	78.00%	80.00%	89.00%	94.00%	94.00%	92.00%
14	**98.00%**	85.00%	76.00%	88.00%	83.00%	91.00%	94.00%
17	**96.00%**	86.00%	79.00%	89.00%	79.00%	95.00%	91.00%
20	91.00%	84.00%	82.00%	87.00%	95.00%	**98.00%**	92.00%
23	93.00%	91.00%	83.00%	89.00%	92.00%	89.00%	**95.00%**
26	92.00%	85.00%	77.00%	79.00%	94.00%	90.00%	**96.00%**
29	94.00%	86.00%	78.00%	82.00%	90.00%	**95.00%**	92.00%
32	**96.00%**	84.00%	76.00%	83.00%	91.00%	93.00%	81.00%
35	**95.00%**	80.00%	83.00%	85.00%	92.00%	93.00%	86.00%
38	**96.00%**	78.00%	82.00%	86.00%	92.00%	88.00%	80.00%
41	**95.00%**	78.00%	83.00%	81.00%	94.00%	92.00%	89.00%
44	**92.00%**	83.00%	77.00%	89.00%	91.00%	**92.00%**	**92.00%**
47	**99.00%**	83.00%	76.00%	80.00%	86.00%	89.00%	88.00%
50	**96.00%**	89.00%	73.00%	89.00%	93.00%	87.00%	84.00%

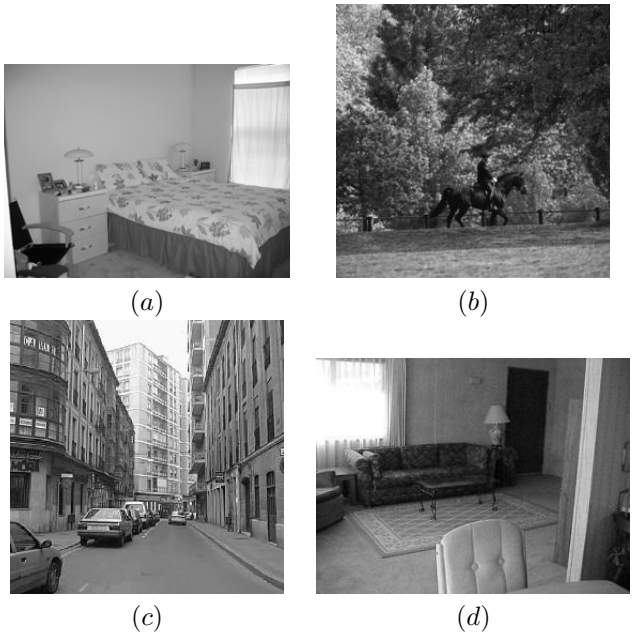

Fig. 4. Four samples from the 15 scenes dataset: (a) bedroom, (b) forest, (c) street and (d) living room

chosen as the metric for a Friedman test [25]. The dataset has been randomly partitioned using, in each class, 70% of the samples for training and 30% for testing. The dictionary size of 50 has been used because it is the maximum size that the current implementation of the grammatical inference algorithm can handle, as basically only lower and upper case letters can be used as the grammar terminal symbols, and because previous experiments with the other three approaches indicated that they perform better with larger dictionary sizes. The keypoints order for the proposed approach, in the second experiment, was chosen based on the performance during the first experiment: SURF order (PYRAMID).

VI. RESULTS AND DISCUSSION

Table II presents the results of the first experiment. The highest F-Measure, of 99%, has been achieved using 47 symbols and the SURF order (PYRAMID). The SURF order implicitly captures scale information, which may explain this superior performance. The random order was consistently inferior to all the other orders, except for the non-quantized reading order (READ), indicating that the grammar inference approach is being able to extract information from the way keypoints are organized in the image. The quantized reading order mitigates small noisy variations in the keypoints positions that should otherwise appear aligned in the image. In Figure 2, for instance, more keypoints would be aligned to the bridge if a proper spatial quantization factor was used. This effect may justify the inferior performance of the pure reading order in this experiment.

TABLE III
F-MEASURES FOR PROPOSED APPROACH, KNN WITH K=15, SVM AND C4.5. HIGHEST VALUES FOR EACH CLASS ARE IN BOLD

Class	Proposed	KNN	SVM	C4.5
bedroom	0.00%	7.80%	6.70%	**11.90%**
coast	14.00%	46.40%	**50.40%**	38.10%
forest	**64.00%**	51.40%	55.70%	57.70%
highway	0.00%	15.00%	**25.70%**	25.00%
industrial	**16.00%**	15.00%	12.30%	15.80%
insidecity	**22.00%**	14.90%	11.50%	11.80%
kitchen	0.00%	**12.50%**	6.70%	7.40%
livingroom	2.00%	13.10%	10.80%	**15.50%**
mountain	**17.00%**	11.40%	16.10%	11.60%
office	0.00%	5.10%	**19.10%**	9.30%
opencountry	2.00%	11.30%	21.50%	**24.70%**
store	**22.00%**	10.70%	16.50%	17.90%
street	15.00%	8.80%	13.00%	15.80%
suburb	16.00%	11.30%	9.20%	**20.80%**
tallbuilding	**25.00%**	6.90%	14.10%	22.70%
Mean	14.33%	16.11%	19.29%	20.40%

Table III presents the F-Measures achieved in the second experiment for each tested approach (columns) and scene categories (lines), with the last line showing the mean F-

Measure among all categories. The p-value using Friedman test on the data presented in Table III is 0.1351 and so we cannot reject the null hypothesis that the algorithms present the same F-Measure performance even in a 0.05 confidence level. In 6 out of the 15 classes, the proposed approach achieved the highest F-measure, the largest number of highest scores per class among all the algorithms tested. C4.5 came in second, with highest scores in 4 classes.

VII. Conclusions and Future Work

A return to the simple approach of converting the image into a string and use string processing algorithms in Computer Vision has been investigated in face of the existence of new strategies for image feature extraction (like SURF and SIFT) and new string processing algorithms that have not been explored when these ideas have been first tested 30 or 40 years ago. The proposed approach compares well with some available algorithms for scene classification problems using smaller dictionary size but the main contribution of this paper is the proposal of a novel method to convert images to strings, which can pave the way for new improvements in the syntactic pattern recognition area when applied to image processing. One main limitation of the BOW method is that it loses important spatial and structural information (the BOW histogram only counts the occurrence of some prototypical keypoints). The string created using the proposed approach preserves, implicitly, some spatial and structural information that can be retrieved using an appropriate grammar inference algorithm (we have only tested the system with a regular language induced, so far).

Suggestions for future research include the use of more powerful grammar inducing algorithms (e.g.:non regular language induction algorithms) and error recovery strategies. Differently from natural language induction problems, finding negative examples can be easier in scene recognition, so it is worth trying grammar induction algorithms that can take advantage of the availability of negative examples (K-Testable explores only positive examples). Using the parser error recovery strategy to devise a way to compare strings derived from images is something that needs further exploration and can be an alternative to the more common metrics used in computer vision. Stochastic grammars are also an alternative to be tried as well as larger dictionaries and different ways to choose the best grammar. Optimizations and experiments regarding execution time performance are also advised.

Acknowledgements

The first author of this paper hold a scholarship from the Brazilian National Counsel of Technological and Scientific Development, CNPQ.

References

[1] R. Damaševičius, "Structural analysis of regulatory dna sequences using grammar inference and support vector machine," *Neurocomputing*, vol. 73, pp. 633–638, January 2010.

[2] K. Probst, "Automatically induced syntactic transfer rules for machine translation under a very limited data scenario," Ph.D. dissertation, Carnegie Mellon University, 2005.

[3] K. Fu, *Syntactic methods in pattern recognition*, ser. Mathematics in science and engineering. Academic Press, 1974.

[4] Z. Tu, X. Chen, A. L. Yuille, and S.-c. Zhu, "Image parsing: Unifying segmentation, detection, and recognition," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 113–140, 2005.

[5] F. Han and S.-C. Zhu, "Bottom-up/top-down image parsing by attribute graph grammar," *Computer Vision, IEEE International Conference on*, vol. 2, pp. 1778–1785, 2005.

[6] B. Yao, X. Yang, L. Lin, M. Lee, and S. Zhu, "I2t: Image parsing to text description," *Proceedings of IEEE*, vol. 98, no. 8, pp. 1485–1508, August 2010.

[7] G. Chanda and F. Dellaert, "Grammatical methods in computer vision: An overview," Tech. Rep., 2004.

[8] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *In Workshop on Statistical Learning in Computer Vision, ECCV*, 2004, pp. 1–22.

[9] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, ser. CVPR '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 2169–2178.

[10] H. I. Akram, C. de la Higuera, H. Xiao, and C. Eckert, "Grammatical inference algorithms in matlab," in *ICGI 2010: Proceedings of the 10th International Colloquium on Grammatical Inference*. Valencia, Spain: Springer-Verlag, 2010.

[11] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Underst.*, vol. 110, pp. 346–359, June 2008.

[12] M. Eden, "On the formalization of handwriting," in *Am. Math. Soc. Appl. Math Symp*, vol. 12, 1961, pp. 83–88.

[13] R. Narasimhan, "A linguistic approach to pattern recognition," University of Illinois, Tech. Rep., 1962.

[14] A. C. Shaw, "A formal picture description scheme as a basis for picture processing systems," *Information and Control*, vol. 14, pp. 9–52, 1969.

[15] A. Rosenfeld, "Array grammars," in *Graph-Grammars and Their Application to Computer Science*, ser. Lecture Notes in Computer Science, H. Ehrig, M. Nagl, G. Rozenberg, and A. Rosenfeld, Eds. Springer Berlin / Heidelberg, 1987, vol. 291, pp. 67–70.

[16] R. A. Kirsch, "Computer interpretation of english text and picture patterns," *Electronic Computers, IEEE Transactions on*, vol. EC-13, no. 4, pp. 363 –376, aug. 1964.

[17] J. L. Pfaltz and A. Rosenfeld, "Web grammars," in *IJCAI*, 1969, pp. 609–620.

[18] H. Fahmy and D. Blostein, "A survey of graph grammars: Theory and applications," *Pattern Recognition*, vol. 2, pp. 294–298, 1992.

[19] V. Shet, M. Singh, C. Bahlmann, V. Ramesh, J. Neumann, and L. Davis, "Predicate logic based image grammars for complex pattern recognition," *Int. J. Comput. Vision*, vol. 93, no. 2, pp. 141–161, Jun. 2011.

[20] S. Aksoy, K. Koperski, C. Tusk, G. Marchisio, J. C. Tilton, and S. Member, "Learning bayesian classifiers for scene classification with a visual grammar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, pp. 581–589, 2005.

[21] W. Wang, I. Pollak, T. shing Wong, C. A. Bouman, and M. P. Harper, "Hierarchical stochastic image grammars for classification and segmentation," *IEEE Trans. Image Processing*, vol. 15, pp. 3033–3052, 2006.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[23] F. C. Crow, "Summed-area tables for texture mapping," *SIGGRAPH Comput. Graph.*, vol. 18, pp. 207–212, January 1984.

[24] C. De La Higuera, *Grammatical Inference: Learning Automata and Grammars*. Cambridge University Press, 2010.

[25] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.