International Journal of Applied Research 2017; 3(8): 368-373

ISSN Print: 2394-7500
ISSN Online: 2394-5869
Impact Factor: 5.2
IJAR 2017; 3(8): 368-373
www.allresearchjournal.com
Received: 27-06-2017
Accepted: 28-07-2017

Tathagat Biswas
3rd Year, B. Tech Computer Science, B.P. Poddar Institute of Management and Technology, Kolkata, West Bangle, India

Saikat Kundu
3rd Year, B. Tech Computer Science, B.P. Poddar Institute of Management and Technology, Kolkata, West Bangle, India

Samir Kumar
Professor Bandyopadhyay Department of Computer Science and Engineering Department, University of Calcutta, Kolkata, West Bangle, India

Correspondence
Tathagat Biswas
3rd Year, B. Tech Computer Science, B.P. Poddar Institute of Management and Technology, Kolkata, West Bangle, India

Analysis of geometric figures- A review

Tathagat Biswas, Saikat Kundu and Samir Kumar

Abstract
A pattern can be used in ascertaining the effect on other related aspects. In the present work, effort has been made to classify geometrical figures using syntactic pattern recognition. For example, in ECG, geometrical patterns can be converted to grammar using pattern primitives and can be used for detection and recognition of related diseases.

Keywords: Geometric figures, ECG, Structural variability

Introduction

Structural variability within an object category may be well explained using a geometric grammar, especially a probabilistic one which can incorporate uncertain and incomplete measurements. In such an approach, object classification is reduced to detection of object parts and verification of the geometric relations among them. Since robust detection of complex object parts is as difficult as object classification itself, the representation of the overall structure of parts is reduced to cuboids, which are independent of appearance and easier to detect in point clouds.

Parsing is used to derive a string using the production rules of a grammar. It is used to check the acceptability of a string. A parser takes the inputs and builds a parse tree.

A parser can be of two types −

- Top-Down Parser − Top-down parsing starts from the top with the start-symbol and derives a string using a parse tree.
- Bottom-Up Parser − Bottom-up parsing starts from the bottom with the string and comes to the start symbol using a parse tree.

In this work we used the approach of Bottom-Up parser, as it felt more convenient to us. Both ways are equally efficient in parsing.

Using the concepts of automata we can design a Bottom-Up parser in this way. A Bottom-Up parser basically uses the terminals to start the parsing and following the production rule parses it's way to the start symbol 'S'.

Initially for theoretical results we started with some grammars and made our own production rules to parse a string of that particular grammar. For bottom-up parsing, a PDA has the following four types of transitions

- Push the current input symbol into the stack.
- Replace the right-hand side of a production at the top of the stack with its left-hand side.
- If the top of the stack element matches with the current input symbol, pop it.
- If the input string is fully read and only if the start symbol 'S' remains in the stack, pop it and go to the final state 'F'.

Two recent trends in technology and research are first, the advent of a novel type of range camera to capture 3D scenes, and second, the regained interest to solve the classification problem with object geometry, in particular, object structure. Object structure has been recognized as a strong characteristic for classification. However, due to the lack of a reliable, compact, and affordable range image sensor, most of the algorithms and systems relied on object structure information extracted from 2D images. The novel type of range camera bears the potential to be compact and affordable as they are based on well-established technologies. The two trends have yet little converged despite of the necessity of object classification in human-robot interaction. This paper attempts to demonstrate the recognition of basic geometrical objects using an algorithm that extracts information from the image.

Review Works

Doing image processing and especially blob analysis it is often required to check some objects' shape and depending on it perform further processing of a particular object or not. For example, some applications may require finding only circles from all the detected objects, or quadrilaterals, rectangles, etc [1]. Human vision seems to make use of many sources of information to detect and recognize an object in a scene. At the lowest level of object recognition, researchers agree that edge and region information are utilized to extract a “perceptual unit” in the scene. Some of the possible invariant features are recognized and additional signal properties (texture or appearance) are sent to help in making the decision as to whether a point belongs to an object or not. In many cases, boundary shape information, such as the rectangular shapes of vehicles in aerial imagery, seems to play a crucial role. Local features such as the eyes in a human face are sometimes useful. These features provide strong clues for recognition, and often they are invariant to many scene variables [2].

Content based image retrieval is one of the topics of interest in the computer vision field which nowadays is at its very peak, due to the growth in the last years of the amount of stored graphical information. For this kind of data, underlying analysis processes mainly lie on graphics recognition, allowing then classification of the images, typically in terms of available symbols. From a general viewpoint, several kinds of recognition approaches can be involved, according to data representation. Bitmap images are usually analyzed with statistical methods, which are time-consuming and quite accurate, but can also be analyzed with structural methods, faster but requiring a pre-vectorization step. In the context of content based image retrieval, the last approach is usually preferred, as the amount of considered data implies the use of efficient processes. One of the most important visual features when classifying images is shape of the represented objects and subsequently a lot of literature deal with object recognition by shape [3].

Some presents new approach of shape recognition from the tactile images by touching the surface of various real life objects. Here four geometric shaped objects (viz. a planar surface, object with one edge, a cubical object i.e. object with two edges and a cylindrical object) are used for shape recognition. The high pressure regions denoting surface edges have been segmented out via multilevel thresholding. These high pressure regions hereby obtained were unique to different object classes. Some regional descriptors have been used to uniquely describe the high pressure regions. These regional descriptors have been employed as the features needed for the classification purpose. Linear Support Vector Machine (LSVM) classifier is used for object shape classification. In noise free environment the classifier gives an average accuracy of 92.6%. Some statistical tests have been performed to prove the efficacy of the classification process. The classifier performance is also tested in noisy environment with different signal to-noise (SNR) ratios [4, 5].

Proposed Method and Results

The main idea behind generating invariant representations of shape is to obtain a compact set of descriptions sufficient to describe all relevant views of an object. It should also be considered as a way of building principled generalization mechanisms into learning systems. Therefore, a representation which is not compact but gives good generalization properties for a trained system would still be considered useful. The invariance properties that we believe to be important for general vision are translation, rotation and scale. This would prevent a learning system from having to discover all of the various manifestations of a rigid object in its vision module. This argument will be valid even for arbitrary occluding boundaries when there may be no true 3D invariance to rotation except in the image plane.

For Example

G = {(A, B), (a, b), P, S}

String: aabb

Step 1: Convert the terminals into non-terminals.
Therefore, string becomes: AABB

Step 2: Use the production rules to parse the string
Production rules: A <- a, B <- b, S <- AB, C <- SB, S <- AC
Parse tree for the string AABB:

AABB
ASB
AC
S

Our aim is to reach the start symbol using the production rules. We start from the terminals, then parse all the way to start symbol following the rules present. Let’s have a look at another example. Parse tree for the string AAABBB:

AAABBB
AASBB
AACB
ASB
AC
S

Any grammar of the form, G= {$a^n b^n$, n>0} can be parsed.

Algorithm:

P is either of the form A -> BC or A -> a. No empty string production is allowed.

Step 1:

Take a variable j, and initialize it to 1. Now t_{i1} as i ranges from 1 to n by placing A in t_{i1} exactly when there is a production rule A -> a_i in P.

Step 2:

Assuming $t_{i, j-1}$ has been formed for $1 \leq i \leq n$, compute t_{ij} where A is placed in t_{ij} when, for any k such that $1 \leq k \leq j$, there is a production rule in P with B in t_{ik} and C in $t_{ik, j-k}$.

Step 3:

Repeat step 2 until the table is completed or until an entire row has only null entries. x is in L (G) iff S is in $t_{in.}$

For parsing a string of length n, the amount of storage required by the algorithm (in worst case) is proportional to $n^{2.}$

The number of elementary operations (such as assigning a value to a variable or testing two variables for equality) is proportional to n^3.

Example:

G = ({S, A, B, C}, {a, b}, P, S)
Production rules, P: S - > AB, S -> AC, A -> a, B -> b, C -> SB

The grammar is in Chomsky normal form with no empty production rules.
The language is $\{x|x = a^m b^m, m \geq 1\}$
Let x = aabb be the input string to be parsed.

Step 1:
Set j = 1 and compute t_{ij} for $1 \leq i \leq 4$, therefore following the steps:
$t_{11} = \{A\}$, $t_{21} = \{A\}$ since A -> a
$t_{31} = \{B\}$, $t_{41} = \{B\}$ since B -> b

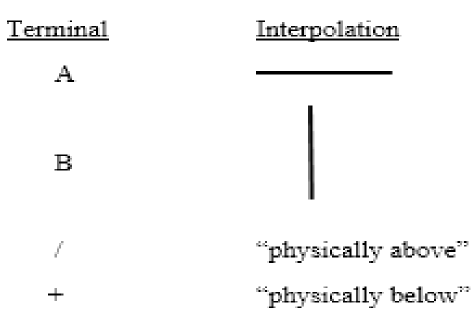
Step 2: First iteration:
Set j = 2 and compute t_{i2} for $1 \leq i \leq 3$
For a_1a_2 = AA, we find t_{12} = Null (since there is no production rule).
For a_2a_3 = AB, $t_{22} = \{S\}$ since a rule S -> AB is present.
For a_3a_4 = BB, $t_{32} = \emptyset$
If at this point t_{22} were also null, the algorithm would be terminated and the input string would be rejected. However, t_{22} is not null.
AA, AB, BB: combination taken two at a time.
For AA and BB there are no production rules.

Second iteration
Set j = 3, and compute t_{i3} for i<i<2,
For $a_1a_2a_3$ = AAB, we have $t_{13} = \emptyset$
For $a_2a_3a_4$ = ABB, we find t_{23} = C since C -> SB, S -> AB, B -> b

Third iteration
Set j = 4 and compute t_{14}. For $a_1a_2a_3a_4$ = AABB, we find $t_{14} = \{S\}$ because S -> AC, A -> a, C -> ABB

Step 3: Halt
With the help of grammars we can denote several problems in the form of geometric figures. Like suppose we have an equation in the form of arithmetic operations then also we can convert it in the form some geometric figure 1.

For example:



For operation:
a/(a+b)
From the above interpolation it means a "physically above" (a "physically below" b) the geometric figure it would be like:

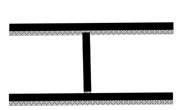


Fig 1: Description of pattern Primitives

G = ({S}, {a, b, +, /, (,)}, P, S)
P: S -> (S), S -> a, S -> b, S -> S+S, S -> S/S
S
⇨ S/S
⇨ a/(S)
⇨ a/(S+S)
⇨ a/(a+S)
⇨ a/(a+b)

Another example is shown in figure 2.

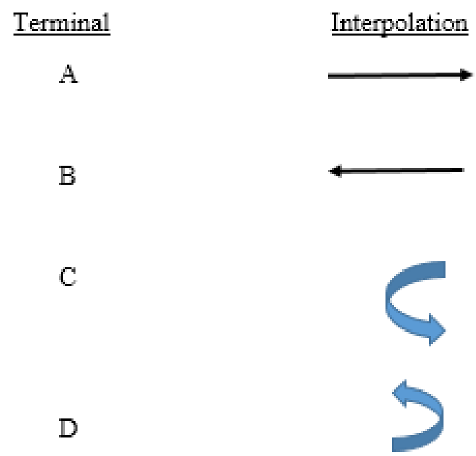
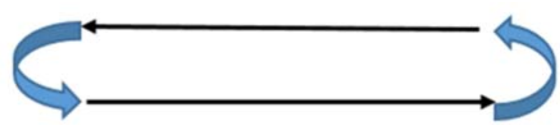


Fig 2: Proposed Pattern Primitives

Suppose we are provided a geometric figure like the following:

Above figure can be represented in the following form:

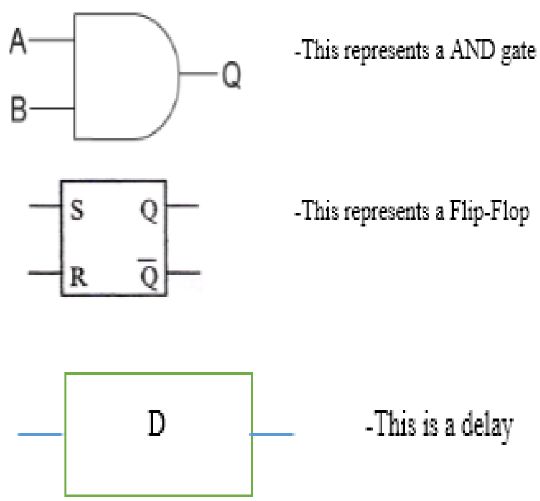


Following the interpolation the figure can be represented as: cadb

G = ({S, A}, {a, b, c, d}, P, S)
P: S <- cA, A <- aAb, A <- d
S
⇨ cA
⇨ caAb
⇨ cadb

Thus for such figures the general form would be: $ca^n db^n$
Several applications of pattern recognition with the use of grammar are described in figure 3.

Shift registers:



<FF> = Flip-Flop <A> = AND gate <D> = Delay

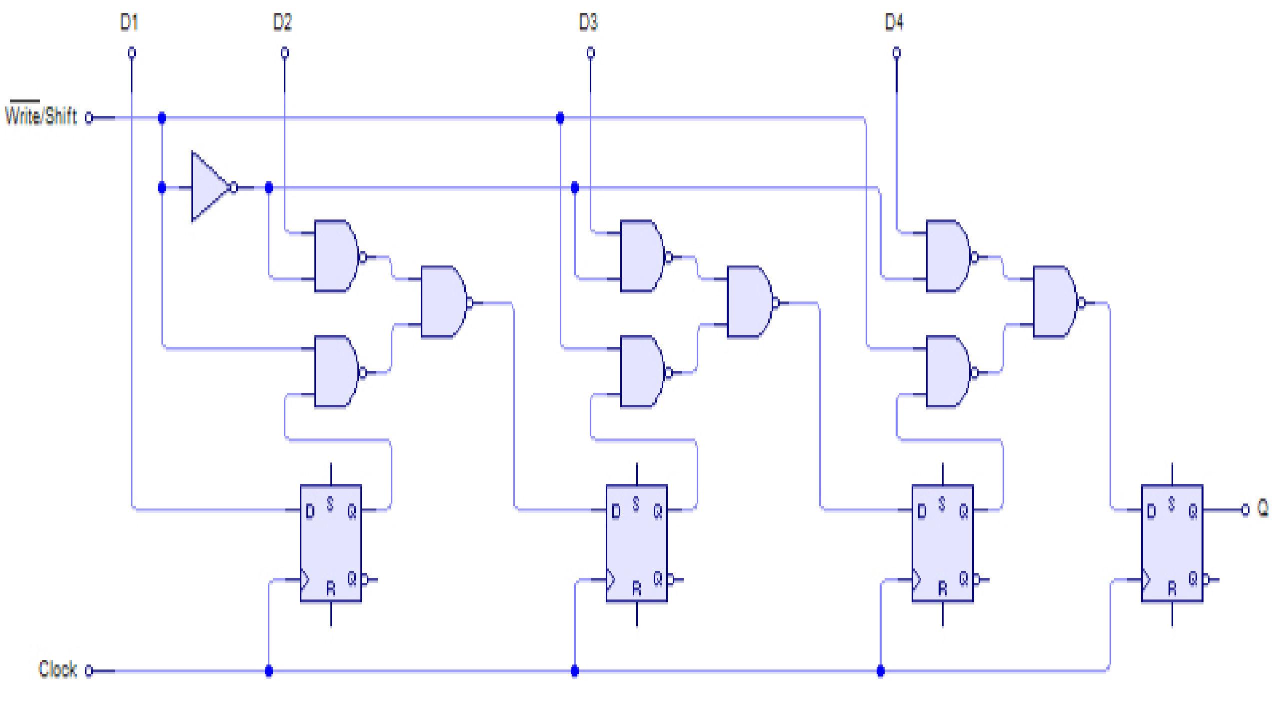


Fig 3: Shift register circuit

This can be represented in the following form:

<Shift register> ----> <Shift stage><Shift register>
<Shift stage> ----> <FF><A><A><D><D>

The electrocardiogram (ECG) is routinely used in clinical practice. Due to the large number of ECG's analyzed each year, it is worthwhile to automate the process to the maximum extent possible.

Computerized ECG processing systems, like manual ECG processing systems, perform two distinct tasks. The first is concerned with pattern recognition and parameter measurement. The second is an interpretation task, which utilizes the results of the first task. In typical systems the pattern recognition and parameter measurement task is the hardest. Attempts to automate this task have been made using non-syntactic methods, syntactic methods and hybrid methods.

Although the syntactic method seems suitable to the problem of ECG pattern recognition and parameter measurement, not much progress has been made to date. In the attempts reported, only specific aspects of this problem have been tackled. Linear and attribute grammars have been proposed for the detection of the QRS complexes. Context-free grammars have been used for the detection of certain ventricular arrhythmias.

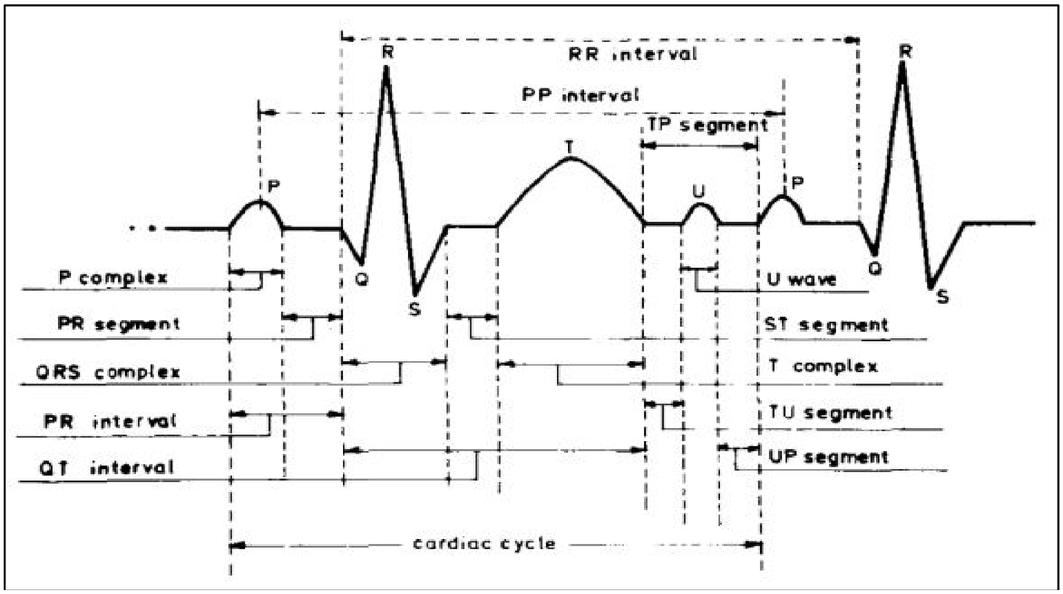


Fig 4: A cardiac cycle and its constituent patterns

The electrocardiographic patterns that constitute a cardiac cycle and must be recognized are the complexes, the inter wave segments, and the cardiac intervals. It is shown in figure 4.The complexes are three: the P complex, the QRS complex, and the T complex. The parameters of these patterns that must be measured are 1) height and duration for the complexes and some of their component waves and 2) duration for the inter wave segments and the cardiac intervals. Thus, there are two types of measurements to be performed: time measurements and amplitude measurements. Moreover, the QRS complexes have to be classified. In most cases they belong to one class but there are cases where they belong to more than one class. The detection and recognition are described below.

1) QRS Detection and Recognition: A series of n ($1 < n < 7$) consecutive peaks is recognized as a QRS complex
If:

a) E: = e, > E, where cl is a threshold value.
b) The angle between the right arm of peak i and the left arm of peak i + 1, i = l (1) n - 1, is less than c2, where c2 is a threshold value.

The first criterion, which is similar to the nonlinear transformation short-time energy used by other investigators, is adopted here due to its suitability in the syntactic approach and because it gives good results. The sample points taken in the summation are the ones of the corresponding QRS complex, while a constant number of sample points is used in the transformation.
The angle criterion prevents peaks belonging to P or T complexes from being merged with QRS complexes. The morphology of the QRS is determined by the alternative of the syntactic rule that matches the QRS.

2) P, T Detection and Recognition: One or two consecutive peaks are recognized as a P or T complex, by thresholding their width and amplitude depending on the syntactic rule being evaluated. They are discriminated from other peaks by comparing their energies. Noisy peaks in a region between two QRS complexes are required to have less energy than the energy of the P and T complexes in that region. The alternative of the syntactic rule that matches the P or T pattern specifies its morphology. It is noted that P and T complexes occurring before the first and after the last QRS complex found are not recognized. This helps to make the grammar simpler.

3) QRS Classification: The classification of the QRS complexes is performed by a nearest neighbor classification algorithm. The distance between a given QRS complex and a given class of QRS complexes is computed as the average of the distances between the given QRS complex and each QRS complex in the given class of QRS complexes. Both morphological (structural) and quantitative features are taken into account in the distance computation. Normalized duration and normalized amplitude are the statistical features used. Morphological features, in the distance computation between two complexes, are taken into account by aligning the complexes so that
The syntactic method to the problem of ECG pattern recognition and parameter measurement, as described above, was implemented and the resultant system named SERAMS (syntactic ECG recognition and measurement system).
The ECG acquisition component of this system is responsible for acquiring one ECG at a time in digital form. The primitive pattern extraction component of this system extracts the primitives of each ECG waveform and encodes them so that each waveform is transformed into a string of symbols (linguistic representation), each symbol accompanied by a set of attribute values. The attribute grammar evaluator component of this system takes as input
1) The pattern grammar of the Appendix and 2) the linguistic representation (together with its attributes) of a waveform. It recognizes the electrocardiographic patterns of that waveform and measures their parameters. Finally, the output formatter component of this system formats the results of the recognition and measurement.
SERAMS is coded in Fortran 77 because the primitive pattern extraction component employs mathematical algorithms that require an algebraic language. The structure of SERAMS is shown in figure 5.

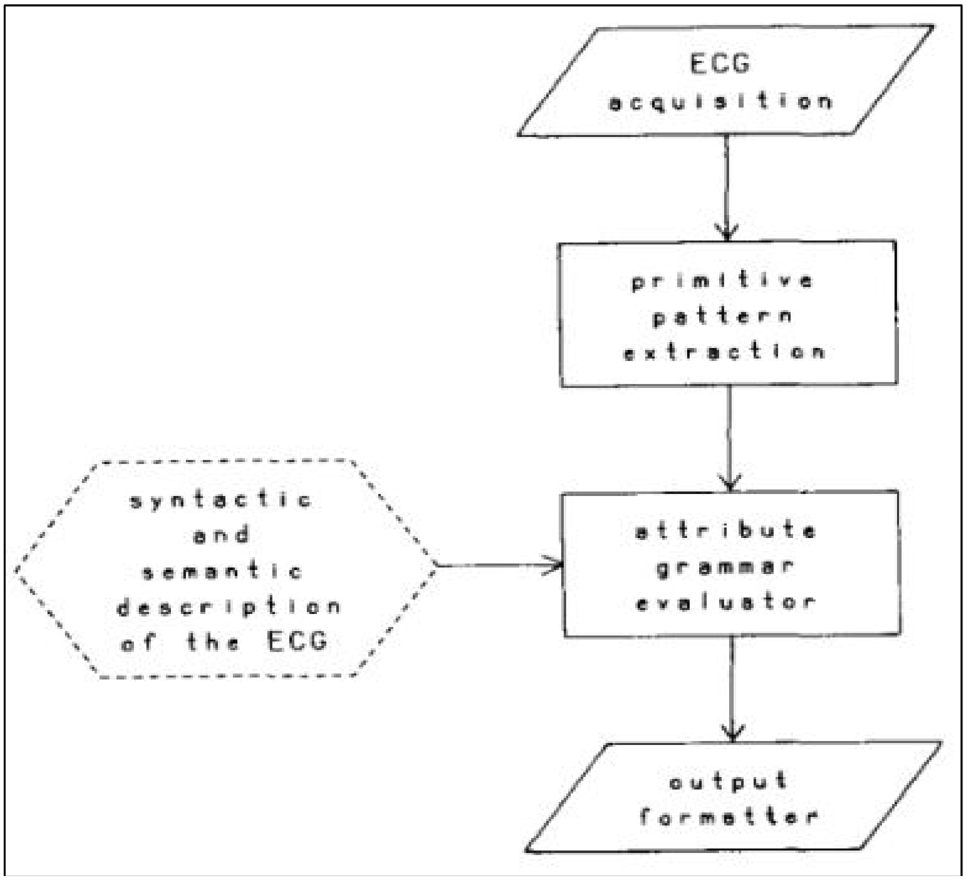


Fig 5: Structure of SERAMS

Conclusion

The application of the syntactic approach to ECG pattern recognition and parameter measurement which has been described has given results that are inferior compared to those reported by some implementations using the non-syntactic approach. However, the non-syntactic approach is fairly mature in this particular problem after considerable research work for many years. On the contrary, this is the first implementation of the syntactic approach and there is much room for improvement of the results by further refinement of the method.

We have observed that the primitive pattern extractor does not always accurately delineate the boundaries of the peak patterns. This type of error is propagated in the next stages and is responsible for many inaccurate results. Removing this deficiency would considerably improve the overall performance of the approach. This is not a trivial task, nevertheless it is tractable. Other than this, a very small percentage of noisy peaks are not rejected but recognized as real ones by the primitive extractor. However, this does not affect the system's performance as this type of error is corrected by the pattern grammar. Errors due to the grammar, i.e., missing or in correct recognition of a complex, were rarely observed. Other than the accuracy of the results, the syntactic approach possesses some very important characteristics, paper. These characteristics are: simplicity, brevity, clarity, understandability, and modifiability of the computer program that implements the syntactic approach. With the exception of the extraction of the primitive patterns and the I/O operations, the rest of the approach is not coded but specified, the pattern grammar being the formal specification.

References

1. Kulikova MS. Shape recognition for image scene analysis, Submitted for the degree of Doctor of Philosophy, Université de Nice - Sophia-Antipolis, 2009.
2. Felzenszwalb Pedro F. Representation and Detection of Shapes in Images, Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of Doctor of Philosophy, Massachusetts Institute of Tecnology, 2003.
3. Krish Karthik, Snyder Wesley. A Shape Recognition Algorithm Robust to Occlusion: Analysis and Performance Comparison. 2007, 1-24.
4. Moon Hankyu, Chellappa Rama, Rosenfeld Azriel. Optimal Edge-Based Shape Detection, IEEE Transcations on Image Processsing NG. 2002; 11(11):1209-1226.
5. Rusi-nol Mar¸ cal, Dosch Philippe, Llad-os Josep. Boundary Shape Recognition Using Accumulated Length and Angle Information, Pattern Recognition and Image Analysis, Springer. 2007; 4478:210-217. doi: 10.1007/978-3-540-72849-8_27.