Graphics Recognition – from Re-engineering to Retrieval

Karl Tombre, Bart Lamiroy
LORIA-INPL, École des Mines de Nancy
Parc de Saurupt, 54042 Nancy CEDEX, France
Karl.Tombre@loria.fr, Bart.Lamiroy@loria.fr

Abstract

In this paper, we discuss how the focus in document analysis, generally speaking, and in graphics recognition more specifically, has moved from re-engineering problems to indexing and information retrieval. After a review of ongoing work on these topics, we propose some challenges for the years to come.

1. Introduction

We are in an era where information is at the core of many industrial strategies. New information-centered services are created, traditional businesses are reoriented towards information services, and new niches appear. This evolution had been predicted by many experts. But they were wrong on one important point: They had foreseen the paperless society, with all-digital information, well structured and logically organized in a uniform way. The truth is very different. The grassroot user, as well as large companies, have a huge amount of information at their disposal, but this information is available in very "poor" formats: paper documents, or low-level, poorly structured digital formats such as Postscript, PDF or DXF.

The challenge is therefore to convert this poorly structured information into enriched information which can be used within an information system. This does not necessarily mean that it must be completely interpreted and converted to higher-level, semantic-based information. But we need at least to add a layer of indexes and navigation links, to allow for easy browsing and information retrieval.

This is a well known evolution in the document analysis community; whereas many efforts were made throughout the years to re-engineer documents, i.e. to come back to the logical and semantic structures carried by the document, the methods developed to perform this have also been proved to be of interest for the problem of indexing the information for efficient retrieval and browsing. Most efforts in this area have been done on textual documents. But the same problem also appears in the presence of graphics-rich documents: technical documentation, maps, schemas. The textual information extracted from these documents must be associated with graphical information: symbols, lines, areas... The input may be scanned paper documents, but also digital data such as those coming from DXF files.

In this paper, we examine how state of the art graphics recognition techniques can be used to address this problem, and we propose some interesting new challenges for further research, coming from the specific problems of indexing and browsing large-scale technical documentation. More specifically, we mention symbol and signature localization and recognition as a topic of interest, and we propose a way to represent context for intelligent document browsing.

2. State of the art

Traditionnally, document image analysis has been a kind of "reverse engineering" activity: Starting with a scanned paper document (the document image), the purpose of analysis has been to retrieve the original characters, words, sentences or structures of the document. In graphics-rich documents, this has led to work on raster-to-vector conversion, also called vectorization [10, 15, 19, 42, 48, 50]. There is also a large number of methods dedicated to the recognition of a given class of symbols [11, 30], in areas such as architecture [2, 29], mathematics [9], diagrams and schematics [22, 56], maps [6, 38], etc.

However, there are a growing number of applications where the need is not for conversion, but for efficient integration of heterogeneous documents into information systems. The problem here is not to have a precise vectorization. It is not either to match potential symbols against a library of existing reference symbols, for the simple reason that such a library does not necessarily exist when you have a large number of diverse documents. The problem is rather to be able to browse these documents, to add a layer of structure, syntax or even semantics on top of the raw data, to build indexes, etc.

There are two major research directions in this area. The first deals with media where there is very little contextual knowledge. Typically, image-based indexing and retrieval methods rely on low-level image features to compute similarity measures [41]; similarly, video sequences can be indexed into subsequences having a number of homogeneous features [7].

The other direction is that of indexing documents. As a document is a medium made by a person for the explicit purpose of conveying some useful information to other persons, using a language known by all actors, there is much more structure, syntax and semantics to rely on when analyzing a document. Still, the usual document analysis tools are not necessarily efficient enough, nor the most appropriate, for dealing with such indexing purposes. Examples of specific document analysis work for the purpose of indexing and information retrieval include indexing textual documents without performing OCR, just by basing the indexing on similarities or signatures of the connected components [43, 47], query on the layout structure [8, 40], or various information spotting techniques [23, 24].

In our case, we are interested in graphics-rich documents, typically technical documentation containing text, but also a lot of graphics. In this case, the usual text-based indexing and retrieval methods are still of interest, but we also need additional ways of accessing the information conveyed by the documents: recurring symbols, connections between textual descriptions and drawing parts, etc. Because of the large variability in the kinds of symbols and graphical entities which can be found in such documents, there has been less work in this area. Let us however mention some interesting contributions.

Syeda-Mahmood uses an adaptation of the topological rules used in the representation of the block world to index a database of maintenance drawings [44]. She uses a structural approach to the problem, with a step where the nodes are labeled by all possible hypotheses, and a discrete relaxation phase to find the most plausible interpretations.

Other groups explicitly rely on the text found in a legend or in the accompanying title box, as a basis for indexing [4, 34]. A more exhaustive variant is to take all symbols found in the legend as the basis for indexing [38, 39]. In a system developed at Boeing, basic text and graphics recognition tools are combined in an interactive environment to allow for information retrieval for maintenance purposes [5, 37].

But the more general problem of graphical information indexing, without a specific application domain, has also become a hot research topic. Müller and Rigoll propose a statistical framework for graphical retrieval in a database of engineering drawings [32, 33]. Fränti et al. [18] use the Hough transform to retrieve graphical parts from images.

We believe that there are plenty of specific research subjects in this general area, when we want to use graphics recognition methods to contribute to indexing and information retrieval in general documentation. In the following, we will propose some topical research directions.

3. Symbols and signatures

Symbol recognition consists in localizing and identifying the symbols which are present in a graphical document. Many symbol recognition methods have been proposed [11, 30]. Symbols are natural features to use in indexing and retrieval applications; however, the existing methods suffer from a number of weaknesses which make them difficult to use in such a context:

- Most symbol recognition methods work with a known database of reference symbols, or with a learning phase. But it is easy to imagine browsing applications where the user delineates an arbitrary symbol in a drawing and queries for similar symbols in the database. In such a case, it is impossible to precompile all possible symbols, or to perform time expensive learning procedures.

- Many symbol recognition methods need a preliminary segmentation phase, where candidate regions for symbol recognition are extracted from the image (be it connected components, or subgraphs, or whatever is most suitable). Thus, we perform *segmentation followed by recognition*. In some cases, segmentation is easy—of course, character recognition comes to mind, but also other methods where connected components can be used as the basis for segmentation (even if we need some post-processing to disconnect parts touching the graphics [51]).

  Another relatively easy domain is that of schemas and diagrams consisting mainly of symbols and connection lines; it is not surprising that these are the application areas where the best results in symbol recognition have been achieved [22, 55].

  But in many other areas, it is difficult to segment before knowing the symbols, while it remains difficult to recognize the symbols with classical methods without prior segmentation! In such cases, we need to develop noise-resistant methods which are efficient (looking for subgraph isomorphisms on the complete vector set of a large drawing would definitely be the wrong idea!) and generic. Thus, there is a need for *segmentation-free symbol recognition* methods or for methods which *perform segmentation while they recognize*.

- A last area where efforts need to be made is the *scalability* of structural symbol recognition methods.

Presently, a number of efficient methods exist for discriminating a symbol among 10 or 20 different models, even if the symbol to be recognized is distorted by noise, other touching graphics, etc. But we don't really know how to scale these structural methods to the recognition of 50, 100 or 500 different symbols. There are both computational complexity issues, and open questions about the power of discrimination of the methods chosen for recognition.

We believe the use of *signatures* to be a key for some of the challenges just described. Signatures are indeed often used for indexing and retrieval purposes; however, most work has concentrated on text-based signatures [43] or image-based signatures [13]. We think that there is also room for graphics-specific signatures, to achieve an efficient localization and recognition of symbols. We currently work in two directions:

Quick and robust symbol localization through image-based signatures. The first idea is to compute signatures on the binary document image. The signatures must be invariant to rotation, translation and scaling. Our group's first work in this area was on $\mathcal{F}$-signatures, based on force histograms [46]. However, these signatures cannot robustly take occlusions into account, and the method still relies on some kind of segmentation. We are therefore investigating the use of the Radon transform, from which R-signatures have been proposed by Tabbone and Wendling [45]. To the contrary of other methods based on this transform [12, 21, 28], the proposed method takes into account the inner part of the symbol, not just its contour. This makes it more robust in the presence of partial occlusions or of connections with other graphical layers.

A theoretical study is currently being made on the shape measurements which can be obtained through the R-signatures. The way the signatures are built should also make it possible to consider spatial relationships between the symbols recognized in a document, in order to compute a relational graph which could be used in later semantical analysis stages.

Vector-based signatures. It is not always necessary to work directly on the raw image data. Vectorization can yield a set of vector data; in many cases, we also actually directly retrieve vector data from CAD files or similar electronic representations. It is therefore interesting to also use signatures computed directly on these vector data.

Such signatures can be computed from various features, in the same way as Huang uses various graphical primitives to index a set of images [20]. One of the few works we are aware of for vectorial data is that of Della Ventura and Scettini [53], whose results are interesting, although they use very constrained, well-segmented CAD data.

In collaboration with the University of Barcelona, our group is currently investigating possible extensions of these ideas to the general framework of technical documents [16]. The approach builds on all the spatial relations which can be found between segments, following an idea initially used in computer vision by Etemadi et al. [17]. The main relations used are parallelism, connectivity and overlapping.

Ultimately, our idea is that signatures can be used for *symbol spotting* and identification of broad hypotheses. This will probably be enough for good retrieval performances, in many cases. If better recognition rates are needed, the preliminary use of signatures can eliminate a number of symbol hypotheses and help segmenting out the candidate region; it can then be followed by classical symbol recognition methods.

In the presence of a large number of symbols, both signatures and structural recognition methods may not be sufficient to discriminate; they could then be used as pre-classification steps, followed by recognition through usual classification methods within the family identified by the signature.

4. Intelligent Browsing of Graphical Documents

The previously described signature methods contribute, in a sense, to the main goal of document analysis, which is to extract the right level of "meaning" from the document. Be it recognition, indexing or information retrieval, the underlying issue systematically boils down to localization and identification of *objects*. However, once we start talking about the meaning of a document, with the aim of manipulating the underlying semantic concepts, we are implicitly at the crossroads between graphics recognition, information retrieval and knowledge representation. Currently, we only try to focus on the aspects that are related to graphics recognition; further work will necessarily and progressively integrate models of the other domains. This is what we mean by *Intelligent Browsing of Graphical Documents*: given a set of unstructured documents (scans, PDF, DXF…), what are the most appropriate means of finding information in a reliable and computationally acceptable manner?

The main efforts during the last decade or so have been directed toward "indexing" [3, 13, 54] (of which the mentioned signature methods form a direct spinoff). The word indexing itself has to be used with care, since it bears quite different meanings, according to the research community. Let us stick to the rather restrictive and computational meaning it has in graphics recognition: the use of discriminant low dimension metrics for rapid identification of patterns.

Numerous indexing methods exist, with fair recognition rates. However, they lack the generic properties required by a real semantic-oriented information retrieval task. The main limitation of these approaches is that the application domain of the metrics they use is very context-sensitive, where the context is to be related to the type of the considered documents. Once we want to extend these methods to a broader context, and therefore richer semantics, we often observe that other metrics and other classifiers are better suited to the particular subdomains of the extended context. This is the main reason why one of the current research trends is to shift efforts towards operator selection, multi-source analysis and classifier combination [1, 14, 25, 36].

The main limitation to generalization is the fact that these indexing methods are based on the same underlying scheme: context + syntax + metric = semantics. In other words, information is encoded as the distance of a measurement to the signature of an identified object, given a reasonable number of constraints and prerequisites. While the signature and metric are usually thoroughly studied and documented, the context is most often embedded in the selection of the test images, the application domain and the used algorithms themselves [26]. When dealing with the problem of retrieving semantics from a set of random documents, the previous relationship proves to be non reversible: Given a semantic concept, one may list a number of possible signatures and associated metrics, but there is no way one can decide whether the contextual prerequisites are met (unless, of course, all documents in the set share the same context).

4.1. Combining Text and Graphics

The most obvious and straightforward step in the semantic labeling of graphics consists in associating textual information with the graphics. We have conducted our first experiences in this direction by analyzing cutaway diagrams [27, 52]. The main reason for choosing this kind of images is that they combine complex graphics (making the problem difficult from a computational point of view) with an easy to define macroscopic structure, and therefore simple semantics. The document contains three main classes of objects: *drawing*, *index* and *legend item*. Elements of each class are related to elements of both other classes: a part of a drawing is identified by its index, which in turn is referenced in the legend. The current state of our research is limited to identifying these relationships. One of the immediate extensions could be to associate OCR results on the legend items with the graphical parts, thus leading to direct referencing of the graphics through keywords, without ever having defined visual constraints on the representation of these graphics. The collaboration between text and graphics is a domain that can be further applied to various problems, as a bootstrap for the (semi-)automated annotation of graphical documents [31, 35].

Of course, these approaches still implement the previously mentioned context + syntax + metric = semantics paradigm, but maybe in a more subtle way. Especially where the context is concerned, one can identify many limitations (shape and structure of the legend, connection between indexes and graphic components, connectivity of graphic components, etc.) that obviously make this initial work a hinder for full generalization. As we already mentioned in [26], the main drawback is not so much the quality of the image, or the appropriateness of the used algorithms, but the lack of formalization of the context in which the approach remains valid!

4.2. Towards a Functional Semantic Framework

The context is actually the difficult part of the problem. It cannot be formalized as being a part of the document, since it includes the usage of the document, among other items. Different usages define different contexts, and in that sense it is related to ontology. On the other hand, one has to admit that a fair number of approaches are coming of age and find numerous industrial applications. They are robust, reliable and fast, notwithstanding the fact that the context + syntax + metric = semantics paradigm in which they have been designed generally limits their use to a well defined environment.

Our idea for trying to solve the issue of encoding context comes from the observation that document analysis is conducted as a stacked pipeline taking syntax as an input, and producing semantics as an output on each layer [26]. Moreover, we introduced the concept of Component Algebra, where syntax and semantics are simply associated to data, and operators are associated to elementary components transforming data of one type into another.

We are currently considering a particular representation of this algebra in the form of a single origin DAG (Directed Acyclic Graph). Each node of the graph represents data, each vertex, the application of a component to the data. The initial document can be considered as the origin of the DAG. Then, given a finite number of components, the fully developed graph of all possible successive applications of the components to the initial and constructed data, represents the intrinsic semantics of the document, the context of which is embedded in the selection of the applicable components. Let us define this more formally:

Components are algorithms (or rather implementations of algorithms) that operate on data. Input data have a certain type, or in other words, belong to a specific application domain. The result of the operation, or output data, belongs to a different domain.

$$\begin{array}{rccc} C: & \mathbb{D} & \rightarrow & \mathbb{D}' \\ & d & \mapsto & C(d) \end{array}$$

Application Context is the set $\mathcal{C}$ of all available components.

Semantics are defined by $\mathbb{D}'$. Both $\mathbb{D}$ and $\mathbb{D}'$ are actually divided into different sub-domains. For instance, a component designed for text labels to legend entries (as used in [52]) would take as input a list of image patches A and a list of legend entries B and would return a list of relations between both lists Z with their associated confidence Y. In this case $\mathbb{D} = A \times B$ and $\mathbb{D}' = Z \times Y$.

More generally, the application domain $\mathbb{D}$ and its image $\mathbb{D}'$ encode the semantics respectively needed and produced by the component.

Intrinsic Semantics of an initial document $\mathcal{I}$, within the context $\mathcal{C}$, is the set of all data that can be extracted from $\mathcal{I}$, using the components in $\mathcal{C}$.

It is important to note that the combinatorial possibilities of successively applying different components to the same document are limited. A given component C can only be used if the data it needs are available, and consequently after another component C' has produced it. Without going into too many details, we can partially order the components in a hierarchical manner with respect to this chronological order, and reasonably prove that the intrinsic semantics are finite when $\mathcal{C}$ is finite. This results in a graph-like structure.

This model has several advantages that can be exploited:

- Each node of the graph has a clearly identified semantic meaning. Furthermore, the relationships between different concepts are explicitly represented by the graph structure itself.

- The structure can be used as a support for automatic reasoning or knowledge inference, or even for cross-verification and confidence measurements.

- There is no need for explicitly taking into account context, since the context is embedded in the components themselves.

- Identical semantic concepts may be associated with different positions in the graph.

- Unlike full semantic document re-engineering, the graph structure can be modeled to produce only the information that is appropriate for a given application. Since the structure is completely open and flexible, components can be added or removed at any moment, thus enhancing the semantic power as needed.

4.3. Application Test Case

An interesting application would be semantic browsing in a loosely structured document corpus. Suppose we attach, to each document, its history with respect to the components it has been through, so that it can be correctly positioned within the graph representation. To that extent the notion of a document would be its initial state (e.g. a binary image) together with all intermediate data generated by subsequent applications of different components. Queries looking for documents containing certain characteristics could then be expanded to shortest path searches through the graph, and even be extended to the automatic application of appropriate components to partially processed documents. This would be the first prototype of a fully semantical browsing application (restricted to the implicitly defined context of the underlying component algebra).

Let us for instance reconsider the cutaway diagrams in [27, 52], and add the fact that we have a large corpus of different documents at our disposal. Different documents, coming from different types of applications, are stored together, and we would like to retrieve *"graphical representations of X"* where X is some known semantic concept (*e.g. "screw bolt ZX48"*). Then, the system could search for documents that have been labeled with X (in other words, documents that have undergone previous processing, and that have been found to contain the considered item). By analyzing the path from the original state of the document leading to X, we can reconstruct the component pipeline that allows for the extraction of our concept, as well as the intermediate semantics that are necessary for its construction (*e.g. text/graphics separation, legend localization, index/legend matching, legend analysis, string matching with "screw bolt ZX48", etc*). The next step would then be to backtrack the pipeline, and find suitable other documents that contain these intermediate semantics (i.e. have successfully undergone parts of the component pipeline) and try to apply the missing components, in order to decide if they are likely to contain X.

4.4. Open Issues

The proposed model still needs to be thoroughly studied and validated, and a number of unsolved obstacles need to be lifted before it can evolve into a truly generic semantic framework:

- Parameters are the first item. Components are usually parametrized to meet their goals. This parametrization is most often context related, and allows for a certain flexibility of use. The way we have modeled the process, parameters belong to the input domain $\mathbb{D}$, which makes them in a sense part of the pre-required semantics. How to correctly distinguish between effective semantics and context parameters is a difficult issue.

- More generally, context itself has to be taken into account if we want this approach to be used for knowledge inference. The graph structure itself does not encode information about the appropriateness of one path with respect to another, to achieve a final goal position. In other words, if different component threads lead to the same semantic concept, the context of the initial data may infer the inadequacy of certain components (designed for different contexts) thus making certain paths through the graph ineffective. This information cannot be deduced without representing the context in a more formal way.

- One possible solution to the quantification of the appropriateness of a component to a given set of input data may be the use of global error, uncertainty or confidence measures. The application context of a component can then be defined as the set of input data for which the confidence measure on the produced output falls within qualified bounds. This, of course, is easier said than done.

- From a more pragmatic point of view, work has to be done on the constraints on constructing components and representing documents, especially if we want to achieve an operational prototype as described in section 4.3.

5. Conclusion and perspectives

A lot of know-how in graphics recognition can be directly used for indexing, browsing and retrieval purposes. But there are also a number of new challenges when it comes to dealing with large databases of technical documentation. We have shown some of them, especially within the area of symbol spotting and recognition.

We have also sketched a plausible framework for establishing semantics oriented document analysis. This framework tries to be as flexible as possible by building upon existing and proved operational bricks. By considering document analysis as a pipelined application of individual components, and explicitly modeling input and output with respect to document semantics, we introduce a DAG as the underlying structure of knowledge inference. The ultimate goal of this work, beyond the resolution of the listed open problems in section 4.4, is to achieve a merge between graphical document analysis, and ongoing work on the semantic web and its associated tools.

References

[1] S. Adam, J. M. Ogier, C. Cariou, R. Mullot, J. Labiche, and J. Gardes. Symbol and character recognition: application to engineering drawings. *International Journal on Document Analysis and Recognition*, 3(2):89–101, Dec. 2000.

[2] C. Ah-Soon and K. Tombre. Architectural Symbol Recognition Using a Network of Constraints. *Pattern Recognition Letters*, 22(2):231–248, Feb. 2001.

[3] E. Appiani, F. Cesarini, A. M. Colla, M. Diligenti, M. Gori, S. Marinai, and G. Soda. Automatic document classification and indexing in high-volume applications. *International Journal on Document Analysis and Recognition*, 4(2):69–83, Dec. 2001.

[4] J. F. Arias, A. K. Chhabra, and V. Misra. A Practical Application of Graphics Recognition: Helping with the Extraction of Information from Telephone Company Drawings. In Tombre and Chhabra [49], pages 314–321.

[5] L. S. Baum, J. H. Boose, and R. J. Kelley. Graphics Recognition for a Large-Scale Airplane Information System. In Tombre and Chhabra [49], pages 291–301.

[6] L. Boatto, V. Consorti, M. D. Buono, S. D. Zenzo, V. Eramo, A. Esposito, F. Melcarne, M. Meucci, A. Morelli, M. Mosciatti, S. Scarci, and M. Tucci. An Interpretation System for Land Register Maps. *IEEE COMPUTER Magazine*, 25(7):25–33, July 1992.

[7] R. Brunelli, O. Mich, and C. M. Modena. A Survey on the Automatic Indexing of Video Data. *Journal of Visual Communication and Image Representation*, 10:78–112, 1999.

[8] F. Cesarini, S. Marinai, and G. Soda. Retrieval by Layout Similarity of Documents Represented with MXY Trees. In D. Lopresti, J. Hu, and R. Kashi, editors, *Proceedings of 5th IAPR International Workshop on Document Analysis Systems, Princeton (NJ, USA)*, volume 2423 of *Lecture Notes in Computer Science*, pages 353–364. Springer-Verlag, Aug. 2002.

[9] K.-F. Chan and D.-Y. Yeung. Mathematical expression recognition: a survey. *International Journal on Document Analysis and Recognition*, 3(1):3–15, Aug. 2000.

[10] J. Y. Chiang, S. C. Tue, and Y. C. Leu. A New Algorithm for Line Image Vectorization. *Pattern Recognition*, 31(10):1541–1549, Oct. 1998.

[11] L. P. Cordella and M. Vento. Symbol recognition in documents: a collection of techniques? *International Journal on Document Analysis and Recognition*, 3(2):73–88, Dec. 2000.

[12] S. R. Deans. *Applications of the Radon Transform*. Wiley Interscience Publications, New York, 1983.

[13] D. Doermann. The Indexing and Retrieval of Document Images: A Survey. *Computer Vision and Image Understanding*, 70(3):287–298, 1998.

[14] D. Doermann, E. Rivlin, and I. Weiss. Applying algebraic and differential invariants for logo recognition. *Machine Vision and Applications*, 9(2):73–86, 1996.

[15] D. Dori and W. Liu. Sparse Pixel Vectorization: An Algorithm and Its Performance Evaluation. *IEEE Transactions on PAMI*, 21(3):202–215, Mar. 1999.

[16] P. Dosch and J. Lladós. Vectorial Signatures for Symbol Discrimination. In *Proceedings of 5th IAPR International Workshop on Graphics Recognition, Barcelona (Spain)*, July 2003. To appear.

[17] A. Etemadi, J.-P. Schmidt, G. Matas, J. Illingworth, and J. Kittler. Low-level Grouping of Straight Line Segments. In *Proceedings of Second British Machine Vision Conference, Glasgow, Scotland*, pages 118–126, 1991.

[18] P. Fränti, A. Mednonogov, V. Kyrki, and H. Kälviäinen. Content-based matching of line-drawing images using the Hough transform. *International Journal on Document Analysis and Recognition*, 3(2):117–124, Dec. 2000.

[19] X. Hilaire and K. Tombre. Improving the Accuracy of Skeleton-Based Vectorization. In D. Blostein and Y.-B. Kwon, editors, *Graphics Recognition – Algorithms and Applications*, volume 2390 of *Lecture Notes in Computer Science*, pages 273–288. Springer-Verlag, 2002.

[20] P. Huang. Indexing Picture by Key Objects for Large-Scale Image Databases. *Pattern Recognition*, 30(7):1229–1237, 1997.

[21] A. Kedyrov and M. Petrou. The Trace Transform and Its Application. *IEEE Transactions on PAMI*, 23(8):811–828, Aug. 2001.

[22] S. H. Kim and J. H. Kim. Automatic Input of Logic Diagrams by Recognizing Loop-Symbols and Rectilinear Connections. *International Journal of Pattern Recognition and Artificial Intelligence*, 8(5):1113–1129, 1994.

[23] K. Kise, M. Tsujino, and K. Matsumoto. Spotting Where to Read on Pages — Retrieval of Relevant Parts from Page Images. In D. Lopresti, J. Hu, and R. Kashi, editors, *Proceedings of 5th IAPR International Workshop on Document Analysis Systems, Princeton (NJ, USA)*, volume 2423 of *Lecture Notes in Computer Science*, pages 388–399. Springer-Verlag, Aug. 2002.

[24] B. Klein, S. Gökkus, T. Kieninger, and A. Dengel. Three Approaches to "Industrial" Table Spotting. In *Proceedings of 6th International Conference on Document Analysis and Recognition, Seattle (USA)*, pages 513–517, Sept. 2001.

[25] L. Lam, Y.-S. Huang, and C. Y. Suen. Combination of Multiple Classifier Decisions for Optical Character Recognition. In H. Bunke and P. S. P. Wang, editors, *Handbook of Character Recognition and Document Image Analysis*, chapter 3, pages 79–101. World Scientific, 1997.

[26] B. Lamiroy and L. Najman. Scan-to-XML: Using Software Component Algebra for Intelligent Document Generation. In D. Blostein and Y.-B. Kwon, editors, *Graphics Recognition – Algorithms and Applications*, volume 2390 of *Lecture Notes in Computer Science*, pages 211–221. Springer-Verlag, 2002.

[27] B. Lamiroy, L. Najman, R. Ehrard, C. Louis, F. Quélain, N. Rouyer, and N. Zeghache. Scan-to-XML for Vector Graphics: an Experimental Setup for Intelligent Browsable Document Generation. In *Proceedings of 4th IAPR International Workshop on Graphics Recognition, Kingston, Ontario (Canada)*, pages 312–325, Sept. 2001.

[28] V. Leavers. Use of the Two-Dimensional Radon Transform to Generate a Taxonomy of Shape for the Characterization of Abrasive Powder Particles. *IEEE Transactions on PAMI*, 22(12):1411–1423, Dec. 2000.

[29] J. Lladós, J. López-Krahe, and E. Martí. A System to Understand Hand-Drawn Floor Plans Using Subgraph Isomorphism and Hough Transform. *Machine Vision and Applications*, 10(3):150–158, 1997.

[30] J. Lladós, E. Valveny, G. Sánchez, and E. Martí. Symbol Recognition: Current Advances and Perspectives. In D. Blostein and Y.-B. Kwon, editors, *Graphics Recognition – Algorithms and Applications*, volume 2390 of *Lecture Notes in Computer Science*, pages 104–127. Springer-Verlag, 2002.

[31] N. Malandain. *La relation texte/image – Essai de modélisation dans un corpus géographique*. Thèse de doctorat, Université de Caen – Basse Normandie, Dec. 2000.

[32] S. Müller and G. Rigoll. Searching an Engineering Drawings Database for User-specified Shapes. In *Proceedings of 5th International Conference on Document Analysis and Recognition, Bangalore (India)*, pages 697–700, Sept. 1999.

[33] S. Müller and G. Rigoll. Engineering Drawing Database Retrieval Using Statistical Pattern Spotting Techniques. In A. K. Chhabra and D. Dori, editors, *Graphics Recognition—Recent Advances*, volume 1941 of *Lecture Notes in Computer Science*, pages 246–255. Springer-Verlag, 2000.

[34] L. Najman, O. Gibot, and S. Berche. Indexing Technical Drawings using Title Block Structure Recognition. In *Proceedings of 6th International Conference on Document Analysis and Recognition, Seattle (USA)*, pages 587–591, Sept. 2001.

[35] M. Nishijima and T. Watanabe. An Automatic Extraction Approach of Road Information on the Basis of Recognition of Character Regions. In *Image Analysis Applications and Computer Graphics — Third International Computer Science Conference, Hong Kong*, volume 1024 of *Lecture Notes in Computer Science*, pages 173–180. Springer-Verlag, Dec. 1995.

[36] A. F. R. Rahman, H. Alam, and M. C. Fairhurst. Multiple Classifier Combination for Character Recognition: Revisiting the Majority Voting System and Its Variations. In D. Lopresti, J. Hu, and R. Kashi, editors, *Proceedings of 5th IAPR International Workshop on Document Analysis Systems, Princeton (NJ, USA)*, volume 2423 of *Lecture Notes in Computer Science*, pages 167–178. Springer-Verlag, Aug. 2002.

[37] M. Robinson, L. S. Baum, J. H. Boose, D. B. Shema, and S. C. Chew. Case Study: Boeing Intelligent Graphics for Airplane Operations and Maintenance. In *Conference XML'98*, Chicago, USA, Dec. 1998.

[38] H. Samet and A. Soffer. MARCO: MAp Retrieval by COntent. *IEEE Transactions on PAMI*, 18(8):783–798, Aug. 1996.

[39] H. Samet and A. Soffer. MAGELLAN: Map Acquisition of GEographic Labels by Legend ANalysis. *International Journal on Document Analysis and Recognition*, 1(2):89–101, June 1998.

[40] C. Shin, D. Doermann, and A. Rosenfeld. Classification of document pages using structure-based features. *International Journal on Document Analysis and Recognition*, 3(4):232–247, May 2001.

[41] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on PAMI*, 22(12):1349–1380, Dec. 2000.

[42] J. Song, F. Su, C.-L. Tai, and S. Cai. An Object-Oriented Progressive-Simplification Based Vectorization System for Engineering Drawings: Model, Algorithm, and Performance. *IEEE Transactions on PAMI*, 24(8):1048–1060, Aug. 2002.

[43] A. L. Spitz. Shape-based word recognition. *International Journal on Document Analysis and Recognition*, 1(4):178–190, May 1999.

[44] T. Syeda-Mahmood. Indexing of Technical Line Drawing Databases. *IEEE Transactions on PAMI*, 21(8):737–751, Aug. 1999.

[45] S. Tabbone and L. Wendling. Technical Symbols Recognition Using the Two-dimensional Radon Transform. In *Proceedings of the 16th International Conference on Pattern Recognition, Québec (Canada)*, volume 2, pages 200–203, Aug. 2002.

[46] S. Tabbone, L. Wendling, and K. Tombre. Indexing of Technical Line Drawings Based on F-Signatures. In *Proceedings of 6th International Conference on Document Analysis and Recognition, Seattle (USA)*, pages 1220–1224, Sept. 2001.

[47] C. L. Tan, W. Huang, Z. Yu, and Y. Xu. Imaged Document Text Retrieval Without OCR. *IEEE Transactions on PAMI*, 24(6):838–844, June 2002.

[48] K. Tombre, C. Ah-Soon, P. Dosch, G. Masini, and S. Tabbone. Stable and Robust Vectorization: How to Make the Right Choices. In A. K. Chhabra and D. Dori, editors, *Graphics Recognition—Recent Advances*, volume 1941 of *Lecture Notes in Computer Science*, pages 3–18. Springer-Verlag, Sept. 2000.

[49] K. Tombre and A. K. Chhabra, editors. *Graphics Recognition—Algorithms and Systems*, volume 1389 of *Lecture Notes in Computer Science*. Springer-Verlag, Apr. 1998.

[50] K. Tombre and S. Tabbone. Vectorization in Graphics Recognition: To Thin or not to Thin. In *Proceedings of the 15th International Conference on Pattern Recognition, Barcelona (Spain)*, volume 2, pages 91–96, Sept. 2000.

[51] K. Tombre, S. Tabbone, L. Pélissier, B. Lamiroy, and P. Dosch. Text/Graphics Separation Revisited. In D. Lopresti, J. Hu, and R. Kashi, editors, *Proceedings of 5th IAPR International Workshop on Document Analysis Systems, Princeton (NJ, USA)*, volume 2423 of *Lecture Notes in Computer Science*, pages 200–211. Springer-Verlag, Aug. 2002.

[52] E. Valveny and B. Lamiroy. Scan-to-XML: Automatic Generation of Browsable Technical Documents. In *Proceedings of the 16th International Conference on Pattern Recognition, Québec (Canada)*, pages 188–191, Aug. 2002.

[53] A. D. Ventura and R. Schettini. Graphic Symbol Recognition using a Signature Technique. In *Proceedings of the 12th International Conference on Pattern Recognition, Jerusalem (Israel)*, volume 2, pages 533–535, 1994.

[54] I. Witten, A. Moffat, and T. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Van Nostrand Reinhold, New York, second edition, 1999.

[55] Y. Yu, A. Samal, and S. Seth. Isolating Symbols from Connection Lines in a Class of Engineering Drawings. *Pattern Recognition*, 27(3):391–404, 1994.

[56] Y. Yu, A. Samal, and S. C. Seth. A System for Recognizing a Large Class of Engineering Drawings. *IEEE Transactions on PAMI*, 19(8):868–890, Aug. 1997.