MVA '92 IAPR Workshop on Machine Vision Applications Dec. 7-9,1992, Tokyo

Technical Drawing Recognition and Understanding: From Pixels to Semantics

Karl Tombre
INRIA Lorraine – CRIN/CNRS
Campus scientifique, B.P. 239
54506 Vandœuvre CEDEX
France
Phone: +33 83.91.21.25
Fax: +33 83.41.30.79
E-mail: Karl.Tombre@loria.fr

ABSTRACT

When dealing with the analysis of technical documents such as engineering drawings, diagrams or maps, it is quite natural for people with a background in pattern recognition and computer vision to consider the problem as a special case of scene analysis. However, at closer look, draftsmanship is also a language, and this linguistic part must be taken into account in the interpretation process in order to achieve true understanding at a semantic level. This paper gives an overview of the methods currently used in technical document analysis by various research teams to achieve such high-level interpretation, and tries to show the newest trends and challenges in this field. We also propose some possible steps towards a methodology for designing in an ordered and efficient way specific interpretation systems for various applications.

1 INTRODUCTION

The use of information systems of various kinds in companies leads to the problem of converting the existing archives of paper documents into a format suitable for the computerized system. In this area, most of the attention has probably been given to structured document analysis, i.e. the automated analysis of business documents such as letters, forms, documentation, manuals etc. Beyond the well-known area of character recognition, the problems to deal with in such systems are to recognize the physical and logical structure of the document, to segment it into homogeneous blocks (text, graphics, pictures...) and to interface it with some office automation tool, comprising text and graphics editors. Despite its importance and interest, we will not address this domain of structured document analysis in this paper; we refer the interested reader to all the published articles on this topic in various pattern recognition conferences, in specialized conferences and workshops such as ICDAR [28], and recently in a book [3] and two special issues of journals [12, 55].

In the last years, we have been mostly working on the analysis of other document classes: technical drawings of various kinds, of which graphics are the most important component. Although at first glance these classes may appear to be quite similar to general structured documents, it quickly becomes obvious that a mere coding of the graphics using low-level primitives such as those used by graphics editors is far from being sufficient. For instance, the information systems to which the drawings have to be converted are CAD/CAM systems or Geographic Information Systems (GIS), which operate with high-level entities having a specific meaning in the related applications. It is therefore necessary to *understand* the document at the same level of abstraction (or semantics) as that of the host application.

This paper gives an overview of the methods currently used in technical document analysis to achieve such high-level interpretation, and tries to show the newest trends and challenges in this field. Although commercially available systems still have limited capabilities with respect to high-level understanding, we see the emergence of methodologies which can lead to very powerful interpretation systems.

2 SO WHAT IS THE PROBLEM?

The digitalization of a drawing creates an image of several million black and white pixels, representing with more or less accuracy the original drawing. The aim of a document analysis system is to locate and recognize in this image high-level entities having a meaning for the related application. This objective is very analogous to the general aim of computer vision; hence, many document analysis systems follow a path similar to that of vision systems: extraction of features (in our case vectorization), grouping of these features into higher-level structures, recognition of various objects by matching feature groups with models of known objects, contextual analysis of the whole scene, etc. Typically, a 1985 feasibility study of the conversion from paper to CAD only mentioned that there was a lot of work to do in structural analysis techniques to achieve such conversion [34].

But technical drawings have actually a twofold nature: they are both an *image*, in the usual meaning of the projection of a three-dimensional object on a plane, and a *language*, i.e. a way of communicating some precise information using specific signs. Trying to include all this linguistic information in a standard pattern recognition scheme may lead to thinking that there are too many problems to solve for real conversion to CAD models [27]. But in fact, this additional source of knowledge can be used to build systems which analyze technical documents at a much higher semantic level than what is currently available in "mainstream" computer vision, if only we use both the spatial (image) view of the document and the "linguistic" view. However, this additional linguistic information also leads to setting much higher goals for the document analysis process: the end user or the host application requires a level of understanding which cannot be reached by usual computer vision techniques. To illustrate this point, here are some typical needs we have met in various discussions with industry:

- **3D CAD conversion**: How to take a drawing, in mechanical engineering for instance, made of multiple views, and reconstruct a 3D CAD model complete with all the "semantic" attributes available in a modern CAD system. This implies among other features the ability to recognize larger entities which are stored as a whole in the CAD library (ball bearing with reference `356B25` from company XYZ, for instance).

- **Understanding of functionalities**: Take the schema of some old electrical or electronics circuitry and analyze its functionalities in order to be able to design a replacement circuitry with today's components. In this problem, it is not sufficient to recognize elementary symbols; expert knowledge must be added to the interpretation system to reach the functionality level (how does this system work, what functions does it perform?).

- **Paper-based maps to GIS**: Geographic Information Systems are used in various areas, such as urban management (cadastral maps), mining, road networks, geology, facilities (telephone, electricity or water distribution), agriculture, etc. Large amounts of maps of many kinds provide useful information; in addition, some applications require the combination of this map-based information with aerial photography or images taken from satellites. A map is a very rich and dense medium and for a given application, only a specific layer may be of interest; thus, the analysis must be able to extract this layer and convert it to information suitable for the host GIS.

- **Indexing large documentation databases**: Technical documentation in a company may include many million sheets of paper: technical specifications, user manuals for various devices, safety regulations, related contracts, financial information, manufacturing instructions, etc. This documentation comprises of course a lot of text, but also a large number of diagrams, synoptics, and both overview and detailed technical drawings. A multimedia documentation system which would be able to store all this information in electronic format and provide easy access to it through various indexing mechanisms would be a great commercial success. But this requires functionalities such as: "when browsing through a technical specification text, click on a word referring to part `AGXP-98` of the machine and bring up the detail drawing of this part", or even: "our company suffers large losses because of repeated breakdowns of part `AGXP-98`; find all other parts designed by our company which have the same type of mechanical setup, as it obviously must be changed".

These are examples for which it becomes obvious that technical drawing analysis is much more than vectorization and extraction of graphics primitives, as a real ***understanding*** of the document is required in each case. In the next section, we will review work going on in this area of technical document understanding. We will not elaborate on the large number of methods proposed for low-level feature extraction (vectorization, graphical primitives...); several excellent surveys on the state of the art in this area have been written, both for available commercial systems [71] and for tools and methods proposed by various research groups, including a complete and well documented survey presented at the previous MVA workshop [38, 39]. Methods have also been proposed for more specialized features of a technical drawing, such as dashed lines [43], circles and circular arcs [51, 56, 58], hatched areas [2, 6], etc.

But there is also a growing interest given to the analysis at advanced levels, beyond vectorization and basic features extraction, to achieve syntactic and even semantic analysis of the drawing. We will give a number of references to various research work, including our own; however, we do not claim to be exhaustive, but we will rather try to give the feeling of the main ideas put forward by different groups to attain high-level interpretation. For additional references, we refer to Kasturi and O'Gorman's recent survey of document analysis techniques [37].

After this review, we propose in § 4 some possible steps towards a methodology, not for achieving a universal system capable of analyzing *any* technical drawing, but rather for designing in an ordered and efficient way specific analysis systems for various applications.

3 STATE OF THE ART IN TECHNICAL DRAWING UNDERSTANDING

The problems given as examples in the previous section actually illustrate 4 main classes of technical documents [4]:

- *Orthogonal projections* are technical drawings which represent planar views of an object. The image is made of a set of lines and symbols, with different thicknesses for the lines in some cases. One part of these lines (usually the thick lines) represents the projection on a plane of the contours of an object's section. This part is typically an "image" part in the usual meaning of computer vision. The other part, often made by the thin lines and the symbols (characters, special annotations...), is much more "linguistic" or symbolic, as it conveys the additional information necessary for full understanding of the drawing: dot-dashed lines representing symmetry axes, dashed lines indicating contours hidden with respect to the section plane, hatching lines symbolizing the presence of matter in the section plane, dimensioning sets comprising additional thin lines, annotations and arrowheads, references to the nomenclature, etc.

- *Schemas and diagrams* represent in a symbolic way electric circuits, printed board electronics wiring, the control flow of a program (flowcharts), the hierarchy in a company, etc. They rarely aim at reproducing the visual aspect of real objects but are rather a convenient way to represent the working principle of some device, program or organization. Their main components are usually a set of symbols having a precise meaning, links between these symbols represented by lines, and attributes given to the symbols and to the lines by text annotation or other symbols.
- *Maps and charts* represent cities, countries, regions... at various levels of detail and with stress laid on different kinds of information, depending on the purpose of the map. Such maps usually contain several *information layers*: road network, facilities (electricity distribution, water supplies...), topographic information (elevation data), color codings for areas of different kinds (agriculture, geology, meteorology...), rivers, annotations (names of cities, of rivers, of streets, attributes of facilities...), etc. For a given application, only a subset of these layers may be of interest, which leads to the additional problem of extracting the right layers from the map, where they are all superimposed.
- *Technical documentation* is actually part of the large family of structured composite documents; the typical drawing understanding problems necessary for efficient indexing are related to the graphics parts of this documentation. These graphics are of one of the three previous categories. The additional feature of technical documentation is the presence of large bodies of text, which may be analyzed at the language level to extract cross reference indexes between the textual part and the drawing part. As far as drawing understanding is involved, however, the problems are those of analyzing the graphics; hence, we will not deal specifically with this category in the following survey.

This great variety of application domains leads to a large number of specific interpretation systems, which are not always easy to compare to one another. Nevertheless, we will try to give the main ideas beyond the various systems developed by different research groups.

3.1 Orthogonal projections

The category of orthogonal projections is mainly that of engineering drawings. Many methods have been applied to the interpretation of such drawings, although most of them remain dedicated to low-level processing, i.e. vectorization and graphics conversion, without true CAD conversion at a semantic level. That is maybe one of the reasons why commercial paper–to–CAD systems have not had the expected success: as low-level processing can hardly be made perfect (problem well known to the computer vision community), companies end up spending large amounts of money on vectorization software packages only to have to employ people to correct the errors made by the vectorization and add the lacking semantic attributes. It is understandable that if someone must edit the vector representation yielded by the software to interactively correct these graphics and then group them to decide that "this is the gearbox referenced `GHUKH67`", it may be a better and more economic idea for the company to forget about the whole vectorization process and have the same employee directly input the drawing again using the CAD system, which allows to add gearbox `GHUKH67` (contained in the CAD library) in approximately 2 minutes! Nevertheless, we claim that knowledge-based techniques, applied to the specific application, and taking into account both pattern recognition methods on the image part and semantic analysis, are able to recognize gearbox `GHUKH67` and to replace it by the corresponding entity taken from the CAD library, even if the results of the vectorization are distorted with respect to the original drawing.

Cappellini *et al.* [7] propose a system which identifies primitives on a hand-drawn drafting. The basic idea is to consider entities in engineering drawings as special symbols and to recognize them by a hybrid approach combining graph matching and classification. However, the level of semantics reachable by such a system remains quite low, as higher-level entities are seldom storable as model symbols!

Lu and Ohsawa [48] use a knowledge based system which vectorizes the drawing by matching opposite line borders, and recognizes various entities specific to technical drawings, especially the components of dimension sets, such as arrowheads. Once again, there is no higher-level analysis of the drawing.

Dimensioning is actually a typical example of the symbolic information conveyed by an engineering drawing. Dimensions follow strict standards, are complete and provide additional information which can be used to check the validity of the drawing [66] or to correct the errors introduced by digitizing and vectorization. Dov Dori has shown that the dimensioning language of engineering drawings can be described by a grammar [16, 15] and proposes hence a syntactical approach to the analysis of dimensions [13, 14]. In order to perform this syntactical analysis, however, the dimensioning layer must be extracted from the drawing, which is not completely trivial, as it consists of a subset of the thin lines, a subset of the textual annotations, and smaller templates such as arrowheads, which must be detected on the image before they are disturbed by vectorization [2]. In our group, Suzanne Collin implemented successfully this dimensioning layer extraction and subsequent syntactical analysis [11, 8, 9].

These works on dimensioning go one step of abstraction higher than the simple pattern recognition methods cited previously. Actually, they allow for the complete analysis of one single *layer* in the drawing. But a drawing is made of the superimposition of several such layers, the most typical example being that of the text superimposed on the graphics. As a general rule is that this text should not intersect the graphics, and that simple "fonts" are used, it is relatively easy to separate text from graphics by simple analysis of the connected components of the binary image, and to aggregate the small connected components considered as text parts into character strings. Several text–graphics segmentation methods adapted to technical documents have thus been proposed [21, 10], so that text parts can be processed apart, usually by some character recognition system.

But this text–graphics separation remains quite crude:

- The determination of the different thresholds necessary for this segmentation is often more or less subjective (maximum size of a character, minimum number of characters in a string, maximum distance between two characters in the same string, etc.).
- As the segmentation procedures do not use *a priori* knowledge about the meaning of the extracted strings, they tend to be more or less "blind" with respect to the difference between a string of hyphens and a dotted line, or between a small graphics symbol and an isolated character.
- Although the rules state that text should not intersect the graphics, reality is that this often occurs. Thus, text–graphics segmentation will find the characters of a string which are connected components on their own, but will miss those which touch a line. Retrieval of the whole string then requires some non-trivial post-processing phase [31].
- In fact, even if we are able to extract all text strings in a drawing, we have just recognized a *physical* layer; it does not necessarily correspond to a *logical* layer. For instance, some parts of the text may make up the legend layer, whereas other strings should be associated with a part of the thin lines to make up the dimensioning layer, and other characters again may be labels of section planes.

The physical layers can be extracted using typical pattern recognition filters; in this way we can separate the large connected components from the small ones, differentiate thick and thin lines, extract the hatching layer, recognize the *inclusion* relation between two parts of the drawing, or find dotted and dot-dashed lines [43]. But the logical layers can only be found using higher-level knowledge, which allows one to analyze the dimensioning layer or the legend, for instance.

In fact, the decomposition of a drawing into layers can be seen as yet another application of syntactical interpretation, where the only composition rule is that of superimposition. But true understanding of the drawing requires to get to some kind of semantics analysis. Few methods have been proposed to achieve this.

The ANON system developed at the University of Sheffield [33, 32] is based on a structural description of engineering drawings, using frames to represent components such as lines, curves, dimension-sets, etc. and the relations between these components. The interpretation itself follows strategy rules written in the `yacc` syntax; the parsing allows the recognition of entities such as dimensions or broken lines. Even if this remains very close to a syntactical approach, the representation of the *a priori* knowledge using frames yields more abstraction power than what is available through a flat set of grammar rules.

Our group has also been interested in this area. The CELESSTIN system [67, 68] is an integrated prototype which performs interpretation of mechanical engineering drawings using a blackboard-based multi-expert system. The first versions of this system were essentially based on structure and syntax to recognize entities such as shafts, screws, ball bearings or gears on a single view of a mechanical device. But in the last version, CELESSTIN IV, we experimented with knowledge rules relative to the semantics, i.e. to the functionalities of the represented object and not only the representation rules. Thus, we designed two experts, one focusing on *disassembling*, based on the assumption that it *must* be possible to disassemble a mechanical setup, the other on the *kinematics* of the whole setup, as it determines the functionalities of various entities from their behavior when a rotation motion is applied around the identified axes in the drawing [69]. Although we are aware that our prototype is far from covering all possible functional interpretations, even in the restricted area of mechanical engineering, we believe that our work suggests a possible methodology for building high-level document interpretation systems, as we will elaborate on in § 4.

One of the ways to conduct still more complex reasoning processes than those proposed in CELESSTIN is to reason on a 3D model of the object and not on a single 2D view. For instance, this would enable to analyze kinematics or disassembly on the whole object, which is more reasonable in real cases than what our system performs, as it is often impossible, even for an engineer, to understand the functionalities of a drawing from a single view. This requires the ability to reconstruct 3D CAD models by matching several views of a drawing. Methods for performing this was already proposed in the beginning of the 80's [25]. Since then, many systems have been designed, building either a 3D B-rep [49, 50, 45, 60] or a CSG assembly [41, 65] by combining several views. But all these methods have two major weaknesses:

- They need *perfect* 2D views, without any distorsions or errors, to perform the matching. This may be possible with machine-generated drawings, but is certainly not with the result of a conversion from paper to electronic format. A possible way to counteract this is to correct the vectorization using the results of dimensioning analysis, as seen previously.

- The matching methods are purely geometric; thus, they work for the "image" part of the drawing but fail on the "linguistic" part. For instance, if a ball bearing is represented by a conventional symbol on one view, the lines making up this symbol will certainly not match with corresponding lines on another view! It is therefore necessary to analyze as much symbolics as possible on each single view and to combine the geometric matching techniques used so far with semantic information yielded by other sources of knowledge.

In conclusion, it is evident that there is still a lot of work to do to achieve really useful automated understanding of engineering drawings, even if one restricts oneself to a limited technical field. But I hope that I have made the challenges interesting enough to convince several people that it is possible and that there is a lot of exciting research to do in this area.

3.2 Schemas and Diagrams

Diagrams have been extensively studied by many groups. Most systems recognize electrical or electronics schemes by using simple knowledge about the representation rules of such documents (a set of possible symbols connected by lines). There are numerous symbol recognition techniques available, based either on classification methods [17, 47], on purely structural attributed graph matching [24, 23, 42] or on hybrid pattern recognition methods [44]. Some of these systems aim not only at recognizing each individual symbol, but also at analyzing the whole diagram. Fahn et al. [19] apply syntactical analysis to the understanding of electronic circuit diagrams. Shimotsuji et al. [62] and Kato et al. [40] use different layers of knowledge to recognize hand-drawn schemes. Futatsumata et al. [22] use a classification-based method to analyze plant diagrams.

Actually, one of the possible uses of contextual knowledge in such systems is the delimitation of candidate areas in the drawing where a symbol has to be looked for. Most existing systems are based on very rudimentary heuristics, like the supposed size of a symbol, the presence of small white loops in it [57], or the simple fact that everything which is not a long line segment is supposed to belong to a symbol [47]. For specific applications, it might be possible to use much more true semantics, such as hypotheses about the expected symbols connected to a symbol already recognized or the general consistency of the represented circuit.

But few systems try to reach this *functionality* level, i.e. to understand how the represented circuitry works. Only limited experiences have been done in that field. Murase and Wakahara [54] analyze flowcharts and logic circuit diagrams. Simple semantic rules are used to recover from erroneous symbol recognition; for instance, basic knowledge about the "meaning" of different flowchart symbols leads to rejection of some inconsistent combinations, such as a *terminal* symbol located at a branch of a process flow. These inconsistencies are defined from knowledge about valid and invalid algorithms. However, these rules remain essentially syntactical.

Benjamin *et al.* use several levels of knowledge in their system for interpretation of telephone outside plant drawings for Bell Canada [5]: syntactical and structural rules describe the various kinds of symbols and relations between them, whereas higher-level rules describe the "meaning" of these symbols and the consistencies which have to exist between the corresponding entities in the real-world application.

It may actually be too ambitious to hope for much higher level interpretations in this class of documents: whereas a mechanical engineer or an architect is able to *understand* a drawing made of orthogonal projections, it is not sure that an electrical engineer is able to tell what a complex electronics circuit does by just looking at its schema! When human expertise fails, can we expect the computer to do better? But there are certainly still new results to get out of this area, by basing oneself on all available knowledge about the way humans understand diagrams [52].

3.3 Maps and charts

The analysis of maps is often aimed at converting part of the information they contain to some Geographic Information System (GIS). We already mentioned the importance of layer segmentation for orthogonal projections; this becomes crucial in the case of cartography, as maps usually contain many superimposed layers, while the host application only requires the information contained in one or two of them. Examples of layers which may be of interest are road networks [35, 26, 29, 30, 36], topographical information such as elevation curves [72] or utilities distribution [70]. In all these examples, the layer of interest is essentially composed of a set of attributed lines and pattern recognition techniques can be used. A typical example of structural analysis is the extraction of drainage vectors from elevation data, using clustering and linking of vector chains extracted from the elevation data map [59].

It is also possible to use other computer-vision related techniques for some applications; for instance, when color is used to code different regions of interest, region segmentation techniques are very efficient to extract the corresponding layers [18]. Another example where several teams have contributed is the matching between aerial or satellite imagery and maps, with all the usual problems of modeling the sensors and the geometry of the observation, finding matching features [20, 46] and integrating all that in the GIS [53].

But in some applications, still higher level information must be retrieved; for instance, the analysis of cadastral information (city maps) may require a very good precision in the results of vectorization and a perfect recognition of all numbers of land parcels, as this information may be used in legal cases [6]. There has been quite a lot of activity precisely in this area of cadastral maps. To the contrary of engineering drawings, there are seldom two classes of line thicknesses, but it is nevertheless important to find the hatched areas, usually representing buildings, and hence to remove the hatching layer from the set of lines to be analyzed at later stages. In our group, Dominique Antoine designed a system based on procedural networks, which extracts the hatching layer first, then finds the parcels on which the buildings are located, and finally groups the parcels into blocks of properties bordered by streets [1, 2]. The Japanese system MARIS is also dedicated to such city maps; it is designed as a set of specialized procedures which cooperate in recognizing houses, streets, elevation curves, road lines and so on [63].

When it is crucial to recognize the hatching layer on the one hand and the complete textual annotations on the other hand (as the numbers of the parcels have a legal meaning), the problem of layer separation becomes very crucial, and it is rendered still more difficult by the fact that on such maps, text tends especially often to touch the graphic lines (usually the hatching). It is therefore not surprising that two of the teams having built the most elaborate systems for analysis of city maps have given special attention to the segmentation of text even when it touches the graphics, by fine-tuning their low-level processing steps, such as vectorization:

- Boatto *et al.* [6] have a system which stresses accuracy in the recognition of the borders between parcels. An intermediate coding of the lines yields a graph which is searched for sets of regularly spaces lines corresponding to hatching. Only the lines which do not belong to this hatching layer are vectorized and interactively corrected when necessary. Symbols touching the graphics are extracted by search for small characteristic subgraphs; characters and other symbols are then recognized by an OCR module. The higher-level analysis can then be performed by simple graph processing methods, with attributes given by the recognized text.

- Shimotsuji *et al.* [61] propose a similar system, but for Japanese maps containing information about electric power distribution. As in the previous method, the vectorization process uses an intermediate data structure, called in the present case *primitive lines*, which describe as exactly as possible the line fragments in the image. Symbols and characters are then recognized by grouping characteristic features in this structure, even if they touch the graphics. The different kinds of lines are also interpreted according to their semantics (cable lines, map line...).

The analysis of various maps to convert them into GIS format may be still much more challenging than that of CAD conversion for engineering drawings, considering the great variety of information which may be extracted from maps and the high level of symbolism used to represent this information. Conceptually, maps are actually not so far from engineering drawings as they both represent a projection of a scene on a plane. But when 3D information is present in maps, it is not represented by the projection from several viewpoints but in much more "linguistic" ways: elevation data curves, altitude written at several points of the map, etc. More generally, the ratio between image and language in a map tends to be much more in favor of language than what is the case in engineering drawings. Hence, it is obvious that real understanding can only be achieved if this linguistic interpretation is correctly taken into account. In addition, we have seen that maps are often a superimposition of many layers, which increases the difficulty of segmentation.

4 TOWARDS A METHODOLOGY?

As we have tried to stress throughout this article, draftsmanship is not only a geometric activity; it also a language. Hence technical document analysis systems aiming at a real understanding of the engineering drawing, the diagram or the map must take into account both its image part and its language part. This compels us to go beyond usual pattern recognition techniques. For instance, after vectorization of an engineering drawing, it is natural to try to connect vectors at junction points in order to extract longer lines; but we must take into account the fact that some vectors belong to the image part, that is to the line drawing representing the orthographic projection of some 3D surfaces, whereas other vectors belong to the language part, for example the hatching lines labeling an area as a section in matter.

The question which comes to mind is then: is there a general methodology allowing to combine these different kinds of knowledge, in order to go all the way from pixels to semantics in an ordered manner? We are far from claiming that we have a definite answer to this question, but our various experiences have led us into decomposing the reasoning process in document interpretation along two axes:

- The first axis corresponds to the spatial (or image) vision; it consists in grouping elementary features into higher-level ones. This grouping can be performed by simple structural matching or by some syntactical pattern recognition techniques.

- The second axis corresponds to the "reduction" of symbolic information (the language part) into new features, which can then be manipulated by new spatial grouping tools, and so on.

The whole reasoning process can then be seen as the progressive use of different levels of semantics to induce new structures, on which an appropriate syntax can be applied [64].

To illustrate this, Fig. 1 summarizes the different steps and levels of reasoning in our CELESSTIN system. On the first level, we have pixel processing, where a set of operations can be performed to clean the image, label connected components, and so on. Once we introduce the low-level knowledge that such drawings are mainly made of lines, we can go up to the second level, where vectorization provides a new basic structure, the vector. Different structural and syntactical rules can be applied on vectors, allowing for separating thin and thick lines, finding dot-dashed lines, etc. We next defined a new basic structure, the *block* [68], which comes from the fact that all closed minimal polygons in thick lines either represent matter or empty space. On these blocks, new structuring operations can be defined, such as a syntax to recognize entities [69]. We could have added still

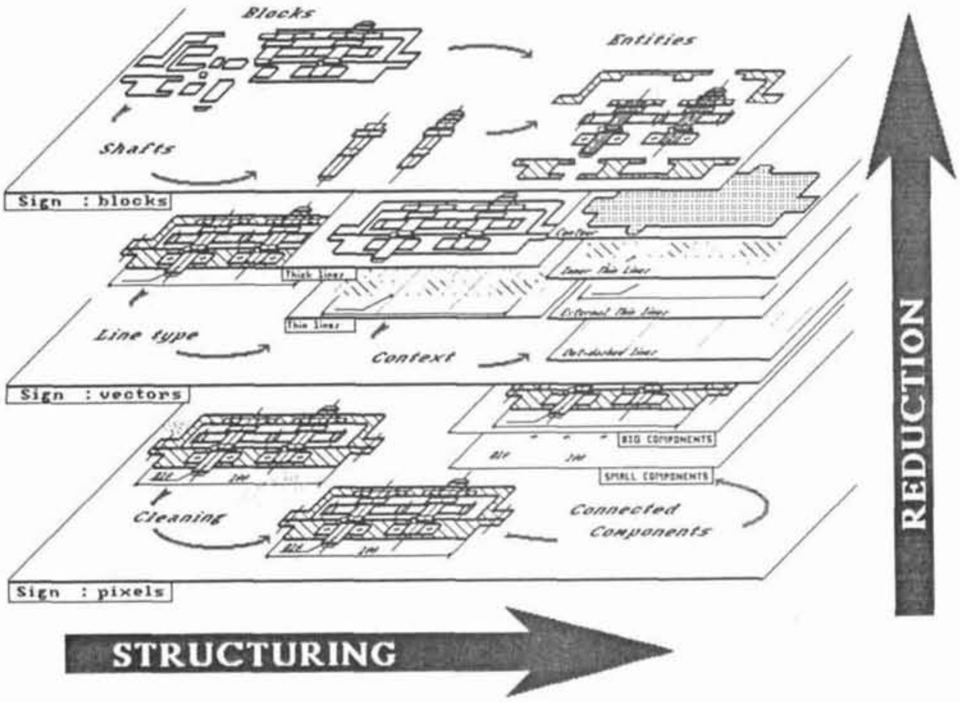

Figure 1: Spatial and symbolic reasoning in CELESSTIN (figure kindly provided by P. Vaxivière).

a higher level of information reduction, where we would have the reasoning about kinematics and disassembling.

However, it is not clear to us if this scheme is general enough to describe any reasoning conducted to analyze a technical document. Even if it is, another open question remains: for a given application with well defined interpretation needs, do we have a general methodology to define the right structures, the right syntax on these structures, and the appropriate symbolics to use from one level to the other? I welcome any answer, even partial, to these questions...

ACKNOWLEDGMENTS

In the past years, several people have been working with me on various aspects of technical document analysis and have hence contributed to our group's expertise in this domain. I would especially like to thank my former and present PhD students Dominique Antoine, Suzanne Collin, Anja Habacha and Pascal Vaxivière for their contribution to this expertise, in various application domains, as can be seen by the bibliographical references to their work in this article. Many long discussions with P. Vaxivière have been especially useful to help maturing the proposed methodology for document understanding. The classification of technical documents into various categories (§ 3) was first proposed by my colleague Abdel Belaïd, whose research group works on structured document analysis and character recognition, when we prepared together a course on document analysis. It has also been rewarding to start collaborating with Dov Dori, from Technion University, Haifa (Israel); this article was written a week after we had completed together a survey on technical document analysis, and it certainly benefited from the spirit of this work. I look forward to still much closer cooperation between our group and Dov's in the coming months and years.

Our group has benefited in the past years from various grants and other financial support for our work on technical document analysis; this support is gratefully acknowledged, with special mention to a "stimulation contract" from *Institut National Polytechnique de Lorraine*, a grant from *Ministère de la Recherche et de la Technologie*, a study grant from IBM France to ESSTIN, an engineering school in Nancy where we built the CELESSTIN system, and of course continuous support from CRIN/CNRS and INRIA Lorraine.

References

[1] D. Antoine. CIPLAN: A Model-Based System with Original Features for Understanding French Plats. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 2, pages 647–655, 1991.

[2] D. Antoine, S. Collin, and K. Tombre. Analysis of Technical Documents: The REDRAW System. In H.S. Baird, H. Bunke, and K. Yamamoto, editors, *Structured Document Image Analysis*. Springer Verlag, 1992. To appear.

[3] H.S. Baird, H. Bunke, and K. Yamamoto, editors. *Structured Document Image Analysis*. Springer Verlag, 1992.

[4] A. Belaïd and K. Tombre. Analyse de documents : de l'image à la sémantique. In *Bigre 80 — Actes de CNED'92 : Traitement de l'écriture et des documents, Nancy*, pages 3–29, Juillet 1992.

[5] D. Benjamin, O. Forgues, E. Gulko, J.B. Massicotte, and C. Meubus. The Use of High-Level Knowledge for Enhanced Entry of Engineering Drawings. In *Proceedings of 9th International Conference on Pattern Recognition, Rome (Italy)*, pages 119–124, 1988.

[6] L. Boatto, V. Consorti, M. Del Buono, S. Di 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] V. Cappellini, F. Flamigni, and A. Mecocci. A System for Automatic Drafting Encoding. In *Proceedings of IAPR Workshop on Computer Vision, Tokyo (Japan)*, pages 173–178, 1988.

[8] S. Collin. Interprétation de la cotation des dessins techniques par analyse syntaxique. Thèse de doctorat, Institut National Polytechnique de Lorraine, Vandœuvre-lès-Nancy, Jan. 1992.

[9] S. Collin. Syntactical Analysis of Technical Drawing Dimensions. In *IAPR International Workshop on Structural and Syntactic Pattern Recognition, Bern (Switzerland)*, August 1992.

[10] S. Collin and K. Tombre. Image Segmentation Using Connected Components for Document Analysis. In *Proceedings of IFACS/IMACS/IFORS International Symposium on Advanced Information Processing in Automatic Control, Nancy, France*, volume 2, pages 166–170, 1989.

[11] S. Collin and P. Vaxivière. Recognition and Use of Dimensioning in Digitized Industrial Drawings. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 1, pages 161–169, 1991.

[12] IEEE COMPUTER Magazine. Special Issue on Document Image Analysis Systems, Volume 25, Number 7, July 1992.

[13] D. Dori. A Syntactic/Geometric Approach to Recognition of Dimensions in Engineering Drawings. *Computer Vision, Graphics and Image Processing*, 47:271–291, 1989.

[14] D. Dori. Self Structural Syntax Directed Pattern Recognition of Dimensioning Components in Engineering Drawings. In *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)*, pages 88–112, 1990.

[15] D. Dori. Symbolic Representation of Dimensioning in Engineering Drawings. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 2, pages 1000–1010, 1991.

[16] D. Dori and A. Pnueli. The Grammar of Dimensions in Machine Drawings. *Computer Vision, Graphics and Image Processing*, 42:1–18, 1988.

[17] K. Eikvil and T. Taxt. Simultaneous Recognition of Class and Rotation Angle of Raster Symbols. In *Proceedings of 7th Scandinavian Conference on Image Analysis, Aalborg (Denmark)*, pages 461–468, 1991.

[18] R. Espelid, N. A. Alvsvaag, J. Eileng, and I. Skauge. Automatic Digitizing of the Colour-Layer of Thematic Maps. In *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)*, pages 299–302, 1990.

[19] C.S. Fahn, J.F. Wang, and J.Y. Lee. A Topology-Based Component Extractor for Understanding Electronic Circuit Diagrams. *Computer Vision, Graphics and Image Processing*, 44:119–138, 1988.

[20] M.A. Fischler, J.M. Tenenbaum, and H.C. Wolf. Detection of Roads and linear Structures in low-resolution Aerial Imagery using a Multisource Knowledge Integration Technique. *Computer Vision, Graphics and Image Processing*, 15:201–223, 1981.

[21] L.A. Fletcher and R. Kasturi. A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images. *IEEE Transactions on PAMI*, 10(6):910–918, 1988.

[22] T. Futatsumata, G. Shichino, J. Shibayama, and A. Maeda. Development of an Automatic Recognition System for Plant Diagrams. In *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)*, pages 207–210, 1990.

[23] A.H. Habacha. Structural Recognition of Disturbed Symbols Using Discrete Relaxation. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 1, pages 170–178, 1991.

[24] A.H. Habacha and K. Tombre. Structural Symbol Recognition. In *Proceedings of 7th Scandinavian Conference on Image Analysis, Aalborg (Denmark)*, volume 1, pages 486–493, August 1991.

[25] R.H. Haralick and D. Queeney. Understanding Engineering Drawings. *Computer Graphics and Image Processing*, 20:244–258, 1982.

[26] T. Hayakawa, T. Watanabe, Y. Yoshida, and K. Kawaguchi. Recognition of Roads in an Urban Map by Using the Topological Road-Network. In *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)*, pages 215–218, 1990.

[27] J. Hofer-Alfeis and G. Maderlechner. Automated Conversion of Mechanical Engineering Drawings to CAD Models: Too Many Problems? In *Proceedings of IAPR Workshop on Computer Vision, Tokyo (Japan)*, pages 206–209, 1988.

[28] *Proceedings of First International Conference on Document Analysis and Recognition, Saint-Malo (France)*, Paris, 1991. AFCET.

[29] M. Ilg. Knowledge-Based Understanding of Road Maps and other Lines Images. In *Proceedings of 10th International Conference on Pattern Recognition, Atlantic City, NJ (USA)*, volume 1, pages 282–284, 1990.

[30] M. Ilg and R. Ogniewicz. Knowledge-Based Interpretation of Road Maps Based on Symmetrical Skeletons. In *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)*, pages 161–164, 1990.

[31] S.H. Joseph. On the Extraction of Text Connected to Linework in Document Images. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 2, pages 993–999, 1991.

[32] S.H. Joseph and T.P. Pridmore. A System for the Interpretation of Images of Graphics. In *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)*, pages 154–164, 1990.

[33] S.H. Joseph, T.P. Pridmore, and M.E. Dunn. Towards the Automatic Interpretation of Mechanical Engineering Drawings. In A. Barrett, editor, *Computer Vision and Image Processing*. Kogan Page, 1989.

[34] M. Karima, K.S. Sadhal, and T.O. McNeil. From Paper Drawings to Computer-Aided Design. *IEEE Computer Graphics and Applications*, 5(2):27–39, 1985.

[35] R. Kasturi and J. Alemany. Information Extraction from Images of Paper-Based Maps. *IEEE Transactions on Software Engineering*, 14(5):671–675, 1988.

[36] R. Kasturi, R. Fernandez, M.L. Amlani, and W.-C. Feng. Map Data Processing in Geographic Information Systems. *IEEE COMPUTER Magazine*, 22(12):10–21, 1989.

[37] R. Kasturi and L. O'Gorman. Document Image Analysis: A Bibliography. *Machine Vision and Applications*, 5(3):231–243, Summer 1992.

[38] R. Kasturi, R. Raman, C. Chennubhotla, and L. O'Gorman. Document Image Analysis: An Overview of Techniques for Graphics Recognition. In *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)*, pages 192–230, 1990.

[39] R. Kasturi, S. Siva, and L. O'Gorman. Techniques for Line Drawing Interpretation: An Overview. In *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)*, pages 151–160, 1990.

[40] H. Kato and S. Inokuchi. The Recognition Method for Roughly Hand-Drawn Logical Diagrams Based on Hybrid Utilization of Multi-Layered Knowledge. In *Proceedings of 10th International Conference on Pattern Recognition, Atlantic City, NJ (USA)*, volume 1, pages 578–582, 1990.

[41] K. Kitajima and M. Yoshida. Reconstruction of CSG Solid from a Set of Orthographic Three Views. In *Proceedings of IEEE International Conference on Systems Engineering, Kobe, Japan*, pages 220–224, September 1992.

[42] P. Kuner and B. Ueberreiter. Knowledge Based Pattern Recognition in Disturbed Line Images Using Graph Theory Optimization and Predicate Calculus. In *Proceedings of 8th International Conference on Pattern Recognition, Paris (France)*, pages 240–243, 1986.

[43] C.P. Lai and R. Kasturi. Detection of Dashed Lines in Engineering Drawings and Maps. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 2, pages 507–515, 1991.

[44] S.W. Lee, J.H. Kim, and F.C.A. Groen. Translation-, Rotation- and Scale-Invariant Recognition of Hand-Drawn Electrical Circuit Symbols with Attributed Graph Matching. In *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)*, pages 273–292, 1990.

[45] R. Lequette. Construction automatique de solides à partir de vues orthogonales de type dessin industriel. Thèse de doctorat, Université de Grenoble, 1987.

[46] S.Z. Li, J. Kittler, and M. Petrou. Matching and Recognition of Road Networks from Aerial Images. In G. Sandini, editor, *Proceedings of Second European Conference on Computer Vision, Santa Margherita Ligure (Italy)*, pages 857–861, 1992. Lecture Notes in Computer Science, Vol. 588.

[47] X. Lin, S. Shimotsuji, M. Minoh, and T. Sakai. Efficient Diagram Understanding with Characteristic Pattern Detection. *Computer Vision, Graphics and Image Processing*, 30:84–106, 1985.

[48] W. Lu, Y. Ohsawa, and M. Sakauchi. A Database Capture System for Mechanical Drawings Using an Efficient Multi-dimensional Graphical Data Structure. In *Proceedings of 9th International Conference on Pattern Recognition, Rome (Italy)*, pages 266–269, 1988.

[49] D.B. Lysak and R. Kasturi. Interpretation of Engineering Drawings of Polyhedral and Non-polyhedral Objects. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 1, pages 79–87, 1991.

[50] E. Martí, J. Regincós, J. Lopez-Krahe, and J.J. Villanueva. A System for Interpretation of Hand Line Drawings as Three-Dimensional Scene for CAD Input. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 1, pages 472–480, 1991.

[51] U.B. Mokate and R. Kasturi. An Algorithm for Recognition of Circles in Graphics. Computer Engineering Technical Report TR-88-061, The Pennsylvania State University, Department of Electrical Engineering, University Park, Pennsylvania 16802, 1988.

[52] F.S. Montalvo. Diagram Understanding: The Intersection of Computer Vision and Graphics. Technical Report 873, Massachusetts Institute of Technology, 1985.

[53] S. Murai, S. Viseshsin, Y. Honda, M. Xi, M. Takagi, and S. Ochi. 3D Application in Remote Sensing and Geographic Information Systems. In *International Symposium on Three Dimensional Image Technology and Arts*, Tokyo, Japan, 5–7 Feb 1992.

[54] H. Murase and T. Wakahara. Online Hand-sketched Figure Recognition. *Pattern Recognition*, 19(2):147–160, 1986.

[55] Machine Vision and Applications. First Special Issue on Document Image Analysis Techniques, Volume 5, Number 3, Summer, 1992.

[56] V. Nagasamy and N.A. Langrana. Engineering Drawing Processing and Vectorization System. *Computer Vision, Graphics and Image Processing*, 49(3):379–397, 1990.

[57] A. Okazaki, T. Kondo, K. Mori, S. Tsunekawa, and E. Kawamoto. An Automatic Circuit Diagram Reader with Loop-Structure-Based Symbol Recognition. *IEEE Transactions on PAMI*, 10(3):331–341, 1988.

[58] D. Pao, H.F. Li, and R. Jayakumar. Graphic Features Extraction for Automatic Conversion of Engineering Line Drawings. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 2, pages 533–541, 1991.

[59] W.W. Seemuller. The Extraction of Ordered Vector Drainage Networks from Elevation Data. *Computer Vision, Graphics and Image Processing*, 47:45–58, 1989.

[60] T. Senda and Y. Arimitsu. Automatic Reconstruction of Solid from a Set of the Orthographical Three Views. In *Proceedings of IEEE International Conference on Systems Engineering, Kobe, Japan*, pages 229–233, September 1992.

[61] S. Shimotsuji, O. Hori, M. Asano, K. Suzuki, F. Hoshino, and T. Ishii. A Robust Recognition System for a Drawing Superimposed on a Map. *IEEE COMPUTER Magazine*, 25(7):56–59, July 1992.

[62] S. Shimotsuji, S. Tamura, and S. Tsunekawa. Model-based Diagram Analysis by Generally Defined Primitives. In *Proceedings of 6th Scandinavian Conference on Image Analysis, Oulu (Finland)*, pages 1034–1041, 1989.

[63] S. Suzuki and T. Yamada. MARIS: Map Recognition Input System. In *Proceedings of IAPR Workshop on Computer Vision, Tokyo (Japan)*, pages 421–426, 1988.

[64] K. Tombre and P. Vaxivière. Structure, Syntax and Semantics in Technical Document Recognition. In *Proceedings of First International Conference on Document Analysis, Saint-Malo, France*, volume 1, pages 61–69, 1991.

[65] K. Tomiyama, T. Nakamura, and T. Koezuka. Auxiliary Lines in Three View Drawings for 3D Shape Reconstruction Using CSG Method. In *Proceedings of IEEE International Conference on Systems Engineering, Kobe, Japan*, pages 250–256, September 1992.

[66] S. Tsujio, T. Ono, and S.S. Lee. Computer-Aided Drawing Check for CAD Systems — A Method for the Checking of Dimensions in Multiview Mechanical Drawings. In *Proceedings of IEEE International Conference on Systems Engineering, Kobe, Japan*, pages 234–237, September 1992.

[67] P. Vaxivière and K. Tombre. Interpretation of Mechanical Engineering Drawings for Paper–CAD Conversion. In *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)*, pages 203–206, 1990.

[68] P. Vaxivière and K. Tombre. Celesstin: CAD Conversion of Mechanical Drawings. *IEEE COMPUTER Magazine*, 25(7):46–54, July 1992.

[69] P. Vaxivière and K. Tombre. CELESSTIN IV: Knowledge-Based Analysis of Mechanical Engineering Drawings. In *Proceedings of IEEE International Conference on Systems Engineering, Kobe, Japan*, pages 242–245, September 1992.

[70] K. Wakimoto, Y. Maeda, M. Shima, A. Maeda, and N. Chikada. A Utility Map Transformation System Based on Map Understanding. In *Proceedings of IAPR Workshop on Computer Vision, Tokyo (Japan)*, pages 220–223, 1988.

[71] L.S. Wolfe and J. de Wyze. An Update on Drawing Conversion. *Computer Aided Design Report*, 8(6):1–11, 1988.

[72] J. Ylä-Jääski and H. Ahonen. Knowledge-based Analysis of Line Drawings. In *Proceedings of 6th Scandinavian Conference on Image Analysis, Oulu (Finland)*, pages 774–777, 1989.