УДК 681.327.12.001.362:0036

S. Ablameyko, V. Bereishik,
O. Frantskevich, E. Melnik

RECOGNITION OF ENGINEERING DRAWING ENTITIES: MAIN PRINCIPLES AND TECHNOLOGY

Introduction

A problem of fast, precise and non-expensive input of engineering drawings (ED) into CAD systems attracts an attention of many researchers. Manual input of drawings into computer is a slow and expensive process. The usage of scanners for this aim allows fast transform of drawings into digital raster form but requires a highly developed software to convert scanned images into high-level CAD models.

There are many papers and systems oriented to solution of this problem. But most of them are restricted by a vectorization of raster data to obtain an image description in terms of simplest graphical primitives, and do not allow more high-level description needed for CAD systems.

We will not analyse the systems and vectorization methods here. Several good surveys of the art state in this area have been published [1-3]. The description of many systems and methods for this task can be found in the Proceedings of the ICDAR'91 and ICDAR'93 Conferences, and MVA'90 and MVA'92 Workshops. A special issue of Computer Magazine (July 1992) gives a description of the systems and techniques.

We also have a big experience in a creation of document vectorization systems. They have been created mainly to process different types of map layers and included not only ordinary vectorization tools but some techniques for interpretation of cartographical objects.

During last some years we are ingaged in application of our techniques for interpretation of ED images (or more precisely - the category of orthogonal proections of ED [1]). The vectorization process for both kinds of graphical documents is similar though has some small differencies. But the recognition process is fully different as we try to "read" two different document types.

The aim of ED recognition we see in obtaining a representation of ED image in terms of universal engineering (CAD/CAM) entities: contour (usually thick) lines, symmetry axes (dash-dotted lines), hidden contour (dashed) lines, hatched areas, dimensions and others. These entities can be used as the base for ED understanding, i.e. for extraction of 2D CAD objects according to the concrete applied field with corresponding CAD library (for example, gearbox, shafts, screws etc.) and/or for the problem of 3D CAD model reconstruction.

In this paper, we show our own viewpoint on the problem of ED image interpretation and concentrate ourself more carefully on the task of CAD entities recognition. We suggest main principles and technology of ED interpretation, introducing basic stages and levels of image representation, discuss the obtained results and their practical usage.

1. Brief overview of existing techniques and viewpoint on the problem

At present, there are no many systems which perform not only conversion of raster ED images to vector form, but allow to recognize more complicated CAD entities than points, lines, arcs and conics. Briefly consider some of them.

The systems developed by R.Kasturi et al [4], V.Nagasamy and N.Langrana [5], M.Ejiri et al [6] are succesfully used for processing and vectorization of ED images. From our point of view, these systems obtain good results but their interpretation level can also be improved. Two following systems are more intellectual and allow to recognize some ED entities.

ANON system for ED interpretation is based on the combination of schemata describing prototypical drawing constructs with a library of low-level image analysis routines and a set of explicit control rules [7]. The system works directly with raster image without prior thresholding and vectorization, combining the extraction of primitives with their interpretation. We suppose, that this peculiarity and used interpretation strategy, based on yacc-machine and very closed to syntactical recognition method, restrict possibilities of this approach for noise sensitive recognition of complex entities.

CELESSTIN system uses high-level knowledge for interpretation of mechanical drawings [8]. The knowledge rules used in the CELESSTIN are the following: technologically significant entities are extracted and the whole setup is analysed with respect to disassembling and kinematic knowledge. The last version of CELESSTIN uses knowledge rules relative to the semantics, i.e. to the functionalities of the represented object and not only the representation rules. But it seems that CELLESTIN is restricted by specific applied domain and not enough universal.

There known some knowledge based systems for interpretation other document types. Methods for recognition of some particular ED elements are given in papers [9-10]. Finally, a paper [1] gives the state of the art of syntactic image understanding and suggest a general methodology of technical drawing recognition.

The next part of this section can be considered as continuation of methodology discussion began by K. Tombre in [1], but other sections contain formulation of concrete principles (key moments) to form a universal CAD representation.

From our viewpoint, the problem of automated conversion of engineering drawings to CAD representation is divided into four main tasks:

(1) vectorization of drawings to obtain a vector image representation in terms of simplest graphical primitives;

(2) recognition of the vector image model to obtain a ED representation in terms of universal CAD entities;

(3) understanding of the obtained image representation to get specific 2D CAD objects cointaining in some CAD library with their parameters and relations;

(4) complete reconstruction of 3D CAD model with all "semantic" attributes.

It is clear that ideal solution of the problem of ED image interpretation could be in full automation of expensive input and preparing all information to solve corresponding applied tasks. But existing level of the image recognition theory does not allow to solve this problem autonomously, i.e. without human intervention. Therefore, we see that the main aim of ED input system design is to minimize this intervention.

In principle, results of each task solution can be used separately without further processing. But the vectorization results take a lot of time and interactive job to create further a required CAD models.

So, we consider (similar to [11]) that a practically useful system of ED interpretation must supply main high-level entities defined by Inititial Graphics Exchange Specification (IGES). Therefore, a corect solution of the second task must satisfy to industrial requriremеnts.

The universal solution of the third task (oriented to any arbitrary CAD library) is very complex, although its particular solution (oriented to concrete CAD library) can be not very difficult. Non correct solution of this task can significantly increase volume of interactive image editing.

Moreover, the third task is very closely connected with the fourth task, because many modern CAD systems operate with complex 3D objects which can be recognized only after analysis of all orthogonal proections and even isometry picture. But the fourth task is more the task of geometrical modelling.

So, we consider that at present, the more important task of ED image interpretation is the task of correct recognition of CAD entities and namely this task is considered in this paper.

2. Main notions and forms of data representation

Defining basic forms of data representation during recognition process we proceed from the fact that the interpretation process can be considered as successive transformation of graphical information about ED image from one level of representation to another one with higher level of abstraction in ED perception. The final aim of the processing is to generalize data obtained from a scanner to the level necessarial for representation of ED in CAD systems.

According to this vision, during the interpretation process we extract three main levels of image representation: raster image representation, intermediate vector representation and image representation in terms of universal CAD entities. To achieve aim of processing we need some auxiliary information, containing apriori data for recognition. Let us consider these components more detaily.

1. Raster representation. We consider that the scanned image is represented in a binary format and thresholding operation is already done by scanner or by any other program way. The main notion of this image is a connected component (CC) which represents a connective set of black pixels.

2. Vector representation. This level is a result of the vectorization process. It contains a description of connected components and their subparts in a vector format. Such level of ED representation is presented in two forms: contour and skeleton.

2.1. Contour form (C-form) of vector representation can be easily received from the initial image. Of course, this form is far from a required output representation. But we consider it as an auxiliary information for further recognition process because in this form can be easily extracted and conviniently represented some elements of ED such as distinct characters and points, potential arrowheads of dimensions and filled areas (two last elements are extracted from images obtaining after morphological operations).

2.2. Skeleton vector form (S-form) of vector representation contains two main data types for further recognition. They are segments (vectorized CC parts of thinned image, bounded by end and node points) and knots describing connections between segments. This form is more structured than the previous one and is used as a basis for the ED image recognition. It contains a detailed information not only about segments location but also usefull data about their characteristics (thickness, length, slope angle etc.) and topologiclal relations (by means of knots). More detaily these notions are described in [15]. Additionally this form includes also an information about segment's curvature (polyline, straight line or circular arc).

S-form can be used for ED archivation, but it is not satisfactory as the CAD model due to its superfluous detalization.

3. CAD representation. Image description in terms of universal CAD elements is considered as a high-level output description of ED which posesses sufficiently high level of abstraction suitable for the CAD systems. On this level, we distinguish between graphical primitives and engineering (or CAD) entities forms.

3.1. Graphical primitives form (P-form) represents ED graphics in a maximal compact way but is more natural from the ED perception point of view when compared with S-form.

The graphical primitives are divided into simple and complex ones. The simple primitives correspond to connected ED elements having the same parameters (lines, arcs, characters). As graphical primitives we consider straight lines and circular arcs (conics and splines can be also added), characters and special symbols. Simple graphical primitives consist of one or more segments, but from "viewpoint" of graphical systems (performing display of the graphical data) they can not be divided into composite parts.

The complex primitives consist of simple ones divided by gaps or having different parameters (text strokes, broken lines, lines with different thickness etc.), Some simple primitives are united to one complex on the base of the same geometric (they have equal coefficients of equation), or logical (words which consist of characters, solid areas etc.) characteristics.

This form can be used for ED archivation too. In this case, volume of data is decreased and contens of data becomes more naturally for CAD systems. We can use this form in double way:

- using only simple primitives we have more detailed description and can "draw" picture very similar to the initial image by using simple graphical system;

- using complex primitives and ignore its composite parts we have more compact (but less detailed) representation oriented to more complex graphical system, but drawn picture will be less similar to the initial one than in the previous case.

3.2. Engineering entities form (E-form) represents the contens of the ED image in terms of universal CAD entities, which are independent on the ED domain and reflect some semantical part of ED. They are (for example [1,3]): contour lines (usually thick lines of ED image representing projections of the object contours on a plane), symmetry axes (represented by dash-dotted lines), hidden contour lines (represented by dashed lines), matter areas (represented by crosshatchings), dimensions (thin lines with arrows, witness lines and so on), annotation texts etc.

We consider simple and complex entities of the E-form. Simple entities are combinations of the graphical primitives (or their pieces) and are used for representation of simple engineering entities like symmetry axis, hatchcrossing line, the border of matter area etc. In some cases, the simple entities can coincide with geometry primitives, but in general case they are more complex notions (for example, simmetry axis can be represented by polyline consisting of a few dot-dashed straight lines and circular arcs; border of hatched area can consist of thick, thin and dot-dashed lines etc.).

The complex entities are used for description of more complex ED structures - scenes - and can be presented as combinations of graphical primitives and set of another simple and/or complex entities. Examples of complex entities are crosshatching (set of hatching lines, bounded by one or more borders); symmetry center; circle or circular arc with denoted (by crossed symmetry axes) center; set of concentric circles having the same center; dimensions of different types etc. It is possible to combine the complex entities; as a result, new (more complex) entities are formed: for example, center of symmetry is intersection of two or more symmetry axes or the first, being combined with corresponding circle can form more complex entity etc.). Geometrical information of engineering entities is projected into P-form of CAD representation and is given in relative coordinates of corresponding geometrical primitives.

All mentioned forms of image representation are connected with each other either physically by coordinates or logically by references. We consider that key moments of output data representation correspond to IGES principles and, moreover, take into consideration recognition specifics.

Using terminology of [1] we can say that under recognition we distinct two "image parts": graphical part (P-form of output representation) and engineering (semantical) part (E-form of output representation).

4. Apriori data.

To convert ED image into CAD object representation, we must have apriori information about recognized elements, i.e. knowledge about rules of building the graphical primitives and CAD entities. Part of this information is included directly in software program codes, realizing the recognition algorithms (for example, "a circle is set of points having equal distance from a centre", "crosshatching is a closed area filled by thin parallel lines which have approximately equal distance one from another" etc.). The second part is extracted on the vectorization stage (for example, maximal width of thin lines and minimal width of thick lines). But, main part of apriori information can be considered as expert's knowledge stored in a knowledge base. We will not consider a structure organization of the knowledge base in this paper. Introduce only the main notions or kinds of information, which it contains.

To recognize graphical primitives and engineering entities, we must know, in a common case, the following information:

- parameters of recognized elements (thickness of segments, length of stroke and space for dashed lines, slope angle for crosshatching etc.);

- the structure of recognized elements, i.e. from what components (segments and/or primitives and/or entities) this element is composed;

- space-logical relations between components of one or different levels of representation and their meanings: joining or crossing under some angles; separation by a gap of some length; lie in some neighbourhood and others. This kind of knowledge can be used either for its initial classification (hypothesis advancing) or for the description of the element under recognition (hypothesis checking).

Using mentioned above information we can describe recognized elements either in a feature space (using only sets of parameters) or by means of special formalized language (similar to [9]) in a grammar form, when consider the complex entities represented by the scenes.

The knowledge give an opportunity to reduce a number of models for object recognition and to speed a process of object recognition.

5. About 2D CAD object representation. We consider that the objects specific for concrete applied domain (for example, gearboxes, shafts, screws of mechanical devices etc.) can be represented in such a way as complex entities. Elements of this abstraction level can be based on P-form and E-form of ED image representation. It is possible, that a description of some 2D CAD objects will require a new and more complex relations between composite components and more complex kinds of control information, but principles of description remain the same.

3. The main principles of engineering drawing recognition

Based on the described above scheme and notions, we introduce four main principles for engineering drawing interpretation. These principles can be considered as different sides of the interpretation process, defining a technology for processing and scheme of main algorithms. But we do not describe the algorithms here because we consider them as subject of another papers.

1. The first principle - "from simple to complex" - determines a sequence of the ED elements recognition and forming of output database. It consist in that the ED is recognized starting from the simple graphical primitives and moving towads the more complex CAD entities.

As one can see from the previous section, the process of image interpretation begins already on the preprocessing and vectorization stage. At first, we analyse the initial (improved) image, build hystogramm of possible lines thickness and calculate width of thin and thick lines. These values are used to perform morphological operations with the aim to extract potential arrowheads and solid areas. After analysis of the C-form, an auxiliary data for recognition of complex entities containing, for example, potential elements of dimensions are formed.

To get the S-form, we perform thining of the image and calculate line thickness at every skeleton point (in this case, we use already calculated maximal line thickness), define the feature points and vectorize the thinned image. Then, small noise remained after thinning is reduced and detailed analysis of thickness and shape for extracted segments is performed. As a result, critical points, potential thin and thick lines, circular arcs and straight lines are extracted.

On the vectorization stage, we extract only some real or potential parts of graphical primitives. Then, we try to join these parts and obtain the P-form, i.e. all mentioned above parts of primitives are used to create another form of data representation with more high level of abstraction. This process is performed by using the formulated principle: at first, we try to extract simple primitives and then complex primitives, though this "moving" is not very straight. Two examples: we do not try to recognize all circular arcs before symmetry axes extraction because process of arc extracting with known centre is more simpler than one without known centre; we "return" to text string extraction after geometrical primitives extraction and try to add not isolated character to strings of isolated ones.

The E-form is built by using the extracted graphical primitives and auxilary data. As early, at first we try to recognize simple entities and then complex entities. Recognition of dimensions - more complex recognized elements - is performed after recognition of all other entities; in this case, we do not check many "false" variants as we have maximal information about ED contents.

A brief scheme of the described technology is shown in Figure 1.

2. The second principle defines rules for checking and advancing of hypotheses for ED entities recognition. It consists in a maximal usage of space-logical relations between elements of different forms of data representation. The majority of such space-logical relations (crossing, crossing under defined angle, joining etc.) are informative enough and can be used to advance hypothesis about presence of some new primitive or CAD entity on the image. For example, thin parallel segments, connected through their node points by thick line, can be considered as initial hypothesis for the hatched area recognition; two straight segments or graphical primitives with free end points distinguished by the gap can be potential parts of axis of symmetry or hidden contour line; two crossing axis of simmetry indicate the point which can be considered as a center of one or more concentric circles etc. It should be mentioned that space-logical relations are fixed and analyzed always when it is possible during the recognition process. And if the relation is useful, i.e. it helps us to continue creation of the recognized element, we use it, otherwise it is recorded into a special stack as potential initial hypothesis for the recognition of another graphical primitive or CAD entity.

3. The third principle - "from local to global analysis" - defines rules of complex entities recognition. It consist in a sequential complication of recognition techniques. The point is, that on the first steps of the ED recognition, when a little amount of additional useful information exists, it is very

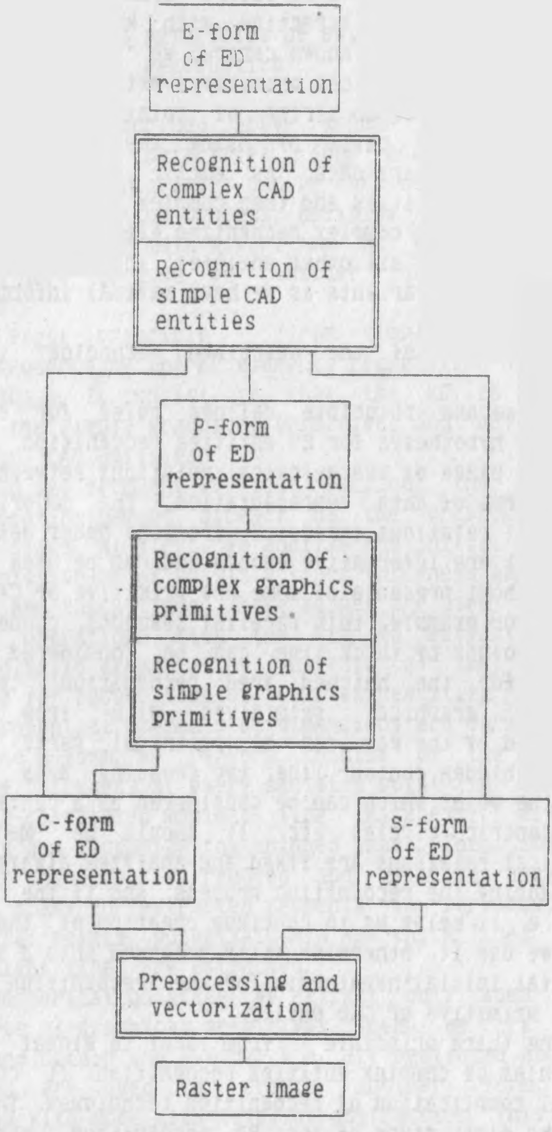

Fig. 1. Brief scheme of ED recognition technology

dangerous to continue recognition of any grapfical element under some contradictory conditions, because a mistake in such doubtful situation can leads to unpredictable results. So, we introduce two stages of ED graphical elements recognition. On the first stage, under "information hunger" condition, so called "clean parts" of corresponding elements are extracted, using only local analysis of situations or features, which are usual for these elements as a rule. And only when all possible "clean parts" are extracted and all possible additional helpful information exists, we perform the second stage of recognition, trying to resolve every doubtful situation by means of more complex and not so dangerous in a given moment global analysis, using some stable criterias to choose a better variant from all possible ones.

This approach is common enough when recognize different types of graphical elements from lines of complex structure to various types of scenes (including dimensions), though methods of situation analysis and criteria to choose the best variants of elements construction can differ from object to object. It is this principle which defines a noise sensitivity and correctness of complex elements recognition.

4. The fourth principle consists in maximal avaiable taking into cosideration of engineering drawing specifics. Using this principle we try to extract all the usefull information about some parts of CAD entities on the earliest stages of ED processing (before recognition). After recognition, we must use peculiarities how the ED graphical primitives are drawn on the document and how they influence one to another. The majority of lines on the ED image are drawn in ortogonal directions. Therefore, with the aim to correct possible distortions on the image after scanning and to reduce amount of interactive editing of resulted CAD representation, we correct this representation making nearly vertical and horisontal straight line graphical primitives as strictly vertical and horisontal. By this, corresponding correction of all another graphical primitives and their contact points is performed. This operation is comparetively easy to perform, because we apply it only for graphical primitives of ED P-form, while the more complex E-form is corrected automatically as having relative coordinates connection with P-form.

Of course, we understand that specifics of ED is used not completely. More complex operation in this direction is to correct placements of some graphical primitives taking into account symmetry axis. Moreover, there is a big and complex problem of CAD model reconstruction in accordance with semantical (dimensioning values, annotations text and so on) information. But these problems are not subject of graphics recognition and belong to other scientific fields such as text understanding and geometry modelling. We plan to investigate these questions in our future works.

Conclusion and iscussion

During some last years we work under development of the ED interpretation system. The first process -vectorization- has been developed early and applied to process map-drawing images. The description of this process is given in [13-16]. Then, it was modified and applied to process engineering drawing images. It allows to obtain the S-form and the C-form of vector image representation with good time and quality characteristics.

At present, we finish the development of an experimental software realizing the recognition process based on the introduced principles which allows to obtain the image representation in terms of universal CAD entities. The most of ideas presented in this paper, were verified on the real ED images and shown good results.

The software has been developed on IBM PC/AT computer in C language. The input binary images are obtained from engineering drawings with size A4-A2, usually digitized with a resolution of 20 pixels per 1 mm and 300 DPI. The raster data for the processing are represented in PCX, TIFF or MSP format. Output data are represented in the IGES or AutoCAD DXB, DXF files. The vectorization process has been tested on many ED images and recognition process has been tested on all mentioned types of graphical primitives and CAD entities.

The main differences of our approach to the task of ED images interpretation from those already proposed we see in the following:

- introduction of distinct notions of graphical primitives and CAD entities what allows to formalize practical ED recognition technology and obtain enough high-level ED representation;

- introduction of the main forms of image representation and their connections with each other during the interpretation process;

- definition of main principles for ED images recognition and their usage for development techology and realization the recognition algorithms;

- practical verifying of the proposed approach on real ED images.

The performed experiments made us sure that we have chosed a right way for solution of the problem of fast conversion and correct interpretation of engineering drawing images.

References

1. Tombre K., Technical drawing recognition and understanding: from pixels to semantics, Proc. IAPR Workshop on Machine Vision Applications, Tokyo, pp. 393-402, 1992.

2. Kasturi R., Siva S., O'Gorman L., Techniques for line-drawing interpretation: an overview, Proc. MVA'90 Workshop, Tokyo, pp. 151ю160, 1992.

3. Hofer-Alfeis J., Maderlechner G., Automated conversion of mechanical engineering drawing to CAD models: too many problems? Proc. IAPR Workshop on CV - Special Hardware and Industrial Applications, Tokyo, pp. 206ю209, 1988.

4. Kasturi R., Bow S.T., El-Mastri W., Shah J., Gattiker J.R., Mokate U.B., A system for interpretation of line drawings, IEEE Trans. on PAMI, 12, pp. 978-992, 1990.

5. Nagasamy V., Langrana N.A., Engineering drawing processing and vectorization system, Computer Vision, Graphics, and Image Processing, Vol. 49, No. 3, pp. 379-397, 1990.

6. Ejiri M., Kakumoto S., Miyatake T., Shimada S., Iwamura K., Automatic recognition of engineering drawings and maps, In Image analysis applications, R.Kasturi and M.M.Trivedi (eds), Marcel Dekker, 1990.

7. Joseph S.H., Pridmore T.P., Knowledge-directed interpretation of mechanical engineering drawings, IEEE Trans. on PAMI, Vol. 14, No. 9, pp. 928-940, 1992.

8. Vaxiviere P., Tombre K., Celesstin: CAD conversion of mechanical drawings, Computer, Vol. 25, No. 7, pp. 46-55, 1992.

9. Dori D., A syntactic/geometric approach to recognition of dimensions in engineering drawings, Computer Vision, Graphics and Image Processing, V.47, pp.271-291, 1989.

10. Lai C.L., Kasturi R., Detection of dashed lines in engineering drawings and maps, Proc. of ICDAR'91, Saint-Malo, France, pp.507ю515, 1991.

11. Espelid R., A raster-to-vector-conversion concept based on industrial requirements, Proc. IAPR Workshop on CV-Special Hardware and Industrial Applications, Tokyo, Japan, pp.224-228, 1988.

12. Initial Graphics Exchange Specification (IGES), Version 3.0, April 1986, U.S. National Bureau of Standards (NEL), Geithersburd, MD 20899.

13. Ablameyko S., Bereishik V., Frantskevich O., Homenko M., Melnik E., Paramonova N., Fast raster-to-vector conversion of largesize 2D line-drawings in a restricted computer memory, Proc. IAPR Workshop on Machine Vision Applications, Tokyo, Japan, pp.59ю62, 1992.

14. Ablameyko S., Bereishik V., Frantskevich O., Homenko M., Melnik E., Okun O., Patsko O., Paramonova N., System for vectorization and interpretation of graphic images, Pattern Recognition and Image Analysis, Vol.3, No.1, pp.39-52, 1993.

15. Ablameyko S., Bereishik V., Paramonova N., Marcelli A., Ishikawa S., Hierarchical vector representation of document images, Proceedings of SPIE, Vol.1977, 1993.

16. Semenkov O., Ablameyko S., Bereishik V., Starovoitov V., Information processing and display in raster graphic systems, Minsk, Nauka i technika, 1989 (in Russian).

Institute of Engineering Cybernetics
Belarussian Academy of Sciences,
Minsk