Pattern Recognition, Vol. 22, No. 1, pp. 1–11, 1989.
Printed in Great Britain

0031-3203/89 $3.00 + .00
Pergamon Press plc
Pattern Recognition Society

PROCESSING OF ENGINEERING LINE DRAWINGS FOR AUTOMATIC INPUT TO CAD

S. H. JOSEPH

Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, U.K.

(*Received* 25 *May* 1987; *in revised form* 1 *February* 1988)

Abstract—An approach to the conversion of images of line drawings to a format appropriate for computer-aided draughting systems is described. A sequential adaptive line finding technique, combined with ordered storage and automatic guided search is used. The performance of the system on good quality originals reaches the limits set by the level of interpretation employed. On poor quality originals the adaptive nature of the line finder produces a line enhancement effect. The method is seen particularly as a contribution to the solution of the problem of interpretation of imperfect data that is always present in engineering drawing conversion.

Line following   CAD   Engineering drawings   Line enhancement   Vectorisation

INTRODUCTION

The widespread introduction of computer-based draughting systems to industries with large stocks of conventional drawings has produced a strong commercial demand for conversion from the old format to the new. This demand has been met partly by manual conversion (with CAD system support) and partly by automatic scanning systems (with interactive post-processing).[1] The conversion process involves the extraction of line structured data from an image and its interpretation and composition into the elements found in CAD systems.[2] Automatic techniques for extraction are well developed, but for interpretation (with the possible exception of text recognition) they are less so. This is reflected in the capabilities of commercial scanning systems, whose output still does not reach the level of interpretation required for CAD input.

The automatic extraction of line structured data can be approached via vector coding of edges or of a skeletonised image (recently reviewed by Smith,[3]) or via heuristic line-following procedures.[4–9] Methods for interpretation are farthest advanced for schematic diagrams, which consist mainly of symbols representing components, lines connecting them and text. Rules for the composition of symbols have been deployed in combination with line following,[8] characteristic pattern detection and feature point graphs[10] and direct primitive detection.[11] Methods of symbol classification by relaxation labelling, decision trees[12] and subgraph clustering[13] have been compared,[14] suggesting that attributed programmed graph grammars have the required flexibility and error tolerance. Approaches to the inference problem for these grammars have been proposed singly[15] and in combination with other methods.[16] Other approaches to document recognition[17] and early vision[18] have avoided the low level error problem by relating the low level processing directly to the interpretation problem at hand. The extraction of higher level structures for simple mechanical engineering drawings of high quality has been demonstrated by rule-based edge coding,[19] but commercially viable interpretation of particular map images has called for the dedicated concatenation and enhancement of many methods.[20,21]

Three guidelines emerge from the above brief review.

(a) The low level line extraction and the high level interpretation should not be separated. Their integration improves performance of the line extraction by providing knowledge-based enhancement, and improves the interpretation by reducing the volume of data and level of errors at the earliest stage.

(b) Although a new method may be developed on small scale or simple images, it should be designed from the outset to be extended to drawings of industrial size (A0) and quality.

(c) Although a new method may be tested against a particular task, it should be designed at the outset so that it can be generalized to others. The variety of drawing disciplines is large and even within one discipline the primitives and their relations are numerous and complex.

EXPERIMENTS IN DRAWING CONVERSION

In order to investigate the possibilities of extending the interpretative power of automatic drawing conversion to a wider variety of technical disciplines we have constructed a system based on the guidelines listed above. To permit integration of interpretation and extraction the original grey level image has been retained, and a line-following approach used. The technique has been aimed initially at linework and its geometric interpretation rather than symbols or text. In this section we outline its main features; they are described in more detail in subsequent sections. The method is then evaluated by a discussion of its response to a range of drawings and the display of results from two of these, together with measurements of performance on artificial targets containing key features from such drawings.

Line-following provides the speed that will be needed for industrial size images, and avoiding irrevocable segmentation by early thresholding allows the line-follower to retain some segmentation capabilities: this is required for images of industrial complexity and quality. The system is constructed in several overlapping modules, with a degree of redundancy and indeterminacy. This provides a robust experimental toolkit for subsequent extension and generalization.

The core of the line-follower works by testing a linear track of pixels and, while they satisfy blackness criteria, extending that track. Then the follower attempts to steer towards the medial axis of the line by a lateral exploration of grey levels and the line direction is adjusted accordingly. This extension and steering is repeated until no more extension occurs. This cyclic procedure is conventional,[6] but here the exploration of grey levels take place overwhelmingly within the line, which provides simplicity, speed and frequent immunity to junctions.

The power and efficiency of the follower is enhanced by starting the searches for new lines at the ends of the previous line. This sequential search technique utilises the information that the previous line is there. This aids the handling of complex regions of intersecting lines, which often pose problems to line-following. The intersection problem is further reduced by tracking lines in from simpler regions outside. This places requirements of flexibility on the line storage module. As successive lines are found they are stored in a manner appropriate to geometric interpretation. Lines are merged into the longest possible accurate line, they are grouped according to slope, and collinear lines are stored as such. This procedure has two added advantages: the line-follower has the freedom to find lines in parts and in any order, and the store is compact and well organized so that geometrical queries (such as whether a newly found line has been found already) can be answered with minimum overhead.

In addition to line detection and storage facilities, a procedure is needed for initiating the search for lines and terminating it when all are found. Experiments with manual initiation indicated that directing the search towards regions of high concentration of lines would give more accurate and complete detection as well as obvious gains in efficiency. There is also a requirement for image statistics on which to base the blackness criteria for line following. The bookkeeping module provides these facilities by dividing the picture up into suitable sub-areas, constructing a grey level histogram in each area and estimating the grey level values corresponding to the local white background. From these, the target number of significantly black pixels in each area are estimated, and as lines come into store the number of pixels they account for are calculated. The search is directed towards areas of large unaccounted targets, and terminated in areas where all are accounted.

INITIAL IMAGE CAPTURE AND BOOKKEEPING

The image is acquired via a commercial vidicon tube camera and frame grabber, giving a $768 \times 576 \times 7$ bit grey level image. Camera resolution is about 400 lines at centre field. Image distortion due to the camera's horizontal and vertical scanning nonlinearity ($\sim 10\%$) is reduced by averaging in and out rows and columns of pixels at the positions necessary to rectify the image of a regular square calibration grid.

The image is assessed and adjusted for light level variation and uneven camera tube response ($\sim \pm 5$ grey levels) by constructing a smoothed grey level histogram for 108 64×64 pixel sub-areas. The local value of the white background is estimated as being at the maximal population of the local histogram, and the image is inverted and offset so that all subsequent work is done on an image for which blacker is positive and the white background level is zero. The half width of the grey level distribution of the background is estimated from the histogram as the separation in grey levels between that giving maximum population and that giving half of maximum population. The average of these half widths is used to scale subsequent grey level measurements: on normally distributed data this halfwidth corresponds to about 1.2 standard deviations. The result of thresholding the image of a high

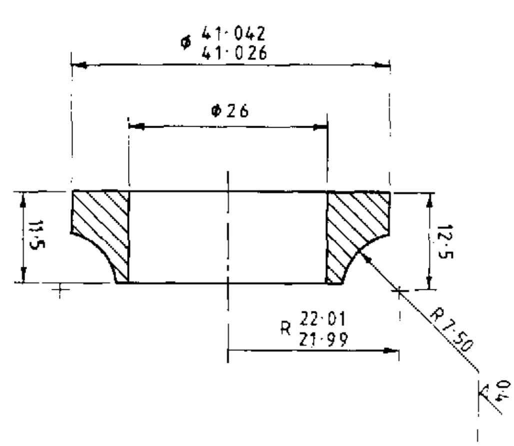

Fig. 1. A high quality drawing original.

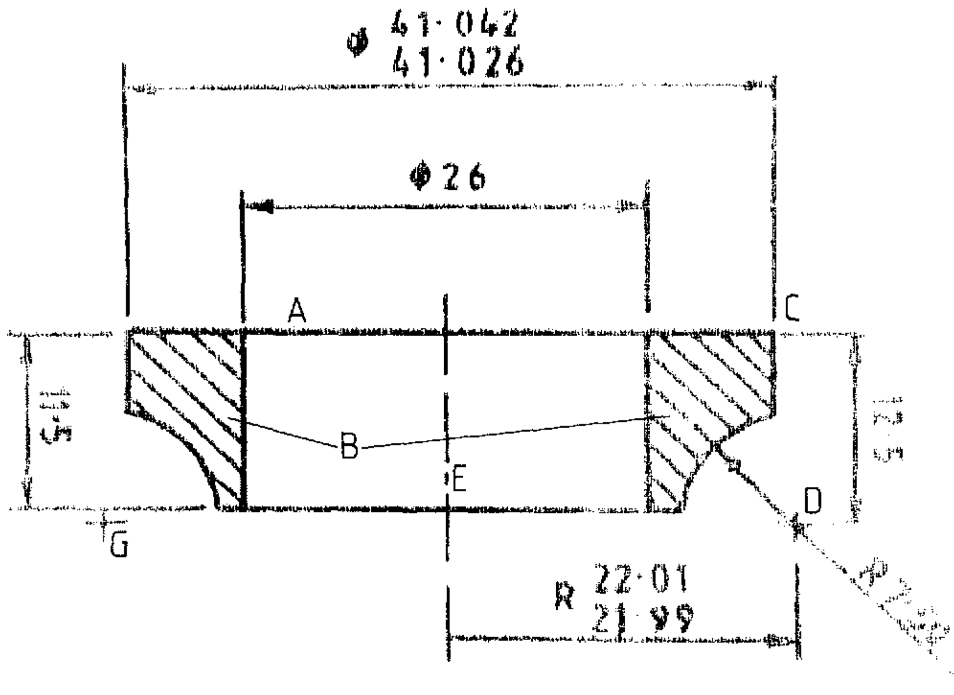

Fig. 2. Results of thresholding an image of Fig. 1 at four standard deviations.

quality drawing, size 135 × 102 mm (Fig. 1) at 4 standard deviations from background is shown in Fig. 2.

The possibility of more refined thresholding procedures[22,23] has not been taken up. Instead, the image assessment provides useful information for subsequent processing but does not make final decisions between black and white. These decisions are particularly critical in the neighbourhood of lines, and are best deferred until the time of line extraction. The adjusted image together with the half width parameter provides a consistent framework for examining the grey level image.

The bookkeeping for search control uses the same sub-area size and same threshold as the image assessment and display described above, although it is not constrained to use these values. Target numbers of black pixels are counted and stored by sub-area. Within the sub-area the sites for search origination are generated from a pattern of uniform distribution; the spacing of search sites is scaled according to the widths of lines found. This use of an appropriate coarseness of search is found to improve both speed and quality of results.

The bookkeeping module is given the end points of lines added to or removed from store. Incoming lines have their widths measured by a "first past the post" histogram technique with memory: the line width is sampled successively at fractions

$$\left\{\left[\frac{2r-1}{2^n}\right] r = 1, \ldots 2^{n-1}\right\} n = 1, \ldots \text{etc.}$$

of its length and a histogram of the widths constructed. The first width value to reach a given population is taken to be the line width. Before the next line is measured the histogram is halved, so as to provide an exponentially fading memory. This procedure overcomes the problem of false width measurements due to intersections, arrowheads, etc., but obviously needs to be tuned if a wide range of genuine widths is to be encountered. Width samples are taken by extending a transverse track of pixels perpendicular to the cartesian axis nearest to the direction of the line while those pixels remain below a threshold; the width is the number of pixels on the track. The threshold is the same as that used for the original target pixel number counting. From the width and endpoints of the line the number of pixels within it that lie in each sub-area are calculated and the appropriate area totals of pixels seen are adjusted accordingly.

When the search is reinitiated it is in the sub-area which has the greatest difference between target and total seen; areas in which the search pattern is exhausted or the total seen exceeds the target are excluded from the search. The search ends when all areas are excluded.

THE LINE-FOLLOWER

When a site for search is received from the bookkeeping section, a circular track of pixels is scanned around that site for "spots" or groups of contiguous dark pixels. If one spot is found the scan is repeated centred on that spot. If a scan produces two spots these are proposed as part of a line and forwarded to the line-follower. If more than two are found the largest two (i.e. containing the largest grey level sum) are forwarded. The threshold for this spot detection is initially set to the whitest possible without incurring excessive noise (2 standard deviations, which gives about 1 noise pixel per scan on blank areas). This threshold is subsequently adjusted to be half the measured blackness of found lines, so long as it does not fall whiter than the initial value. The diameter of the scan is adjusted to be twice the width of the lines found. This, coupled with the adjustment of initial site spacing described above, gives fineness of examination appropriate to the line widths encountered.

On receiving a pair of spots, the line growing routine attempts to follow the line they are presumed to lie on. If no line is found between them, the search is terminated. To give added ability to follow lines of uneven density there are two blackness criteria for line extension. These take the form of a fatal threshold at which a single pixel on the line terminates extension, and another somewhat blacker threshold at which the provisional end point of the line is marked. This means that small gaps can be jumped so long as blacker pixels follow; if not, the end point remains at the last provisional point marked. Both thresholds are set at fractions of the average grey level of pixels so far located on the current line (remembering that mean white background is at zero grey level). Experiments with varying these fractions indicate that the fatal threshold should be one-third to one-quarter of line greyness. The provisional threshold is set at half the line greyness so as to provide a good location of its end point, as is the threshold for steering. The steering step generates a transverse track like the width sampling step described above; the mid point of the transverse track is used as the estimate for the medial axis. When growth by extension and steering is exhausted in one direction it is attempted in the other; when both directions are exhausted the line is regarded as complete. The above procedures are a powerful way of seeking out faint lines, but are still capable of producing spurious ones too. A final check is made that the number of pixels on the line, taken with their average grey level, amounts to a statistically significant object before the line is returned.

In initial experiments the extension step was performed by tracking from the forward end of the line only, and geometric criteria were used to prevent the follower growing round gentle corners. This was later revised to employ the method of checking the extension track from the back of the line through the front after each steering step. As well as being simple and robust, this is a more exact criterion, in that the specification of a line of a certain average greyness is a well defined assertion about the grey levels all along that line.

STORAGE

The storage section receives the data on a completed line in the form of end coordinates, width estimate and depth (i.e. average grey level). Before new lines enter the store they are checked against previous lines, overlapping sections are removed and continuations of previous lines are merged. Initial geometric criteria for this were found to be unsatisfactory and better performance was obtained by regarding possible splits or merges as hypotheses to be verified against the original pixels. Thus, for example, if two lines are genuinely part of the same straight line then their ends should be connected by a track of sufficient blackness.

Having determined the novel portion of a line, if any, it is entered into the store. The storage is hashed according to line slope: each hash table entry points to a linked list of pointers; each of these points to a list of collinear lines at the head of which is the end to end data of their common line. The sense of the lines is adjusted to bring them into the slope range ($-45°$, $+135°$), and collinear lines are stored in their correct sense along the common line. All slope measurements are approximate, and lines of low aspect ratio (such as occur in text) have poorly defined slope. It is necessary to ensure that a line is classified as being within a known error of its true position in the hash table. This implies that lines of below a minimum aspect ratio must be stored in a separate list. The more finely divided the slope classification the larger this minimum ratio must be to maintain a given error in hashing. The values used here were 16 hash table entries (about $10°$ intervals of slope) and a minimum aspect ratio of 5, giving an error of ± 1 table entry. The data of the line as stored is returned to the line-follower module.

SEQUENTIAL LINE FINDING

Many of the procedures described above (image assessment, search initiation, tracking and storage) form the basis of any line-following vectoriser, but in themselves are not adequate to produce complete and accurate automatic conversion of the original.[4,6–8] Our initial experiments with sequential line-following were prompted by the quest for efficiency: in engineering drawings lines are found at, or carefully offset from, the ends of other lines. It soon became apparent, however, that the ability to track the successor to a line around a curve or profile, or across an intentional gap, contributed to the quality of the final result. For example, there is no obvious best method for collinearity assessment of a static group of vectors,[24,25] but it is clear that if the dashes on a broken line are always found in sequence along the line then assessment under this restriction becomes simpler and more accurate.

The procedure for finding successive lines consists of making circular scans centred on sites at the end points of the previous stored line (the parent line) and identifying peripheral spots. For each spot a child line is hypothesized running from the centre of the scan to the spot, and appropriate data is forwarded to the line-follower. This is similar to the way an initial search site is handled, except that the knowledge that the site is the end of a confirmed line has been used to refine the search procedure. The threshold for the scan is half the depth of the parent line (within the previous restriction to limit noise) and its radius is a multiple of parent line width. The line-follower first attempts to track a child line from the centre of the scan in the direction of the peripheral spot, but if this fails then it tracks from the periphery in the same direction. This procedure contributes to the successful extraction of broken lines and witnessed geometry. The generation of child lines has a natural tree structure, and the potentially complex network of dependent parameters is handled by recursive programming.

Although the above method produces very rapid collection of data and good tracking of successive vectors and dashes, it poses problems in eliminating looping. Drawings contain complex networks of lines, and these need to be traversed without repetition. This could be achieved if the data on hypothesized lines forwarded to the line-follower were checked against the lines already in store, and the data due to lines already found were eliminated. Initial attempts to do so showed that even if false rejection of data were at a significant level, repetition and looping still occurred. The solution was found by using the feedback from the storage module that returns the net length of line added to store at each storage operation. This length may be zero (if the new line has been entirely merged into a previous one) or negative (if the new line has prompted the merging of previously unrecognized overlaps). A line that returns a non positive net length on storage is not propagated. Furthermore, the net lengths are totalled at each level of the propagation tree, and if this total becomes non positive the propagation of the level is halted.

EVALUATION

In developing the system a range of engineering drawings was used so that the generality of the methods could be monitored. Drawings vary as to content and to quality and the latter is compounded with the quality of the imaging system before our methods are applied. As far as content is concerned, the present system attempts to extract long straight lines from complex networks to classify them according to slope, and to compose them into collinear lines. The response of the system to varying content must be considered not only insofar as it succeeds in doing these things but also in that it deals with extraneous content (e.g. arcs, text) in a reasonable way. Examination of the drawing archives of both suppliers and users of engineering equipment[26] indicated two main factors in drawing quality: line density and background noise. The former is determined principally by the medium (pencil or ink) and the latter by whether the drawing is an original or a reproduction. In practice, the complexity and poor quality of drawings will extend to the limits of human readability. An important task in evaluation is to indicate whereabouts in this wide range of possible targets our system might apply.

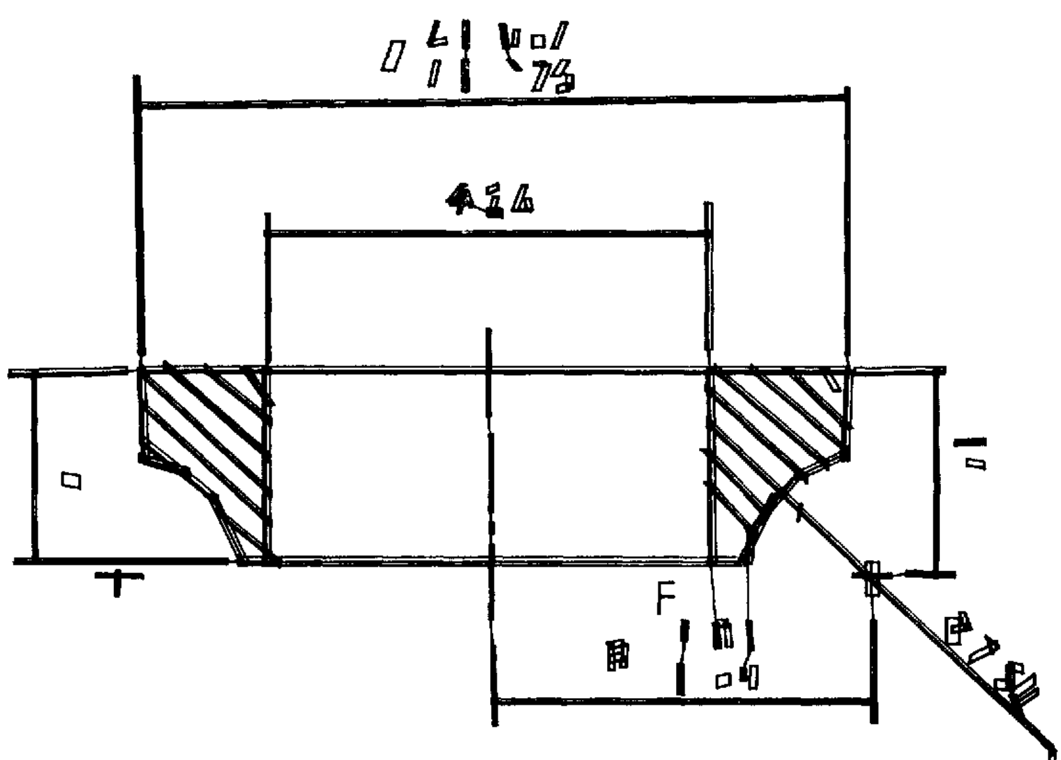

Fig. 3. Results of sequential line following on the image of Fig. 1.

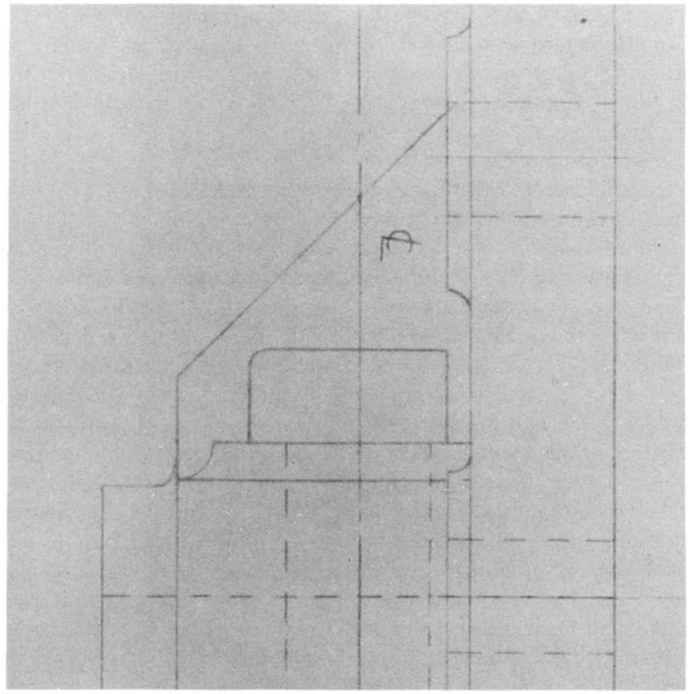

Fig. 4. A dyeline drawing original.

The primary aim of system development was in the handling of content, rather than poor quality. We first present results for an image which demonstrates the limits of the method in the handling of content but is of good quality, then those for one that demonstrates the limits of poor quality, but will not be considered for its content. Having thus crudely situated the capability of the system on real images, we then establish detailed criteria for certain aspects of performance and measure them more accurately on artificial images.

RESULTS ON THE HANDLING OF CONTENT

The sequential line follower was run on the image shown thresholded in Fig. 2; the results are shown in Fig. 3. Lines are plotted as an outline (showing the width) and a core line (showing the actual pixels tracked). Collinear lines are joined by a tie line. Lines of low aspect ratio have no core line: these are 45 in number, out of a total of 124 lines found. A profile of execution times on a Whitechapel MG1 (32016 CPU at 8 MHz) showed that approximately 67 s was taken to generate these lines, roughly equally divided between initial search, following, ordered storage and sequential search.

The image of Fig. 2 contains a number of important features: a physical outline (A), areas of crosshatching (B), text, dimensions witnessing the physical outline (e.g. at C) and the centres of arcs (at D), a dashed line (E). The method appears to work well for graphics linework and to produce full length vectors with high immunity to intersections. Even the dense areas of linework around cross hatching are largely successful. The sequential following procedure and flexible storage has allowed lines in these areas to be extracted and associated with the overall linework. It is observed that although the line-follower may halt at an intersection due to the failure of the recentring procedure, leaving a partial line, the complete line is successfully captured by being grown in from the other direction and the partial lines merged.

The use of collinearity to associate objects in engineering drawings is particularly important: examples here are at C, D and E. The success or otherwise of the method is governed more by the sequential nature of the linefinder than by the details of the geometrical criteria of collinearity. For example, the arms of the centreline cross (at D) have been extracted and correctly associated with their collinear dimension witness lines by the follower tracking in along the witness line, moving over the gap, then continuing along the arm of the cross. Reference to Fig. 2 shows how difficult this extraction would be if it had to be performed on the cross alone without benefit of the witness lines. A similar effect is found with dashed lines: if two dashes are examined at the opposite ends of the line they may well fail the collinearity criteria, but if the dashed line is "grown" dash by dash the correct association into a collinear object is straightforward.

The limitations of the collinearity procedures are seen in that coincidental collinear lines have been found in text, and between text and the physical outline (at F, Fig. 3), yet the arms of the crosses which are collinear (at D and G, Fig. 2) have not been detected as such. Thus the success of collinearity

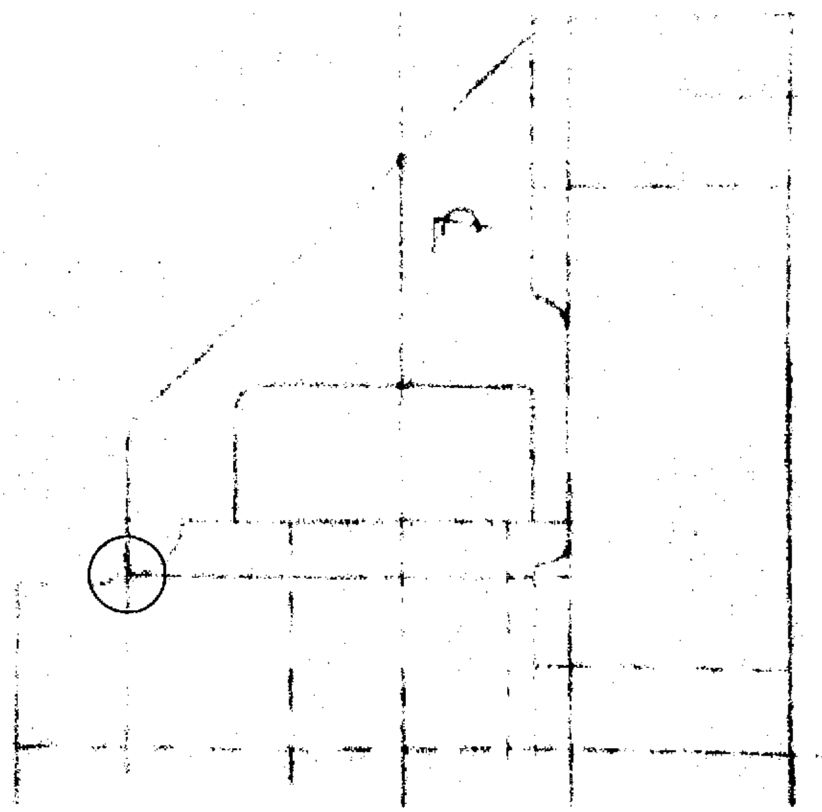

Fig. 5. Results of thresholding an image of Fig. 4 at three standard deviations.

assessment requires a better use of context than is employed here; it seems unprofitable to measure the performance of the criteria in geometrical terms while the cause of their failures lie elsewhere.

The accurate detection of small gaps in linework is essential to its correct extraction. Reference to Fig. 3 (at C for example) shows that these gaps may only be a few pixels wide. This is particularly demanding in that there may be lines of lighter weight in the same area of the drawing. Thus the line follower has to correctly register the intentional gap in a bold line, whilst being able to detect a faint line as continuous. This is achieved in the present system by making the threshold of line tracking depend on the grey levels sampled along the line length. The tracking of faint lines is considered further in the next section and is measured, together with the performance on gap detection, on artificial images in the section after that.

Some essential limitations of the method are evident: text is handled poorly, and is in effect no more than noise; arrowheads are not recorded. An example of content beyond the basic sequential capabilities of the present system is at the almost tangential arc and hatch line in Fig. 2. Reference to the original image shows that this could be decoded only if the system sought out cross hatching as such and sought out arcs of circles as such. It is again unprofitable to measure the performance of the system in tasks such as these for which it has not been designed. It is, however, a useful pointer to further work.

RESULTS ON HANDLING POOR IMAGE QUALITY

A common form for drawing storage is the dyeline copy, in which contrast may be poor and density uneven. An example of such a drawing is shown in Fig. 4; It would be possible to process this by using high resolution, low noise digitization, followed by noise removal. To test the response of our system to poor quality images, we use the image direct from the low cost CCTV camera described above, and apply the sequential line following methods. The result of capture and threshold at three standard deviations is shown in Fig. 5. An enlarged view of part of the image is shown in Fig. 6. It is important to note that the system parameters remain the same for this test as for the previous one, as we are particularly concerned to investigate the ability of the system to adapt to differing images. Results from the adaptive sequential line tracking system are shown in Fig. 7.

It is clear that this image shows the limits of the system performance in the extraction of faint lines. To assess the faintness of a line we compare its depth (the mean grey level along the line as measured by the line follower) with the noise standard deviation of the white background (as measured by the image assessment module). The mean depth of lines extracted here is 2 standard deviations, as compared with a mean depth of 12 standard deviations in the previous image. It appears that the dynamic thresholding that was developed to ensure reliable extraction of content on

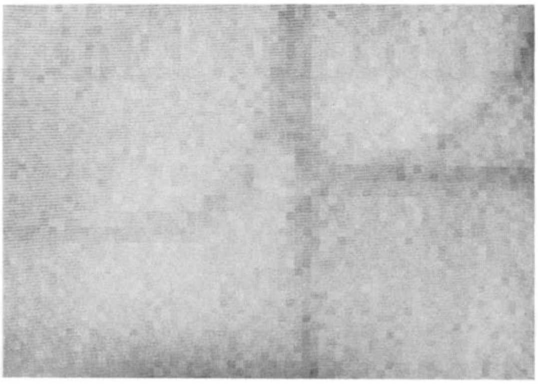

Fig. 6. Enlarged view of part of the grey level image of Fig. 4.

good quality images has also provided a line enhancement effect on a poor quality image.

TESTS ON ARTIFICIAL IMAGES

From the previous two sections we have seen that certain aspects of the performance of the line-follower can be quantified. These are: the complete tracking of lines, the extraction of faint lines, the detection of small gaps in bold lines and the handling of intersections. We wish to investigate the effects of noise on these aspects: in the present system the background noise level is first assessed and subsequent grey level measurements are all scaled relative to this noise level. We can therefore use line depth as our independent variable, and report results as a function of line depth. This depth will be quoted as a multiple of noise standard deviation, σ. Thus the mean depth of lines from the real images in the previous sections are 12σ and 2σ respectively.

Having identified a particular feature of interest, e.g. a horizontal line with a small gap in it, many copies of it were presented to the system for processing. The location of an original with respect to the digitization grid and with respect to the initial search pattern is essentially random, so with the artificial features each copy was assigned random offsets horizontally and vertically. These offsets were uniformly distributed over an interval at least as large as the separation between search sites. The fractional part of this offset was used to map each pixel of the feature into 4 pixels of the final image, proportioned as by ideal digitization. The final image had a superimposed uniform Gaussian noise level of $\sigma = 3$ grey levels, and the line depths vary from 2σ to 8σ. For each set of experimental conditions the line-following system was run on between 63 and 108 copies of the feature and the lines

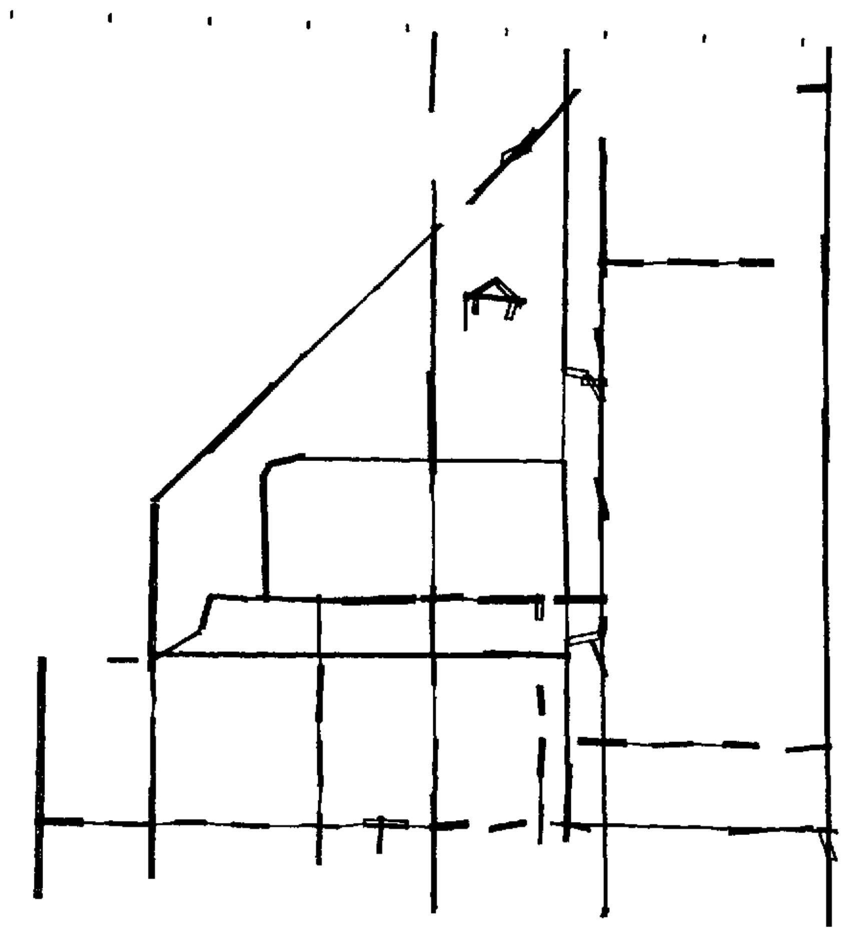

Fig. 7. Results of sequential line following on the image of Fig. 4.

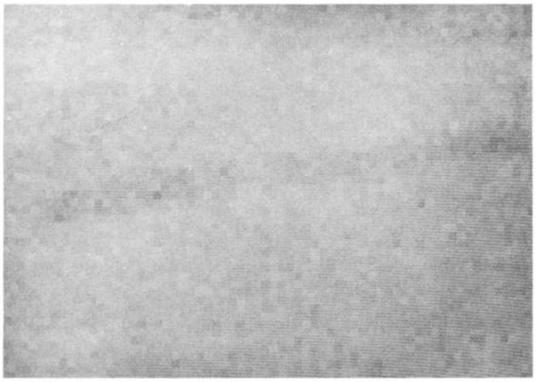

Fig. 8. Enlarged view of part of an artificial image containing lines of width 4 pixels and depth 3σ with a gap 3 pixels in extent.

found were compared with those in the original.

Some general characteristics of the response were noticeable. The mean observed depth of found lines was less than the artificial depth by about σ: results will be quoted in terms of the artificial depth, rather than the observed depths used for real images. The line width measurement, bookkeeping and line detection, which are based on fixed thresholds, degraded sharply as line depth went below these thresholds. The performance was not sensitive to angular orientation of the feature.

Line detection rates are shown as a function of depth in Fig. 9 for lines 60 pixels long and 4 pixels wide. The degradation at low depth corresponds to the fixed detection threshold of 2σ. The mean and standard error of the end position of the best found line compared to the actual end position of the artificial line are also shown in Fig. 9. It is clear that as line depth decreases the tracking tends to stop prematurely, producing short lines. The mean error in end position is also shown as a function of line width for a depth of 3σ in Fig. 10.

The gap detection failure rate is shown in Fig. 11 as a function of line depth, for lines of length 60 and width 4 containing gaps of: (a) 2 pixels and (b) 3 pixels extent. Part of the latter image is shown enlarged in Fig. 8 for a line depth of 3σ. Gap detection is contingent on line detection, and is thus only shown where line detection rates are high.

Measurements on intersecting lines were carried out for line pairs which intersected near their midpoints and for line pairs which intersected 7 pixels from the end of one line. The latter tested the immunity to intersection of the line tracking alone, the former tested for the combination of tracking and merging in store. Failure rates for the latter were 7% and for the former 2%.

CONCLUSIONS AND DISCUSSION

The present work is an experiment in the integration of the extraction and the interpretation of line structured data. It has been applied to the automatic

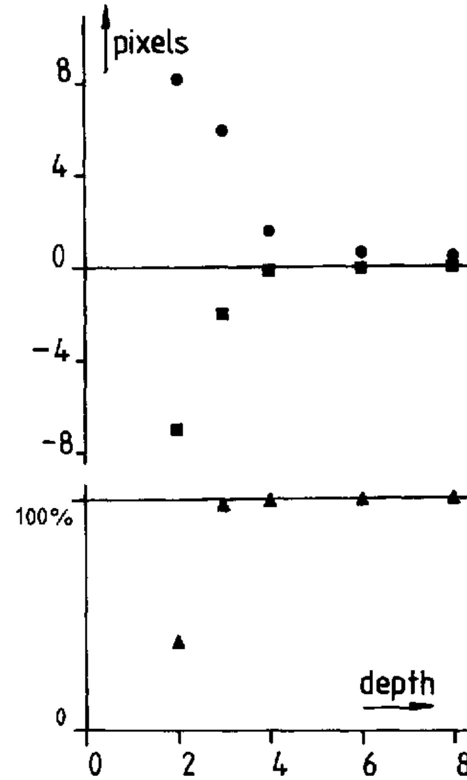

Fig. 9. Standard error (●) and mean error (■) in tracking, and detection rate (▲) vs line depth.

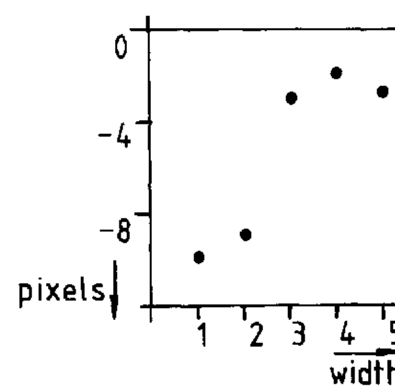

Fig. 10. Mean error in tracking vs line width at a depth of 3σ.

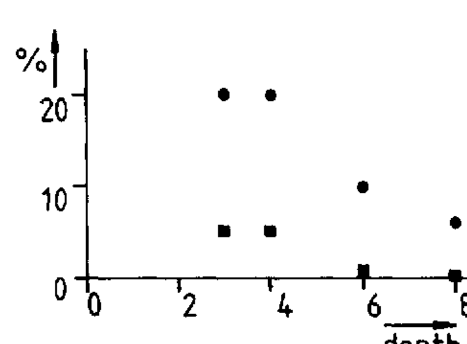

Fig. 11. Gap detection failure rate vs line depth. (●) 2 pixels gap; (■) 3 pixels gap.

extraction and geometrical classification of long straight lines and broken lines from mechanical engineering drawings with some success. Firstly a line primitive has been sought, extracted and classified in the context of the grey level image. This has permitted extraction from regions containing many intersecting lines. Complex areas of linework and broken lines have been included in the scheme by sequential search conditioned by primitives already found. This has enabled the extraction of broken lines from regions of multiple intersection. Collinearity assessment has been found to be limited by the level of interpretation employed rather than by the goodness of the geometrical criteria. Similarly, the successful extraction of parts of higher level drawing entities seems to depend on the explicit seeking of these entities.

The passing of data on lines already found down the sequential search path together with use of the full grey level image has produced a line enhancement effect. This has been measured on particular features in artificial images of different qualities. In particular, this has enabled the detection of small gaps in dense lines together with the following of faint lines.

Let us review the present method in the terms set out in the introduction. In terms of line extraction it must face comparison with skeletonization and tracking procedures, which have the advantage of guaranteeing production of complete networks of lines from text or graphics. The apparent disadvantage of line-following in this respect is annulled when quality of data is taken into account: in order to extract useful vectors and reject spurious ones from complex areas of linework or text, the skeletonized data requires as great and as fallible an application of heuristics as are deployed in the present approach. Moreover, the sequential adaptive line-follower deals directly with the original image and employs enhancement and verification techniques which further improve data quality and reduce interpretation problems. It is important to recognize that there is no ideal general line extraction method: the extraction of meaningful data from drawings must depend on the level of interpretation used.

In terms of interpretation, the present method is based on the implicit assumption (or prior knowledge) that the image contains intentionally straight lines related to each other geometrically. Insofar as this is correct, it contributes strongly to the performance of the system via the integration of interpretation and extraction. For future work, the key question is how to assemble and run a conversion system for complex line drawings of mixed variety and quality. Successful systems have employed ad hoc specialized methods,[8,10,20,21] implying at best onerous reprogramming and at worst complete rebuilding if they are to be adapted and generalized. This has been recognized in the use of rule based and grammatical techniques[11-16] to investigate more general approaches. These latter have encountered the problem of fallibility in primitive extraction, with incomplete success. It seems that the combination of the present integrated methods with a hierarchical control structure would provide a valuable new route to its solution.

SUMMARY

Despite advances in automatic recognition of symbols and text, the transfer of existing engineering drawings from paper or film onto computer-aided draughting systems is still performed by largely manual methods. This paper describes the development of a line-following approach applied to digitized grey level images of engineering drawings, aimed at extraction of geometrically significant relations from the linework. The system is fully automatic, so that image assessment, search initiation, line finding, ordered storage of found lines and their composition into collinear lines takes place without operator interaction or adjustment of parameters. It is thus a model system, which can be judged in terms of overall performance in conversion to CAD format.

The grey level image is not thresholded explicitly, but the grey level corresponding to the white drawing background is measured and its noise level assessed. These figures are used as the basis for a dynamic bi-level thresholding procedure by the line-follower, which improves performance on poor and variable contrast images. The search for lines takes a branching sequential form, and properties of lines already found are passed on to the current finding process. The initiation of search sites also adapts to the widths and darkness of found lines.

The composition of collinear lines is improved by the incorporation of gap jumping in the line finding sequence, and the limitation of this particular interpretative step are explored. The line-follower employs direct validation of pixel tracks on the original image, as does the merging procedure for adding new lines to the store.

The results of running the system on high and low quality images are discussed and handling of key drawing features noted. Performance on these features as a function of noise is measured using artificial images.

It is concluded that the integration of the direct finding of higher level objects with their storage and interpretation provides a powerful method for tackling drawings of high complexity and poor quality.

Acknowledgements—The author wishes to thank Mr M. E. Dunn for many useful discussions, and the ACME directorate of the SERC for funding the latter part of this research.

REFERENCES

1. J. Hofer-Alfeis, Automated conversion of existing mechanical engineering drawings to CAD data structures: state of the art, *CAPE '86: Conference on Computer Applications in Production and Engineering*, Copenhagen, May 1986, E. A. Warman, K. Bo, L. Estensen (eds), North Holland, Amsterdam (1986).

2. Initial Graphics Exchange Specification (IGES), Version 3.0, NBSIR 86-3359, National Bureau of Standards, Washington, DC (1986).
3. R. W. Smith, Computer processing of line images: a survey, *Pattern Recognition* **20**, 7–15 (1987).
4. M. C. Fulford, The Fastrak automatic digitizing system, *Pattern Recognition* **14**, 65–74 (1981).
5. I. Nackunstz, An automatic digitizing system for line drawings, *6th International Conference on Pattern Recognition, IEEE Computer Society*, p. 1196 (1982).
6. W. Black, T. P. Clement, J. F. Harris, B. Llewellyn and G. Preston, A general purpose follower for line structured data, *Pattern Recognition* **14**, 33–42 (1981).
7. T. P. Clement, The extraction of line structured data from engineering drawings, *Pattern Recognition* **14**, 43–52 (1981).
8. T. Sato and A. Tojo, Recognition and understanding of hand drawn diagrams. *6th International Conference on Pattern Recognition, IEEE Computer Society*, pp. 674–677 (1982).
9. L. Caponnetti, M. T. Chiaradia, A. Distante and M. Veneziani, A track-following algorithm for contour lines of digital binary maps, *Digital Image Analysis*, S. Leviddi (ed.), pp. 149–154, Pitman, London (1984).
10. X. Lin, S. Shimotsuji, M. Minoh and T. Sakai, Efficient diagram understanding with characteristic pattern detection, *Comput. Vision Graphics Image Process.* **30**, 107–120 (1985).
11. D. S. Tudhope and J. V. Oldfield, A high-level recognizer for schematic diagrams, *IEEE Comput. Graphics Applic.*, 33–40 (1983).
12. H. Bunke, Automatic interpretation of lines and text in circuit diagrams, *Pattern Recognition Theory and Applications*, J. Kittler, K. S. Fu and L. F. Pau (eds), pp. 297–310, D. Reidel, Dordrecht (1981).
13. H. Bley, Segmentation and preprocessing of electrical schematics using picture graphs, *Comput. Vision Graphics Image Process.* **28**, 271–288 (1984).
14. H. Bunke, Experience with several methods for the analysis of schematic diagrams, *Proceedings of the 6th International Conference on Pattern Recognition, IEEE Computer Society*, pp. 710–712 (1982).
15. G. Maderlechner and J. Kreich, Knowledge based pattern recognition in disturbed line drawings, *Proceedings of the 2nd International Conference on Image Processing and Applications*, IEE publ. no. 265, pp. 139–143 (1986).
16. P. Kuner and B. Ueberreiter, Knowledge based pattern recognition in disturbed line images using graph theory, optimization and predicate calculus, *Proceedings of the 8th International Conference on Pattern Recognition*, pp. 240–243, Paris (1986).
17. T. Pavlidis, A vectoriser and feature extractor for document recognition, *Comput. Vision Graphics Image Process.* **35**, 111–127 (1986).
18. P. H. Mowforth, J. Jelinek and Z. P. Jin, An appropriate representation for early vision, *Pattern Recognition Lett.* (in press).
19. U. Cugini, G. Ferri, P. Mussio and M. Protti, Pattern-directed restoration and vectorization of digitized engineering drawings, *Comput. Graphics* **8**, 337–350 (1984).
20. J. F. Harris, J. Kittler, B. Llewellyn and G. Preston, A modular system for interpreting binary pixel representation of line structured data, *Pattern Recognition Theory and Application*, J. Kittler, K. S. Fu and L. F. Pau (eds), pp. 311–351, Reidel, Dordrecht (1981).
21. T. Hoshino, S. Suzuki and M. Kosugi, Automatic input method for large scale maps, *Proceedings of the 8th International Conference on Pattern Recognition*, pp. 449–453, Paris (1986).
22. J. S. Weszka and A. Rosenfeld, Histogram modification for threshold selection, *IEEE Trans. Systems Man Cyber.* **9**, 38–52 (1979).
23. J. Kittler, J. Illingworth and J. Foglein, Threshold selection based on a simple image statistic, *Comput. Vision Graphics Image Process.* **30**, 125–147 (1985).
24. A. Broder and A. Rosenfeld, A note on collinearity merit, *Pattern Recognition* **13**, 237–239 (1981).
25. A. Scher, M. Shneier and A. Rosenfeld, Clustering of collinear line segments, *Pattern Recognition* **15**, 85–91 (1982).
26. S. J. Cheetham, *Ph.D. Thesis*, University of Sheffield, U.K.

About the author—S. H. JOSEPH is a lecturer in engineering design at Sheffield University. He obtained a B.A. in Natural Sciences at Cambridge University in 1971 and a Ph.D. in Applied Physics at Leeds University in 1976. Following research in machinery design at Imperial College, London, his interest in CAD and its industrial applications led to research at Sheffield University on machine reading of engineering drawings. He is currently leading a project on automatic drawing conversion funded by the Application of Computers to Manufacturing Engineering directorate of the Science and Engineering Research Council.