FINDING PICTURE EDGES THROUGH COLLINEARITY OF FEATURE POINTS

by

F. O'Gorman and M. B, Clowes

Laboratory of Experimental Psychology
University of Sussex
Brighton, Sussex, England

Abstract

The recovery of straight picture edges from digitisations of scenes containing *polyhedra* ('line finding') is central to the functioning of scene analysis programs. While recognising that recovery properly involves a computational mobilisation of a great deal of knowledge-supported context, there remain some "basic issues of representation which govern the way in which the primary data - grey levels - are addressed. The paper describes a parametric representation of straight picture edges and its procedural deployment in the recovery of edges from digitisations *of scenes* whose contents arc essentially polyhedra with strong visible shadows.

1. Scene Analysis

Contemporary thinking in scene analysis is focussed primarily upon efforts to deploy higher level knowledge, e.g. *of* support, lighting and scene composition12 to achieve an 'integrated' interpretation of pictures in the form of grey scale data from TV cameras or comparable picture digitisers. The need to involve such knowledge has "become progressively more apparent as line and region finders10 5 2 1 have demonstrated that a pass-oriented approach to the recovery of edges and surfaces is only partially successful.

Our own studies are focussed primarily upon this same endeavour: a crucial feature of our research is the formulation of a task in which there is a rich "but hopefully tractable range of knowledge which could "be mobilised in such an integrated analysis. The task had its origins in part in the dictum we attribute to Guzman7 'if you recognize a foot you know where (in the picture) to look for a leg. . . . .' The scenes we are working on are polyhedral in content but polyhedra constrained to form a puppet figure (Fig.l) In so doing we hope to bring to bear just that sense of *context* which Guzman evokes, in order to assist the processing of some picture or scene fragment. Contemporary programs e.g. Falk£ do just this but within the restriction that context can only be mobilised to help in identifying the elements of a single body (of which Falk has 9 distinct models) except for the contextual inferences provided by the requirement that bodies be simply supported. Winston's account12 of Finin's work goes beyond Falk in precisely the direction we have adumbrated although it is unclear what range of architectures is being contemplated beyond the 'arch' he describee.

Notwithstanding these attempts to reformulate approaches to scene analysis as integrated or ' heterarchical' (as epposed to •hierarchical' pass-oriented schemes) there remains a clear requirement for procedures which are capable - with or without context - of recovering three-dimensionally significant entities in the primary picture data. One recent attempt to mobilise context to assist a line finder is described by Shirai11 . After an initial 'pass' to recover an outer contour of the group (or groups) of objects visible in the scene, the line finder is d'reefed to significant locations e.g. concavities, in this contour as the origin of a search for further edges. Heuristics of various sorts e.g. 'Jock for an *edge* parallel to one already found' are used to further constrain the direction of the search. The search itself proceeds from the immed iate vicinity of that location. Inspection *of* typical scenes suggests that at many such locations on an edge, the evidence may be meagre although elsewhere along its length the evidence is much stronger. Thus a search initiated by a program such as Shirai's may i'ail because it has a very myopic way of addressing the picture space to find edges. This paper is concerned primarily with a study of an alternative to this locai search strategy, by exploiting the collinearity of the feature points making up a straight picture edge as an addressing mechanism.

The exploitation of this property to find straight lines is not in itself original. Both Duda and Hart* and Griffith6, describe a parameter!sation of feature-point data which is *of* assistance in recovering straight lines. Indeed our own work derives directly from that of Duda and Hart. These earlier workers however leave many problems unresolved (e.g. Duda and Hart p.13 col. *2)* not least of which is the nature of the methods by which the parameterised data is to be concatenated into 1ines. The study which we have carried out provides a fresh perspective from which to view the 'myopia' of line finders and gives at the same time a more detailed picture of the difficultjes that remain.

2. The Line Finding Process

2.1. The Data.

The data with which we have worked derive from positive photographs of scenes lit by a single light source. Our primary concern is with scenes involving the puppet figure which is painted a uniform matt grey, althougi we have done some work on matt white polyhedral blocks. The photograph is digitised by a flying spot scanner* with a linear

+We are grateful to Dr. J.R.Parks of the National Physical, Laboratory for making this Scanner available to us.

response to reflected light quantised into 61+ grey levels. In practice the range is atout 1+0 grey levels. Inspection of the digitisations reveals that over perceptually uniform regions of the picture the variation in grey level between adjacent picture points ('noise') increases approximately linearly with average grey level to an extent (about 1+ units peak-peak at a grey level of 1+0) which becomes a nuisance in computing the brightness differential within the bright areas of the picture. We have not attempted to identify the source of this 'noise' nor the extent to which it originates in the photomultiplier system or in the photographic process.

2.2. Mappingthe data into r.a space.

As indicated in Section I we wish to use the 'global' collinearity of feature points as a major factor in recovering straight picture edges rather than their contiguity. Following Duda and Hart4 we capture this property by using a variant of the so-called Hough transformation. A straight picture edge is treated as line in the picture plane, parameterized "by a the orientation of, and r the length of the normal from the origin to the line. *The* equation *of* the line is then

$$x \cos \alpha + y \sin \alpha = r \qquad 1$$

A line is thus completely specified (except for its extent - an important consideration in the present task) by a (r,a) pair. A point (x,y) in the picture plane may be thought of as the *common* point *of* intersection of a family of lines, each member of this family having an orientation a in the range (0,180°) and a value of r determined by (1). A point therefore corresponds to a family of (r,a) pairs. A set of collinear points in the two-dimensional Space of the picture has a set of line families wjth. a. common (r,a). member. The basic principle of the technique described by Duda and Hart4 and related methods such as the so-calledHough transformation is to accumulate evidence for such common members by mapping picture points into (r,a) space.

The picture edges of interest are associated with local changes in picture brightness (feature points) and thus the points which form these edges are (x,y) locations in the picture where thegradient G of the picture brightness significantly exceeds the values of G associated with locations in uniform areas of the picture. We can compute G at the location (m,n) by obtaining a measure of the gradient in the x and y directions (DX^DY^) as estimated from a 3 x 3 grid centered on (m,n):

$$DX_{mn} = \sum_{i=-1}^{i=+1} I_{m+1,n+i} - \sum_{i=-1}^{i=+1} I_{m-1,n+i} \qquad (2)$$

$$DY_{mn} = \sum_{i=-1}^{i=+1} I_{m+i,n+1} - \sum_{i=-1}^{i=+1} I_{m+i,n-1} \qquad (3)$$

The magnitude of G takes the form

$$G_{mn} = \frac{\sqrt{DX^2_{mn} + DY^2_{mn}}}{S_{mn}} \qquad (4)$$

Where

$$S_{mn} = \text{constant} + \sum_{i,j=-1}^{i,j=+1} I_{m+i,n+j} \qquad (5)$$

The term Smn is a scale factor which reduces the magnitude of G for brighter picture points to combat the noise (described earlier) encountered in bright areas of the picture. In the pictures we have studied well over 75% of the picture locations give effectively zero values for G as can be seen by imposing a threshold which eliminates these points. The result displayed two-dimensionally (Fig.2) contains most of the picture edges of interest. It is only the values of G for these locations that are used in the recovery of picture edges. The threshold in a fixed one which can be regarded as a constant in the picture-taking and picture-digitising *process.* For each picture location so identified we could compute the family of r,a pairs it determines and begin to accumulate evidence by incrementing the appropriate counters in a two dimensional array, where each counter will record the number of locations giving rise to a value of r,a lying within the range appropriate to that counter, a range Dr,Da, which is determined by the quantisation we impose on the r,fi- space. We depart from this method of accumulating evidence (the method described by Duda and Hart4) in two important respects.

Firstly by making use of local evidence to determine uniquely the value of a. This local evidence can be derived from the ratio DX/DY *Which* gives the direction in which the gradient lies. Since the gradient should be perpendicular to the picture edge a is given by

DY
tan a = *ffc* (6)

There is of course a risk that this estimation will be imprecise especially for example at junctions between contiguous picture edges, but much of the uncertainty arises for picture points where the value of G is low, which we have already excluded from consideration by thresholding G. That these local estimates are consistent and reasonably reliable may be Judged from Fig.*2,* which displays the direction of G at each of the points where G exceeds threshold. If a, x and y are all known then r is determined, it can however be computed in two ways given that a is going to boquantised. That is, r can be computed either directly from the value of a given by (6) or after that value has been quantised. The two alternatives give the r,a counters differing geometrical fields of view: we shall not pursue this point of detail here save to say that we have adopted the second approach, namely we quantise a before computing r. The quantisation steps used are Da=10° and Dr=3 picture rows (columns).

The second point of departure relative to the method of Duda and Hart4 is to increment the counter "by the magnitude of G rather than "by one for each contributing picture location. Our motivation in so doing is to ensure that we readily find strong picture edges (at the risk to be sure of masking weak ones) and to ensure that those counters for which a is accurately computed from the local evidence (because G is large) are emphasised.

The resulting histogram H(r,a) for the digitisation of Fig.I is given in Fig.3. The origin of coordinates is in the centre of the picture (Row 133, Column 129 of Fig.2), it is this p-oint from which r is measured. In the piurely algebraic interpretation of the transformation a ranges between 0° and 160°; the method of computing a from the local evidence yields values which range between 0° and 3600 since we wish to distinguish which side of the picture edge is the darker. Thus for a given orientation of the picture edge there are two possible values of a corresponding to one or other side being the darker. We have labelled the *a* axis *of* the histogram according to the convention i'or displaying a used in Fig.2 as well as in the more conventional way, so that it is possible to pick out *correspondences* between peaks in the histogram with bands of picture points in Fig.2. When one does this e.g. for the bin r = 72, a = 180° (identifier 'I') corresponding to the shadow of the puppet's shoulders, we immediately grasp the fact that the bin in no sense represents the lines we see in Fig.2. For this bin contains both these picture edges although they are not joined, because their constitutent picture locations have the same r,a values associated with them: they are collinear. The quantisation also has its effect too in dividing a single edge between adjacent histogram "bins; for example the points forming the lower edge of the leg shadow in Fig.2 do not all have the same orientation and are therefore allocated to different histogram bins. Elsewhere in the picture parallel edges which are close together have been allocated to the same bin. The latter two defects can be exposed more readily if we consider the locations contributing to each histogram bin. For conveniens let us restrict attention to only those bins of Fig.3 whose magnitude exceeds 5. We can now assign an identifier, chosen again froiii the character set of the line printer, to each bin and display in the two-dimensional format of Fig.2 the locations which belong to each of these major bins. The result (Fig.4) illustrates the points we have been making only too well. At the same time the lists of contributing points for each bin, referred to hereafter as the contributing points list(cpl), provides a basis for surmounting the first ' two problems posed by the (r,a) transformation. The data structure we refer to as the histogram H(r,a) contains for each bin a list (the cpl) of the (x,y) locations contributing to that bin as well as the summed magnitude H of the contributions of those locations.

2.3. Recovering straight lines from the r,q histogram.

The recovery of bounded straight lines from the histogram centres upon a process - at this stage full of crude rather ad, hoc devices - of recovering sub-sets of the locations contributing to the histogram: subsets which correspond as closely as possible with the lines we would sketch in on Fig.I. The starting point for the search is a cpl from a major bin. The method is iterative: search for a new line always begins with the largest bin remaining in the histogram, and search terminates when no bin exceeds the arbitrary value of 5. These subsets which we shall call putative line lists (pll), are recovered in a series of steps.

2.3.1. Recovery *of* putative line lists. The cpl of the largest bin in the histogram is taken as a basis for forming a pll. A list of the points in the cpl is made, ordoral on either x value (for a</|f)° or a>l-J3°) or y value ()|fj° < a <l3;j°). This list is now scanned from one end with a view to finding other cpl's containing points which contribute to the edge. For each point in the list we compute G, a for those neighbours of the point which are not already in the list. By neighbours of a point we mean the eight locations which immediately bound the point. For each of. those neighbours having a G above threshold and a value of *a.* not more than 10° different from that of the initial cpl, the cpl containing each such neighbour is merged into the list and the scan is resumed. During this scan gaps in the list are tested i'or. A gap occurs when the next point on the list has an x value (if the list is ordered on x) or y value (if the list is ordered on y) which differs from thatof the current point by more than some constant which at present is set at 1 unit. If a gap is found then the points in the list beyond the gap are assumed to be from a d if fere: it, collinear, edge, and are discarded.

Since this scan adds cpl's to the list the *beginning of* the list may have been extended, and it is possible that this extension contains gaps. It is therefore necessary to scan the list again in the reverse direct3on looking for a gap. As before, points (if any) beyond the first gap found are discarded.

The points remaining in the list after these two scans are the required pll. They are removed from the appropriate cpl's of the histogram and the value of H for each cpl decremented accordingly. Recovery of pll's then continues with the cpl from the new largest bin in the histogram as a basis. *The* recovery process in terminated when there art- no bins left with magnitude greater than 5.

The set of pll's extracted from the digitisation of Fig.I and the histogram of Fig.J *is* illustrated in Fig.5), again using a unique (new) identifier for each pll;

2.3.2. Converting putative line lists into lines. The ordered list of points emerging from the procedure described above is subjected to two tests to determine its acceptability. Its 'strength* is assessed by summing the values of G for the points contained in the list. If their summed value is less than *2,* the pll is rejected.

The next step is to fit a line to the points *of* the pll using a least squares approximation with each point weighted by its G value.

The quantisation on r sometimes causes points from adjacent parallel edges to fall into the same "bin. The algorithm for recovering pll's cannot at present handle this situation and produces a single pll for the edges. Such pll's can be detected from the large spread of points about the fitted line. The measure used for this is the variance of the perpendicular distances of the points from the line. Whe $\sum G_{pll}$ is used to reject a pll, the value of the variance computed is not, since the pll and the line fitted are structurally significant objects which other programs can be expected to make use of.

The accepted pll's now have to be assigned end points. The technique involved first reorders the points of the pll in the direction of the line fitted to them, i.e. orders them according to their perpendicular distance from a normal to the line. The extremum points of this list are not usually very good candidates as end points; the lists tend to have tails with extremum points some distance off the fitted line. The method adopted is to scan the list noting on which side of this line the points lie, until the first cross-over is noted. This cross-over point approximates to the intersection of the line with a boundary around the set of points, and is taken as the endpoint. The procedure is repeated at the other end of the list.

The line segments superimposed on the pll's in Pig.5 were obtained from those pll's by the above method. Pig.6 is a plot of the same lines produced using the GROATS graphical system8; for ease of inspection lines whose variance exceeds 10 units are shown dotted. Results for other pictures of this same type, using the same parameter settings, are illustrated in Fig.7.

2.3.3. Program details. The program was implemented in Algol 66R3 on an ICL 1906A, and uses 35-45K of core memory. The processing time depends on the complexity of the pictures, and varies from about 30 seconds for simple blocks scenes to about 90 seconds for the more complicated puppet pictures.

3. Results

Perhaps the most obvious question to ask of this line finder is 'does it work?' Unfortunately such a question can only be answered relative to the success or failure achieved in using the line data for some recognition or interpretation task. At the time no such evaluation has been attempted; the interpretation routines simply do not exist. However, we can adopt the requirements that some reported scene analysis system would appear to impose upon its line finder, and assess performance on that basis. A good candidate is Shirai's program11 since the context free phase of line finding is rather well defined in that program, being aimed at the recovery of the scene contour. That is the set of 'outer' edges of the object or group of overlapping objects visible in the scene. The pictures we are working with, unlike those used by Shirai, contain substantial areas of visible shadow which can be regarded as augmenting this contour, for example the lines in Pig.6 numbered 1, 36, 19, 16, 17, 9, 24......Ignoring internal contours (e.g. 27, 22, 20 in Fig.6) we can form a simple numerical measure of success by comparing the number of picture edges in the outer contour estimated by eye for the data, with the number of these edges appearing as lines in the output data. In Table 1 these counts are listed for five scenes.

Scene No.	Number of contour picture edges in scene.	Number of corresponding lines in output.
211	25	17
212	24	14
214	28	19
217	25	23
225	29	17

Table 1.

The results suggest - relative to Shirai who apparently never fails to recover the complete outline - a rather disappointing performance. There is however a major difference between the data for the two line finders namely that Shirai was working with matt white blocks on a black velvet support plane whereas we are working with matt grey polyhedra (the puppet) on a matt white support plane. Moreover the optical quality especially focus of some of our photographs is much inferior to that of Shirai (our scenes extend over a greater spatial extent). Thus our worst cases 212, 225, suffer from very badly defocussed shadows where many outline picture edges are missed,

A more fragmentary analysis of performace can be given by looking at some well-defined mistakes. Specifically a number of picture edges are missed due to design defects of the algorithm which are potentially easily rectified. We can identify three main causes for missing (or mis-positioning) a picture edge.

(a) Edges which are short and/or lacking in contrast. The vast bulk of missed edges fall into this category. Por example the edge completing the top of the puppet's head in Fig.I (the missing line joining the junction of 16 and 17 with the junction of 9 and 2k in Fig.6). This is clearly visible in the differentiated data (Fig.2) its magnitude however is too low for the histogram bin (r = +57, a- = 130°) to exceed critical threshold for line finding, so that it never appears in the cpl's for thiB data (Fig.k).

(b) A similar case of considerable interest because of its potential value in any 3-D interpretation of the data, occurs as part of 'TEE junction' formed by the occlusion of the trunk shadow by the nearer 'arm' in Fig.I. Two of the edges - lines 13 and 18 in Pig.6 - are recovered; the third is clearly visible in the differentiation Pig.2, and at first 3ight it seems odd that it does not get picked up as part of 18. The reason is that these, the two collinear pieces of the 'TEE', are of opposed contrast as close inspection of Fig.2 shows. (α = 'X','Y' for line 18; but α = 'F','G' for the missing edge), this interesting fragment is thus 'on its own' and fails to exceed the H = 5 threshold. This case, and there are quite a lot more, points to an intrinsic defect in distinguishing between (α) and ($\alpha \pm 180°$) in recovering edges: but to discard the distinction in accumulating H (r,α) would undoubtedly degrade the quality of the histogram data. What seems more relevant is to *modify the criterion for merging* cpl's to form pll's so that the permitted variation in α becomes

$$\alpha \text{ new} = \alpha \text{ orig} \pm 10°$$

$$\text{or} \quad \alpha \text{ new} = (\alpha \text{ orig} \pm 180°) \pm 10°$$

The effect of such a change in the restriction on 'acceptable α' would be to accommodate a particular way in which an edge stops.......because it has reversed its contrast.

(c) Yet another edge in Fig.1 which appears quite clearly in the differentiated picture (Fig.2) is the lower (supported) edge of the puppet's leg. (The missing line between 25 and 35 in Fig.6). Here the problem is not immediately one of low contrast although it is very weak, as close inspection of G and α (Fig.8) shows, for there are sufficient points to secure a value of H exceeding the criterial value of 5 as is evidenced by the presence of at least one substantial cpl (identifier '8') in Fig.9. The listing of pll's in Fig.9 provides an explanation for this loss: some 15 out of the 53 members of cpl '8' in Fig.9 are 'stolen' by other pll's during the merger of cpl's. Typically the pll which eventually emerges as line 25 in Fig.6 acquires some of them, (they contribute a small error in positioning of the fitted line 25) and does so at an early stage. The residual cpl is then progressively dismembered (starting at the 'hip joint') into small sub-critical pll's ('D', '9', '8', '7', etc. in Fig.9) which remain small precisely because the requirement for strict picture adjacency cannot be met. Eventually the original cpl is eroded until its histogram magnitude is less than 5 at which point it drops out altogether. This is why the sizeable group of '8's' in Fig.9 near the feet never appear as pll's.

A major part of this problem can be traced to the myopic nature of the line-following criteria, more generally perhaps to the formulation of the task of assessing the continuity of a picture edge, as a serial search of sequentially ordered points. This would in fact have been successful for this picture edge had the search commenced from the <u>other end</u> of the cpl i.e. the '8's' near the feet in Fig.9. But a deeper issue is at stake here in deciding just how appropriate a connectivity search is when a set of contributing points is available. We return to this topic in the next section. The use of the connectedness criterion is not helped moreover by the 'capture' of picture locations by mergers with strong neighbouring edges (in this case the edge represented by line 25 in Fig.6), a process which punches additional holes into the already straggly string of locations. A much more dramatic case of this 'poaching' can be seen for scene 214 (Fig.7) where the low confidence line 22 has captured the larger part of the picture edge marked by line 8 with the result that <u>two</u> picture edges are effectively missed. The problem here stems from the pass-oriented character of the algorithm, line 22 being apparently a big peak in H(r,α) is formed early on and acquires through merger with a picture-adjacent cpl a substantial subset of the cpl which properly belongs to line 8. When the peak in H(r,α) for line 8 is eventually considered, the points which 'rightfully' belong to it have been removed. The remedy here seems to lie - like the contrast reversal problem - in modifying the criteria for gap acceptance. Specifically it would seem possible to extend adjacency tests to include relevant pll's <u>already formed</u> (e.g. the pll for line 22 in scene 214) although the decision mechanism to resolve 'disputed' data is unclear at this stage. It might be sufficient to return ambiguous analyses or to flag pll's with a 'disputed' marker.

4. <u>Discussion</u>.

The account in the previous section gives a partial analysis of performance and weakness of the algorithm. It is by no means exhaustive

Perhaps the best way to assess the technique is not solely in terms of lines recovered and missed. Rather it is to see how its method of addressing the picture data differs from that of other techniques. Duda and Hart[4] discuss the efficiency of parameterisation methods of this general type by contrasting it with a method based upon fitting lines to all possible pairs of feature points. This, as inspection of Fig.2 suggests, is quite prohibitive and can hardly be considered as a serious candidate. Rather the comparison should be with line followers such as that described by Horn[9]. In the so-called 'Binford-Horn line finder', feature points derived from an edge-marker due to Binford are formed into lists on the basis of <u>proximity</u> and agreement of attributes such as the *direction* (α) *and magnitude* (G) *of the* intensity step. The crucial characteristic of the procedure is that it <u>follows</u> or tracks along feature points because it is using picture adjacency as the method of addressing possible new members of the list.

In the method we have described, the addressing scheme is in effect inverted since it is proximity in (r,α) space which is used to form the contributing points lists rather than proximity in (x,y) space. Of course we then have to add the spatial proximity requirement which in its current form is a simple adjacency test (Section 2) corresponding to the test on α and G applied <u>after</u> spatial proximity in the Binford-Horn approach. At this point our process becomes a line follower, but with the difference that the points it addresses belong to lists of locations of the same (r,a). It is clear that the adjacency criterion we have adopted makes it a very crude edge-follower.

There are basically two things one is trying to do with the cpl in its simple form or its form as augmented by other cpl's merged due to proximity in (r,a) space or $(x_{f}y)$ space. The first is to find the end points of edges. At present we do this by examining the list. However most interesting picture edges end in junctions. Thus the fact that an edge terminates usually if not always implies that another edge is 'pointing' at that termination point. Thus many 'gaps' perhaps a majority could be marked not from a consideration of the edge cpl per se but from a consideration of 'neighbouring' "[Tn some sense) cpl's. In Horn's paper he describes a phase which we have not yet studied called •concocting vertices'. This phase follows on from the formation of bounded lines to represent recovered edges: the proposal sketched informally above would in the context of the Binford-Horn line finder amount to a merge of two processes (line fitting and vertex concoction) which are at present sequential. The procedures for carrying this out in the r,a parameterisation have not been detailed as yet.

The second objective in processing a significant cpl is to ascertain whether the picture locations which lie in the 'mark' as against the 'space' between gaps, satisfy some criterion of'edge-ness' realised in the present program by strict sequentially-addressable picture contiguity. Perhaps less search-dominated criterion would be simply to assess the spatial density of points on the cpl in the putative 'mark' between gaps. Again we have not pursued this notion to the point of programming it.

The crucial question is that *of* deciding how much work should be done on attempting to remedy some of these defects - perhaps along the lines suggested - as against accepting the existing performance and leaving other routines to remedy these omissions and errors. Our current preference and intention lies almost wholely in the latter direction.

Acknowledgements

We are grateful to Alan Mackworth and John Francis for discussions and especially for suggesting the use of locally-computed values of a in forming the histogram and to John Francis who implemented an early version of that suggestion.

We are grateful to both Alan Mackworth and Stuart Sutherland for comments on the draft of this paper.

The research is part of a project supported by the Science Research Council.

References

1. Barrow, H.G. and Popplestone, R.J. (I971). Relational Descriptions in Picture processing. Machine Intelligence. 6.,377-396. B. Meltzer and D. Michie (Eds), University Press; Edinburgh.

2. Brice, C.R. and Fennema, C.L. (1970).Scene Analysis Using Regions, Artificial Intelligence. 1, 3, 203-226.

3. Currie, I.P., Bond, S.G., and Morison, J.D. (1971). Algol68-R in Algol 68 Implementation. J.E.L. Peck (Ed) pp.21-34 North Holland.

4. Duda, R.O., and Hart P.E. (1972). Use of the Hough Transformation to Detect Lines and Curves in Pictures. Communications of the A.CM. 15*, 1 11-15.*

5. Falk, G. (1972). Interpretation of imperfect line data as a three-dimensional scene. Artificial Intelligence. 3*,* 2. 101-114+.

6. Griffith, A.K. (1970). Computer Recognition of Prismatic Solids, Ph.D. thesis, Project MAC, MIT.

7. Guzman, A. (1971). Analysis of Curved Line Drawings Using Context and Global Information, Machine Intelligence;, 6.. 323-376. Meltzer and Michie (Eds) University Press: Edinburgh.

8. Hopgood, P.P.A. (1969). GROATS, A Graphic Output System for Atlas using the SCi|.020$_f$ Proceedings 8th UA1DE Conference, San Diego.

9. Horn, B.K.P. (1971). The Binford-Horn Line Finder. Vision Flash 16. Artificial Intelligence Laboratory, MIT.

10. Roberts, L.G. (1963). Machine Perception of Three-dimensional Solids in Optical and Electro-optical Information Processing. Tippett et al (Eds) pp.159-197, Cambridge, MIT Press.

11. Shirai, Y. (1972). A Hetcrarchical Program for Recognition of Polyhedra. Memo No.263. Artificial Intelligence Laboratory, MIT.

12. Winston, P.H. (1972). The MIT Robot, Machine Intelligence, 2» ii31-U63.B. Meltzer and D. Michie (Eds), University Press: Edinburgh.

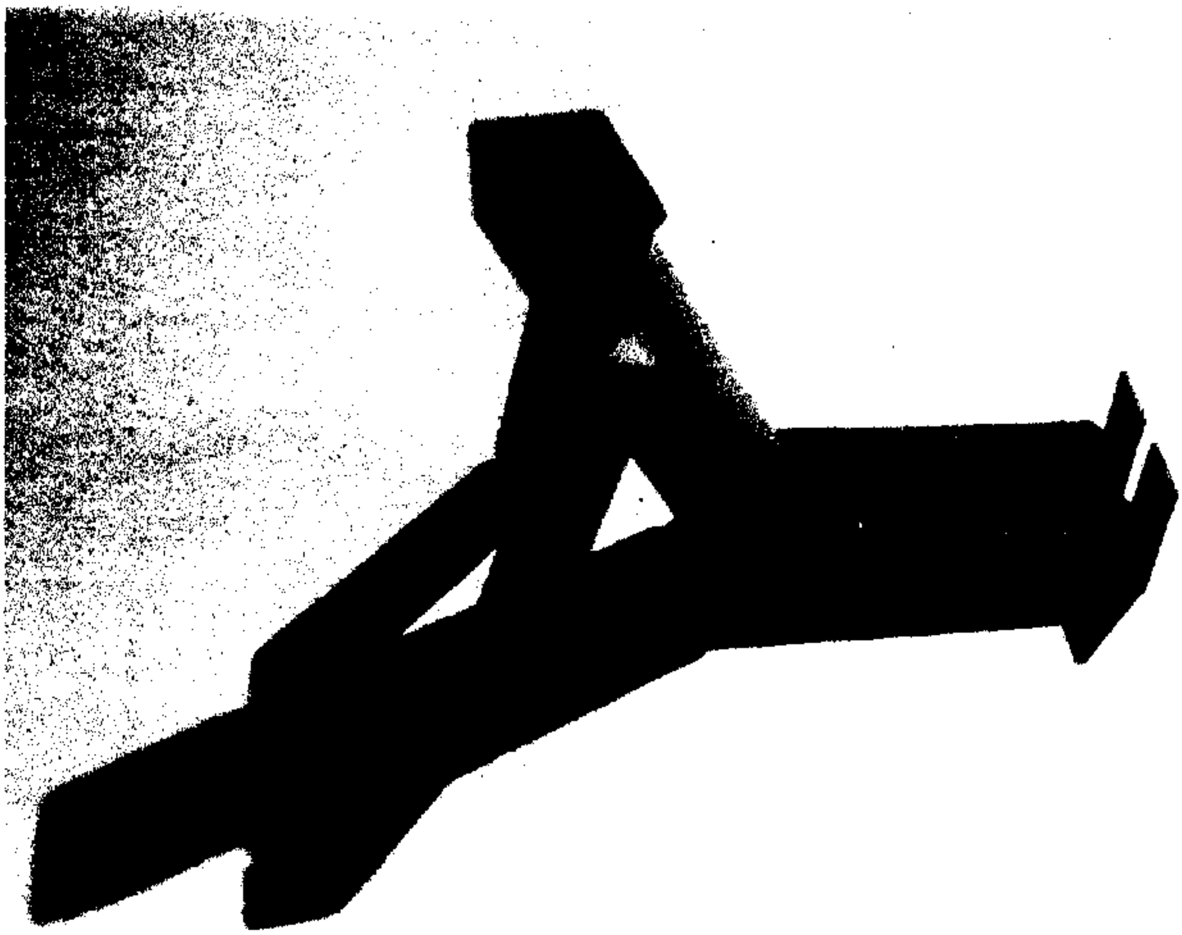


Fig.1. Puppet scene No. 217.

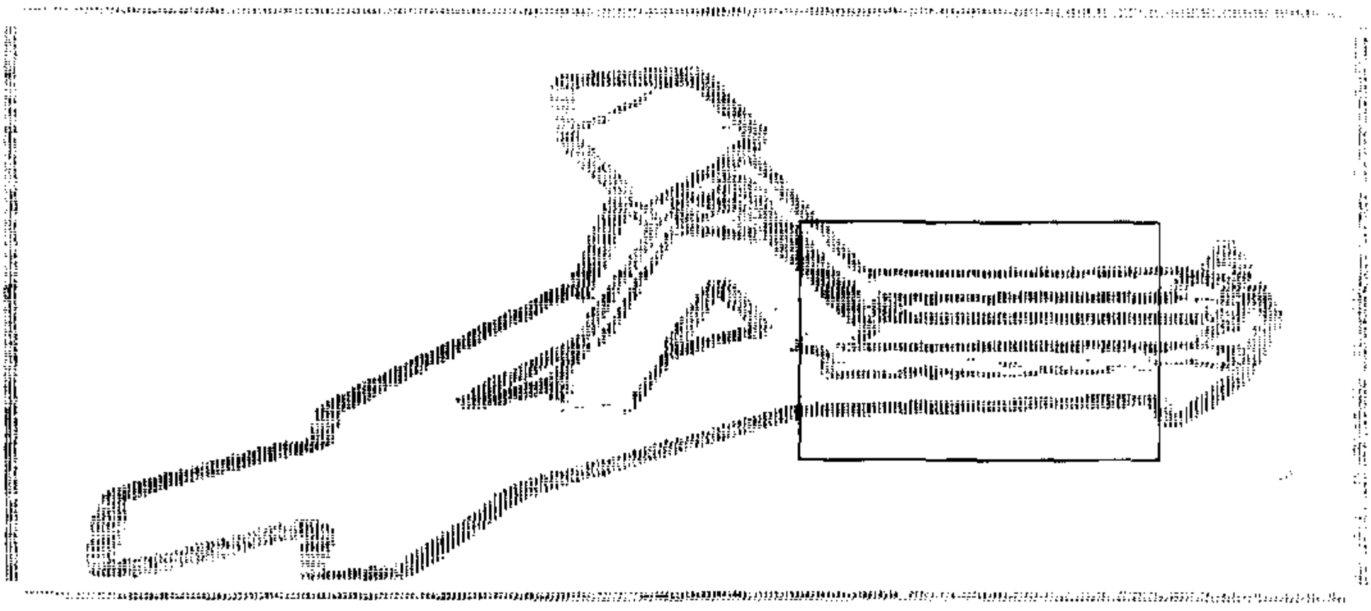


Fig.2. Computed values of α at picture locations where G is supra-threshold. Each character (0-9, A-Z) identifies the local direction in 10° steps.

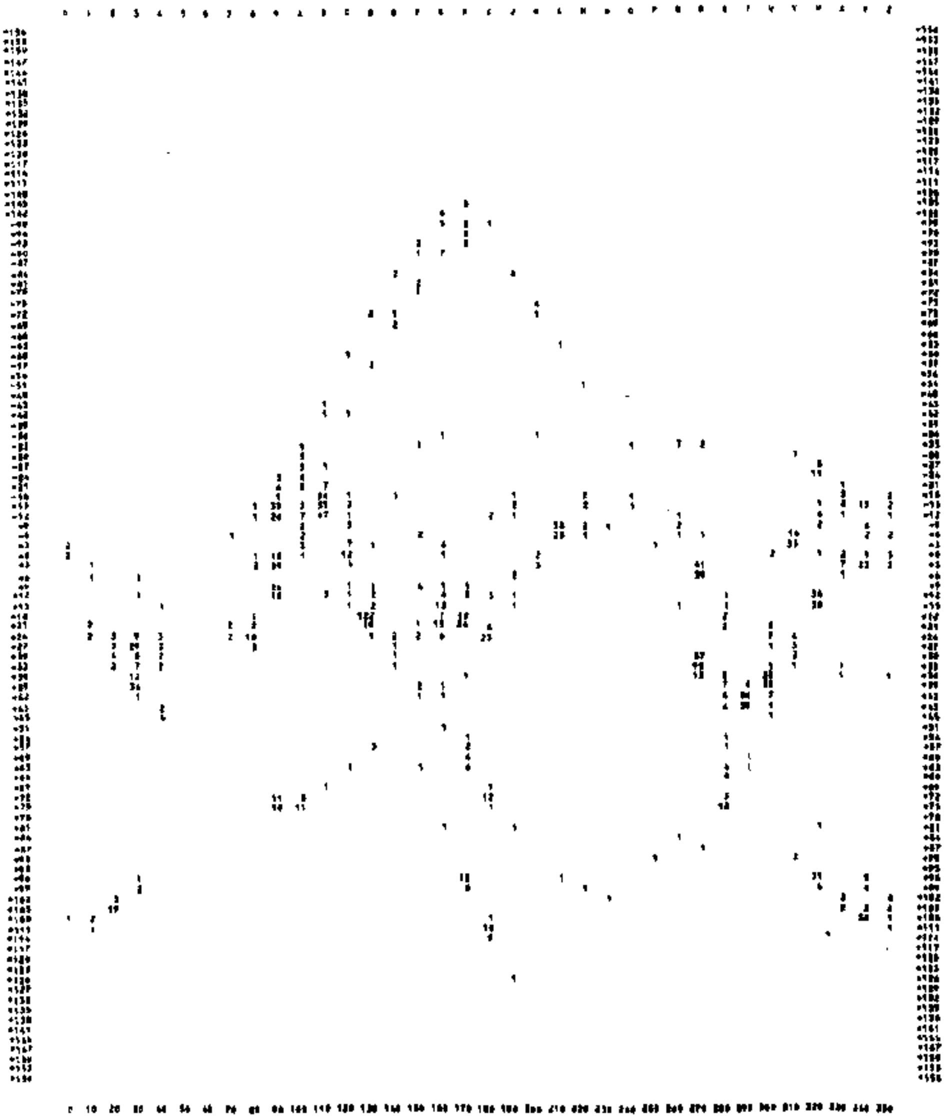


Fig.3. Histogram H(r,α) accumulated for the picture locations of Fig.2.

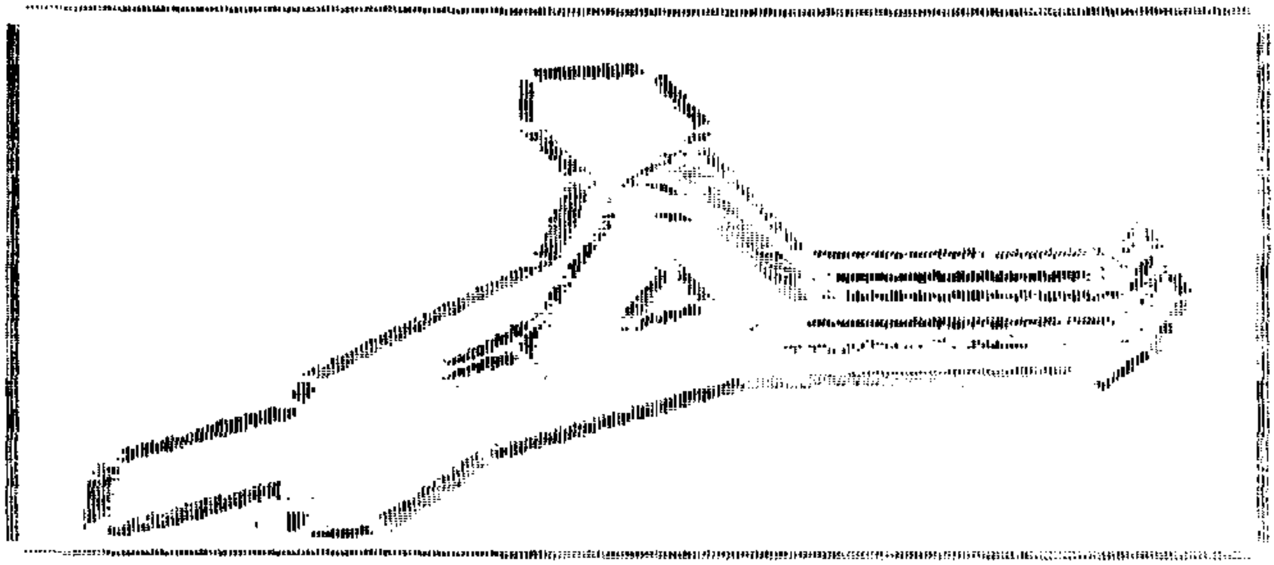


Fig.4. Contributing-points-lists in the format of Fig.2 for histogram entries exceeding 5 units magnitude. Each character now identifies a unique location in the histogram.

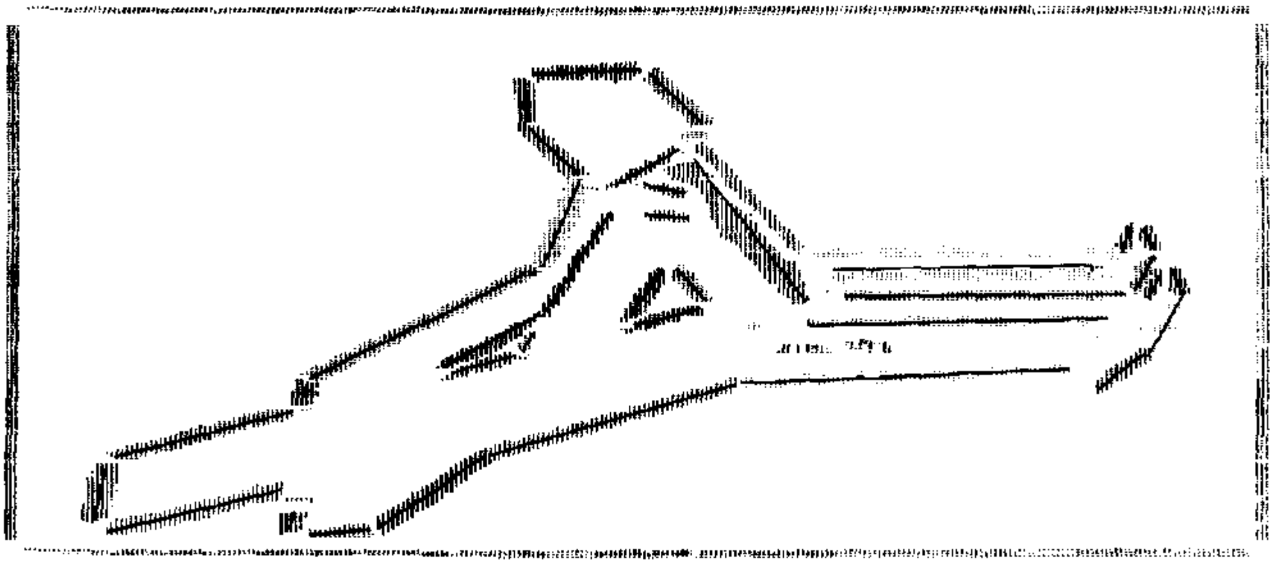


Fig.5. Putative-line-lists extracted from Fig.4 with fitted lines inscribed.

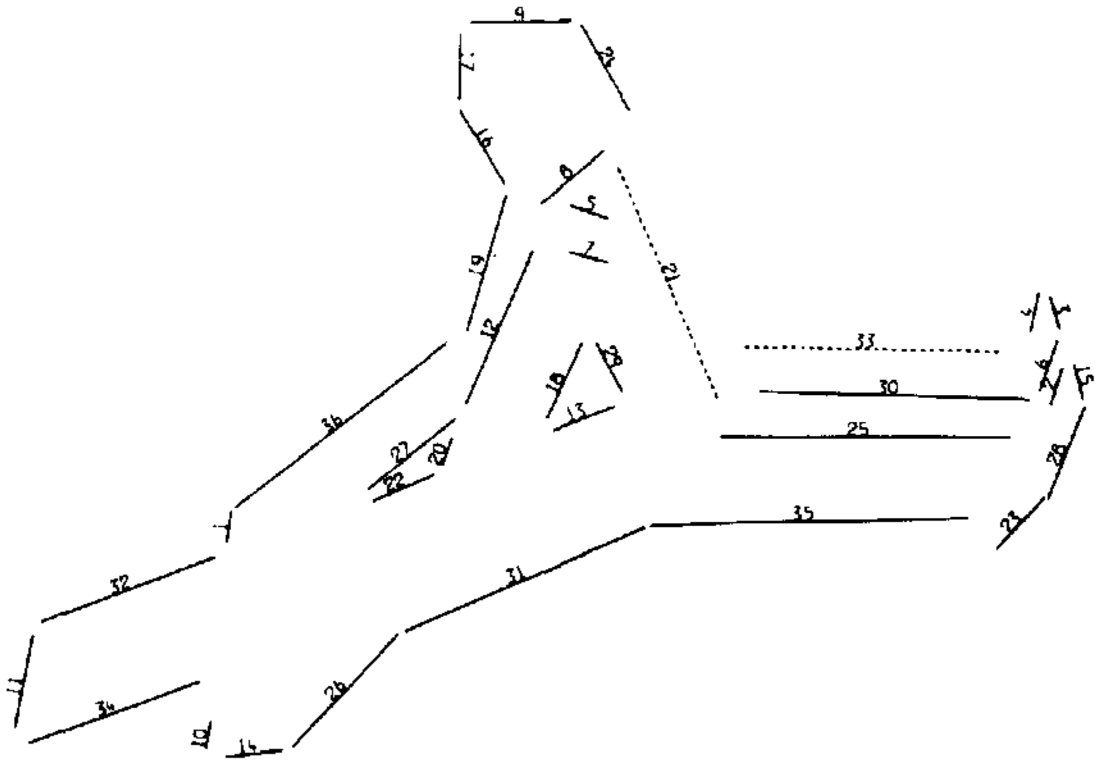


Fig.6. Graph plotter output of the lines inscribed on Fig.5. The lines are numbered in inverse order of recovery.

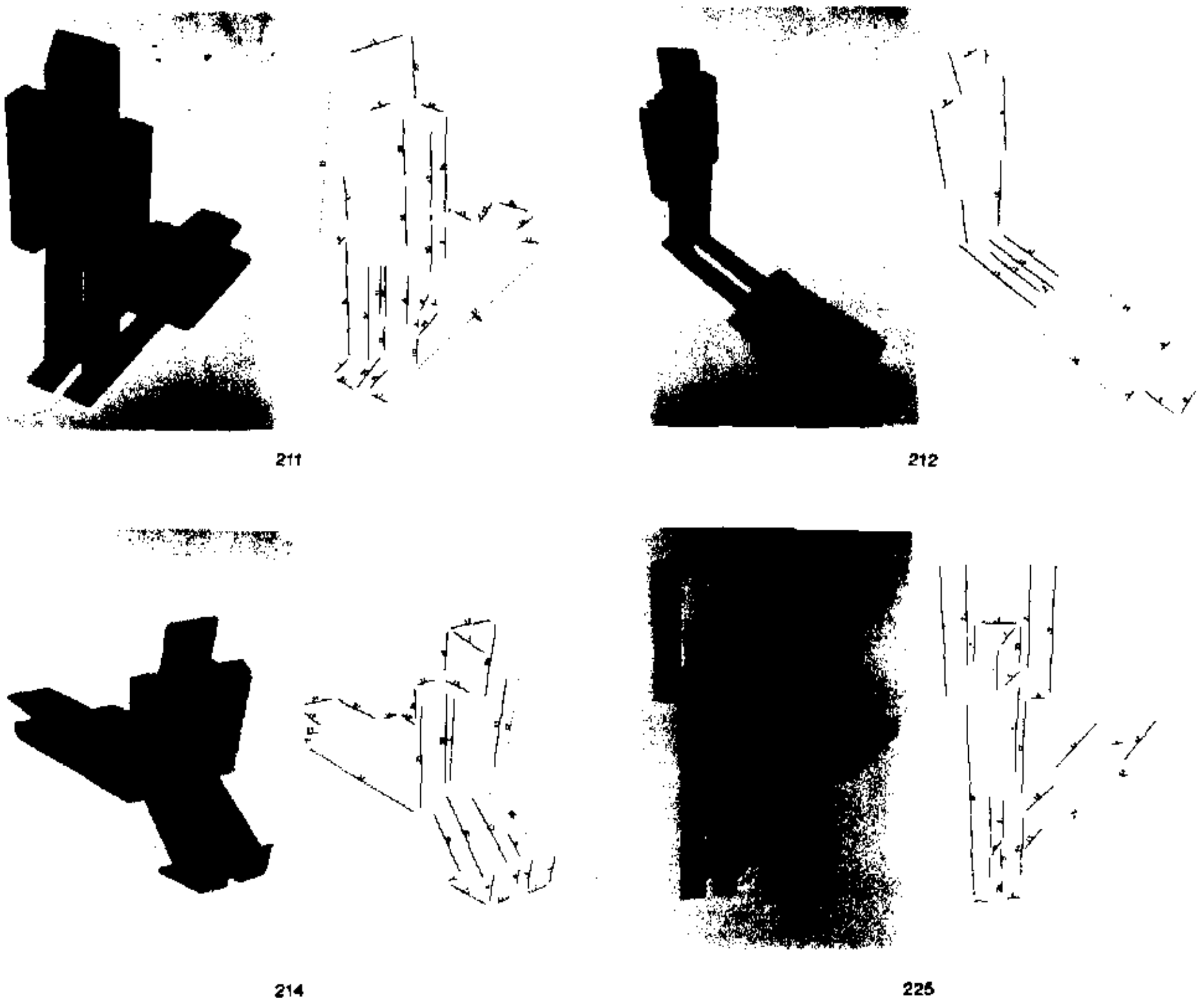
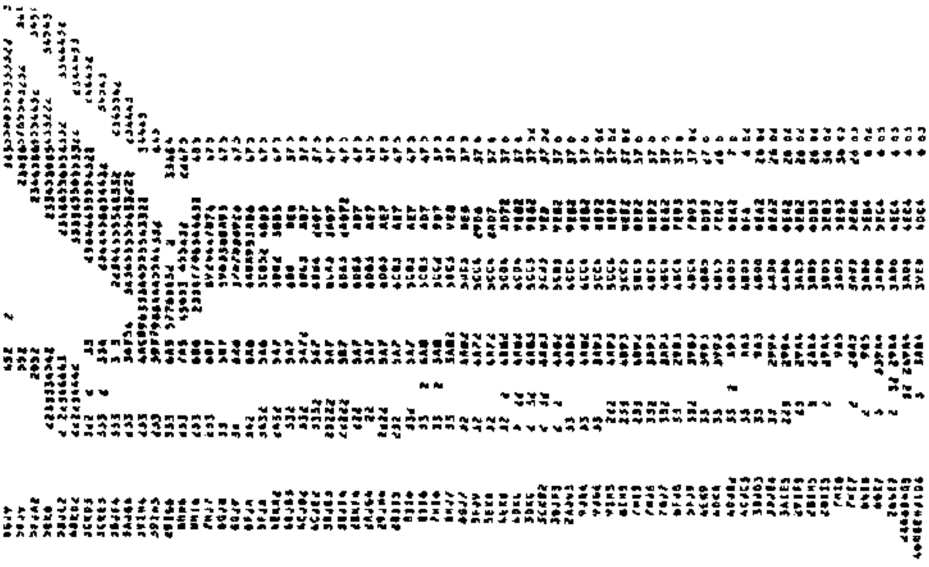
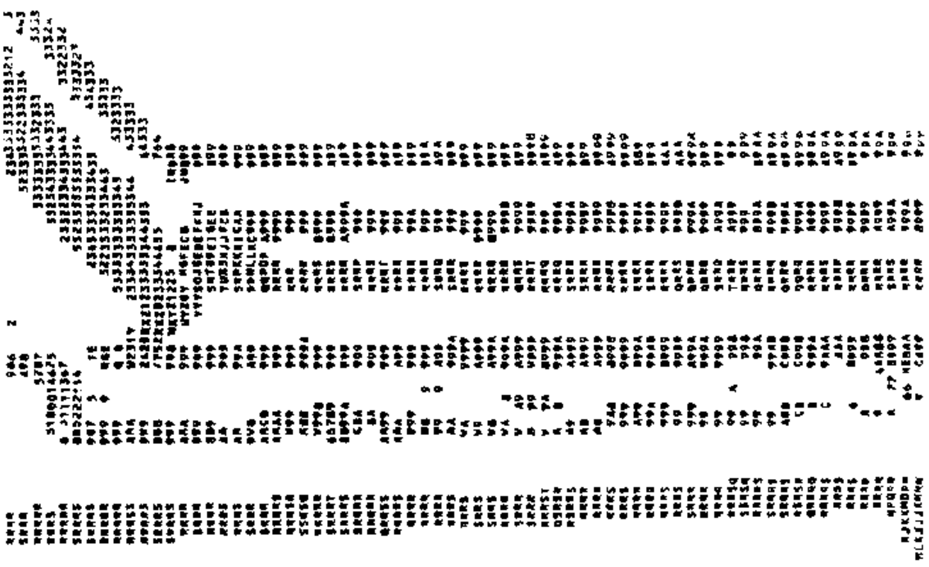


Fig.7. Puppet scenes 211, 212, 214, 225 and the edges recovered by the algorithm.



A



B

Fig.8. Computed values of G(a) and α(b) for the locations within the inscribed area of Fig.2. Key for α as in Fig.2. For G linear steps in magnitude indicated by the same ordering viz. 0-9, A-Z....

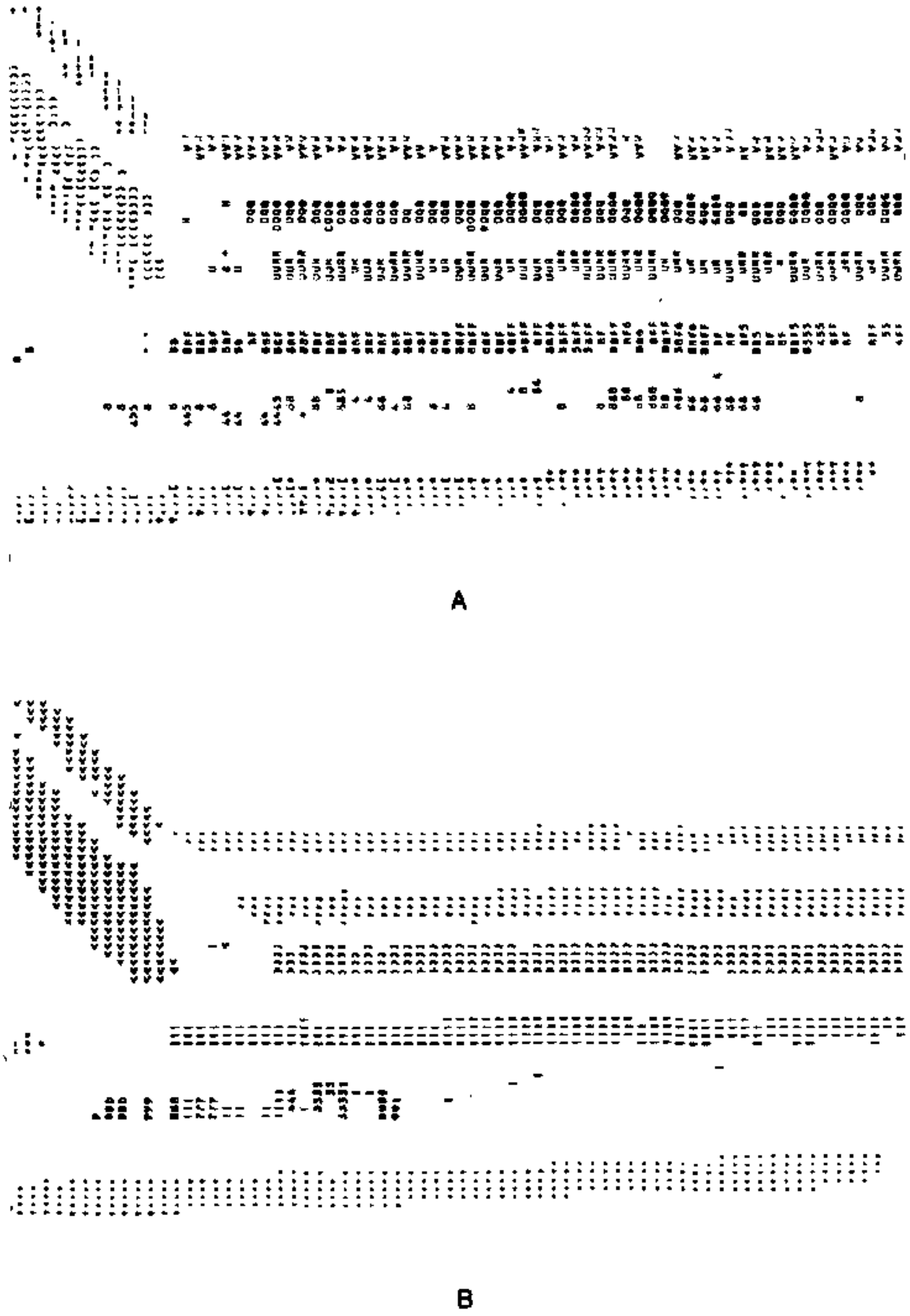


Fig.9. Contributing-points-lists (a) and putative-line-lists (b) for the inscribed area of Fig.2.