IMAGE SEGMENTATION BASED ON HOUGH TRANSFORMATION *

Prof. Zhang ZhuXun, Mr. Min Yiren, Prof. Zhang Jianqing
Wuhan Technical Universty of surveying and mapping
P.R.China
Commission III of ISPRS

ABSTRACT

In this paper the standard algorithm of the straight-line Hough Transformation and the modified algorithm presented by F.O.Gorman and M.B.Clowes are analysed. The author presents a new modified algorithm of Hough Transformation, which solves the contradiction between the running time and accuracy. In order to get the position of the straight-line , a straight-line recovering system is defined. On this base, region segmentation using Graph Theory is presented. The satisficatory experiment results with aerial photographs are shown finnaly.

key words: Image segmentation, Hough Transformation, straight-line recovering, Graph theory, region segmentation

1.INTRODUCTION

For the large scale image of a city area, existent methods of image matching may be not suitalbe. The edge extraction and image segmentation based on the description of image structure seem to be the first step of the new strategy, not only in matching of city images, But also in image analysis and interpretation. One of image segmentation methods, which divides the image into some regions with even grey level(colour, or texture), is the edge detection, and Hough transformation is the importent techniqne of edge detection. It is used in many fields to extract various curves successfully. but, there is a contradiction between the running time and the accuracy in the transformation procedure, and the description of the regions is necessary for image processing in the higher level, after edges are extracted.

Based on the analyses of the traditional algorithm of straight line Hough transformation and the modified algorithm proposed by Gorman and Clowes(1976), a new algorithm is proposed, in which the information of gradient direction is used more reasonably.

A straight line recovering system is designed in order to locate the straight line. then, image segmentaion can be carried out through the regions are described by the edges using Graph Theory. The experiment results with real images will be shown finally.

2. ALGORITHM OF STRAIGHT LINE HOUGH TRANSFORMATION AND ITS MODIFICATION

The straight line equation used by Hough is

$$y=kx+b$$

Because k may be unlimited, the normal line equation was proposed by Duda and Hart(1972)

$$\rho=x\cos\theta+y\sin\theta$$

where ρ and θ are the direction angle and length of the normal line of the straight line.

2.1. Traditional Algorithm

step 1: extracting feature points with edge detection operator.

step 2: quantizing the parameter plane as h(m,n) with initial value zero, where $m=INT(\pi/d_\theta)+1$, $n=INT(2R/d\rho)+1$, d_θ and $d\rho$ are the intervals of quantization, $R=sqrt(L^2+W^2)/2$, L and W are the length and width of the image.

step 3: for each feature point(x_k,y_k),

$$h(i,j)=h(i,j)+1$$

where $i=0,1,...,m$, $\theta_i=i\cdot d\theta$, $\rho_i=x_k\cos\theta_i+y_k\sin\theta_i$, $j=INT(\rho_i/d\rho)$.

step 4: detecting the extreme maximum points of the parameter plane h. The extreme maximum point (i,j) corresponds to the straight line parameters $\theta_i=i\cdot d\theta$ and $\rho_j= j\cdot d\rho$.

In this algorithm, the calculation is carried out from θ_0 to θ_m. If the high accuracy is expected, then, the $d\theta$ should be small and m will be large. The running time must be much. Beside, determination of the threshold of extreme point detection is difficult, and there are some false straight lines.

2.2. Modified algorithm of Gorman and Clowes

Other steps are the same as in traditional algorithm except step 5: for each feature point(x_k,y_k), calculating the direction θ_k of its gradient with Roberts operator or Sobel operator and $\rho_k=x_k\cos\theta_k+y_k\sin\theta_k$, $i=INT(\theta_k/d\theta)$, $j=INT(\rho_k/d\rho)$, h(i,j)=h(i,j)+1.

In this way, (x_k,y_k) corresponds to only one point (θ_k,ρ_k) on parameter plane. The summation is only performed in one cell of parameter plane h. However, there is noise on

* The investigation is supported by National Nature funds and National Bureau of Surveying and Mapping of P.R. China

image. Thus, the direction of gradient computed by gradient operator is not precise. Therefor, the extreme points are not distinct, and their detection is difficult.

2.3. New modified algorithm

step1: The sobel differences are calculated for each pixel:

$$dx = [(g_{i+1,j-1}+2g_{i,j+1}+g_{i+1,j+1}) - (g_{i-1,j-1}+2g_{i-1,j}+g_{i-1,j+1})]/4$$

$$dy = [(g_{i-1,j+1}+2g_{i,j+1}+g_{i+1,j+1}) - (g_{i-1,j-1}+2g_{i,j-1}+g_{i+1,j-1})]/4$$

The direction of the gradient is

$$\theta = \mathrm{arctg}(dy/dx)$$

The feature points are extracted by edge detection operator, and the edge is thinned along the direction of the gradient by selecting the extreme maximun gradient.

step 2: The parameter plane is quantized.

step 3: For each feature point (x_k,y_k) with its gradient direction θ_k,

$$h(i,j)=h(i,j)+1$$

where $i=i_{bk},i_{bk}+1,...,i_{ne}$, $i_{bk}=INT[(\theta_k-\theta_0)/d\theta]$, $i_{ne}=INT[(\theta_k+\theta_0)/d\theta]$, θ_0 is a experimental value between 5 degree and 10 degree.

step 4: The extreme maximum points of the parameter plane are detected, and the straight line paremeters are computed

$$\theta=i\cdot d\theta\ ,\ \rho=j\cdot d\rho$$

where$(i,j)$is an extreme point of h plane.

The new algorithm solves the contradiction between the running time and the accuracy, because the summation is porformed in a small area and the $d\theta$ can be small. What is more, the gradient information is used more reasonably in the case of the noise.

3.STRAIGHT SEGMENT RECOVERING

The position of straight segment is still unknown after the parameters are acquired by Hough transformation, because there is no end-point, In order to determine the position of a straight segment, the pixels corresponding to certain parameters are recorded respectively.

The set of the parameters and corresponding set of pixels are

$$L=\{(\theta_i,\rho_i) \text{ and } S_i \mid i=1,2,......,M\}$$

where $S_i=\{(x_k,y_k) \mid k=1,2,......,N_i\}$

There is false information in L, due to the quantization errors on both of the image and parameter space. Those are:

(1).The set includes some false pixels.

(2).Two straight segment are in the same set.

(3).The pixels in the intersection of two line are not in the set.

3.1. Delete false pixel

If $|\rho_i-x_k\cos\theta_i-y_k\sin\theta_i|<=d\rho$ then the pixel (x_k,y_k) is removed from the set S_i.

3.2. Segmentation and merging

After ordering the pixels of the set S_i the break-points are searched, and interpolation is employed between two broken points(x_i,y_i) and (x_{i+1},y_{i+1}) If there is feature point in the window with size 1*3 pixels centering at the interpolated point, it is a supported point.

The ratio of the number of the supported points and the interpolated points is

$$K=m/M$$

where m is the number of the supported points, and M is the interpolated points. If K>T(threshold), ponit(x_i,y_i) is the end point and (x_{i+1},y_{i+1}) is the begin point of next segment. After that, the ovrerlapped segments are merged into one.

3.3. Intersection

The intersections of the staight lines may not be included in the set S_i due to the errors of the gradients at the corners or crosses.

The intersection coordinates of two segments S_i and S_j can be computed. If computed intersection is not in S_i and S_j and the distances between it and nearer end-point are smaller than T_1(threshold) it is received and the end-points are connected. If d_1 and d_2 lagrre than T_2(another threshold), it will be rejected. Othewise, the method used in 3.2 is applied to detrmine whether the computed intersection is rejected .

3.4.Extension of end-point

Because of the strict criteria of straight line detection and noise, some real end-points are not included in the line set S_i. The end-point, namely (x_1, y_1) should be extended. The extrapolated points are processed progressively, when there is not any feature point in the window centering at the extrapolated point with size 1*3 pixels, the extension is completed.

From the straight segment recovering system, the straight line parameters, two end-points and all pixels of each straight segment are acquired.

4. REGION SEGMENTATION USING GRAPH THEORY

If there are only straight segments on the image, they divide the image plane into different regions. For the further application, the straight segments comprising the regions should be determined. The Graph Theory (Mayeda, 1972) can be used in the extraction of the regions on images.

4.1 Background of Graph Theory(Mayeda, 1972)

The following introduction is only constrained in the relative parts.

4.1.1. Linear graph. Let E and V be sets of edges and vertices respectively. If every edge e∈E corresponds to exactly one pair (v,v') of vertices, then the set G(V,E) comprised V and E is a linear graph. The two vertices v and v' are called the end-points of edge e, and v,v' and e are incident or connected each other. If v and v' are the same, then edge e is called a self-loop. If the pair (v,v') is not ordered, then e is a nonoriented edge. If all edges are nonoriented, the linear graph is called a nonoriented (linear) graph.

4.1.2. Paths and Circuits. The degree of a vertex v is defined as

$$d(v) = 2n_s + n_n$$

Where n_s is the number of self-loops incident at vertex v and n_n is the number of edges other than self-loops incident at v.

A route from vertex v_0 to vertex v_n consists of a sequence of edges and the incident vertices. When each edge in such a sequence appears only once, this sequence is called an edge train. The vertex v_0 is intial vertex and v_n is final vertex. If the initial vertex and the final vertex are distinct, the edge train is an open edge train. Otherwise it is a closed edge train.

A path between vertices v_0 and v_n is a open edge train which satisfies that every vertex other than v_0 and v_n is of degree 2.

4.1.3. Subgraph and connected graph. A linear graph G' is called a subgraph of a linear graph G, if G' consists only of edges and vertices of G.

A linear graph is called a separated graph if there exist two vertices such that there are no paths between them. Otherwise it is a connected graph.

A set of maximal connected subgraphs of a linear graph G is a set of connected subgraphs $g_1, g_2, ..., g_p$ such that every edge and every vertex in G is in exactly one of these subgraphs where an isolated vertex is a connected subgraph by difinition. The number p is the number of maximal connencted subgraph of linear graph G.
The rank of a linear greph is n_v-p where n_v is the number of vertices and p is the number of maximal connected subgraphs in the linear graph.

4.1.4. Tree and fundamental circuit. A tree of a connected linear graph G of n_v vertices is connected subgraph having n_v vertices and n_v-1 edges.

For a tree t in a linear graph G, an edge that is not in t is called a chord. A set of all edges in G which are not in t is called a set of chords with respect to tree t.

Let $(e_1, e_2, ... ,e_r)$ be a set of chords with respect to a tree t of a connected linear graph G. Also let c_i be a circuit in t U(e_i) for i=1, 2, .., k. Then the collection of circuits $c_1, c_2, ..., c_k$ is a set of fundamental circuits with respect to the tree t, and $k=n_e-n_v+1$, where n_e and n_v are the numbers of edges and vertices of G respectively.

Theorem 1: If and only if a linear graph has no circuit and is connected, the linear graph is a tree.

4.1.5 Incidence matrix and circuit matrix
(1) An exhaustive incidence matrix and circuit matrix A_e is defined by

$$a_{ij} = \begin{cases} 1 & \text{if edge } e_j \text{ is incident at vertex } v_i \\ 0 & \text{otherwise} \end{cases}$$

where a_{ij} is the (i,j) entry of A_e.

Theorem 2: The rnak $R(A_e)$ of an exhaustive incidence matrix A_e of a connected linear graph G is equal to the rank of G, that is

$$R(A_e) = R(G) = n_v - 1$$

where n_v is the number of vertices.

(2) A fundamental circuit matrix B_f of a linear graph is defined by

$$b_{ij} = \begin{cases} 1 & \text{if edge e is in fundamental circuit i} \\ 0 & \text{otherwise} \end{cases}$$

where b_{ij} is the (i,j) entry of B_f in the form

$$B_f = [U\ B_{f12}]$$

consisting of n_e-n_v+p rows where p is the number of maximal connected subgraphs in G, and the columns and rows of unit matrix U correspond to the chords and fundamental circuits.

(3) Suppose t is a tree of a linear graph G(p=1). We arrange the columns of the incidence matrix A of graph G such that we can partition A:

$$A = [A_{11}\ A_{12}]$$

where the columns of A_{11} correspond to the chords with respect to t.

Theorem 3: An incidence matrix A and a fundamental circuit martix satisfy

$$B_f = [U\ \ A_{11}^T \cdot (A_{12}^T)^{-1}]$$

4.2 Region segmentation

From above, the region segmentation of image by extracted straight segment is equal to determination of fundamental circuits in linear graph and the fundamental circuits in linear graph consist of the tree and corresponding chords.

For example, the tree t(Fig.2) is selected from the linear graph G(Fig.1). The corresponding chords are b, c, g.

$$A = [A_{11}\ A_{12}] = \begin{array}{c} \begin{matrix} b & c & g & a & d & e & f & h & i & j \end{matrix} \\ \begin{pmatrix} 0&0&1&1&0&0&0&1&0&0 \\ 1&1&0&0&0&0&0&0&1&0 \\ 0&1&0&0&1&0&0&0&0&0 \\ 0&0&0&0&1&1&0&0&0&1 \\ 0&0&0&0&0&1&1&0&0&0 \\ 0&0&1&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&0&1&1&1 \end{pmatrix} \end{array} \begin{matrix} 1 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{matrix}$$

Accoording to modulo 2 algebra, the inverse matrix of A_{12}^T is

$$(A_{12}^T)^{-1} = \begin{pmatrix} 1&0&0&0&0&0&0 \\ 1&0&0&0&1&1&0 \\ 1&1&0&0&1&0&1 \\ 1&0&0&0&1&0&1 \\ 1&0&1&0&1&0&1 \\ 1&0&1&1&1&0&1 \\ 1&0&0&0&1&0&0 \end{pmatrix}$$

Thus, the fundamental circuit matrix is

$$B_f = \begin{array}{c} \begin{matrix} b & c & g & a & d & e & f & h & i & j \end{matrix} \\ \begin{pmatrix} 1&0&0&1&0&0&0&1&1&0 \\ 0&1&0&0&1&0&0&0&1&1 \\ 0&0&1&0&0&1&1&1&0&1 \end{pmatrix} \end{array}$$

The B_f shows the boundaries of region I, II, and III clearly

$$C_I = (b,a,h,i)$$
$$C_{II} = (c,d,i,j)$$
$$C_{III} = (g,e,f,h,j)$$

The procedure of region segmentation using Graph Theory is

(1) Calculating the intersections of straight segments and image frame,forming the exhaustive incidence matrix A.
(2) rearranging A as a diagonal p-block matrix in order to divide the linear graph into p maximal connected subgraphs .
(3) For each connected graph G
--- Generating the tree from G and constructing the incidence matrix according to the order of the chords and tree-branches (edges).
--- Calculating the fundamental circuit matrix to acquire the boundary of the region.
(4) Judging the inclusive relationship
--- Finding the inclusion between maximal connected subgraphs.
--- Determining the inclusive relationship of regions.

5. EXPERIMENTAL RESULTS

The relative experiments are carriedd out based on the test images of Comm.Ⅲof ISPRS.

5.1. Three algorithms of Hough Transformation

The running time of each algorithm is shown in table 1. The tranditional algorithm is the most one. The parameter spaces are shown in Fig.3. From Fig.3 b).c), it can be seen that the Gorman and Clowes algorithm causes the extreme points to spread, and the new algorithm avoids that problem.

5.2. Straight segment recorve system

Fig.4(a) shows the set of points corresponding to the parameters of Hough Space.

Fig.4(b) is the result of segmentation and merger.

Fig.4(c) is the result of end-point extension.

5.3. Region segmentation

Fig.5 shows the reaults of region Segmentation with different grey levels,and the relative region, edge, initial and final end-points are in Table 2.

6. Conclusion

The new algorithm of Hough Transformation overcomes the contradiction between time and accuracy. The straight segmentation is necessary and successful. The region segmentation based on Graph Theory is a new hopeful way to discribe the image structure. And further, Extraction of arbitrary curve with straight line Hough Transformation, relationship matching and image interpretation based on region segmentation should be investigated.

CONFERECES

[1] Davies E.R.: "Image Space Transforms for Detecting Straight Edges in Industrial Images" Pattern Recognition Letter, 4,1986
[2] Dyer C. B.: "Gauge Inspection Using Hough Transform" IEEE Trans. PAMIS,621-625,1983
[3]Engelbrecht J. R. and Wahl F. M.: "Polyhedral Object Recognition Using Hough-space Features" Pattern Recognition 21,155-167,1988
[4] Gorman F.O. and Clouse M. B.: "Finding Picture Edge Through Collinearity of Feature Point" IEEE Computers,Vol.c-25,No.4, April,1976
[5] Forstner W. and Gulch E.: "A Fast Operator for Detection and Precise Loction of Distinct Point, Corners and Centres of Circular Features" Proceedings of Intercommision Conference of ISPRS on Fast Proceedings of Photogrammetric Data, Interlaken, June,1987
[6] Kushuir,Abe,K. & Matsumoto K.: "Recognition of Hand-printed Hebrew Characters Using Features Selected in the Hough Transform Space" Pattern Recognition 18,173-193,1985
[7] Mayeda W.: "Graph Theory" John Wiley and Sons Inc. 1972
[8] Mckenzie D. S. and Protheroe S. R.: "Curve Description Using the Inverse Hough Tansform" Pattern Recognition, Vol.25 No.34,pp.883-290,1990
[9] Richard O. Dude and Peter E. Hart: "Use of the Hough Transformation to Detect Lines and Curves in Pictures" Comm. of ACM, Vol.15, No.1,1972
[10]Van Veen T. M. and Groen F. C. A.: "Discretization Errors in the Hough Transform" Pattern Recognition, Vol.14, No.1, pp.137-146,1981

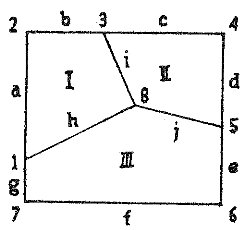

Fig.1 A Linear Graph

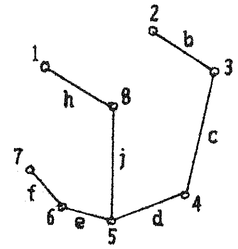

Fig.2. The Tree of the Linear Graph(Fig.1)

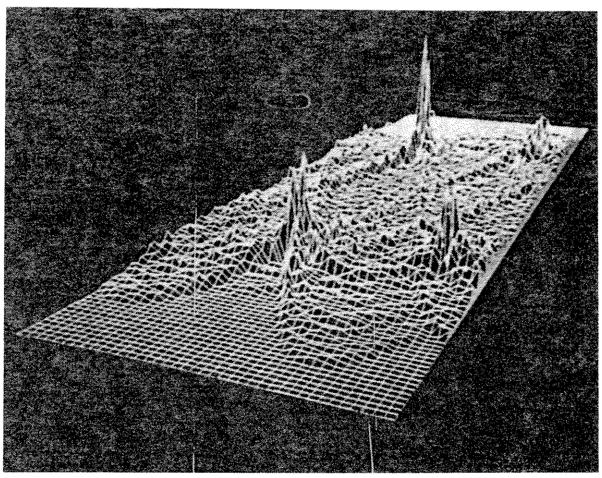

Fig.3.a) Parameter Space of Algorithm 1

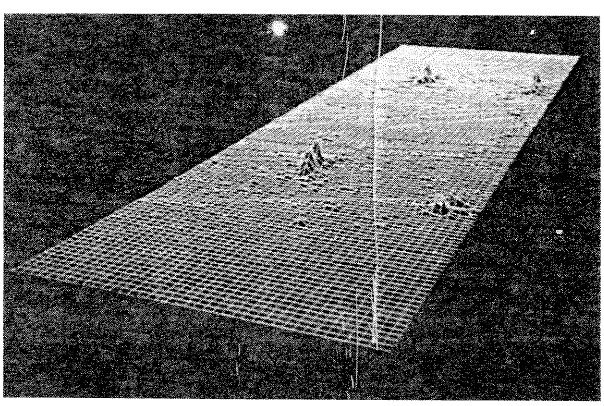

Fig.3.b) Parameter Space of Algorithm 2

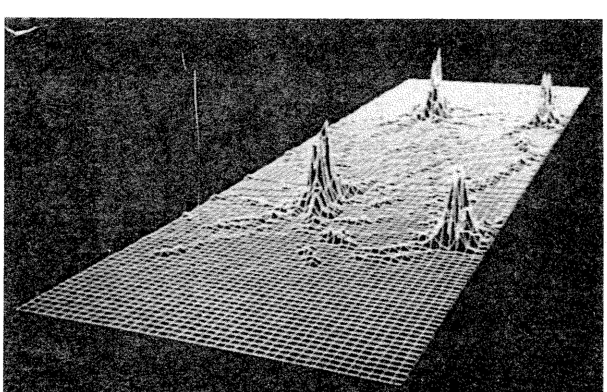

Fig.3.c) Parameter Space of Algorithm 3

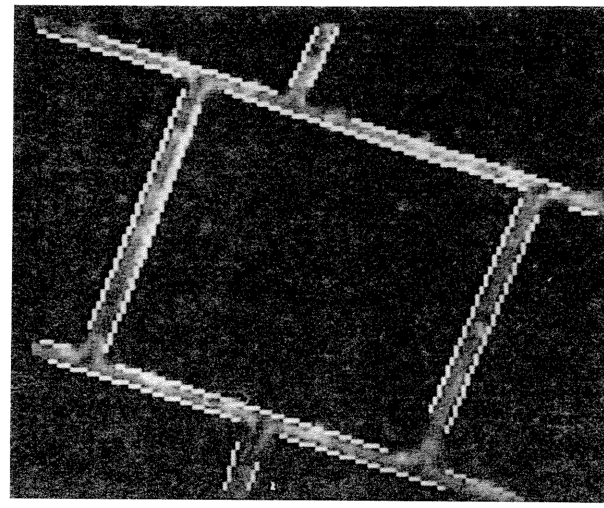

Fig.4.a) Straight line of Hough Transformation

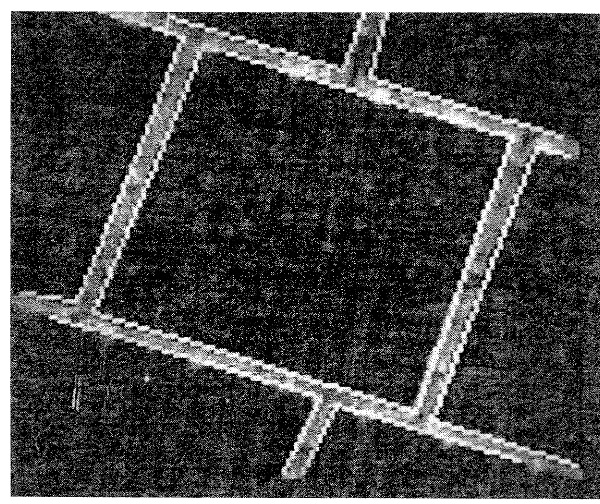

Fig.4.b) Intersection processing

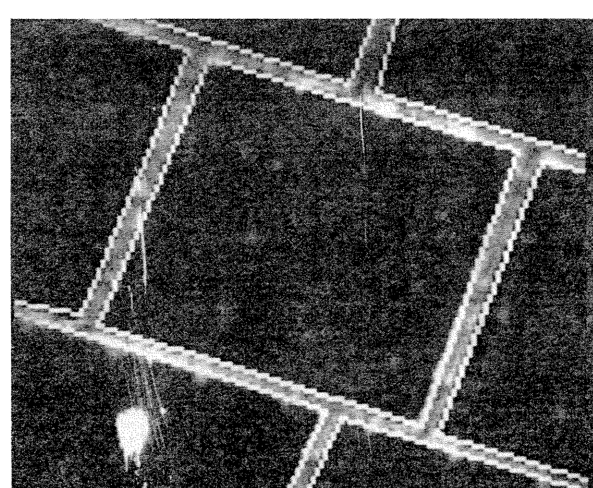

Fig.4.c) End-point extension

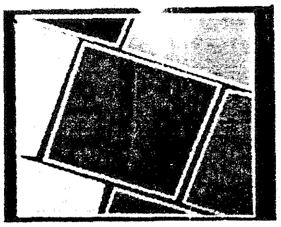

Fig.5 Region Segmentation

Algorithm	I	II	III
Time	49.3	15.3	19.4

Unit: second

Table 1

Plane	edge	star __ end
1	1	(33, 16) __ (10, 93)
	2	(10, 93) __ (77, 120)
	3	(103, 40) __ (77, 120)
	4	(33, 16) __ (103, 40)
2	1	(0, 0) __ (0, 46)
	2	(0, 46) __ (18, 51)
	3	(34, 0) __ (18, 51)
	4	(0, 0) __ (34, 0)
3	1	(0, 46) __ (0, 52)
	2	(16, 58) __ (0, 52)
	3	(16, 58) __ (0, 110)
	4	(0, 110) __ (0, 123)
	5	(8, 98) __ (0, 123)
	6	(8, 98) __ (74, 124)
	7	(74, 124) __ (71, 139)
	8	(71, 139) __ (74, 139)
	9	(99, 64) __ (74, 139)
	10	(99, 64) __ (119, 73)
	11	(119, 68) __ (119, 73)
	12	(101, 60) __ (119, 68)
	13	(119, 0) __ (101, 60)
	14	(116, 0) __ (119, 0)
	15	(116, 0) __ (105, 36)
	16	(36, 10) __ (105, 36)
	17	(38, 0) __ (36, 10)
	18	(34, 0) __ (38, 0)
	19	(34, 0) __ (18, 51)
	20	(0, 46) __ (18, 51)
	21	(33, 16) __ (10, 93)
	22	(10, 93) __ (27, 120)
	23	(103, 40) __ (77, 120)
	24	(33, 16) __ (103, 40)

Plane	edge	start __ end
4	1	(0, 52) __ (0, 110)
	2	(16, 58) __ (0, 110)
	3	(16, 58) __ (0, 52)
5	1	(0, 123) __ (0, 139)
	2	(0, 139) __ (71, 139)
	3	(74, 124) __ (71, 139)
	4	(8, 98) __ (74, 124)
	5	(8, 98) __ (0, 123)
6	1	(38, 0) __ (116, 0)
	2	(116, 0) __ (105, 36)
	3	(36, 10) __ (105, 36)
	4	(38, 0) __ (36, 10)
7	1	(119, 0) __ (119, 68)
	2	(101, 60) __ (119, 68)
	3	(119, 0) __ (101, 60)
8	1	(119, 73) __ (119, 139)
	2	(74, 139) __ (119, 139)
	3	(99, 64) __ (74, 139)
	4	(99, 64) __ (119, 73)

Table 2