Using Fractal Model in Documents' Security

Ali .A. Saeid*

Abstract

The research aims to build graphical models of complex formation and morphology which can be used to secure paper documents from attempts of fraud, especially those trying to expose them to scanning. Therefore, the scanned version will differ in features from the original one, particularly the part that contains the seal (the decorative form). The forms were generated by using a model of Koch's star; keeping in mind that points of the lines of the new form will be generated by using Brsenham's theory in order to generate the straight line.

استخدام النماذج التشعبية في أمنية الوثائق

الخلاصة

يهدف البحث إلى بناء نماذج رسومية معقدة في التكون والتشكل يمكن استخدامها في تـامين الوثـائق الورقية من محاولات التزوير. خاصـة الأمر المتعلق بمحاولة تعريضها لعملية المسح الضـوئي وبالتـالي ستكون النسخة المصورة تختلف في معالمها عن النسخة الأصلية وخاصـة الجزء المحتـوي علـى الخـتم (النموذج ألزخرفي).تم توليد الأشكال باستخدام نموذج كوخ ألنجمي .علما أن خطوط الشكل المتولد يتم توليد نقاطها باستخدام النظرية العامة لبرزنهام لتوليد إلخط المستقيم.

1-Introduction

Spend a lot of government agencies and non government funds, to support research projects related in document security. especially after the availability of new technologies in printing and electronic imaging [5][6] .On this basis the deal of this research for the first time the use appropriate of mathematical techniques to limit the attempts to imitation and forgery of documents. We have been relying on a Koch's model to generate the required figures which depends generated based on the concept of middle third cantor set. the research mention the innovative way in determining the ideal point to head form under the conditions specified leaving the other points where the unwanted as for the straight lines between points generated has been relying on the theory Bresnham in the generation of string –line with its public conditions and give reasons. The rationales underlying the choice of theory Brsenham related to limit points to be generated without the other and which together form the decorative containing the words written. Therefore this research and innovative in his own way that is consistent and compatible with the technical methods used in the security of documents and to generate accurate models conformation preventing imitation [6][7] see figure (ß).in

*Information & Communication Technology Center, University of Technology/Baghdad

conclusion it must be noted that this research has come a complement to what has been previously adopted in the construction of design models based on spline functions[9][10].

2-Bresenham Line Algorithm

Bresenham'salgorithm [Bresenham 1965] is the classic technique for plotting lines on bitmaps: it approximates linear segments defined by rational coefficients using only integral points. The algorithm calculates a finite set of points on the integer lattice with minimum total vertical distance from the original line segment [8].

The Cartesian *slope-intercept equation* for a straight line is:

$$=m^.x+b \qquad \dots..(1)$$

with *m* representing the slope of the line and b as the y intercept. Given that the two endpoints of a line segment are specified at positions (x_1, y_1) and (x_2,y_2)as shown in Fig.1, we can determine values for the slope m and y intercept b with Algorithms for displaying straight lines are based on the line equation (1) and the calculations given in Esq. (2) and (3).

For any given *x* interval Δx along a line, we can compute the corresponding y interval Δy from eq (2) as:

$$\Delta y=m \Delta x \qquad \dots\dots.(2)$$

Similarly, we can obtain the *x* interval Δx corresponding to a specified Δy as:

$$\Delta x= \Delta y/m \qquad \dots\dots(3)$$

To illustrate Bresenharn's approach, we first consider the scan-conversion process for lines with positive slope less than 1. Pixel positions along a line path are then determined by sampling at unit *x* intervals. Starting from the left endpoint (xo, yo) of a given line, we step to each successive column *(x* position) and plot the pixel whose scan-line y value is closest to the line path[4]. Figure(2a)demonstrates the kth step in this process. Assuming we have determined that the pixel at (x_k, y_k) is to be displayed, we next need to decide which pixel to plot in column x_{k+1}, Our choices are the pixels at positions (x_{k+1}, y_k) and(x_{k+1}, y_{k+1}).At sampling position $x_{k+1,}$ we label vertical pixel separations from the mathematical line path as d_1, and d_2 Fig. (2-b). The y coordinate on the mathematical line at pixel column position x_{k+1} is calculated as,[4]

$$y=m(x_k+1)+b \qquad \dots\dots (4)$$

then

$$d_1 = y-y_k$$
$$= m(x_k+1)+b-y_k$$

And

$$d_2 = (y_k+1)-y$$
$$=y_k+1-m(x_k+1)-b$$

The difference between these two separations is.

$$d_1-d_2=2m(x_k+1) -2y_k+2b-1 \dots\dots.(5)$$

A decision parameter p_k for the kth step in the line algorithm can be obtained by rearranging Eq.5 so that it involves only integer calculations. We accomplish this by substituting m= $\Delta y/\Delta x$, where Δy and Δx are the vertical and horizontal separations of the endpoint positions, and defining:

$$P_k= \Delta x (d_1-d_2)$$
$$=2\Delta y.x_k-2 \Delta x.y_k+c \qquad \dots\dots.(6)$$

The sign of p_k is the same as the sign of d_1-d_2 since $\Delta x > 0$ for our example. Para meter c is constant and has the value $2\Delta y +\Delta x (2b - l)$, which is independent. of pixel position and will be eliminated in the recursive calculations for p_k. If the pixel at y_k is closer to the line path than the pixel at y_{k+1} (that is, $d_1 < d_2)$, then decision parameter p_k is negative. In that case, we plot the lower pixel; otherwise, we

plot the upper pixel. Coordinate changes along the line occur in unit steps in either the *x* or *y* directions. Therefore, we can obtain the values of successive decision parameters using incremental integer calculations. At step $k + 1$, the decision parameter is evaluated from Eq. (6) as

$P_{k+1} = 2 \Delta y \cdot x_{k+1} - 2 \Delta x \cdot y_{k+1} + c$

Subtracting Eq.(6) from the preceding equation, we have

$P_{k+1} - p_k = 2 \Delta y (x_{k+1} - _k) - 2\Delta x (y_{k+1} - y_k)$ **(7)**

But $x_{k+1} = x_k + 1$ so that

$P_{k+1} = p_k + 2\Delta y - 2\Delta x (y_{k+1} - y_k)$ **(8)**

where the term $y_{k+1} - y_k$ is either 0 or 1, depending on the sign of parameter p_k. This recursive calculation of decision parameters is performed at each integer *x* position, starting at the left coordinate endpoint of the line. The first parameter, p_0 is evaluated from Eq. 6 at the starting pixel position (x_o, y_o) and with (M) evaluated as Δ_x / Δ_Y:

$P_0 = 2 - \Delta_X - \Delta_Y$ (9)

We can implement Bresenham's line algorithm for a line with below cases [11].

- *Implementation of Algorithm for First Octant.*
- *Implementation of Algorithm for Four Octants with Slopes -1 to 1*

- *Generalized Implementation for all Directions.*

2-1 Generalized Implementation for all Directions

Bresenham's algorithm is generalized to lines with arbitrary slope by considering the symmetry between the various octants and quadrants of the *xy* plane. For a line with positive slope greater than 1, we interchange the roles of the x and y directions. That is, we step along the *y* direction in unit steps and calculate successive x values nearest the line path. Also, we could revise the program to plot pixels starting from either endpoint. If the initial position for a line with positive slope is the right endpoint, both *x* and y decrease as we step from right to left. To ensure that the same pixels are plotted regardless of the starting endpoint, we always choose the upper (or the lower) of the two candidate pixels whenever the two vertical separations from the line path are equal $(d_1 = d_2)$. For negative slopes[4].

2-2   General   Bresenham's algorithm

A   full   implementation   of Bresenham   algorithm   requires modification for lying in the other octants.   These   can   easily   be developed by considering quadrant in which the line lies and its slope.

X=X1
Y=Y1
dX=Abs (X2-X1)
dY=Abs(Y2-Y1)
S1=Sign (X2-X1)
S2=Sign (Y2-Y1)
If dY > dX Then
Begin
T=dX   :   dX=dY   :   dY=T   :
Interchange=1
End
Else
Interchange =0
End If
po= 2 dy - dx
For I=1 to dX
Plot (X,Y)
While (po ≥ 0)
Begin
If Interchange=1 Then X=X + S1
Else Y= Y + S2

End If
E= E - 2 dx
End While
If Interchange=1 Then Y=Y + S2
Else X=X + S1
End If
po = po+ 2 dy
Next I
Finish
Based on aforementioned and according to the theory of General Bresenham's algorithm, points of the straight line passing from the point (106, 124) to the point (130, 138) will be generated. The data in the table (a) list the points generated and the values of p_o that are being generated.

A set of lines of multiple directions can be generated starting from the point of300,350 () as shown in Figure (4-a),(4-b). A set of parallel lines passing through parallel points can also be generated, Figure(C) illustrates that. The previous parallel lines could be re-generated with the exception of generating points that passing through other points being relatively to the image as shown in figure D or being generated with another color as shown in figure E

3-Middle third Cantor set
The middle third Cantor set is one of the best known and most easily constructed fractals; nevertheless it displays many typical fractal characteristics. It is constructed from a unit interval by a sequence of deletion operations[a], see figure (3) Let E_0 be the interval [0, 1]. (Recall that [*a, b*] denotes the set of real numbers *x* such that $a < x < b$.) Let E_1 be the set obtained by deleting the middle third of E_0, so that E_1 consists of the two intervals [0, 1/3] and [2/3, 1]. Deleting the middle thirds of these intervals gives E_2[3];

thus E_2 comprises the four intervals [0, 1/9], [2/9 , 1/3], [2/3 , 7/9], [8/ 9 , 1]. We continue in this way, with E_k obtained by deleting the middle third of each interval in E_{k-1}. Thus E_k consists of 2*k* intervals each of length 3−*k*. The *middle third Cantor set F* consists of the numbers that are in E_k for all *k*; mathematically, *F* is the intersection K=E_K. The Cantor set *F* may be thought of as the limit of the sequence of sets E_k as *k* tends to infinity. It is obviously impossible to draw the set *F* itself, with its infinitesimal detail, so 'pictures of *F*' tend to be pictures of one of the E_k, which are a good approximation to *F* when *k* is reasonably large[1][2]; see figure (5). At first glance it might appear that we have removed so much of the interval [0, 1] during the construction of *F*, that nothing remains. In fact, *F* is an infinite (and indeed uncountable) set, which contains infinitely many numbers in every neighborhood of each of its points. The middle third Cantor set *F* consists Construction of the middle third Cantor set *F*, by repeated removal of the middle third of intervals. Note that F_L and F_R, the left and right parts of *F*, are copies of *F* scaled by a factor 1/3[1].

precisely of those numbers in [0, 1] whose base-3 expansion does not contain the digit 1, i.e. all numbers .
*a*13−1 + *a*23−2 + *a*33−3 + ·· · with *ai* = 0 or 2 for each *i*. To see this, note that to get E_1 from E_0 we remove those numbers with *a*1 = 1, to get E_2 from E_1 we remove those numbers with *a*2 = 1, and so on**.**

Von Koch model
The von Koch curve has features in many ways similar to the middle third Cantor set. It is made up of four

'quarters' each similar to the whole, but scaled by a factor 1/3,[1][2].

The fine structure is reflected in the irregularities at all scales; nevertheless,this intricate structure stems from a basically simple construction. Whilst it is reasonable to call F a curve, it is much too irregular to have tangents in the classical sense.

the von Koch curve, will also be familiar to many readers; see figure (6). We let E_0 be a line segment of unit length. The set E_1 consists of the four segments obtained by removing the middle third of E_0 and replacing it by the other two sides of the equilateral triangle based on the removed segment. We construct E_2 by applying the same procedure to each of the segments in E_1, and so on. Thus E_k comes from replacing After fragmenting the line into three parts, the top point of the triangle (1β) will be determined as shown in figure (5). The selection of this point is made by generating circle's points according to the equation (6), its diameter (R) starts from the point (2β) and ends to the point (3β); its radius is set equal to the value of section S1. After that, and in order to determine the ideal point which represents triangle's top point , the process is done by calculating the length of distance (L_1, L_2) between the points of 3β, 2β and the points of the generated circle ($β_X$, $β_Y$) according to the equation of distance (7). The appropriate point will be selected if the two distance lengths are equal, i.e. when $L_1 = L_2$, which are the sides of isosceles triangle Y=R.Sin(θ)

$$Y = R.\mathrm{Sin}(\theta)$$
$$X = R.\mathrm{Cos}(\theta) \qquad (6)$$

$$L = \sqrt{(\beta - \beta x)2 + (\beta - \beta y)2} \qquad (7)$$

After that, the selected point will be connected with the both sectors at the end of sector (S_1) and the form generated will be similar to the sector (E_1) of the figure (6). This is a problem related with the selection of point 1β because there is another point, α see figure(7), in which the same conditions applied on. To avoid such selection, the first point that the condition of $L_1 = L_2$ applied on it will be selected, beginning from 2β to 3β on the direction of clockwise. The previous steps could simply be repeated to obtain a new form like that of figure (6) sector E1. Hereby, the new generated form is in figure (8-a) in the 1st step and (8-b) in the next second step.

4-Results

Depending on what was mentioned about the theory of Bresenham (general model) to generate a straight line as well as Koch's model, the hereunder forms will be generated on the basis that the generated form would depend on the concept of Koch and its straight lines are generated according to the Bresenham's. Results will be reviewed in two phases: the first one is related to Koch model, similar to figure 8-a) while the second phase will be similar to the model of Koch; figure 8-b).

First phase: the generation of the form in the figure (11) depends on Koch's star, nothing that generation will be repeated from left to right. The points generated based on Bresenham's will be done so in relatively to the image of figure (10) of the newly generated form, which contains the word (windows); as it has

3442

been discussed in the beginning of the research.

Second Phase: The other model, which is generated on a *multi-phase,* depends on Koch's star with repeated generation of the points of the form, from left to right side like layers repeat themselves … i.e. the figure (12) is horizontally repeated of the last figure. Keeping in mind that the points of generation . based on the theory of Bresenham for straight line generation).

Or the following statement can be generated based on the corresponding image in the same way as in the previous generation … Figure (13).

5-Conclusions

a-Depending on (Brezenham) algorithm, especially its general condition, to generate the form points; so that the points generated or drawn, which their coordinates are apposite to form's image, would be excluded, as described in the previous stages. Noting that it cannot be depending on the ready functions, which are available in most software applications, to draw a straight line within the instructions of drawing functions. The reason is that they completely generate the points of the straight line confined between two points and without using any pre-condition determines the size of the area to be excluded or drawn.

b-The lines of the forms were generated depending on the general condition of (brezenham) algorithm; because this condition restricts neither the location of points of the area to be painted within any quarter of circle nor the slope value of the points' coordinates.

c-Decoration models are being built according to complex mathematical

techniques and on multiple stages in order to increase the complexity of knowing the mathematical base adopted by the counterfeiters.

d- model, generated based on design characteristics of complex and interlocking lines,is being depended. It includes forms or phrases which shapes are relatively set out to the less dotted density. The above mentioned specifications work together to form an obstacle to counterfeiters when they attempt to re-copy the original document using scanner devices. Herewith the new model would be contrary to the original one in terms of its specifications.

References

[1]Kenneth Falconer.2003. FRACTAL GEOMETRY Mathematical foundations and Applications . John Wiley & Sons Ltd.

[2]Benoit B. Mandelbro.1983. THE FRACTAL GEOMETRY OF NATURE. w. H. FREEMAN AND COMPANY

[3] Gerald Edgar.2008. Measure Topology and Fractal Geometry Springer

[4] Donald . Hearn, Pauline .barker 1989 Computer graphic.

[5] Cormack Herley, Poorvi Vora and Shawn Yang .2005. Detection and deterrence of counterfeiting of valuable documents.University, Washington DC

[6] Marcela M.Williams and Richard G. Anderson Currency Design in the United States and Abroad Counterfeit Deterrence and Visual Accessibility

[7] U.S. Department of the Treasury . Security Features Introduced in Series 1996 Notes ©The U.S.

Department of the Treasury - Bureau of Engraving and Printing

[8] MITCHELL A. HARRIS, EDWARD M. REINGOLD , Line Drawing, Leap Years, and Euclid ACM Computing Surveys, Vol. 36, No. 1, March 2004.

[9] Abdul.Mohssin jabber AL-Maali 2005. modification the mathematical model of curve university of technology Al-Rasheed college

[10] Ali .A. SAEID ,2002 developing mathematical algorithm of curve for using in computer design university of technology Al-Rasheed college .

[11] Kenny Hoff . September , 1995 ,Computer Graphics, Derivation of Bresenham's Line Algorithm

**Table of the straight line points from point(107,124) to(130,137)
by using bresinham algorithm**

NO	X	Y	Po	NO	X	Y	Po
1	107	124	14	13	119	131	18
2	108	125	-14	14	120	132	-10
3	109	125	26	15	121	132	30
4	110	126	-2	16	122	133	2
5	111	126	38	17	123	134	-26
6	112	127	10	18	124	134	14
7	113	128	-18	19	125	135	-14
8	114	128	22	20	126	135	26
9	115	129	-6	21	127	136	-2
10	116	129	34	22	128	136	38
11	117	130	6	23	129	137	10
12	118	131	-22	24	130	137	-18

Figure (β) Graphicl models used in documents security
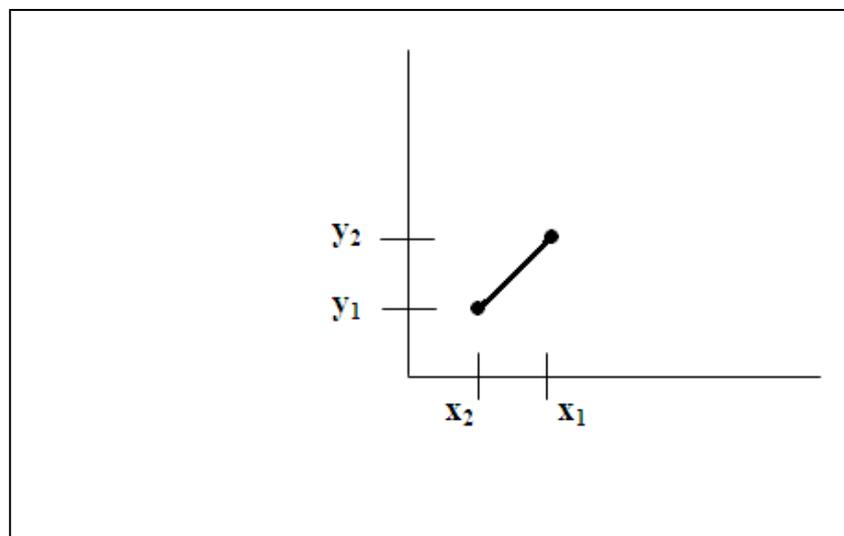
Figure (1) Straight line segment with sampling positions along the x ax' s between x, and x2

(a)   (b)
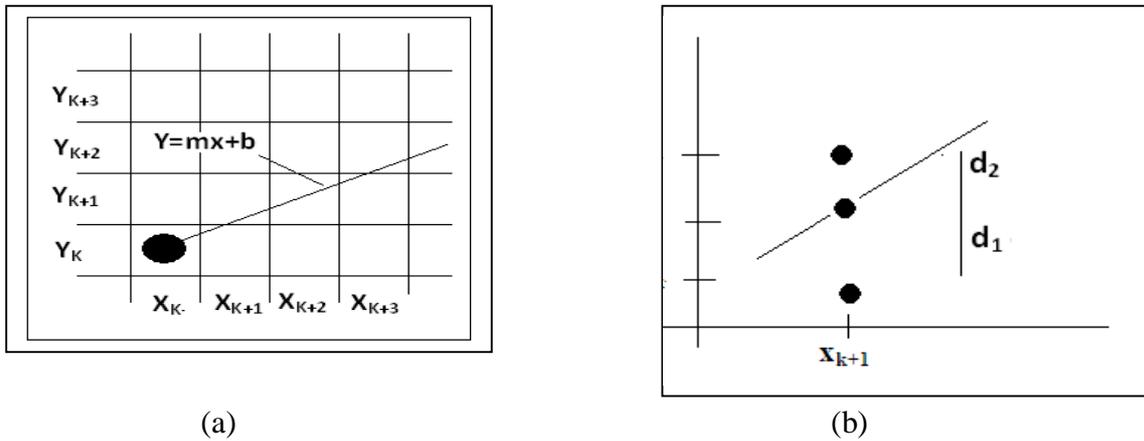
Figure (2) Section of the screen grid showing a pixel in on column x_k scan line yk that is to be plotted along the path of a line segment with slope `0<m<1`.
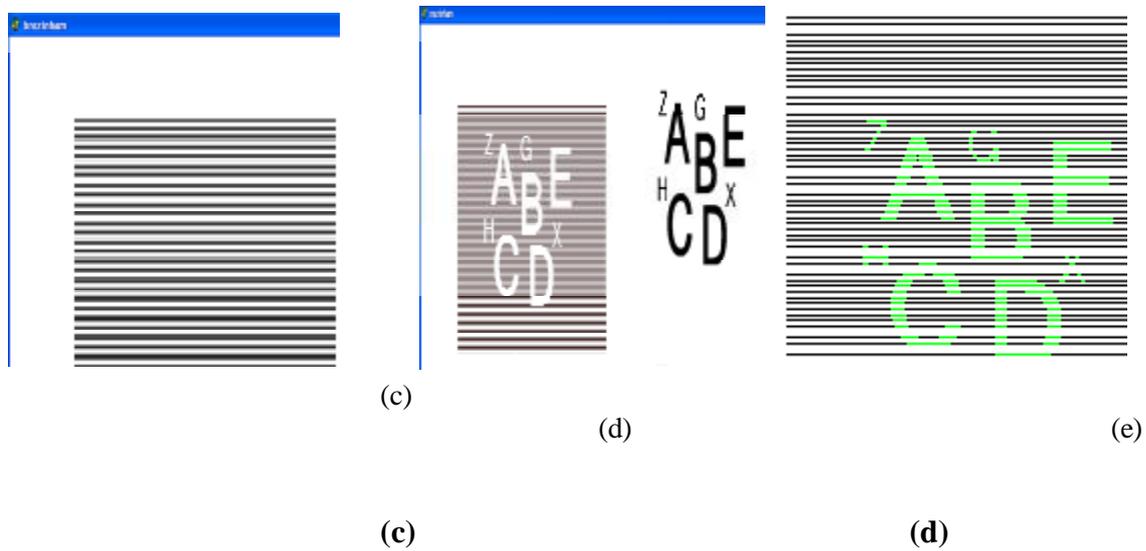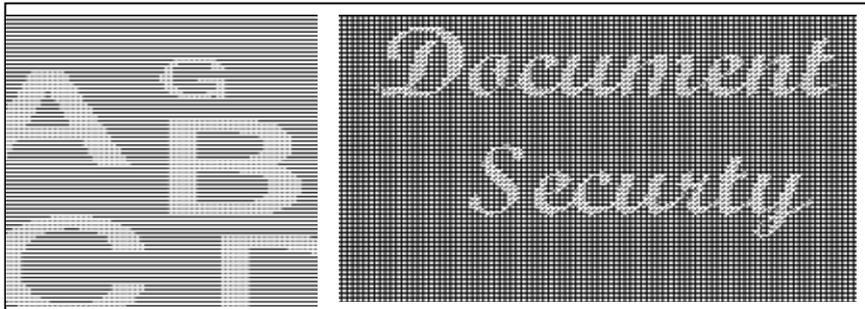
(a)   **(b)**

(c)

(d)   (e)

(c)   **(d)**

Figure (3) Models generated using the theory of lines Brsenham

3446

Figure(4)Parallel lines generated points are generated relative to the image
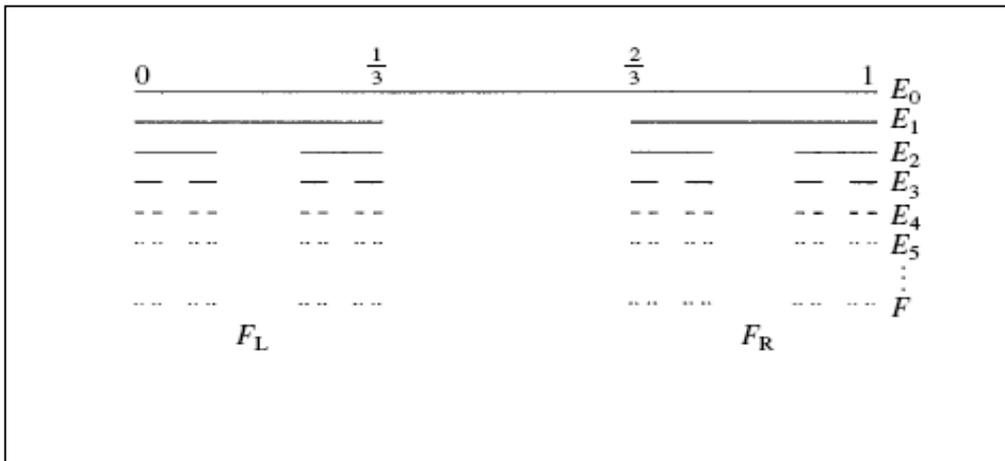
Figure (5) Construction of the middle third Cantor set F by
repeated removal of the middle third of intervals
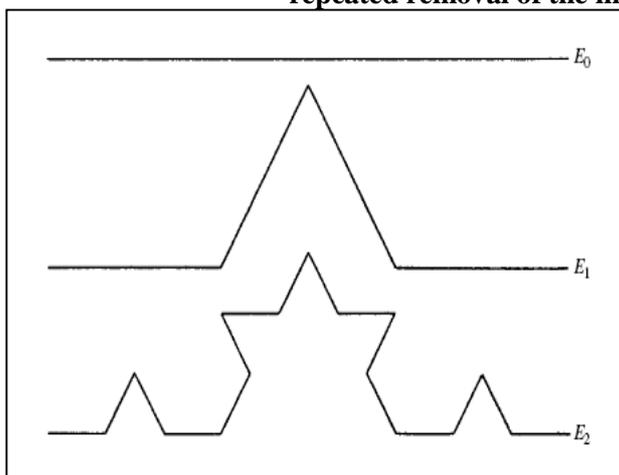
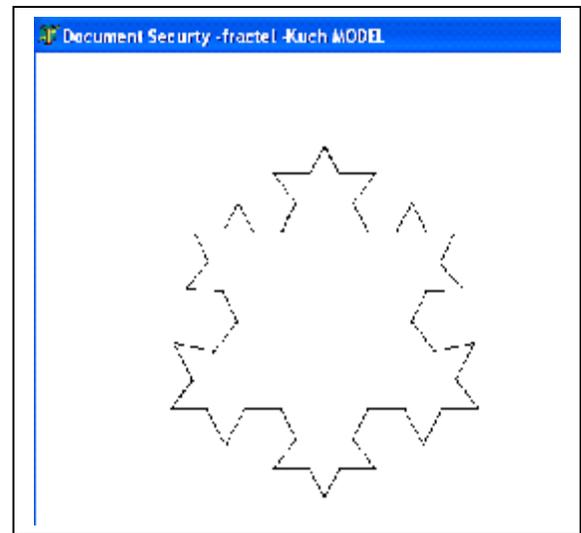**Figure (6) Koch curve model**   **Figure(7) Choosing the ideal point**
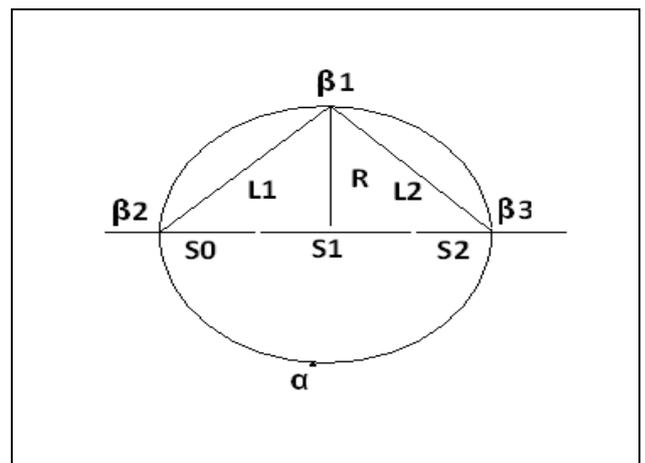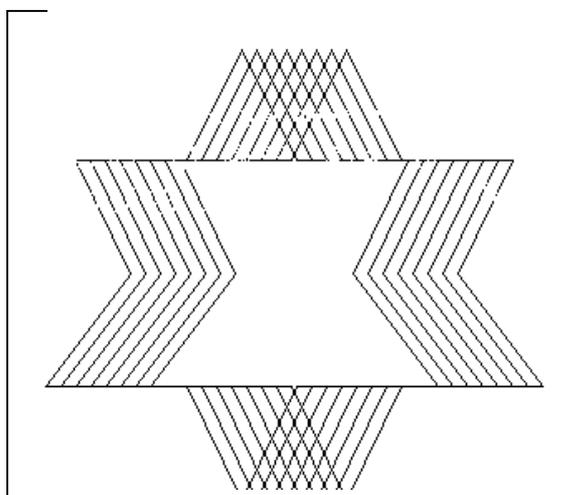
3447
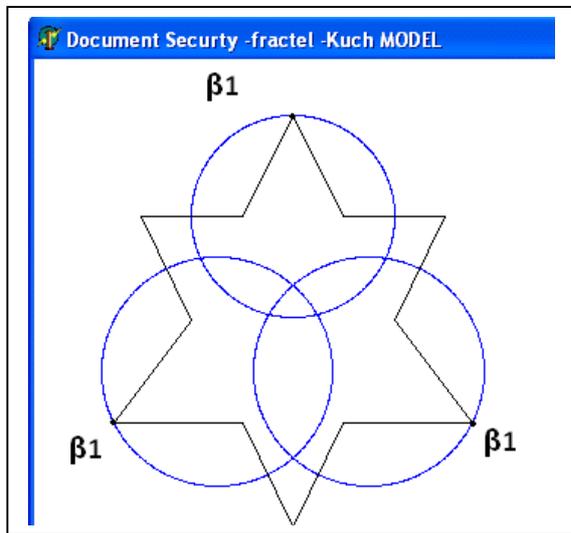
Figure (9) Generating a horizontal
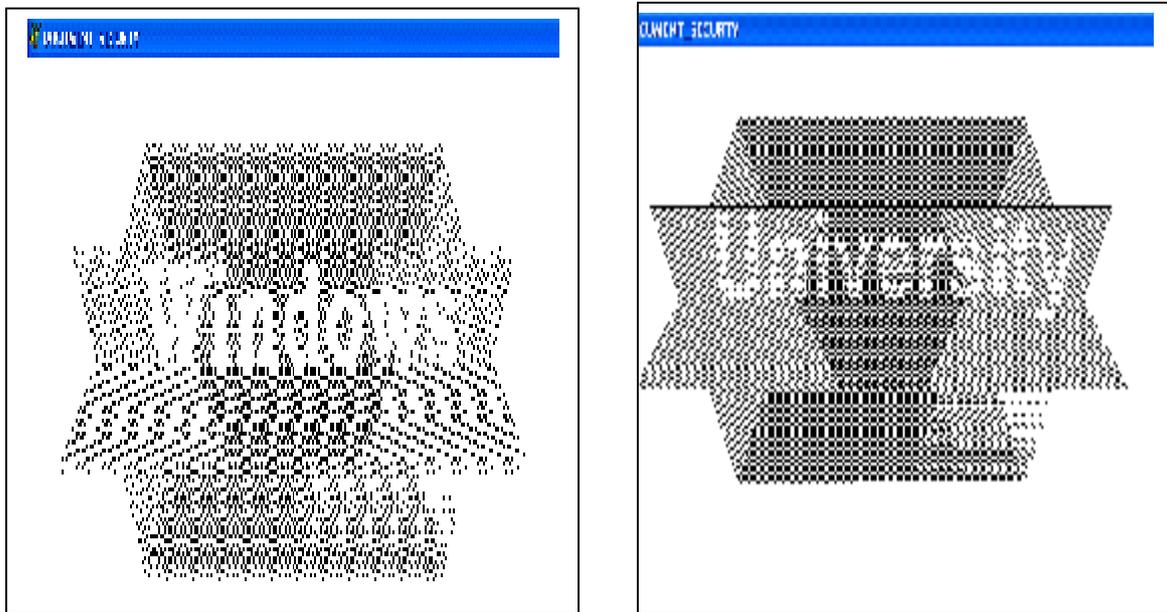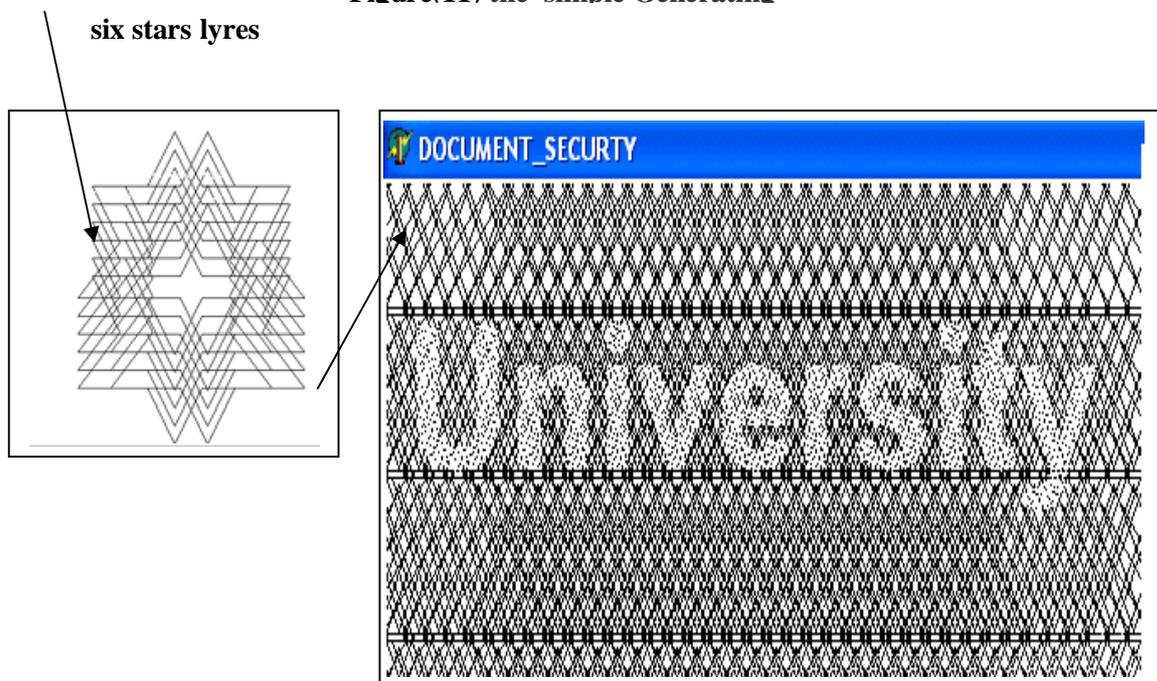

Figure(10)   the corresponding Image

(a)

(b)

Figure(8) Koch curve model, Simple and compound

3448

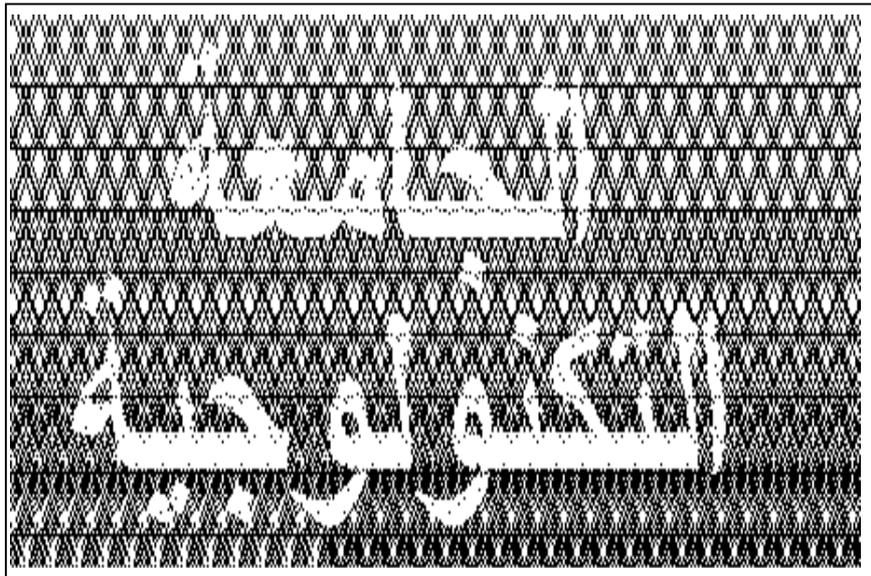**Figure(11) the simple Generating**

six stars lyres

Figure (12) the compound Generating

Figure (13)   Generating words model

3450