KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

Advanced Computer Graphics
(ICS – 535)

Term Project Report

Scan Conversion of Spirals

Submitted To

Dr. Muhammad Sarfraz

By

Syed Nazim Nawaz
(ID # 210291)

Abstract:

Drawing straight lines and curved primitives is an important part of computer graphics. Due to the success of raster displays, scan conversion algorithms are fundamental in computer graphics. However complicated curve primitives such as spirals are less considered for direct scan conversion. In Cartesian coordinates they are typically transcendental functions, which make the evaluation on Cartesian grids an inefficient process. This project discusses the issues concerning the scan conversion of Archimedes spiral. A simple algorithm based on the piecewise circular approximations is implemented and investigated. Variations of the algorithm to scan convert other types of spirals are also investigated.

Keywords: spirals, scan conversion, circular approximations, raster display.

1. Introduction

Straight lines and cuved primitives have always been an integral part of Computer Graphics [1]. Scan conversion algorithms such as the Bresenham's algorithm and the midpoint technique are fundamental to computer graphics. More complicated curve primitives, however, have been less considered as objects for direct scan conversion. Most of these curves could only be scan converted using general purpose graphing techniques[2]. This work concentrates on scan conversion of Archimedes spirals.

A *spiral* is a curve that traces out a path around a point called the *pol*e. The pole is the center of the tracing activity. The distance from the pole to a point on the trace of the curve is called the *radiu*s [3]. The radius for a spiral is an increasing function of the angle the trace makes with respect to a coordinate axis embedded in the plane. The actual function that controls the radius determines the type of spiral [3].

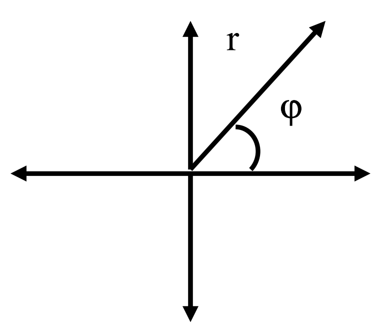


Figure 1: Polar Coordinates

In Polar coordinates this spiral is a line $r(\varphi)=m*\varphi+b$. One can distinguish several classes of spirals such as Polynomial (e.g, Archimedean class of spirals) spiral and

Exponential or Logarithmic spiral [1]. In general a spiral is characterized by the equation $r(\varphi)=m*\varphi^{p}+b$ where $p \in R$ and is sometimes called an Archimedean spiral[1]. The “spiral of Archimedes” is one of the spirals that belong to this family (if p=1). When p= -1 the spiral is called a Hyperbolic spiral. The Fermat spiral is also a kind of Archimedian spiral. For a Fermat spiral p=1/2. The inverse of Fermat spiral is called a “Littus spiral” i.e. for a Littus spiral p=-1/2. Figure 2 Below shows the different forms of spirals. For Logarithmic or exponential spirals the radius grows exponentially with the angle. The logarithmic relation between radius and angle represented with the equation $r(\varphi)=e^{a\varphi+b}$ where the relation with the angle φ is non-linear.

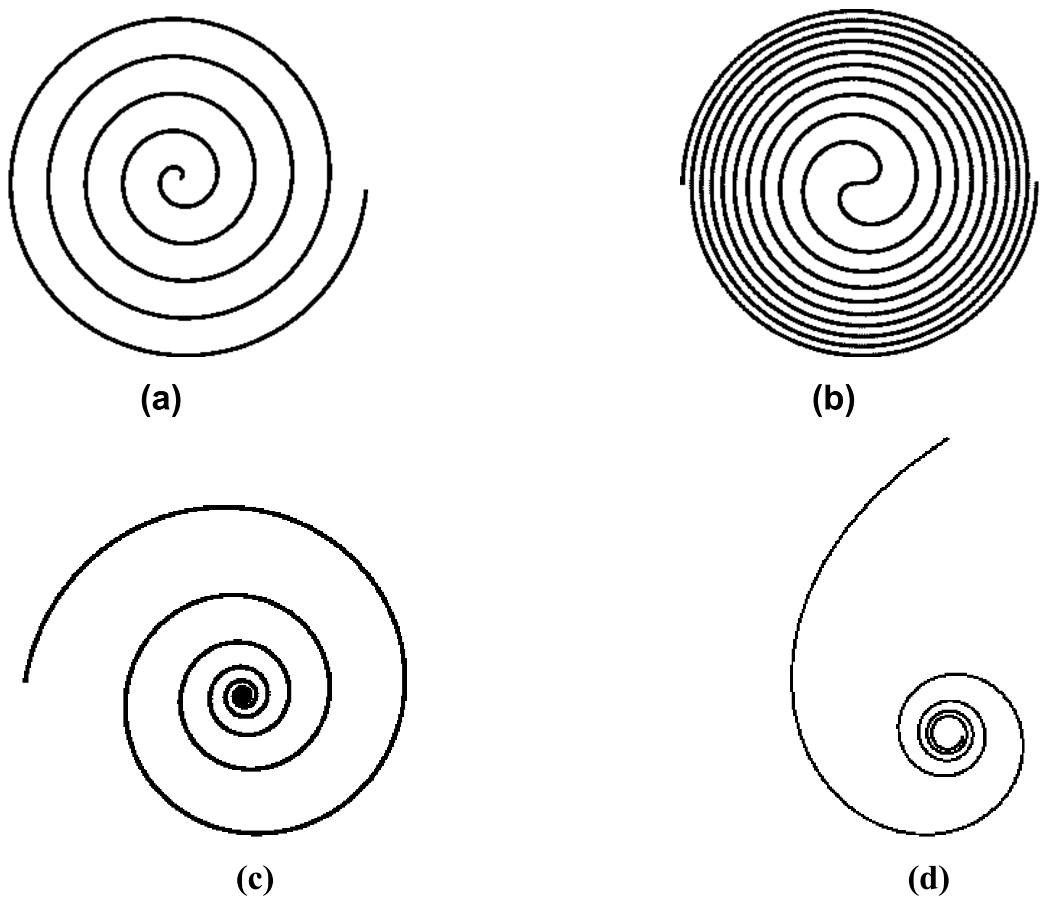


Figure 2: Different classes of Spirals
(a) Archimedes Spiral (b) Fermat Spiral (c) Logarithmic Spiral (d) Hyperbolic Spiral

Spirals are found in nature in many different forms. Some of the common forms of spirals in nature are shown in figure 3. All the images in Figure 3 are dominated by spiraling patterns.

Figure 3: Different spiral patterns: web, fruits and vegetables: succulenta, pinecone, caulis.

A major reason for spirals to be common in nature and the arts is the efficient use of space. Growing processes of several organisms have always been studied and exhibit spiral patterns. The spiral has been named “the curve of life” [6], as exponential functions model reproduction processes in biology, explosions in chemistry, evolution processes in economy, etc.

Spirals are also being employed for visual data mining: [5]. “Spirals” are one of the available options and permits a different approach in visualizing information A well developed spiral tool offers the possibility of giving a practical view of significant aspects of the data; nevertheless it permits a very compact representation of the data, which is obtained by utilizing all the space that is available on a display device.

As most spirals are transcendental functions in Cartesian coordinates, simple drawing implementations are difficult to realize and incremental algorithms are likely to suffer from drift, as scan conversion of high-degree polynomials is cumbersome and error-prone [4]. A piecewise approximation using a simple primitive in polar coordinates is more promising [1].

In the following, some straightforward approaches for drawing spirals on raster displays are discussed and their suitability with respect to the characteristics accuracy, drift, and speed is evaluated. We conclude that among these approaches only piecewise elliptical and circular approximations deserve attention. However, piecewise elliptical approximation is shown to have systematic derivative error and varying approximation quality depending on the spiral’s slope m. A simple and fast approximation algorithm based on circular arcs is presented. This algorithm is discussed in detail and its relative performance is analyzed. [2]. The main contributions of this work are strategies to efficiently compute the centers and radii of circular segments and the generalization of algorithm to generate any type of spiral.

The rest of the paper is organized as follows section 2 discusses why direct scan Conversion algorithm like midpoint line algorithm cannot be applied to Curved primitives like spirals. Section 3 discusses the piecewise elliptical approximation technique section 4 discusses the circular approximation technique Section 5 covers the project goals and Implementation issues and Finally we conclude in Section 6.

2. Direct Scan Conversion algorithm

A general approach for scan converting curves is the midpoint algorithm [4]. The idea is to represent the curve in an implicit form F(X,Y)=0. First the midpoint of the two candidate pixel coordinates is taken. F is applied to this midpoint and depending on the above /below (Positive or Negative value of F) information one of the candidate pixel is chosen. However before finding the midpoint the curve is split into regions which meet at points where the patial derivatives of F are equal. In these regions X or Y Component is increased by one and the other component is increased or not increased depending on the above/below information (Decision variable).

The general steps of the Midpoint algorithm for Direct Scan Conversion are as follows.

1. Take the curve equation F(x,y)=0;
2. Split the curve to regions which meet at points where the partial derivatives of F are equal.
3. Identify the Midpoint of the two candidate pixel coordinates.
4. Apply the curve equation to the midpoint and Identify the value of F.
5. Depending on the value of F Identify whether the curve lies above or below the midpoint i.e the decision variable.
6. change the value (Increased or not increased) of the other component based on the decision of Step 5.
7. Identify all the points for the curve in the first regions.

8. Using Symmetry considerations Identify the Corresponding points in all the other regions.

2.1 Case Study: Generation of a spiral using the circle equation $x^2+y^2=r^2$

Consider the Cartesian equation of a circle $x^2+y^2=r^2$. Considering φ can be expressed as φ=arc tan(y/x) the Cartesian equation of a linear spiral can be expressed as

$$F(x,y)= x^2+ y^2-(m.\ \text{arc}\tan(y/x) +b)^2$$

$$\left[\frac{\partial F(x,y)}{\partial x},\frac{\partial F(x,y)}{\partial y}\right]=\left[2x+\frac{y}{x^2}.\left[\frac{2mb}{1+\left(\frac{y}{x}\right)^2}+\frac{2m^2\arctan(y/x)}{1+\left(\frac{y}{x}\right)^2}\right].2y-\frac{1}{x}.\left[\frac{2mb}{1+\left(\frac{y}{x}\right)^2}+\frac{2m^2\arctan(y/x)}{1+\left(\frac{y}{x}\right)^2}\right]\right]$$

$$\arctan(s)\cong s-\frac{1}{3}s^3+\frac{1}{5}s^5-\frac{1}{7}s^7+.......(|S|<1)$$
$$\arctan(s)\cong\frac{\pi}{2}-[\frac{1}{s}-\frac{1}{3s^3}+\frac{1}{5s^5}-\frac{1}{7}+.......(S>1)$$

From the above equation it can be noticed that errors cannot be avoided as the arctan function is transcendental for some arguments. Thus the algorithm is almost prone to drift. To avoid drift the inverse tangent function can be expanded to the following power series

$$\arctan(s)\cong s-\frac{1}{3}s^3+\frac{1}{5}s^5-\frac{1}{7}s^7+.......(|S|<1)$$
$$\arctan(s)\cong\frac{\pi}{2}-[\frac{1}{s}-\frac{1}{3s^3}+\frac{1}{5s^5}-\frac{1}{7}+.......(S>1)$$

This expansion need to be inserted in 3 and 4 to yield polynomial expressions for F. However this expansion compromises accuracy for complexity. The above method of Scan Conversion has the following disadvantages.

- First, the implementation of forward differencing for high degree polynomials are cumbersome and error- prone.
- The algorithm does not converge even when if we use many terms and the error accumulates so quickly that the overall scan conversion fails.

Due to the reasons mentioned above it is concluded that Scan Conversion of spirals cannot be done using the incremental algorithms.

3. Piecewise Interpolation

A piecewise approximation is usually built by exactly computing several points of the curve and then interpolating these points with primitives that are easy to scan-convert. The simplest primitive for Interpolation is a line segment; more complex alternatives are circular arcs, then elliptical arcs, general conics, and so on. However it is found that the interpolation of Piecewise linear and Piecewise circular arcs is difficult due to the following reasons

1. The approximation quality of linear pieces depends on the length of the line segments. Using a fixed set of angles, the length of a line segment (and, thus, the error) is unbounded.
2. Circular arcs could not have the origin as their center and it is not clear what other point could be used as center.

While linear or circular pieces (used in a naive way) are inadequate, elliptical arcs are an interesting alternative. The following algorithm provides such an approximation:

- Define a quarter ellipse having $d=m*\pi/2$ as the distance between the semi major and the semi minor axis.
- Draw the ellipse from $(A_0+b,0)$ to $(0,A_0+d+b)$ as shown in Figure.

Subsequent elliptical quarters increase every semi-axis' value by *d* and the center remains constant in the origin. Thus, all elliptical arcs are joined yielding a continuous increment of the radius in the resulting spiral (see Figure 4).

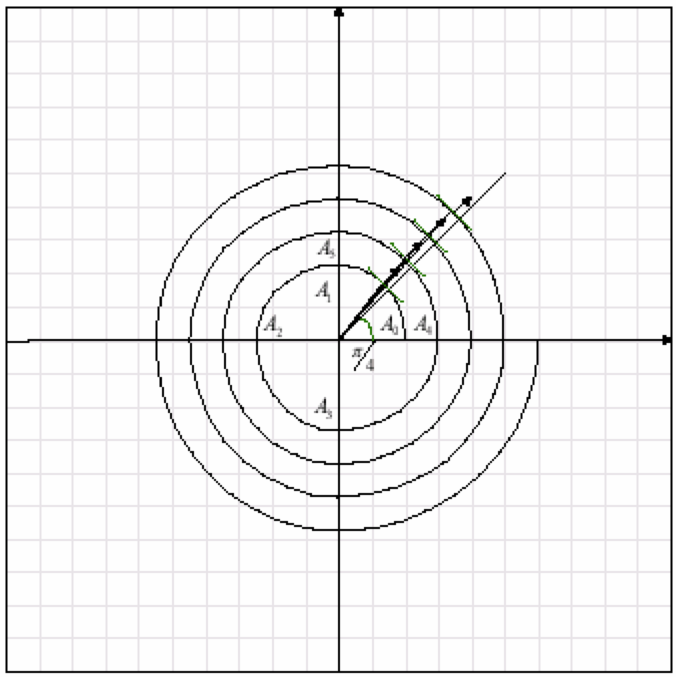


Figure 4: Quarter Ellipse spiral, iterative construction

3.1 Analysis

- In each of the four quadrants quarter-ellipses have a constant distance of d 4 between each period, as in Archimedes' spiral.
- A general major semi-axis has the value of d $A_n=A_{n-1}+d$ therefore, elliptical arcs meet with the same semi-axis length and the resulting curve is C^1 .

3.2 Demerits

The main disadvantage of this technique is that the Approximation does not converge against Archimedes spiral. The first quarter of the approximation shows most notably the difference between the two semi axes. If m is large, the difference between a spiral quarter and an ellipse is significant.

In the next section It is shown this problem of spiral Convergence is solved using the Quarter Circle Algorithm

4. Quarter Circle Algorithm

This algorithm is inspired by ancient technique by Fibonacci and Padovan. This technique assembles a set of triangles and use circular arcs. See Figure 5 .

Consider the equation of an Archimedian spiral. $r(\varphi)=m.\varphi+b$. The Quarter circle algorithm is as follows.

- Let $d=m*\pi/4$ define four points $c_i=(\pm d, \pm d)$ in the four quadrants of the coordinate system.
- Draw a circular arc with center $c_1=(-d,d)$ and radius $A_1=2d+b$ from $(d+b,d)$ to $(-d,3d+b)$.
- Starting from the last point of the previous quarter circles are drawn increasing radius by 2d in each step.
- The expression for the quarter circle radii is $A_{n+1}=A_n+2d=A_1+2d.n$ with $A_1=2d+b$ and $n\in N^+$.
- Relationship between the radius in each period of the spiral is $A_{q+4i}=A_q+8i.d$ where i is the period number and $q\in[1,2,3,4]$ (Quadrant information**).**

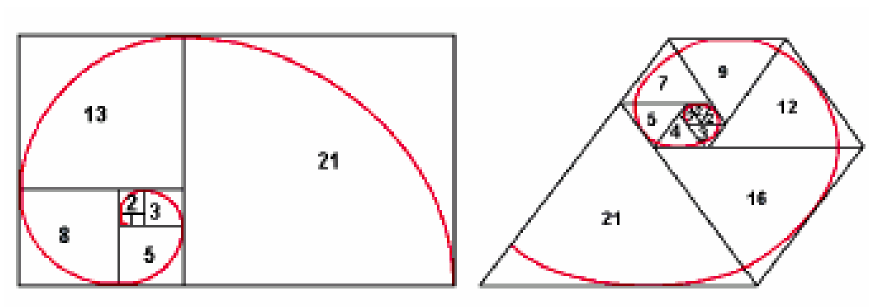


Figure 5: Fibonacci and Padovan spiral.

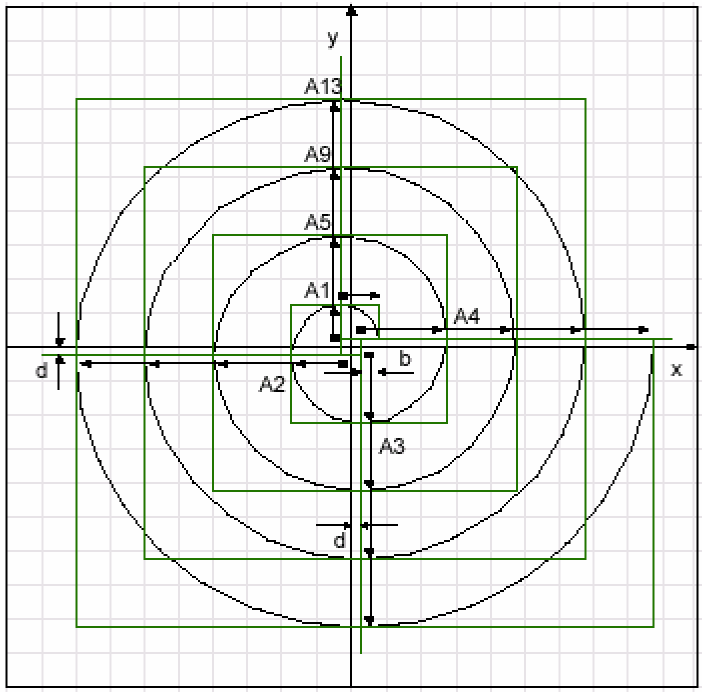


Figure 6: Illustration of the piecewise circular approximation to a spiral

4.1 Analysis

If *d* and *b* are chosen to be integer, the algorithm can be implemented using only integer addition. The circular arcs require (after set up) only two additions, and the step from each circular arc to the next requires another two additions.

The following tables analyzes the increase in radius from one quadrant to the next.. Here the steps are analyzed for each of the four quadrants and the Quarter points represent the center of the circle in each of the four quadrants. From Table 2 it can be easily noticed that each quarter starting point is automatically given by the quarter end point of the previous step.

Table 1: Initializing table for the algorithm's step

Step	Quarter Circle Radius	Quarter center points
$(1+4i)^0$	**2*d+b+8i*d**	**(-d,d)**
$(2+4i)^0$	**4*d+b+8i*d**	**(-d,-d)**
$(3+4i)^0$	**6*d+b+8i*d**	**(d,-d)**
$(4+4i)^0$	**8*d+b+8i*d**	**(d,d)**

Table 2: Initializing table for the algorithm's step

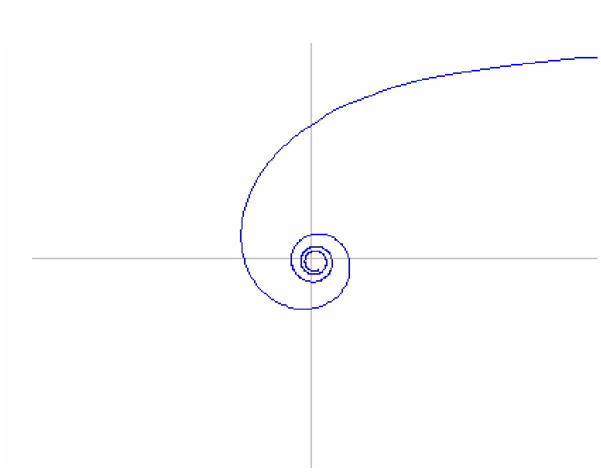
Step	Quarter Starting Point	Quarter end points
$(1+4i)^0$	(d+b+8i*d,d)	(-d,3*d+b+8i*d)
$(2+4i)^0$	(-d,3*d+b+8i*d)	(-5*d-b-8i*d,-d)
$(3+4i)^0$	(-5*d-b-8i*d,-d)	(d,-7*d-b-8i*d)
$(4+4i)^0$	(d,-7*d-b-8i*d)	(9*d+b-8i*d,d)

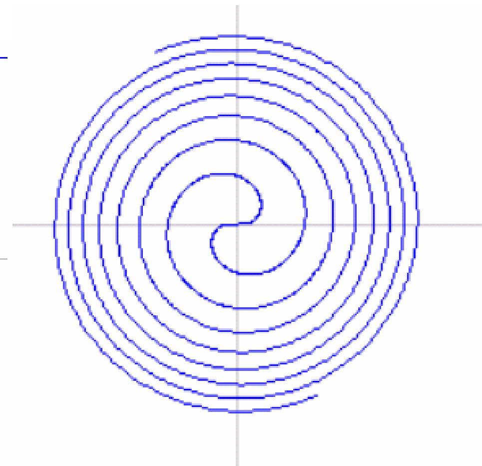
The algorithm is easy to implement and the error bound of the algorithm is given by $e(i) = 1 - \sqrt{1 - \frac{1}{64i^2}}$ Where i is the period number. As the Period Number (Winding Number of the spiral) Increases the error bound decreases.

circular segments meet with horizontal or vertical tangent. Thus, the resulting approximation is G^1 . Note that it is not C^1 as the meeting circular arcs have different radii.

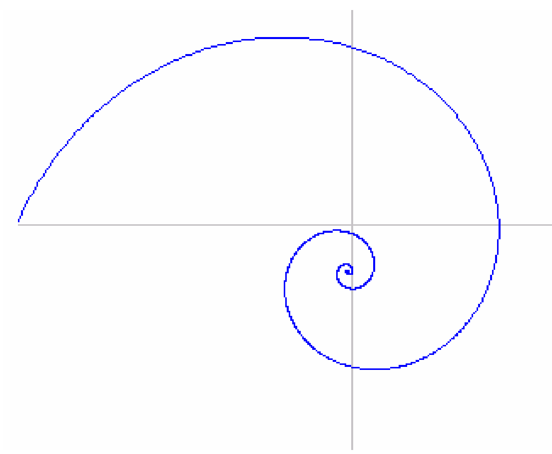
5. Implementation and Results

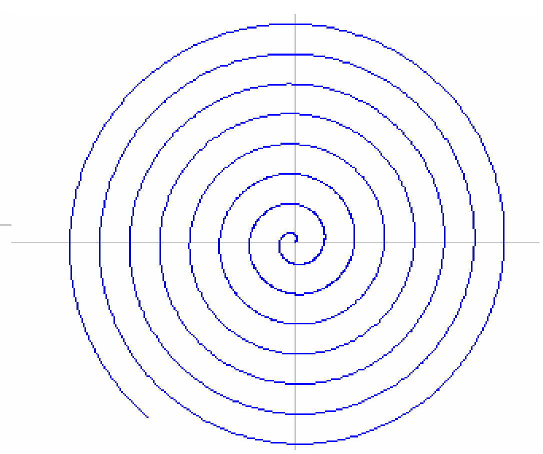
The system is implemented in Java Programming Languages and Java Applets is used to create the Graphical User Interface. The Archimedian spiral is created The spiral is generated using the Circular Approximation Technique and the Technique is generalized to create other types of spirals . The other types of spirals created are the Equiangular spiral, Fermat spiral and Hyperbolic spiral. Figure 7 below shows the different types of spirals generated using the above technique . The system is still under implementation with some developments to indicate how the spiral curve change their shapes when the radius and the angle of curvature is increased or decreased.





(a) Hyperbolic Spiral **(b) Fermat Spiral**





(c)Equiangular Spiral **(d) Spiral of Archimedes**

Figure 7: The Different Types of Spirals generated using the Circular Approximation Technique

6. Conclusion:

Methods to render Archimedes spiral have been analyzed and discussed. While it is proved that scan conversion techniques to generate spirals and other curved primitives are not efficient Approximation techniques provide a better alternative. The quarter ellipse algorithm produces a C^1 curve but the resulting approximation has systematic error. The quarter circle algorithm is simple in implementation and execution; the relative error of the resulting curve diminishes as the winding number increases. While the Elliptical approximation technique fails as the difference between major and minor axis increases , spiral generation using the Circular approximation technique tend to be more robust and the relative error decreases as the winding number increases. The technique can easily be generalized to generate different types of spirals and this has been proved by generating four different types of spirals. The system is still under development to experiment the change in shape of the spiral as the radius and angle of curvature changes.

References:

[1] F.Taponecco and M.Alexa, (2002), Piecewise circular approximation of spirals and polar polynomials.

[2] F.Taponecco and M.Alexa, (2002), Scan converting spirals, Proc. of WSCG 2002, Plzen, Czech. Republic.

[3] D.Lyon, show.docjava.com:8086/book/cgij/backup/ch27.pdf

[4] J.R. Van Aken, M. Novak,1985, *Curve Drawing Algorithms for Raster Displays, ACM Trans. On Graphics, 4,147-169.*

[5] T.Rieger, F.Taponecco, Interactive Information visualization of Entity-Relationship Data, Proc. of WSCG 2002, Plzen, Czech. Republic

[6] A.S.Cook. 1972, The curves of life: Being an account of Spiral formations and their application to growth in Nature, to Science and to Art: with special Reference to the Manuscripts of Leonardo da Vinci. Newyork: Dover, 1979.