BRAC UNIVERSITY

Inspiring Excellence

Department of Computer Science and Engineering

Course Code: CSE 423	Credits: 1.5
Course Name: Computer Graphics	Semester: Fall 2020

Lab 02

Digital Differential Analyzer (DDA) Algorithm

I. Topic Overview:

In any 2-Dimensional plane if we connect two points (x0, y0) and (x1, y1), we get a line segment. But in the case of computer graphics we can not directly join any two coordinate points, for that we should calculate intermediate point's coordinate and put a pixel for each intermediate point. Digital Differential Analyzer or DDA is a simple line drawing algorithm that enables us to do it.

II. Lesson Fit:

a. JOGL is a prerequisite to this lab which was taught in the previous lab.
b. We also learned how to draw line using dots in JOGL.

III. Learning Outcome:

After this lecture, the students will be able to:

a. Understand lines and its characteristics.
b. DDA Algorithm.
c. Implementation of the DDA Algorithm using JOGL.

IV. Anticipated Challenges and Possible Solutions

a. Students might confuse the proper use of slope.

Solutions:

1. Study the use of slope if forgotten.

V. Acceptance and Evaluation

Students will start implementing the algorithm after we finish our lecture. If a student fails to complete the implementation he/she will have to complete by that night and show it to the class teacher on his/her consultation period. If a student also fails to do that then instead of 10 we will evaluate the student on 7.

1. Lab evaluation marks: **out of 10**
2. Late Lab evaluation marks: **out of 7**

VI. Activity Details

a. **Hour: 1**

Discussion:

1. **Characteristics of a line:**
    a. The line should appear as a straight line and it should start and end accurately.
    b. The line should be displayed with constant brightness along its length independent of its length and orientation.
    c. The line should be drawn rapidly.
2. **Line Drawing:**

    Most line drawing algorithm uses incremental methods. In an incremental method a line starts with a starting point and then a fixed increment is added to the current point to get the next point on the line and continued till the end of the line.
3. **Digital Differential Analyzer (DDA):**

    The slope of a straight line can be given as,

$$m=\frac{\Delta y}{\Delta x}=\frac{y_2-y_1}{x_2-x_1} \quad \ldots\ldots\ldots 1$$

    The above differential equation can be used to obtain a rasterized straight line. For any given x interval Δx along a line, we can compute the corresponding y interval Δy from equation 1 as

$$\Delta y = \frac{y_2 - y_1}{x_2 - x_1} \Delta x \quad \text{..... 2}$$

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

$$\Delta x = \frac{x_2 - x_1}{y_2 - y_1} \Delta y$$

Once the intervals are known the values for next x and next y on the straight line can be obtained as follows

$$x_{i+1} = x_i + \Delta x = x_i + \frac{x_2 - x_1}{y_2 - y_1} \Delta y$$

and

$$y_{i+1} = y_i + \Delta y = y_i + \frac{y_2 - y_1}{x_2 - x_1} \Delta x$$

The equation 4 and 5 represents a recursion relation for successive values of x and y along the required line. Such a way rasterizing a line is called a digital differential analyzer. For simple DDA either Δx or Δy, whichever is larger, is chosen as one raster unit,

if $|\Delta x| \geq |\Delta y|$ then $\Delta x = 1$

else $\Delta y = 1$

With this simplification

if $\Delta x = 1$ then

we have $x_{i+1} = x_i + 1$

and $y_{i+1} = y_i + \frac{y_2 - y_1}{x_2 - x_1}$

if $\Delta y = 1$ then

we have $x_{i+1} = x_i + \frac{x_2 - x_1}{y_2 - y_1}$

and $y_{i+1} = y_i + 1$

4. DDA Line Drawing Algorithm

a. Read the line end points (x_1, y_1) and (x_1, y_1) such that they are not equal.

b. $\Delta x = |x_2 - x_1|$ And $\Delta y = |y_2 - y_1|$

c. if $(\Delta x \geq \Delta y)$ then
length = Δx
else
length = Δy

d. $\Delta x = (x_2 - x_1)/length$ and $\Delta y = (y_2 - y_1)/length$ [This makes either Δx or Δy equal to 1]

e. $x = x_1 + 0.5 * Sign(\Delta x)$ and $y = y_1 + 0.5 * Sign(\Delta y)$ [Here, Sign function makes the algorithm work in all quadrants. It returns -1, 0 , 1 depending on whether its argument is < 0,= 0, >0 respectively. The factor 0.5 makes it possible to round the values in the integer function rather than truncating them.]

f. Setting i to 0
While $(i \leq length)$
{
Plot(Integer (x),Integer (y))
$x = x + \Delta x$
$y = y + \Delta y$
$i = i + 1$
}

g. STOP

b. **Hour: 2 and 3**

Discussion:

Students will implement the DDA algorithm that was discussed in the first hour of this class.

VII. Home tasks

a. Using the implemented DDA Algorithm draw the last 2 digits of your student ID. For example, if your student ID is 17123456, then you should draw '56' on your output screen using the DDA Algorithm.

Lab Activity List

Task 1

Implement the DDA Algorithm using JOGL.

Reference

1. **Computer Graphics** by A.P.Godse, D.A.Godse.