

Computer Graphics Lab Assignment – Readme file

Introduction:

This assignment based on Parametric Surfaces was very interesting and challenging. Firstly the control points were read from the given file. After loading the points, a polyhedron cage was made by using triangles and lines separately. 'SnLines' class was used to draw lines between defined coordinates. Next step was to make a teapot by using the control points and coefficients of the Bezier equations.

Normal vectors were successfully generated each vertex. Firstly, separate vectors were generated on each point in the U and V direction. After getting values of these 2 vectors, cross product was performed between them to get a new Vector for that specific point.

In order to meet requirement 2 of the assignment, Shadowing is done. The vector for light source is kept unchanged. A triangle is being projected onto the XY plane using the light vector.

Functionalities in the project:

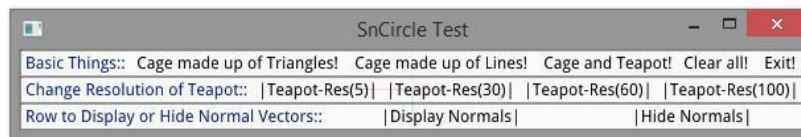


Fig 1: Controls for User

- First Row: Display Cage made by triangles and lines. Display Cage with teapot.
- Second Row: These buttons allow user to change the resolution of the teapot.
- Third row: The button in this row can be used to display or not the normal vectors.

Machine Specification on which this assignment was made and code testing was done:

Processor	Intel i3 - 3217U CPU @ 1.80GHz (8GB RAM)
Graphic card details	Internal Intel® HD Graphics 4000

Project Behavior on different machines:

Processor	Number of Triangles and Lines	Creation time (sec)
Intel i3	9,40,816	1.384193
Intel i5	9,40,816	1.382199
Intel i7	9,40,816	1.374656

Code Structure:

Lines	Description
0 – 13	Header files
14-22	MyViewer Constructor
23-67	add_ui function
68– 81	handle_keyboard function
82 - 102	uievent()
103 - 108	all() function
109 - 112	Clearall()
113 - 168	CageLines()
169– 220	CageTriangles()
221– 372	Teapot()
373 - 461	TeapotNormals()
462– 543	NormalFunction()
544 - 573	Shadow()

Time evaluation of the project:

Functions	Resolution	Number of Triangle's/Lines	Creation time (Sec)
add_ui()	-	7040 Lines, 954108 Triangles	1.38419
Polyhedron Cage (Triangles)	-	288 Triangles	0.011214
Polyhedron Cage (Lines)	-	768 Lines	0.001984
Teapot	5	6,400 Triangles	0.139175
Teapot	30	57,600 Triangles	0.145342
Teapot	60	2,30,400 Triangles	0.149654
Teapot	100	6,40,000 Triangles	0.152366
Teapot with Shadow	15	19820 Triangles	1.358789
Teapot with Normal Vectors	15	6272 Lines	0.048608

By carefully examining the above numbers, it can be concluded that Lines take less time than triangles for creation of any primitive. Second Conclusion is that as the resolution of the teapot increases the time for creation of the model also increases. The time function has been used to get the above values. The code has not been deleted from the program. It has been commented.

Result:

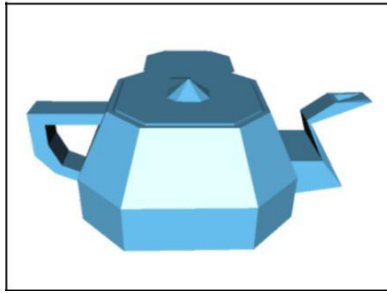


Fig 2: Cage with Triangles

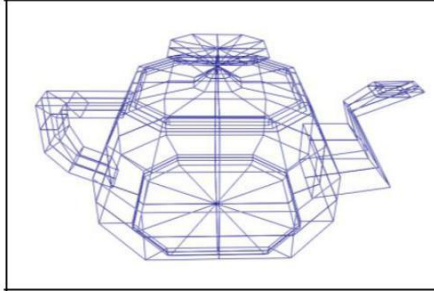


Fig 3: Cage with Lines

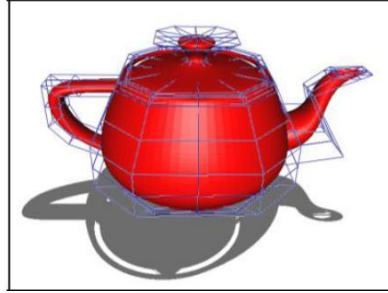


Fig 4: Cage with Teapot and shadow.

These figures show the illustrations of the project. Figure 2 is made when the triangles are been constructed between the control points. Figure 3 shows up a cage that is made with lines between respective control points. In Figure 4, Teapot is being generated with its shadow being projected on the XY plane.



Fig 5: Teapot with Low resolution

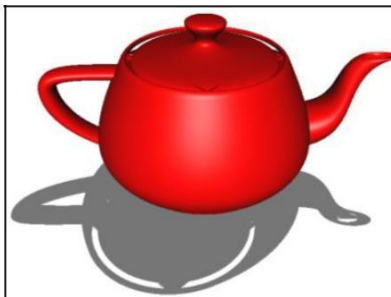


Fig 6: Teapot with High resolution

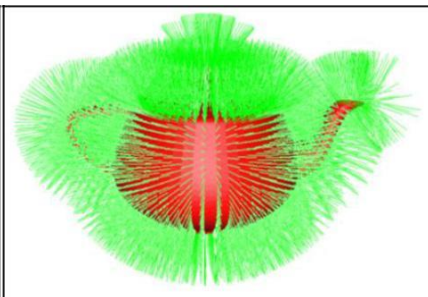


Fig 7: Teapot with correct Normal Vectors

The resolution of the Teapot can be changed by adjusting the number of discretization. For example in Figure 5, each patch of the teapot is being divided in 5 rows and 5 column i.e. there will be 25 vertices in each patch according to which new Bezier coordinates will be generated. As the number of rows and columns in each patch increase, the resolution of the teapot will accordingly increase. In Figure 6, the teapot is very smooth and is of high resolution. Here, each patch is being divided in 100 rows and 100 columns. Figure 7 is to show the normal vectors that are being generated in the project.