IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 2, April-May, 2013
**ISSN: 2320 - 8791**
**www.ijreat.org**

# Optimization of Mesh Reconstruction using Delaunay and Ball Pivoting Algorithm

**Bharti Sood[1], Kamaldeep Kaur[2]**
**[1, 2] Department of Electronics & Communication Engineering, BBSBEC, Fatehgarh Sahib, India**

**Abstract**
It is very important to approximate a shape from the coordinates of a given set of points for surface reconstruction. The specific shape deals with curves in two dimensions and surfaces in three dimensions. In this paper we have presented a hybrid model of the ball pivoting and Delaunay algorithm for mesh reconstruction from 3D point cloud. The ball of a diameter d pivots around an edge keeping in contact with the edge's endpoints until it touches another point, forming another triangle and the diameter of the ball is increased according to the requirement, when the pivoting ball is not able to reach to some of the sample points, then the pivoting ball is rotated in 3D. This improvement will lead to more effective representation of 3D image. The hybrid model of Ball Pivoting and Delaunay Algorithm is efficient in terms of execution time and storage requirements, number of triangles created, and time spend for computing triangles.
***Keywords:*** *Ball Pivoting Algorithm, Delaunay Triangulation Algorithm, Surface Reconstruction, 3D point cloud.*

## 1. Introduction

Surface reconstruction from range data involves four major steps. First step is data acquisition of surface samples with a laser range scanner. Second is data registration where alignment of several scans is done into a single co-ordinate system. Third step is data integration, the separate registered range maps are integrated into a single surface representation (often a polygon mesh). Fourth step is optimization, the single surface representation can be fitted better to the data, it may be further simplify, or the representation can be converted to some other format (e.g., smooth surfaces).

The Ball Pivoting Algorithm makes two mild assumptions about the samples that the samples are distributed over the entire surface. The method is conceptually simple. Starting with a seed triangle, the ball of a radius p pivots around an edge keeping in contact with the edge's endpoints until it touches another point, forming another triangle, which is added to the mesh, and the algorithm considers a new boundary edge for pivoting [9].

The output mesh is a manifold subset of an alpha-shape of the point set [12]. Some of the nice properties of alpha-shapes can also be proved for our reconstruction. The
\
hybrid model of Ball Pivoting and Delaunay Algorithm is efficient in terms of execution time and storage requirements, number of triangles created, and computation of CPU time for creating triangles. It exhibited linear time performance on datasets consisting of millions of input samples. It has been implemented in a form that does not require all of the input data to be loaded into memory simultaneously. The resulting triangular mesh is incrementally saved to external storage during its computation, and does not use any additional memory. This paper focuses on the hybrid model of the ball pivoting and Delaunay algorithm for mesh reconstruction from 3D point cloud. In this paper we have presented a new method for finding a triangle mesh from unorganized set of points and improved the results in terms of execution time and storage requirements, number of triangles created, and time spend for computing triangles. The paper is structured as follows: In section 2 we discuss the triangulation algorithms for surface reconstruction. Section 2 also describes the problem and related work. Section 3 describes the hybrid approach of Ball pivot algorithm and Delaunay algorithm in detail. Section 4 describes the flow of our hybrid approach of algorithms. We present results and discussions in section 5, and conclusion is in section 6.

## 2. Triangulation Algorithms for Surface Reconstruction

The algorithms for surfaces reconstruction from large data have been proposed in the past. Delaunay triangulations are important geometric data structures that are built on the notion of nearness. Many differential properties of curves and surfaces are defined on local neighborhoods. Delaunay triangulations provide a tool to approximate these neighborhoods in the discrete domain. They are defined for a point set in any Euclidean space. These are defined in two dimensions and mention the extensions to three dimensions since the curve and surface reconstruction algorithms are concerned with these two Euclidean spaces. The Delaunay triangulation is formed by joining all the circum circles.

IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 2, April-May, 2013
**ISSN: 2320 - 8791**
**www.ijreat.org**

The hardware projection is used to compute a triangulation from each range scan and then stitch them together. This gives a very fast reconstruction for each range scan. It faces difficulty if the projection of a scan self-overlaps. The consistent stitching of adjacent scans possesses problems. The current methods for stitching are based on stimulating interest as a means of furthering investigation and often create artifacts in the surface. The ball pivoting algorithm of was used to handle millions of sample points. This algorithm builds the surface incrementally by rolling a ball over the sample points. But It requires that the sample points be equally dense everywhere and each sample point have a surface normal.

The Ball-Pivoting Algorithm computes a triangle mesh interpolating a given point cloud. Typically the points are surface samples acquired with multiple range scans of an object. The principle of the BPA is very simple: Three points form a triangle if a ball of a user-specified radius p touches them without containing any other point. Starting with a seed triangle, the ball pivots around an edge that means it revolves around the edge while keeping in contact with the edge's endpoints until it touches another point, forming another triangle. The process continues until all reachable edges have been tried, and then starts from another seed triangle, until all points have been considered [9].

The problem is formulated that when the sampling density is too low, means point is located at a larger distance than radius r then some of the edges will not be created, leaving holes and when the curvature of the manifold is larger than 1/p, some of the sample points will not be reached by the pivoting ball, and features will be missed. Standord vrip (Volumetric Range Image Processing) program was used to connect the points within each individual range data scan to provide estimates of surface normals. The plane carvers, large planes of triangles were also removed by using hole-filling algorithms.

Our approach fills the holes by increasing the diameter of the ball according to the requirement and the ball pivots around an edge keeping in contact with the edge's endpoints until it touches another point, forming another triangle. In our method Delaunay algorithm is also considered to resolve the problems occurred due to low or high density points. When some of the sample points will not be reached by the pivoting ball, then the pivoting ball is rotated in 3D. The hybrid model of the ball pivot and Delaunay algorithm will lead to improve more effective representation of 3D image.The hybrid model of Ball Pivoting and Delaunay Algorithm is computing the number of triangles very efficiently. Our method is also computing the I/O time, Memory usage and CPU time. I/O time is the time to read and write the file. Memory usage is the total space occupied to store the file. CPU time is time taken for creating one triangle. In our paper these parameters are compared with the past results. Table 1 depicts the results carried out in the past research. We refer a review of research in the field [9].

## 3. The Ball Pivot & Delaunay Algorithm

The hybrid model of the Ball pivoting algorithm and Delaunay algorithm is implemented for mesh reconstruction from 3D point cloud. The Delaunay triangulation is formed by joining all the circum circles. It is assumed that the distance between the vertices and the points a, b, c in the plane is measured using the standard distance. For a matrix:

$$\begin{pmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ - & - & - \\ X_n & Y_n & Z_n \end{pmatrix}$$

The distance between the vertices of the matrix and the points a, b, c can be computed by the equation:

$$d_i = \left| \sqrt{(X_j - a_i)^2 + (Y_i - b_i)^2 + (Z_j - c_i)^2} \right|_{j=1}^{i=n} \quad (1)$$

Delaunay triangulations are data structures that are built on the notion of nearness. Many differential properties of curves and surfaces are defined on local neighbourhoods. The algorithm to approximate the neighbourhoods in discrete domain is:

```
{
    No =0
    for No<3
    Neg=Neg ∪ |d_i < r|_{i=0}^{i=n}
    No=0
    No=Σ_{i=1}^{len(Neg)} No + 1
    r=r+.1
    end
}
```

In this an approach is made to fill holes that the diameter of the ball is increased according to the requirement and

IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 2, April-May, 2013
**ISSN: 2320 - 8791**
**www.ijreat.org**

the ball pivots around an edge keeping in contact with the edge's endpoints until it touches another point, forming another triangle. In the second approach the Delaunay algorithm is followed, Delaunay triangulations are data structures that are built on the notion of nearness, so when some of the sample points will not be reached by the pivoting ball, then the pivoting ball is rotated in 3D as shown in Fig. 1.

## 4. Implementation

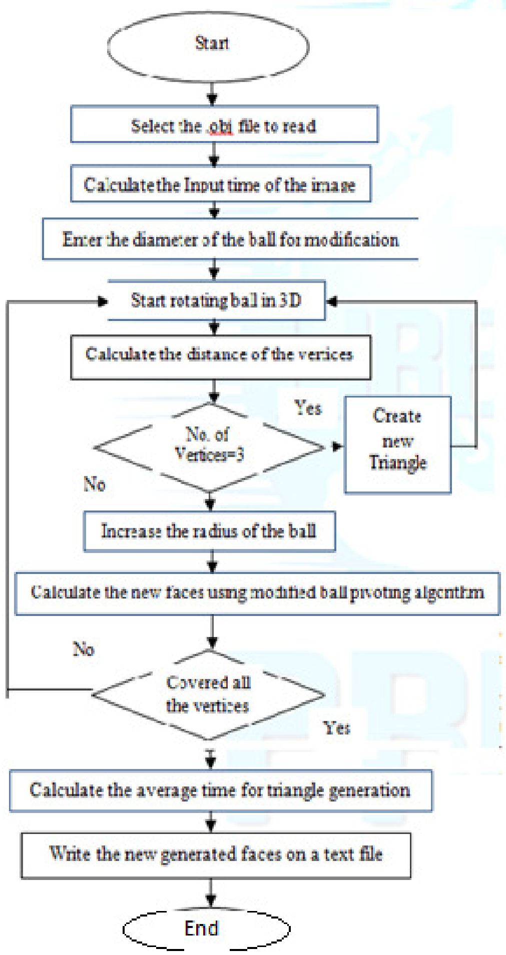


Fig.2. Flow chart of hybrid model of Ball Pivoting Algorithm and Delaunay algorithm

## 5. Results & Discussions

Our experiments for this paper were conducted on a 2.20 GHZ Core i3-2330M processor of Sony Vaio E series running windows 7. In our experiments we used several datasets like bunny, dragon, Buddha. The datasets are taken from the Stanford scanning database. In order to confirm the effectiveness of the research, we modified the ball pivot algorithm. The hybrid model of the Ball pivoting algorithm and Delaunay algorithm was implemented in MATLAB. The whole code is less than 150 lines.

In dataset Bunny the number of points i.e. vertices are 34835, in dataset Dragon the total vertices are 50000 and in dataset Buddha the number of vertices are 49990. d lists the diameter of the pivoting balls, in mm. The initial diameter of the pivoting ball for each dataset Bunny, Dragon, Buddha is fixed. The results are taken by considering a fixed initial value of diameter. With increase in diameter number of triangles will also increase. During simulation of the model the diameter of the ball will automatically increase according to the requirement. The total triangles are the number of triangles created by the hybrid approach of Ball Pivot and Delaunay algorithm. Memory usage is the maximum amount of memory used at any time during mesh generation, in MB. I/O Time is the time taken to read the input binary files and it also includes the time to write the output mesh, in seconds. CPU Time is the average time spent computing the triangulation, in seconds.

Fig. 3, Fig.4, Fig. 5 represents the mesh reconstruction of the datasets Bunny, Dragon, Buddha respectively by our method. Table 2 gives the results of calculation of number of triangles, memory usage, I/O time, CPU time in case for Bunny, Dragon and Buddha by hybrid approach of both the algorithms. The hybrid model of Ball Pivoting and Delaunay Algorithm is computing the number of triangles very efficiently. I/O time is the time to read and write the file. Memory usage is the total space occupied to store the file. CPU time is time taken for creating one triangle. In our paper these parameters are compared with the past results.

## 6. Conclusion

The hybrid model of Ball Pivoting and Delaunay Algorithm is efficient in terms of execution time and storage requirements, number of triangles created, and time spend for computing triangles. It works on datasets consisting of millions of input samples. It has been implemented in a form that does not require all of the input data to be loaded into memory simultaneously. The resulting triangle mesh is incrementally saved to external storage during its computation, and does not use any additional memory.

## References

[1] B. Curless and M. Levoy, 1996. "A volumetric method for building complex models from range images", Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, New Orleans, Louisiana USA, pp. 303 –312.
[2] Chuan-Chu Kuo and Hong-Tong Yau, 2005. "A Delaunay-Based Region-Growing Approach to Surface Reconstruction from Unorganized Points", Journal of Computer-Aided Design, Vol. 37, No. 8, pp. 825-835.
[3] Dominique Attali, 1997. "r- regular shape reconstruction from unorganized points", Proceedings of the thirteenth annual conference on Computational geometry, Nice, France pp. 248–253.
[4] Ernst Peter Mucke, 1993. "Shapes and Implementations in Three-Dimensional Geometry", PhD dissertation, University of Illinois, Urbana-Champaign.
[5] Eric Bittar , Nicolas Tsingos , Marie-paule Gascuel, 1995. "Automatic reconstruction from unstructured data: Combining a medial axis and implicit surfaces", Proceedings of Eurographics in Computer Graphics Forum, Vol. 14, No. 3, pp. 457–468.
[6] Esdras Medeiros, Luiz Velho and Helio Lopes, 2004. "Restricted BPA: Applying Ball-Pivoting on the Plane" Proceedings of 17th Brazilian conference Computer Graphics and Image Processing, Curitiba, PR, Brazil, pp. 372 - 379
[7] F. Bernardini, C. Bajaj, J. Chen, and D. Schilore, 1997. "Triangulation-based object reconstruction methods," Proceedings of the thirteenth conference on Computational Geometry, Nice, France, pp. 481-484.
[8] Fausto Bernardini, and Chandrajit L. Bajaj, 1997. "Sampling and reconstructing manifolds using alpha-shapes", Proceedings of 9th Canadian Conference Computational Geometry, Kingston, Ontario, Canada, pp.193-198.
[9] Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Claudio Silva and Gabriel Taubin, 1999. "The ball-pivoting algorithm for surface reconstruction", IEEE Journal of Transactions on Visualization and Computer Graphics, Vol. 5, No. 2, pp. 349-359.
[10] Gabriel Taubin, 1995. "A signal processing approach to fair surface design", Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, New York, USA, pp. 351–358.
[11] H. Hoppe, T.DePose, T.Duchamp, J.McDonald and W.Stuetzle, 1992. "Surface reconstruction from unorganized points", Proceedings of the 19th annual conference on Computer graphics and interactive techniques, New York, USA, pp. 71-78.
[12] H. Edelsbrunner and E. P. Mücke, 1994. "Three-dimensional alpha shapes", Journal of ACM Transactions on Graphics, Vol. 13, No.1, pp. 43–72.
[13] J. D. Boissonnat, 1984. "Geometric structures for three-dimensional shape representation", Journal of ACM Transactions on Graphics, vol. 3, pp. 266-286.
[14] J. D. Boissonnat and F. Cazals, 2000. "Smooth surface reconstruction via natural neighbor interpolation of distance functions", Proceedings of the sixteenth annual conference on Computational geometry, Hong Kong, pp. 223-232.
[15] Ji Ma, Hsi-Yung Feng, and Lihui Wang, 2010. "Delaunay-Based Triangular Surface Reconstruction from Points via Umbrella Facet Matching", Proceedings of IEEE Conference on Automation Science and Engineering, Toronto, Canada, pp. 580-585.
[16] L. Kobbelt and M. Botsch, 2000. "An interactive approach to point cloud triangulation", Journal of Computer Graphics Forum, Vol. 19, No. 3, pp. 479–487.
[17] N. Amenta, M. Bern and M. Kamvysselis, 1998. "A new Voronoi-based Surface Reconstruction Algorithm", Proceedings of 25th annual Conference on Computer Graphics and Interactive Techniques, Florida, USA, pp. 415–421.
[18] N. Amenta, S. Choi and R. Kolluri, 2001. "The power crust", Proceedings of the sixth ACM conference on Solid modeling and applications, Michigan, USA, pp. 249-260.
[19] Peer Stelldinger, 2008. "Topologically Correct Surface Reconstruction Using Alpha Shapes and Relations to Ball-Pivoting", Proceedings of 19th International Conference on Pattern Recognition, Tampa, Florida, USA, pp.1-4.
[20] Tamal K. Dey, Joachim Giesen and James Hudson, 2001. "Delaunay Based Shape Reconstruction from Large Data", Proceedings of the IEEE conference on parallel and large-data visualization and graphics, SanDiego,CA, USA, pp. 19-27.

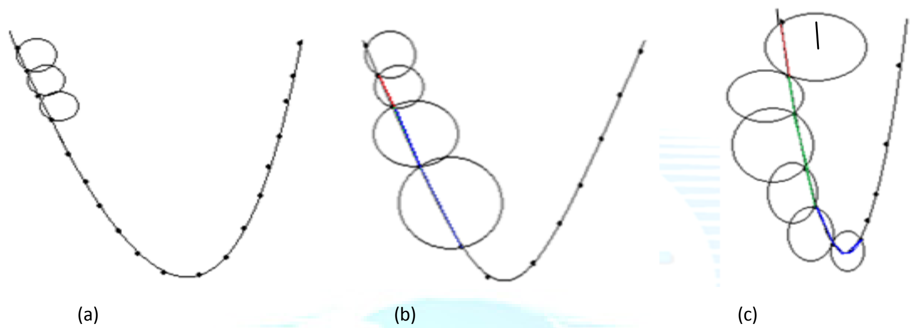


Fig. 1. The Ball Pivoting Algorithm in 3D (a) A circle of diameter d pivots from sample point to sample point, connecting them with edges. (b) When the sampling density is too low, the edges are created by increasing the diameter of the ball. (c) When the pivoting ball is not able to reach to the sample points, then ball starts rotating in other direction.

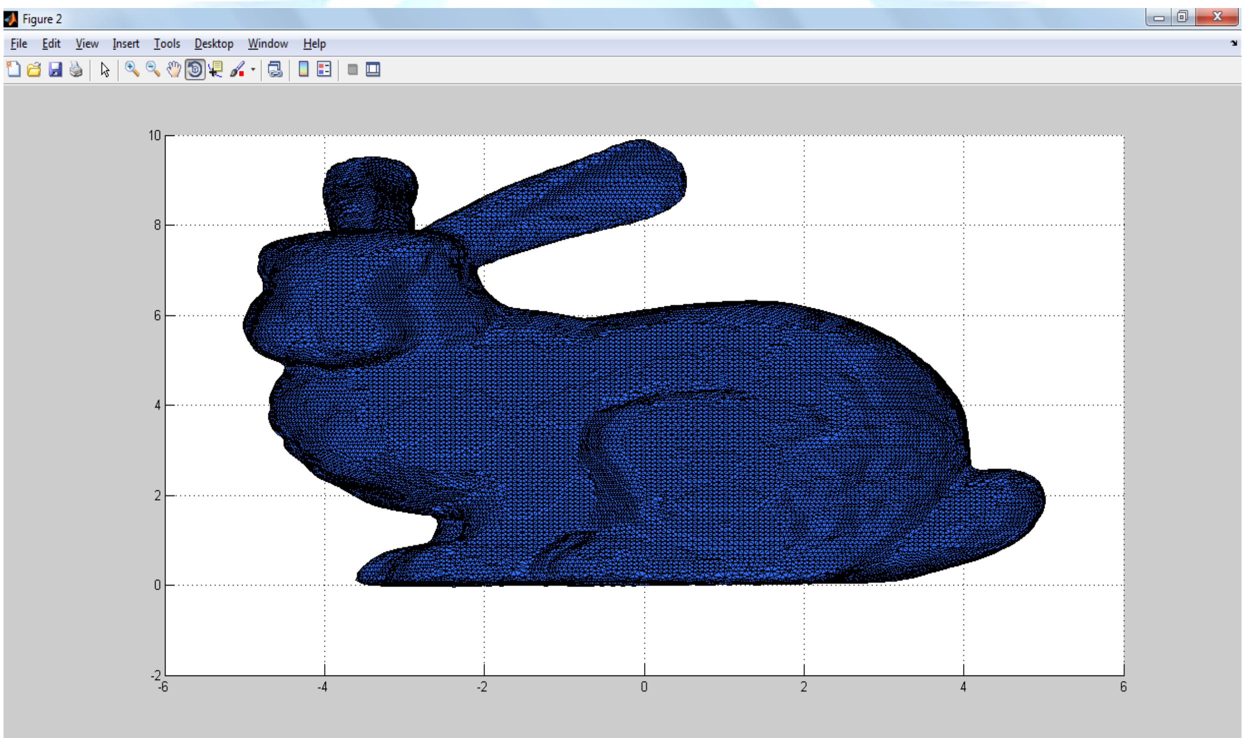


Fig. 3 Bunny

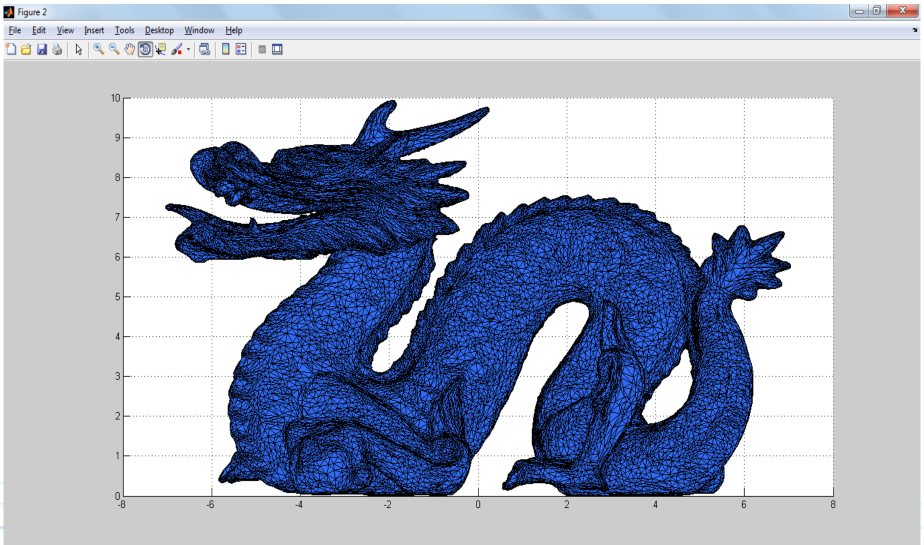

Fig. 4 Dragon

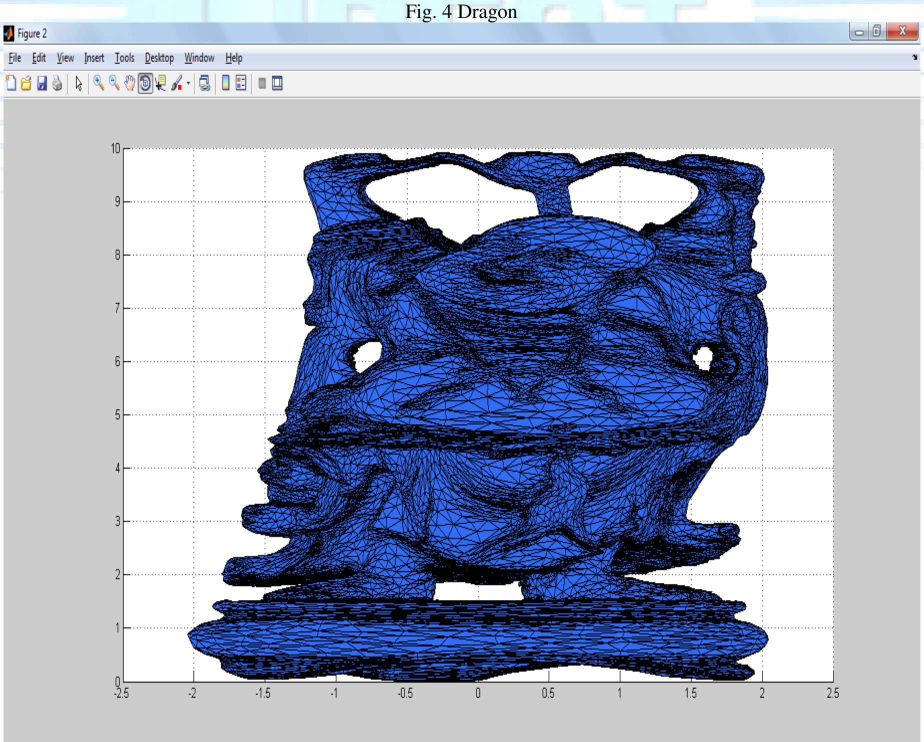

Fig. 5 Buddha

Table 1: Summary of Results of Ball Pivoting Algorithm

Table 1: Summary of Results of Ball Pivoting Algorithm

| *Dataset* | *Points* | *Scans* | *Radius (mm)* | *Slices* | *Triangles* | *Memory Usage* | *I/O Time* | *CPU Time* | *Ratio (Faces/Points)* |
|---|---|---|---|---|---|---|---|---|---|
| Bunny | 361K | 10 | 0.3,0.5,2 | - | 710K | 86MB | 4.5 secs. | 2.1 mins. | 1.97 |
| Dragon | 2.0M | 71 | 0.3,0.5,1 | - | 3.5M | 228MB | 22 secs. | 10.1 mins. | 1.75 |
| Buddha | 3.3M | 58 | 0.2,0.5,1 | - | 5.2M | 325MB | 48 secs. | 16.9 mins. | 1.58 |

Table 2: Summary of Results of hybrid Approach of Ball Pivoting and Delaunay Algorithm

| *Dataset* | *Points* | *Scans* | *Diameter (mm)* | *Slices* | *Triangles* | *Memory Usage* | *I/O Time (Sec)* | *CPU Time (Sec)* | *Ratio (Faces/Points)* |
|---|---|---|---|---|---|---|---|---|---|
| Bunny | 34835 | 1 | 0.048 | - | 0.1M | 9.34 | 0.0962 | 0.0209 | 3.86 |
| Dragon | 50000 | 1 | 0.032 | - | 2.3M | 98 | 1.03923 | 0.081 | 45.19 |
| Buddha | 49990 | 1 | 0.018 | - | 1.4M | 62.4 | 0.5142 | 0.0625 | 27.84 |