Asian Journal of Information Technology 5(6): 566-572, 2006

Enhanced Configuration of OBB Orientation Based on Heuristic Approach

A. Bade, [1]M. Zin A and [1]T. Sembok T.M.
FSKSM, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
[1]FTSM, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor Darul Ehsan, Malaysia

Abstract: Oriented Bounding Box (OBB) is a rectangular volume generated based on arbitrary orientation. The formation of this orientation is frequently obtained from the calculation of the covariance matrix. The previous work shown that OBB is one of the preferred bounding volumes due to its capability on enhancing the performance of object intersection algorithm especially in the field of virtual reality, animation, simulation, games and robotic. In this paper, we describe a variant approach to compute a tight OBB volume using heuristic. Therefore, to adjust the covariance matrix value, our method manipulates mean squared heuristically. The generated covariance matrix value is then handed over to eigenvector function as an input to derive a new OBB orientation. Our heuristic strategy has been implemented and we compare its performance in terms of volume ratio and collision contact between objects composed of hundreds polygons at interactive rates. From the conducted test, our conjecture on a slight adjustment to OBB orientation is probably potential to construct an optimal OBB volume compared to previous approach.

Key words: Bounding volumes, collision detection, virtual environments

INTRODUCTION

The purpose of this work was to develop an optimal Bounding Volume approach for virtual environment application (MELAKA: A historical walkthrough application) based on 3D objects generated from triangulated surface points. Bounding Volumes (BV) contribute much benefit to improve the performance of the intersection detection in large scaled environment. Although, current hardware to some extent offers faster intersection computation for complex data, the quest for producing a real time and realistic intersection detection in computer graphics applications has never come to an end.

OBB frequently used in computer graphics application as a mechanism to detect collision between objects. Former research about OBB typify that simpler shape, fast intersection testing and good tight-fitting volume type are the promising criterions that made OBB as effective BV to check collision rather then the bounded objects[1].

Constructing the smallest OBB from a set of triangulated surface points is a hard problem. Moreover, poorly aligned principal axes of OBB will give quite bad OBB fitting. As mentioned in[2,3], if our bounded object is vertices of a cube, the OBB orientation was quit unpleasant. This could be an isolated case due to the equal statistical spread of vertices in all directions. This condition suggests some heuristic to be applied such as using weighted spread of vertices of the triangles on convex hull facets[2]. This is due to the fact that convex hull is the smallest convex set containing all the points. Thus, the return is a good fit OBB, but $O(n \log n)$ time are required just to compute the convex hull.

The usage of principal component analysis to compute principal axes of OBB commonly correlates to maximal distribution of vertices, line segments or triangles. If the input data has outliers even in small quantity, the direction of maximal spread will change and enlarge the OBB. Furthermore, the area of empty corners also increases. As reported in[2], uniform distribution of vertices work convincing for simple models as well as the distribution of triangle facet where the tessellation not necessarily uniform. Both kind of distribution compute covariance matrix in O(n) time. Therefore, we conduct some experiments to probe the orientation of OBB on several objects formed by hundred to thousand triangulated surface points (Fig. 1). The result seems unpleasant since the generated orientation sometimes formed unfit OBB. As shown in Fig. 1, the former covariance approach creates unfit OBB for triangulated surface model such as Santa and Horse. Once again, we do check the possibilities to enhance the OBB volume by making minor alteration to the generation of OBB principal axes formed by hundred triangulated surface points. Thus, we apply several modifications heuristically and use the result as an input to the eigenvector function. The results seem effective and we able to produce fit OBB

Corresponding Author: A. Bade, FSKSM, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia

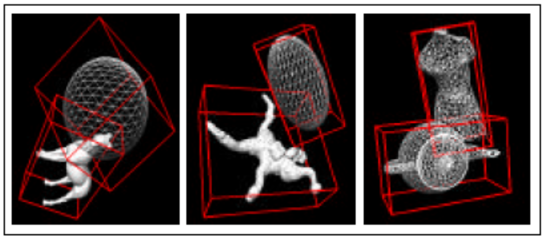


Fig. 1: Several objects enclosed by OBB

volume for the mentioned models. On the other hand, it is common to use heuristic that utilizes bounding box in the field of animation, rendering and modeling[4].

PREVIOUS WORK

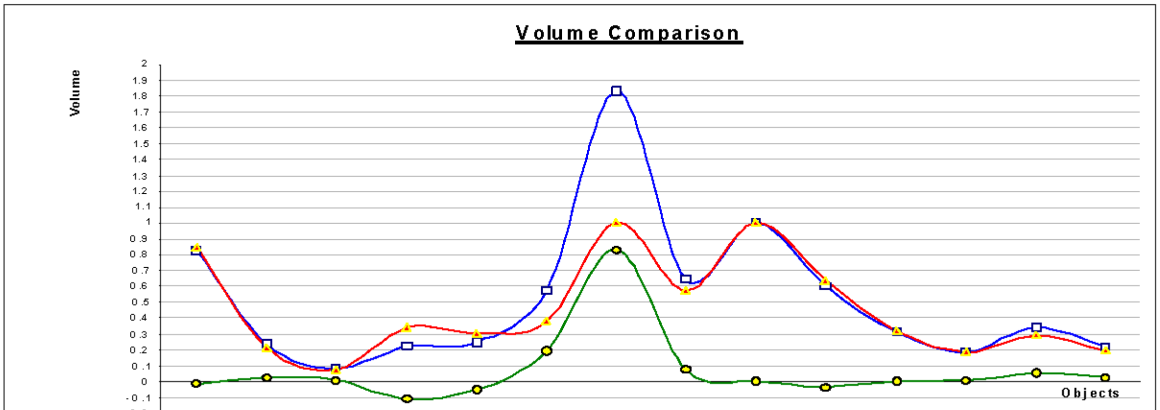
Most of the common approach in Collision Detection between n-body system is Bounding Volumes (BV). By surrounding complex objects with BV, a number of expensive primitive collision checking can be reduced. It is the reason why certain researcher placing BV as a coarse testing before implementing more accurate collision checking[5,1,6]. The intersection between pairs of object simply can be confirmed by checking their intersection of BV. Consequently, if BV intersects the bounded object has a potential to collide. Otherwise, the bounded object will not collide at all. Beside its greater function in collision detection, BV also play major role in other field in computer graphics. Ray Tracing, Viewing Frustum culling and Level of Detail are the range of examples that manipulates and utilizes BV for the sake of performance development[7-10]. On the other hand, first exploration of BV was in the field of Ray Tracing[11]. Probably, Moore and Wilhems[12] extend the usage of BV in Collision Detection.

Finding tight-fitting bounding box or OBB in particular is not straightforward. The previous solutions for the optimal OBB orientation basically depend on statistical spread of the polygons. In two-dimensional space, rotating calipers algorithm seems promising to calculate minimum area of rectangular[13,14]. The operation of the algorithm is $O(n)$ time. In order to compute tight-fitting rectangular, four lines are constructed. These lines are derived from four extreme points of the polygon. After that, they will be rotated until one of them coincides with an edge of the polygon. The area of new rectangle is then compared to the previous rectangle area to determine the smallest rectangle.

In three-dimensional space, the best possible known algorithm for finding an optimal OBB is by O'Rourke[15]. Unfortunately, this $O(n^3)$ algorithm (where n is the number of vertices in the model) is not easy to implement and inappropriate for real-time application[2,16]. Instead of this algorithm, a heuristic solution is cited in[17]. Once the OBB orientation not optimal or in other words loose-fitting, the solution is to realign the OBB to only one principal axes which is given the smallest volume. The other two axes are constructed from the projection of all vertices onto the perpendicular plane to the selected axis. Generally, there are three ways to elect the mentioned axes. First, OBB is constructed based on all principal axes. This is similar to the original algorithm of OBB. Second technique uses the largest eigenvalue as the principal axes. The other two axes are then created from the projection of all vertices in the model. The last approach manipulates the shortest principal component to be the main orientation. The remaining two axes are built in the same way of Max-principal-Component Box. The conducted experiment shows that, Min-Principal-Component Box performs better compared to two previous strategies. Meanwhile, Max-principal-Component Box expands the volume of OBB due to the longest possible direction.

Barequet[18] offers two other approximate strategies on the subject of computing tight-fitting Box. As stated in[18], the first algorithm is not easy to implement. It needs $O\ (n+1/\epsilon^{4.5})$ time in order to compute approximate minimum-volume bounding box of n points in space. The second algorithm presented by Barequet[18] is a coarse approximation, less efficient but easy to implement compared to the first algorithm. The running time of the alternative algorithm is $O(N\log N+N/\epsilon^3)$. At first, a tight-fitting Axis-aligned Bounding Box (AABB) is computed. Afterward, a farthest distant pair of points on two parallel sides of AABB is then picked to be the main OBB orientation. Likewise, the second axis is then computed from the minimum sides of the AABB. Finally, the last axis is assembled from cross product of the first two axes. From the given steps, the algorithm seems to be easier to implement. However, the usage of exhaustive grid based search research needs prior calculation quite a few properties such as convex hull and initial approximation of minimum bounding box[19]. Moreover, according to Ericson[20], using original algorithm cited in[2] will produce near optimal OBB compared to given algorithm.

Another way of algorithm to handle minimum tight-fitting OBB is presented in[19]. The algorithm manipulates Powell's quadratic convergent optimization method. The convergence rate to produce minimum OBB depends on initial conjecture for axis and angle of each direction[20]. Moreover, the authors state that the method is simple and does not need special prior calculation



	Bunny	Cow	Diamond	Dinosour	Elips	Horse	Kdops	Santa	Sphere	Skull	Teapot	Venus	Dragon	Hand
Previous OBB	0.82553	0.2372648	0.0833094	0.224259	0.2459294	0.5732436	1.8310892	0.6447863	0.9971081	0.6002006	0.3138178	0.1873224	0.3433903	0.2150378
Enhanced OBB	0.8432018	0.2131484	0.0776094	0.33733	0.3004963	0.3806897	1.0029962	0.57155	0.9966227	0.6383447	0.3161703	0.1827252	0.2915114	0.1956818
Diff	-0.0176718	0.0241164	0.0057	-0.1130711	-0.0545669	0.1925539	0.828093	0.0732363	0.0004853	-0.038141	-0.0023525	0.0045972	0.0518789	0.019356

Fig. 2: Comparison of bounding box volume for 14 objects based on previous and enhanced OBB orientation

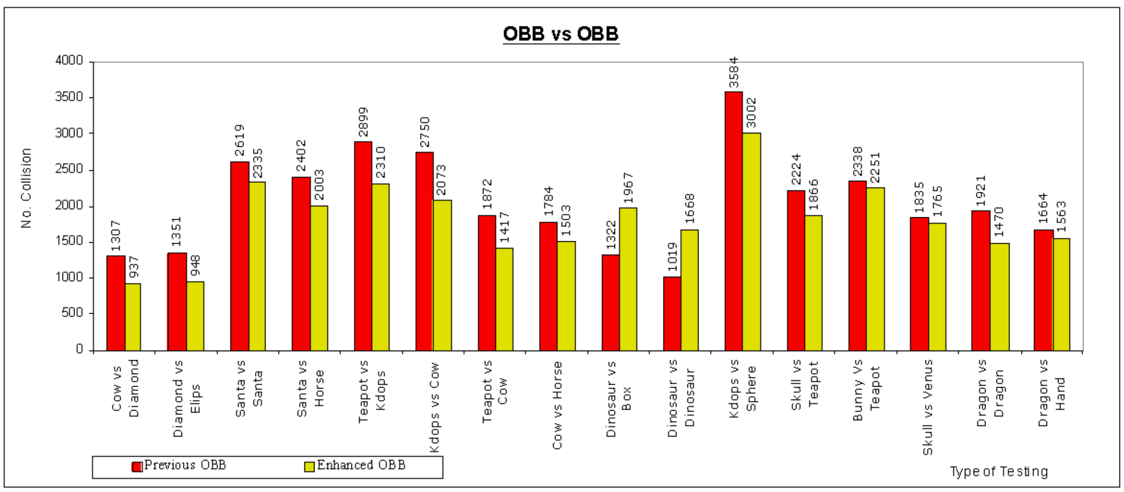
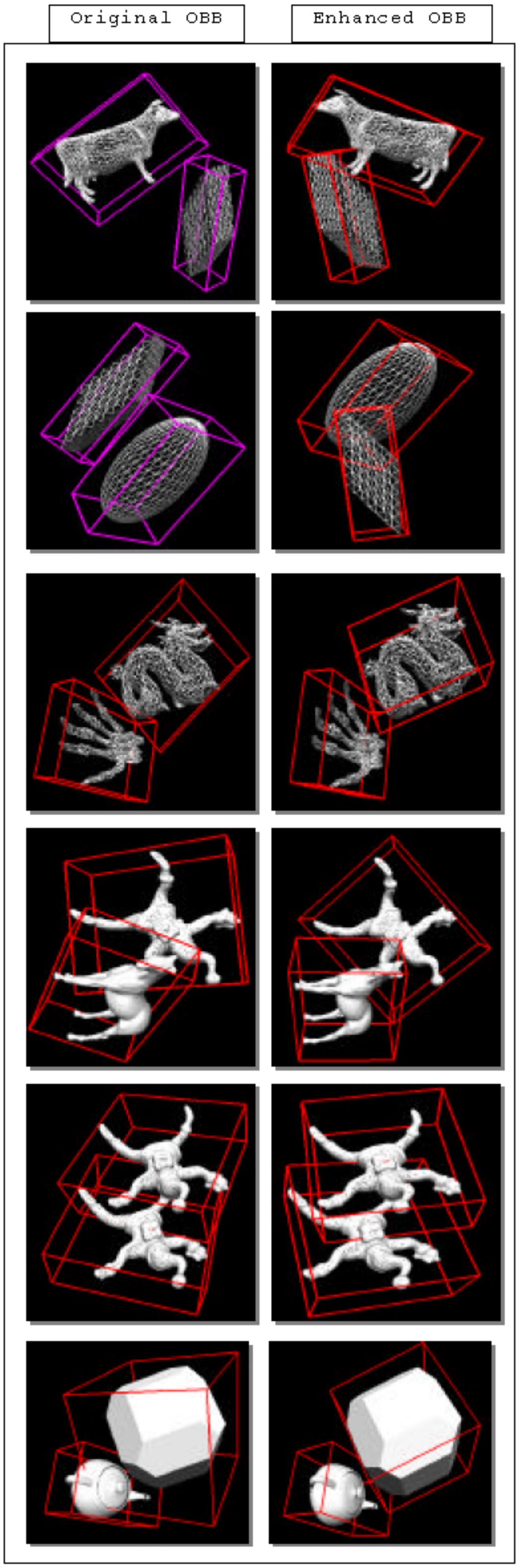


Fig. 3: A number of detected collisions between two complex objects enclosed using previous and enhance OBB

such convex hull or initial approximation of bounding box[19]. Brute-force approach is probably an extra algorithm to compute OBB fitting[20]. By using certain interval resolution, the rotation of space orientation is sampled and the best sample should be saved. Then, the OBB orientations can be improved by sampling them through subinterval resolution until no changes can be discovered.

CONSTRUCTION OF HEURISTIC OBB ORIENTATION

In this section, we describe the process of finding OBB orientation derived from heuristic approach. The main difference of our method from the methods so far is the generation of covariance matrix before computing the eigenvectors. To start with, all polygons composed from more then three edges should be triangulated. Using Principal Component Analysis method (PCA), prior calculation is required in order to build OBB orientation. Initially, we calculate *mean* and *covariance matrix* of the triangulated surface points models. Refer to the previous study conducted by Gottschalk *et al.*[1] and Gottschalk[2], there are three types of elements used to compute covariance matrix namely as vertex points, triangle facets and convex hull facets. In our case, we focus to 3D models formed by triangulated surface points. Now, denote x^i, y^i and z^i correspond to the points of the surfaces, μ, C_{ij} and n represent mean, 3 by 3 covariance matrix and number of triangles, respectively.



Appendix 1

The following steps explain the above process (Eq. 1 and 2) and the procedure to construct the principal axes of OBB and its bounding box.

$$\text{Let } \mu = \frac{1}{3n}\sum_{i=0}^{n} p_i + q_i + r_i \quad (1)$$

$$\text{Let } C_{ij} = \frac{1}{3n}\sum_{i=0}^{n} [p_i\, p_j + q_i\, q_j + r_i\, r_j] \quad (2)$$

- Replace each p, q and r with p-μ, q-μ and r-μ, respectively.
- Compute the eigenvectors of symmetric matrix C_{ij} and normalize the value before using them as a principal axes.
- The eigenvectors basically will be likely to align the principal axes with the geometry of object. For example we have a set of triangulated surfaces of an ellipse; an eigenvector of the covariance matrix will be aligned with the long axis of the ellipse (Fig. 1).
- In order to construct bounding box, project all vertices onto the principal axes and find the extreme vertices along each axis.

As mentioned in the previous section, the covariance matrixes really depend on the distribution of surface points and contribute to the nearly approximate approach. The sensitivity to the presence of the outliers sometimes returns very unpleasant OBB direction. The only guarantee that mentioned in[1,2] is that the construction of OBB orientation should be within the convex hull of the point clouds. Therefore based on heuristic conjecture, we enhance the construction procedure of the OBB orientation variant from covariance matrix proposed in[2,19].

It is known that covariance is a measure of how much the deviations of two variables linked. It can take any values range from negative infinity to positive infinity. Given two such random variables X_1 and X_2, where both of the variables are measured from the specific reference points namely as mean μ_1 and μ_2. Thus if $(X_2-\mu_2)$ and $(X_1-\mu_1)$are strongly (positively)matched, they indicate that both $X_2$and X_1 are increase or decrease concurrently together. If the value is negative, one of the variable increase and the other decrease. Once the value is zero, it indicates that both two variables are independent of each other. Meanwhile, a larger positivevalue of covariance is a good detector of a strong link between two variables. On the other hand, a smaller positive value is a sign that one of the variables has a fair chance to be closed to its mean when the other variable takes infinity values (positive or negative). It is obvious here that the above properties deduce that covariance is quite sensitive to the scales of the variables under investigation.

Heuristic approach: Our strategy basically manipulates mean squared instead of standard deviation to form a similar concept as Correlation Coefficient (Eq. 3). This is to ensure that a larger positive or negative value of covariance can be produced. Our conducted experiment shows that the eigenvector function tends to produce justify orientation of OBB if we use larger covariance value as an input. Total operation required to alter the covariance matrix is O(n) time. Meanwhile it is known that eigenvector is a non-null vector wherein if we scale it by some amount, the direction of the vector is unchanged rather than making it longer. In this case, it is pointless to scale the eigenvector values in order to obtain justify orientation. The details about eigenvector are beyond the scope of this study. Therefore, instead of using original covariance matrix explained in Eq. 2, we replace each of the covariance matrix elements with the version of covariance quoted in Eq. 3. The rest follows easily as mentioned in the given steps (Step 1-5).

$$\mathrm{Cov}(X_1, X_2) = \left[\frac{\sum_{i=1}^{n} (X_{1i} - \mu_1)(X_{2i} - \mu_2)}{\mu_1^2 . \mu_2^2} \right] \quad (3)$$

IMPLEMENTATION AND RESULT

To evaluate our approach, all experiments given in this section have been conducted on a PC with 2.8 GHz Pentium IV 1 GB main memory. In general, we performed two types of experiments. The first experiment measures bounding box volume derived from OBB orientation both previous and enhanced approach. The following experiment quantifies number of intersection recorded on a pair of objects within 5000 frames.

Bounding box volume: Prior to quantifying the bounding box size, Table 1 shows properties of each bounding box. The table also illustrates number of vertices and triangles for each bounded object. In Fig. 2, we calculate bounding box size for 14 objects namely as Dragon, Santa, Bunny, Diamond, Cow, Horse, kDop, Ellipse, Sphere, Dinosaur, Venus, Hand, Skull and Teapot (Appendix 1). The given graph (Fig. 2) shows that the enhanced OBB derived from slight modification to the OBB orientation tend to produce a justify volume compared to previous approach. This especially true to irregular objects that have gradual outliers such as Santa, Dragon, Horse, Cow and Hand. The volume ratio between two different approaches recorded as 11.35, 15.10, 33.59, 10.16 and 9.00 %, respectively. For nearly symmetry object such as kDop, our approach performs superior where the volume ratio is 45.22% smaller compared to previous approach. Likewise, the situation also happened for Diamond and Sphere, though the volume ratio is fairly small.

Intersection test between two objects: In order to quantify the needs of OBB to be tight-fitting, we have conducted a head-to-head collision testing between two rotating objects. Both objects have been enclosed by the OBB volume. Figure 3 demonstrated that our approach performs well for almost every head-to-head intersection testing. For enhanced approach, about 14 experiments out of 16 show number of detected collisions recorded not as much of the previous OBB orientation. This is about 3-30% collisions. Moreover, the number of detected collision for irregular objects bounded by enhanced OBB work as good as the symmetry objects. In our experiment, we sometimes tried to detect collision between the combination of irregular and symmetrical objects and the result is fairly well. Therefore, we probably can ascertain that the tightness OBB be likely to contribute more accurate intersection testing. The tightness OBB almost benefit to test nearly collide situation due to small number of empty corners. On the other hand, the worst case scenario for our approach is for the combination Dinosaur vs Dinosaur and Dinosaur vs Box. This is happened since our enhanced OBB orientation produced OBB volume larger then the previous approach.

CONCLUSION

In this study we have presented a heuristic method to alter the orientation of OBB principal axes by using heuristic. Our approach is made possible by slight modification to the scaled version of covariance. We have implemented our enhancement OBB orientation and the overall results seem fairly good. The results show the volume ratio is about 9-45% smaller compared to the previous OBB approach particularly for irregular type objects. On the other hand, the number of detected collision for head-to-head collision is about 3-30% fewer then the original OBB due to smaller volume of OBB.Currently, we are investigating the potential of our version OBB to be embedded to our historical walkthrough application. Moreover, we are also probing

Table1: The properties of objects and bounding box for enhanced and original OBB

			Height		Width		Length	
Objects	Vertices	Triangles	Previous OBB	Enhanced OBB	Previous OBB	Enhanced OBB	Previous OBB	Enhanced OBB
Diamond	320	588	0.222343	0.222336	0.356889	0.340775	1.051348	1.024323
Bunny	453	948	0.969898	1.005375	0.688816	0.798426	1.235673	1.050434
Cow	2903	5804	0.696178	0.628157	0.32861	0.325818	1.046015	1.041451
Dinosaurs	14050	28096	0.582603	0.967378	0.32861	0.408615	1.171376	0.853384
Dragon	1257	2730	0.792891	0.690179	0.43370	0.419751	0.998569	1.006241
Ellipse	482	992	0.448604	0.548176	0.548176	0.548176	1.000063	0.999998
Hand	1055	2130	0.294874	0.266625	0.691888	0.701635	1.054081	1.046016
Horse	48485	96966	0.791957	0.827638	0.362445	0.453129	1.99708	1.0151
Kdop	1639	2550	1.293402	1.001497	1.206684	1.00000	1.173228	1.001497
Santa	18946	37888	0.460625	0.485559	1.179239	0.999999	1.187043	1.177098
Sphere	467	992	0.998747	0.997976	0.998359	0.998644	1.0000	1.00000
Skull	30224	60339	0.819837	0.844627	0.69104	0.69104	1.059414	1.093672
Teapot	1177	2256	0.501743	0.505622	0.62212	0.62212	1.005361	1.005127
Venus	711	1418	0.384088	0.37479	0.486836	0.490418	1.001789	0.994132

the potential of our approach to be merged with another bounding volume based on discrete orientation polytopes concept.

ACKNOWLEDGEMENT

Thanks to Norhaida Suaib (GMM) for his promising proof and discussion. Best gratitude to Mohd Shahrizal Sunar (GMM), Azmir Muttalib (Virtual Reality Lab, MDC) and Syaiful Nizam Yahya (GMM) for their supportive ideas, critiques and motivations.

REFERENCES

1. Gottschalk, S., M.C. Lin and D. Manocha, 1996. OBB-Tree: A hierarchical structure for rapid interference detection. In computer graphics proceedings, annual conference series, ACM SIGGRAPH. New Orleans, Louisiana, pp: 171-180.
2. Gottschalk, S., 2000. Collision queries using oriented bounding box, Ph.D Thesis, Department of Computer Science, University of North Carolina, Chapel Hill.
3. Bergen, G.V.D., 2004. Collision Detection in Interactive 3D Environments, Morgan Kaufmann Publishers. USA.
4. Suri, S., P.M. Hubbard and J.F. Hughes, 1998. Analyzing Bounding Boxes for Object Intersection.
5. Bergen, G.V.D., 1998. Efficient Collision detection Of Complex Deformable Using AABB Trees. J. Graphics Tools, 2: 1-13.
6. Hubbard, P.M., 1995. Collision detection for interactive graphics application. IEEE Transactions on Visualization and Computer Graphics, 1: 218-230.
7. Müller, G., S. Schäfer and D. Fellner, 1999. A rapid clustering algorithm for efficient rendering. Eurographics Association, ISSN, pp: 1017-4656.
8. Larsson, T. and T. Akenine-Möller, 2003. Strategies for bounding volume hierarchy updates for ray tracing of deformable models. MRTC Report, ISSN, pp: 1404-3041.
9. Assarsson, U. and T. Möller, 2000. Optimized view frustum culling algorithms for bounding boxes. J. Graphics Tools 5: 9-22.
10. Moller, T. and E. Haines, 1999. Real-Time Rendering. A. K. Peters Ltd.
11. Kay, T.L. and J.T. Kajiya, 1986. Ray tracing complex scenes. Computer Graphics (SIGGRAPH '86 Proceedings, 20: 269-278.
12. Moore, M.P. and J. Wilhemls, 1988. Collision detection and response for computer animation. SIGGRAPH '88 Proceeding, 22: 289-298.
13. Toussaint, G., 1983. Solving geometric problems with the rotating calipers, Proceedings of Ieee Melecon Athens, Greece.
14. Freeman, H. and R. Shapira, 1975. Determining the minimum-area encasing rectangle for arbitrary closed curve. Commun. ACM, 18: 409-413.
15. O'rourke, J., 1998. Finding minimal enclosing boxes. Intl. J. Computer and Information Sci., 14: 183-199.
16. Ericson, C., 2005. Real Time Collision Detection, Morgan Kaufmann Publishers. USA.
17. Barequet, G., C. Bernard, G. Leonied, M. Joseph and T. Ayellet, 1996. Boxtree: A hierarchical Representation for Surfaces in 3D. Computer Graphics Forum, 15: 387-396.
18. Barequet, G. and Sariel Har-peled, 1999. Efficiently approximating the minimum-volume bounding box o fa point set in three dimensions, Proceeding of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp: 82-91.

19. Lahanas, M., K. Thorsten, N. Milickovic, K. Karouzakis, D. Baltas and N. Zamboglou, 2000. Optimized bounding boxes for three-dimensional treatment planning in brachytherapy. Medical Physics, 27: 2333-2342.
20. Eberly, D., 1999. Dynamic collision detection using oriented bounding boxes. Technical Report, Magic Software.