International Journal Of Interactive Digital Media, Vol. 1(3), ISSN 2289-4098, e-ISSN 2289-4101
http://www.ijidm.org

A Framework on Hierarchical Self-Collision Detection for Multiresolution

Siti Hasnah Tanalol and Abdullah Bade

Abstract—Challenges of cloth simulation have attracted researchers to improve issues related to computational cost in resolving collision. Proximity detection is performed by applying collision detection on the triangulated data of the cloth particles to the coordinates of the close points within accepted range. The collision detection step can be accelerated using hierarchical bounding volume. Constructing good hierarchies, however, is difficult because the number of hierarchies grows exponentially with the number of triangles. In this paper, a framework to enhance the conventional approach in producing highly efficient, robust and fast proximity computation of self-collision checking is presented. The collision checking procedure between cloth surfaces is divided into two phases: broad phase and narrow phase. The goal of these phases is to apply successive filters in reducing computational complexity. The enhanced procedure starts in the broad phase by encapsulating the arbitrary triangle nodes with multiresolution sphere bounding volume (BV). A bounding volume hierarchy is then automatically generated using modified evolutionary algorithm (EA) by recursively splitting it into sphere tree. Collisions can then be detected by recursively traversing the hierarchies of two colliding triangle. Any cloth surface points beyond the BV region is ignored and removed from the collision calculation list. Meanwhile, in the narrow phase, the improved proximity detection will be applied to detect self-collision in order to improve overall performance and preserve the visual realism of cloth simulation. It is expected that the proposed technique will able to avoid penetration among cloth surfaces and self-collision handling could be more efficient and robust with the EA method.

Index Terms— self-collision detection, cloth simulation, multiresolution sphere bounding volume, bounding volume hierarchy.

1 Introduction

CLOTH simulation has been widely used not only in the digital animation industry, but also in computer games and textile industries. As the power of the average personal computer increases, the real-time application has added the realism to the simulation of the real world. Interaction of objects in the virtual world is often an important aspect of computer simulation. In order to handle the interaction, the collision between two objects needs to be tested for intersection and once detected, dynamic equations are applied to simulate appropriate collision response. The main problem in collision detection is to control the computational complexity due to the discretization. The surface of the cloth is represented by triangle meshes that can have several thousand triangles each. Testing each pair of triangle for potential collisions is an unrealistic task.

In the real world, cloth sometimes touches its own surface when it is in motion. When this occurs in computer graphics, penetration will happen on the cloth created and the simulation will be seen as unrealistic. To prevent cloth interpenetration, the simulation system must be able to detect touching or self-collision of the cloth. Figure 1 shows how self-collision occurs in cloth simulation.

- *S.H. Author is a postgraduate student with the GRAVSLab, Universiti Malaysia Sabah, Malaysia, Sabah 88400. E-mail: hasnah@ ums.my*
- *A.B. Author is with the Mathematics with Computer Graphics Department, Universiti Malaysia Sabah, Malaysia, Sabah 88400. E-mail: abb@ ums.com*

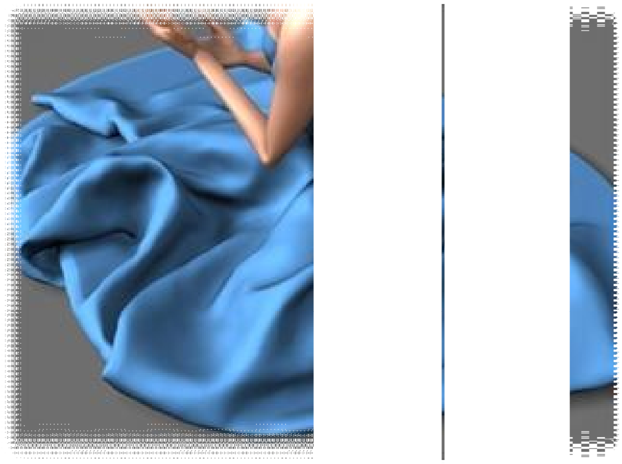

Fig. 1 : Self-collision in cloth simulation [1]

Generally, collision detection of deformable objects is a demanding task, and therefore interactivity is not easy to reach. Implementing cloth simulation in interactive environments require efficient and robust methods for basic simulation tasks, such as deformation, collision detection and collision response. Thus, efficient collision detection is the fundamental problem which leads to the bottleneck in cloth simulation.

In this paper, we propose a framework designed to detect self-collision for multiresolution cloth surface using hierarchical bounding volume. Besides, an overview of self-collision detection and how essential it is in achieving a realistic interaction in real-time system are also presented. The key problem in cloth simulation has always been the expensive computational cost and also the accuracy. When the cloth animated, the most obvious problem which arises is the $O(n^2)$ problem of detecting collisions between all *n* triangles. This is clearly undesirable property of any collision detection algorithm. One of the efficient solutions is using hierarchical bounding volume (BVH) [2]. The major advantages of bounding volume hierarchies are the fast query for interaction testing which will reduce the computation cost.

2 PREVIOUS WORK

Many different approaches have been proposed to accelerate the performance of cloth simulation. They are mainly proposing the new methods include allowing larger time steps [3], GPU-based parallelization [4], wrinkle synthesis [5], faster collision detection [1], [6], multi-resolution approaches [7] and bounding volume hierarchy [8]–[11] .

2.1 Bounding Volume Hierarchy

Generally, collision detection algorithm can be divided into two main approaches: space subdivision and bounding volume hierarchies. Among this two different approaches, the bounding volume hierarchies (BVH) proved to be the most successful in contemporary system [12]. Numerous researchers have used BVH as a means of performing efficient process of determining the collision point and interpenetration test can be accelerated. BVH are one of the simplest, most widely used and a very efficient data structures for performing collision detection for rigid bodies and to some extent, for complex models such as deformable objects [13]. Hubbard [14] and Bradshaw [15] had explored the hierarchy construction phase to obtain a better approximation of the object's shape whilst James [16] and Guibas et. al [17] focused on the hierarchy traversal phase to speed up the process. Somchaipeng [18] claims that BVH are widely used in computer graphics because they are applicable of handling more general shapes than most feature-based and simplex-based algorithm. Besides, using BVH tend to generate smaller hierarchies compared to spatial subdivision algorithms and able to produce highly accurate bounding volume with a smooth degradation of objects. Besides it is the most popular method for answering proximity and collision queries and therefore, often possible to reduce the running time of a geometric query from, e.g., O(n) to O(log n) [19].

There are many different geometric primitives have been used for constructing the Bounding Volume Hierarchies (BVH) used to perform narrow phase processing. These include: Spheres [14], [20], Axis Aligned Bounding Boxes (AABB) [21], Oriented Bounding Boxes (OBB) [22], Discrete Oriented Polytopes (*k-DOP*) [23], Quantised Orientation Slabs with Primary Orientations (*QuOSPO*) [24], Spherical Shells [25] and Sphere Swept Volumes (*SSV*) [26]. Common BV used in BVH are as depicted in Figure 2 below.

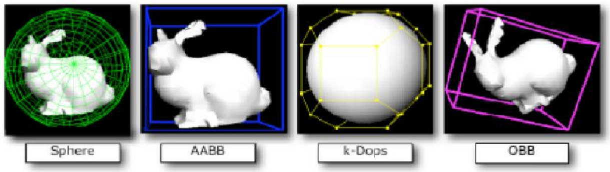

Fig. 2. Common BV in previous researches [27]

A bounding volume hierarchy is a tree data structure on a geometric model H that stores all the geometric primitives of H in the leaf nodes. Each node in the tree stores a volume that encloses all the primitives located below it, i.e., in its subtree. In this way, the root node stores a bounding volume (BV) enclosing all the primitives or the entire model. And the children nodes, store BVs enclosing various subsets of the primitives or parts of the model in a wrapped hierarchical way. The following Figure 3 shows an example of BVH with primitives containing in BV.

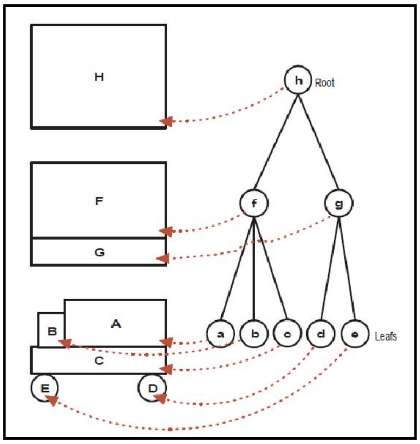

Fig. 3. An example of a Bounding Volume Hierarchy with primitives containing Bounding Volumes. [28]

The collision detection will be done by visiting the all the nodes one by one according to traversal algorithm such as Breadth-First or Depth-First traversal. The bounding volume hierarchies of two objects are traversed recursively. The recursion stops at the leaves and at disjoint bounding volumes. Whenever two BVs overlap, one of them is tested against the children of the other one.

2.2 Multiresolution Cloth

Using physically based simulations method, the cloth is modelled as a regular grid of point particles (Figure 4) of articles using triangles and simulates the cloth dynamics using complex mathematical equations. The computational time highly depends on the resolution of the meshes or particle grid. As the resolution of meshes get higher, the quality of cloth simulation will also increase to capture the detailed features such as folds and wrinkles. Although, using a coarse particle grid reduces the realism of the simulation significantly. Hence there arises a need to use particle grids with different resolution so that enough particles mapped on the grid such that visual appeal is not compromised.

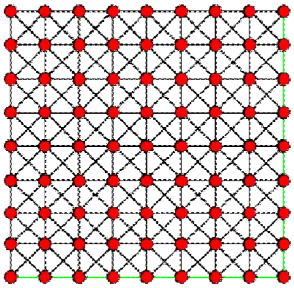

Fig. 4. Cloth is modelled as a regular grid.

Existing multi-resolution techniques [7], [29] achieve a higher simulation performance by providing adaptive meshes. These techniques identify mesh regions that require high accuracy and use more high resolutions only for those regions instead of using a uniformly refined mesh. However, existing multi-resolution techniques do not attempt to simplify mesh regions where can be represented with lower resolutions while providing plausible simulation quality.

However, this will lead to a significant drop of computation performance as the time complexity of most high-quality cloth simulations is higher than linear functions with the number of vertices of the mesh [30]. Thus there arises a need to use adaptive refinement of meshes which identify mesh region that require high accuracy in some particular part of the cloth and use higher resolution only for those regions instead of using a uniformly refined mesh.

2.3 Cloth Model

In this paper, geometric structure and physical property of cloth are represented in the spring-mass system. Mass-spring model consists of network of a lot of mass and spring, and it can represent mechanical properties of cloth by network structures of mass [31]. Mechanical properties of cloth influence mainly stretching-compression, shearing, and bending. Fig.5 shows representations of each deformation.

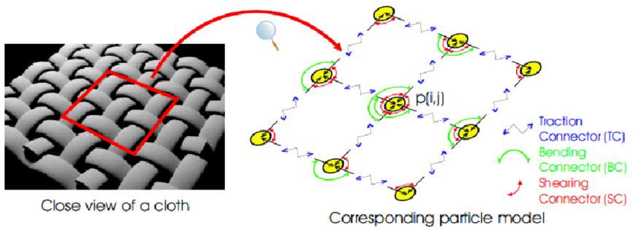

Fig. 5. Structure of mass-spring model (Source: INRIA - Nonlinear Cloth Simulation 2003)

3 COLLISION DETECTION

Collision detection can be represented and built as a pipeline [14] as shown in Figure 6. Collision detection can be divided in to two main parts: broad-phase and narrow-phase. The goal of this pipeline is to apply fast culling of the pairs of primitives that do not collide [14] in order to break down the O(n^2) complexity. Therefore, the

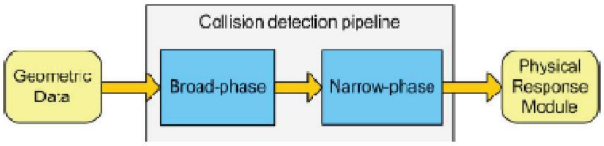

Fig. 6. Collision detection pipeline. [32]

goal of any collision detection algorithm is to first reduce the number of object pairs that must be considered using an efficient algorithm.

The primitive can be wrapped in a bounding volume (BV) to improve the computing performance. By using BV the surface representation is able to be simplified for a fast approximate collision detection test. This could be done by checking the information of the BV whether the bounded primitives could interfere with each other.

The basic idea is to apply a test first to a simple bounding volume before using a time consuming and exact test for the original primitive. If the simple test fails the complex test does not need to be performed as it will fail as well.

3.1 Self-Collision

Detecting self-collision has long posed a challenge to computer animation, and it often becomes the bottleneck in simulations, in particular for cloth animation [33]. In the real world, cloth sometimes touches its own surface when it is in motion. When this occurs in computer graphics, penetration will happen on the cloth created and the simulation will be seen as unrealistic. To prevent cloth interpenetration, the simulation system must be able to detect touching or self-collision of the cloth. In dynamic environment, in which millions of geometric primitives may be involved, highly efficient collision detection algorithms are necessary. Proximity checking or interpenetration prediction is a procedure to identify nearly intersects nodes which constitute the cloth model. The interpenetration detection has become more challenging once the generated cloth or garments placed or draped over the objects and at the same time the running cloth simulation system tries to animate the cloth simulation properties such as wrinkle, fold, bend and stretch in order to preserve visual realism of cloth simulation, especially in the dynamic, real-time environment. Therefore, the animation will be more believable if collision detection or self-collision detection techniques are added into the simulation [11] in order to prevent cloth penetration.

International Journal Of Interactive Digital Media, Vol. 1(3), ISSN 2289-4098, e-ISSN 2289-4101
http://www.ijidm.org

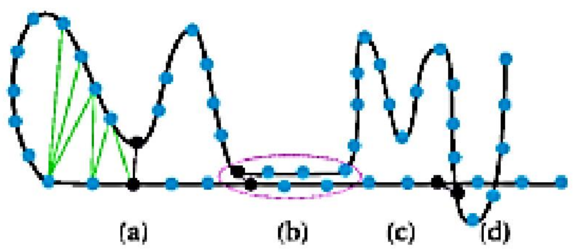

Fig. 7. Self-collisions of a deformable object. The active pairs of points are shown in black. (a) An active pair has converged iteratively to a local distance minimum. (b) A collision has been detected and the associated collision cluster is encircled. (c) No active pair is present yet. (d) A collision has been detected too late and an intersection already occurred. [34]

4 PROPOSED TECHNIQUE

We propose a framework for transforming low resolution cloth simulations into higher resolution cloth animations in real time. The process is divided into two enhanced phases; broad-phase and narrow-phase as in Figure 8 below.

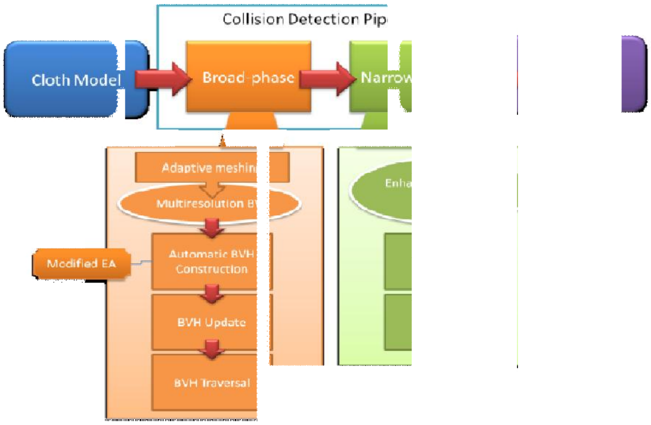

Fig. 8. The proposed framework

In order to filter the potential self-collision, the arbitrary triangle mesh of cloth surface will be enclosed with sphere bounding volume as in Figure 9.

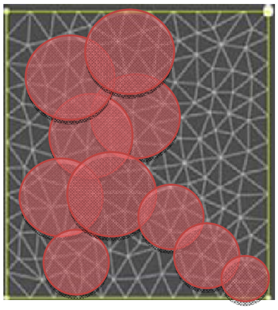

Fig. 9. Multiresolution bounding volume

The generic method for incremental mesh adaptation based on hierarchy of semi-regular meshes (Figure 10) will be used as the cloth meshes. The method supports any refinement rule mapping vertices onto vertices such as 1-to-4 split or √3-subdivision [35] will be the foundation of the proposed multiresolution bounding volume. Semi-regular mesh is obtained from a coarse triangular mesh applying recursively refinement rule, such as 1-to-4 split [36] or √3-subdivision [37]. Semi-regular meshes are inherently multiresolution, have good aspect ratio of triangles and are easy to handle [35].

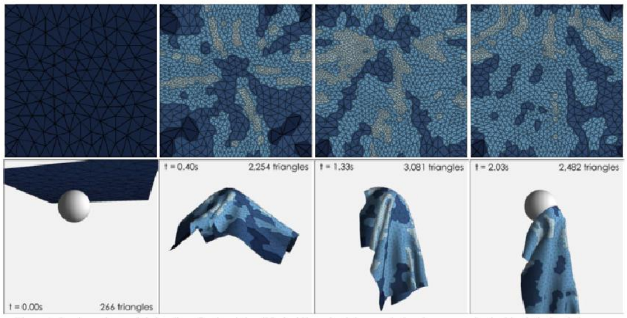

Fig. 10. Mesh adaptation based on hierarchy of semi-regular meshes [35].

Using adaptive meshing allows us to reduce the number of mesh elements and then to reduce computational time. In addition, cloth surface is more realistic than in methods which employing uniform meshes.

After that, the BVH will be automatically construct and update according to the most optimal cost determined by the learned tree traversal using the evolutionary algorithm. It is also a desirable for height of the hierarchies to be kept low [38]. This ensures that traversing the hierarchy from root to leafs can be done in few steps.

In the narrow phase, a hybrid proximity queries is proposed to enhance the accuracy of point of collision. The algorithm will compute the minimum distance between potential colliding triangles. The Gilbert-Johnson-Keerthi (GJK) and Lin-Canny (LC) algorithm are well-known fast solution to the problem. The hybrid between GJK's and LC's algorithm can be used to improve the shortcoming of the existing algorithm which is the instability.

5 CONCLUSION

This paper discussed a framework to enhance the conventional approach in producing highly efficient, robust and fast proximity computation of self-collision checking. The enhanced procedure starts in the broad phase by enclosing the arbitrary triangle nodes with multiresolution sphere bounding volume (BV). A bounding volume hierarchy is then automatically generated using modified evolutionary algorithm (EA) by recursively splitting it into sphere tree. In the narrow phase, the hybrid proximity detection algorithm will be applied to detect point to point self-collision in order to improve overall performance and preserve the visual realism of cloth simulation. It is expected that the proposed technique will able to avoid penetration among cloth surfaces and self-collision handling could be more efficient and robust with the EA method.

REFERENCES

[1] N. K. Govindaraju, D. Knott, N. Jain, I. Kabul, R. Tamstorf, R. Gayle, M. C. Lin, and D. Manocha, "Interactive collision detection between deformable models using chromatic decomposition," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 991–999, 2005.

[2] M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M. P. Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, and others, "Collision detection for deformable objects," *Comput. Graph. Forum*, vol. 24, no. 1, pp. 61–81, 2005.

[3] D. Baraff and A. Witkin, "Large steps in cloth simulation," in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, 1998, pp. 43–54.

[4] S. Green, "Nvidia□: Cloth simulation.," 2003.

[5] H. Wang, F. Hecht, R. Ramamoorthi, and J. O'Brien, "Example-based wrinkle synthesis for clothing animation," *ACM Trans. Graph.*, vol. 29, no. 4, p. 1, 2010.

[6] D. Kim, J. Heo, H. J, J. KIM, and S. Yoon, "HPCCD: Hybrid parallel continuous collision detection.," *Comput. Graph. Forum (Pacific Graph.*, vol. 28, 2009.

[7] J. Villard and H. Borouchaki, "Adaptive meshing for cloth animation," *Eng. Comput.*, vol. 20, no. 4, pp. 333–341, May 2005.

[8] J. Mezger, S. Kimmerle, and O. Etzmuß, "Hierarchical Techniques in Collision Detection for Cloth Animation," *J. WSCG*, vol. 11, no. 2, pp. 322–329, 2003.

[9] W. Feng, Y. Yu, and B. Kim, "A deformation transformer for real-time cloth animation," *ACM Trans. Graph.*, vol. 1, no. 212, pp. 1–9, 2010.

[10] N. S. M. Shapri, A. Bade, and D. Daman, "Hierarchy Techniques in Self-Collision Detection for Cloth Simulation," *2009 Second Int. Conf. Mach. Vis.*, pp. 325–329, 2009.

[11] M. Hutter and A. Fuhrmann, "Optimized Continuous Collision Detection for Deformable Triangle Meshes," *Computer (Long. Beach. Calif).*, vol. 15, no. 1–3, pp. 25–32, 2007.

[12] F. Zhigang, J. Jianxun, and X. Jie, "Efficient Collision Detection Using a Dual K-DOP-Sphere Bounding Volume Hierarchy," *Inf. Technol. Appl.*, pp. 185–189, 2010.

[13] J. Klein and G. Zachmann, "ADB-Trees□: Controlling the Error of Time-Critical Collision Detection," in *8TH International Fall Workshop Vision, Modeling, and Visualization*, 2003.

[14] P. M. Hubbard, "Approximating polyhedra with spheres for time-critical collision detection," *ACM Trans. Graph.*, vol. 15, no. 3, pp. 179–210, 1996.

[15] G. Bradshaw and C. O'Sullivan, "Adaptive medial-axis approximation for sphere-tree construction," *ACM Trans. Graph.*, vol. 23, no. 1, pp. 1–26, 2004.

[16] D. James and D. Pai, "BD-tree: output-sensitive collision detection for reduced deformable models," *ACM Trans. Graph.*, 2004.

[17] L. J. Guibas, A. Nguyen, and L. Zhang, "Zonotopes as bounding volumes," *Proc. 14th Annu. ACMSIAM Symp. Discret. algorithms*, pp. 803–812, 2003.

[18] K. Somchaipeng, K. Erleben, and J. Sporring, "A Multi-Scale Singularity Bounding Volume Hierarchy," in *International Conference in Central Europe on Computer Graphics Visualization and Computer Vision*, 2004, pp. 179–186.

[19] T. Larsson and T. Akenine-Möller, "Bounding Volume Hierarchies of Slab Cut Balls," *Comput. Graph. Forum*, vol. 28, no. 8, pp. 2379–2395, 2009.

[20] S. Quinlan, "Efficient distance computation between non-convex objects. pages," *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 3324–3329, 1994.

[21] Gino Johannes Apolonia van den Bergen, *Collision Detection in Interactive 3D Environments (The Morgan Kaufmann Series in Interactive 3D Technology)*. 2004.

[22] S. Gottschalk, M. C. Lin, and D. Manocha, "OBBTree: A Hierarchical Structure for Rapid Interference Detection," *Proc. SIGGRAPH*, vol. 30, pp. 171–180, 1996.

[23] J. T. Klosowski, M. Held, J. S. B. Mitchell, H. Sowizral, and K. Zikan, "Efficient Collision Detection Using Bounding Volume Hierarchies of k-DOPs," *IEEE Trans. Vis. Comput. Graph.*, vol. 4, no. 1, pp. 21–36, 1998.

[24] T. He, "Fast Collision Detection Using QuOSPO Trees," in *Proc of the Symposium on Interacitve 3D Graphics*, 1999, pp. 55–62.

[25] S. Krishnan, "Spherical shell: A higher order bounding volume for fast proximity queries," in *Proc. of Third International Workshop on Algorithmic Foundations of Robotics*, 1997.

[26] E. Larsen, S. Gottschalk, M. Lin, and D. Manocha, "Fast proximity queries with swept sphere volumes," pp. 1–32, 1999.

[27] A. Bade, N. M. Suaib, A. M. Zin, and T. M. Tengku Sembok, "Oriented convex polyhedra for collision detection in 3D computer animation," in *Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia - GRAPHITE '06*, 2006, p. 127.

[28] K. A. Andersen and C. Bay, "A survey of algorithms for construction of optimal Heterogeneous Bounding Volume Hierarchies," 2006.

[29] D. Hutchinson, M. Preston, and T. Hewitt, "Adaptive refinement for mass/spring simulations," *Eurographics (1996)*, vol. 45, pp. 31–45, 1996.

[30] R. Goldenthal, D. Harmon, R. Fattal, M. Bercovier, and E. Grinspun, "Efficient simulation of inextensible cloth," *ACM Trans. Graph. (Proceedings SIGGRAPH 2007*, vol. 26, no. 3, p. 49, 2007.

[31] X. Provot, "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior," *Integr. VLSI J.*, pp. 147–154, 1995.

[32] Q. Avril, G. Valerie, and B. Arnaldi, "A Broad Phase Collision Detection Algorithm Adapted to Multi-cores Architectures," in *Proceedings of Virtual Reality International Conference (VRIC 2010)*, 2010, no. April.

[33] S. Schvartzman, Á. Pérez, and M. Otaduy, "Star-contours for efficient hierarchical self-collision detection," *ACM Trans. Graph. (Proc. ACM SIGGRAPH)*, 2010.

[34] S. Kimmerle, M. Nesme, F. Faure, and others, "Hierarchy accelerated stochastic collision detection," in *Proceedings of Vision, Modeling, Visualization*, 2004, pp. 307–314.
[35] V. Volkov and L. Li, "Real-time refinement and simplification of adaptive triangular meshes," *IEEE Vis. 2003*, pp. 155–162, 2003.
[36] C. T. Loop, "Smooth Subdivision Surfaces Based on Triangles," University of Utah, 1987.
[37] S. Bischoff, L. Kobbelt, and H. Seidel, "Towards Hardware Implementation Of Loop Subdivision," *Work. Graph. Hardw.*, 2000.
[38] J. Klosowski, "Efficient collision detection for interactive 3D graphics and virtual environments," State University of New York at Stony Brook, 1998.

Siti Hasnah Tanalol obtained her BIT in Computer Science from Universiti Kebangsaan Malaysia in 1998 and MSc from Strathclyde University, Glasgow, UK. Currently, she is pursuing her doctorate degree at the School of Science & Technology, Universiti Malaysia Sabah. She is also a member of Real-time Graphics & Visualization research group (GRAVS) at School of Science & Technology, Universiti Malaysia Sabah. Her research interests include scientific visualization and real-time cloth simulation.

Dr. Abdullah Bade received his BSc in Computer Graphics from Universiti Teknologi Malaysia, in 1999 and MSc in Computer Graphics & Scientific Visualization, Universiti Teknologi Malaysia in 2003. Hereceived his PhD in Computer Graphics & Scientific Visualization from Universiti Kebangsaan Malaysia, in 2008. He joined Universiti Malaysia Sabah as a senior lecturer in 2011. Currently,he is the program head of Mathematics of Computer Graphics at the School of Science and Technology. His research interests include virtual& augmented reality, and surgical simulation.