# Efficient Culling Criteria for Continues Collision Detection Using a Fast Statistical Analysis Method

Fengquan Zhang[*], Jiaojiao Guo, Jianfei Wan and Junli Qin

*Department of Computer Science, North China University of Technology, Beijing, 100144, China*

**Abstract:** Continuous Collision Detection (CCD) between deforming triangle mesh elements in 3D is significant in many computer and graphics applications, such as virtual surgery, simulation and animation. Although CCD is more accurate than discrete methods, its application is limited mainly due to its time-consuming nature. To accelerate computation, we present an efficient CCD method to perform both inter-object and intra-object collision queries of triangle mesh models. Given a model set of different poses as training data, our method uses Statistic Analysis (SA) to make regression on a deformation subspace and also on collision detection conditions in a pre-processing stage, under a uniform framework. A data-driven training process selects a set of "key points" and produces a credible subspace representation, from which a plug-in type of collision culling certificate can be then obtained by regression process. At runtime, our certificate can be easily added to the classic BVH traversal procedure, as a sufficient condition of collision free cases, providing efficient culling in overlapping test and reducing hierarchy updates frequency. In the end, we describe performance and quality of our method using different experiments.

**Keywords:** CCD, Deformation body, Regression, Statistical analysis, Canonical correlation analysis, Linear discriminant analysis.

## 1. INTRODUCTION

In recent years, virtual reality technology continues to expand in depth and breadth, and increasingly widely used in many areas of our lives such as digital entertainment, industrial manufacturing, education, biomedicine, medical surgery and so on. Simulation of complex models (such as high-definition model) and the study of the deformation process of collision detection is one of the current research hotspot. Compared with classic collision detection method, deformable body collision detection with high accuracy, which is possible to obtain the correct realistic simulation results. Continuous Collision Detection (CCD) is a key part of many algorithms in virtual surgery. But, the computational complexity is a new challenge for CCD technology especially in deformable body.

In this paper, a unified framework based on statistical analysis is designed. As for fine models have great data dimensions, we designed a unified framework, under which we realized the deformation driven by key points for data preparation and trained a threshold for collision culling. Dimension reduction method was used to promote efficiency. Our framework included a pre-process stage and a run-time stage. During the pre-process stage, we calculated and saved the information of subspace, clustering and culling criteria, all of which were performed under unified data-driven subspace building progress and a regression process. In the run-time stage, with the support of pre-process, we use key points to update model position and culling criteria,

driving deformation for acquiring data and performed collision detection [1, 2].

## 2. RELATED WORK

We first review applications in Statistical Analysis in computer graphics. Then we discuss on some methods with regression. Lastly, we introduce some recent works about CCD.

Statistical Analysis: There are a few statistical analysis approaches widely used in machine learning and then in computer graphics. Principal Component Analysis (PCA), Canonical Correlation Analysis (CCA) and Linear Discriminant Analysis (LDA) are usually considered to be the most popular ones. The PCA technique extracts the most several important axes given a set of scattered data points, in order to recover these data by reduced coordinates [3, 4], while the middle computes the maximum correlation between two sets of variables, thus capturing data dependency [5-7]. In this paper, we also use PCA with the non-linear kernel version for Subspace construction. Feng proposed a novel regression method for deformation simulation, which is also based on CCA method [8]. A fully automatic method is designed to obtain bones and weights [9]. Then Kavan improved above method, which tended it to highly deformable object surface [10].

CCD for deformable models is extensively studied in computer graphics, and there has been a sound recent survey [11]. CCD provides more accurate collision results than discrete collision detection (DCD). Therefore, CCD has been studied relatively recently for rigid models [12], articulated models [13], and deformable models [14]. Since CCD requires longer computation time than DCD, different

*Address correspondence to this author at the Department of Computer Science, North China University of Technology, Beijing, 100144, China; E-mail: fqzhang@ncut.edu.cn
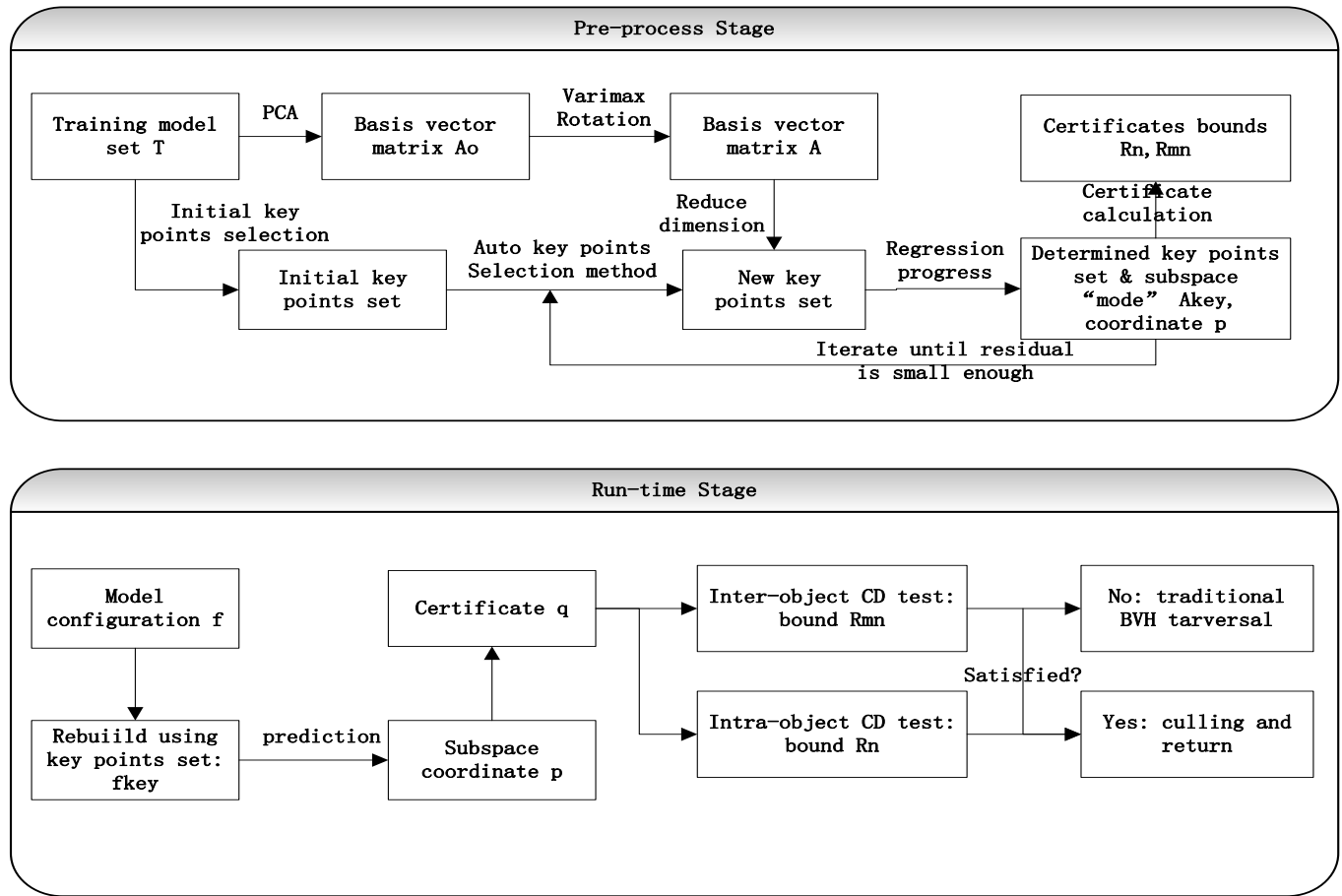
**Fig. (1).** The workflow of our method.

acceleration techniques have been proposed. Some of them are based on normal cone culling and coplanarity-based culling [15, 16]. The elementary tests between the primitives (e.g., triangles) are performed using local advancement methods [17].

## 3. OVERVIEW

Our primary contribution is a novel uniform framework using SA scheme to make prediction for both deformation representation and collision detection criteria. We also introduce PCA method to update criteria at real-time, and build a credible subspace representation for original models to achieve efficiency. The workflow of this work is shown in Fig. (**1**).

## 4. STATISTIC ANALYSIS

The subspace is built using with the similar method of [1]. Such subspace models take the form:

$$f = Ap \tag{1}$$

where $A \in R^{3V \times r}$ contains the $V = O(N)$ vertex displacement of an N triangle mesh, is the basis matrix where each of the $r \langle\langle V$ columns gives one subspace mode, and $p \in R^r$ are the reduced coordinates. Usually, a set of characteristic

key points would be chosen, so that the above form becomes:

$$f_{key} = A_{key}p \tag{2}$$

If the key points are well chosen to have a good representativeness of features on original models, the reduced coordinate $p$ could have quite low residual over origin models. After successfully building a subspace, we are able to map the origin models to reduced-order deformable models.

As for our algorithm of automatic key points selection need to calculate the residual matrix at each iteration, an initial key points set should be given in advance. In our experiments, a uniform sampling method [1] is introduced to choose initial key points set. This method is shown as follows. Firstly, distributing points densely over the surface; Secondly, repeatedly removing all points within a given radius. The radius can be chosen in accordance with the size of model used; Repeat progress above until a desired sampling density is reached. For meshes, the above selecting method produces satisfying results when the triangle aspect ratio is restricted to avoid long and skinny triangles. The number of initial key points doesn't have to be large. About $1\% \sim 4\%$ could be fine, as is shown in Fig. (**2**).

Cueing error is due to using an inadequate set of key point values as distance proxies, and we used the method

mentioned in Meyer's work [1], which introduces an iteration progress to select key points for minimizing this error. The key points choosing workflow can be seen from the Fig. (**3**).
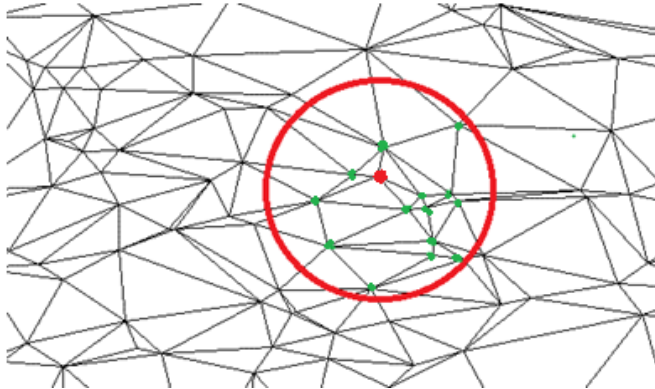


**Fig. (2).** The initial key points choosing method. All the vertices (green) will be removed and in one radius and only one point (red) will remain.
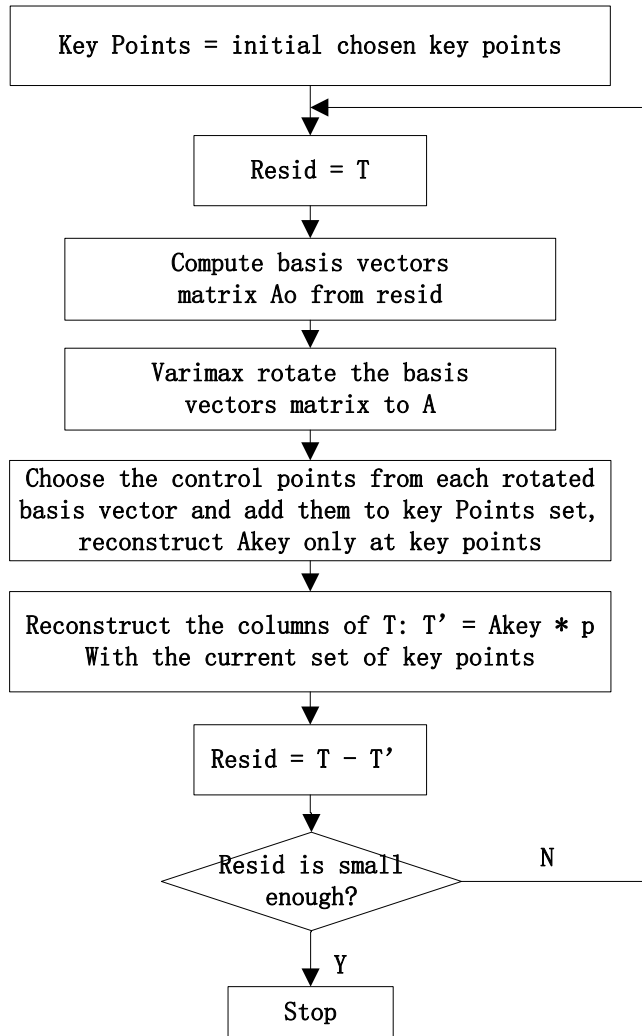


**Fig. (3).** Workflow for choosing key points.

In this training stage, we construct a mapping function between key point positions for example poses and the corresponding certificates of each pose. Unlike PCA which can find the principle component in the data set, Canonical Correlation Analysis is capable of finding the maximum correlation between two data sets. In our case, it can be computed by

$$\min_F \left\| F*K\text{-}C \right\|^2 \qquad (3)$$

where matrix $K$ includes key point positions in each column, and matrix $C$ includes the corresponding certificates computed for each pose. At runtime, given a novel set of key point positions, we can obtain the new certificates by matrix multiplication, which should be more accurate than those computed for the training data.

When using the SA technique there are always two potential sources of error. The first of these is the projection error that results from the fact that the target pose itself may not be in the subspace [1]:

$$e_{proj} = \left\| f - A p_{proj} \right\| \qquad (4)$$

$$p_{proj} = \arg\min \left\| f - A p \right\| \qquad (5)$$

we choose two points from each basis vector $V$, the first point $i$ is the point with the largest magnitude in the basis vector and the second $j$ is the point whose inner product with our first point has the largest negative value:

$$i = \arg\max \left\| V_K \right\| \qquad (6)$$

$$j = \arg\min(V_i \cdot V_k) \qquad (7)$$

## 5. RUNTIME COMPUTATION

And in the run-time stage, given the initial and final configurations, we take advantage of the same key points set to build new subspace mode A for each frame. The certificates based upon A will also be updated at each configuration. Considering Bounding Volumes Hierarchy (BVH) is a common choice in collision detection, we use axis-aligned bounding boxes (AABBs) as our bounding volumes (BVs) to build BVH. In the BVH traversal step, the certificates are applied to reduce overlap tests on two levels: both the tests of two non-adjacent triangles and tow BVs. If the certificates are satisfied, the current traversal brunch will be culled. The process can be described as follows, based on paper [1] method:

1)     Compute values only at the key points: $f_{key}$;

2)     Least squares project the key points onto the subspace to find the subspace coordinates p with Equation (2);

3)     Reconstruct all points by using Equation (1), to get predicted coordinates $\hat{f} = Ap$ .

The collision-free certificates are predicted by the regression results in the training stage. Note that, in the offline stage, we build a linear mapping between key point positions and the certificates. At run time, the current certificates are computed by matrix multiplication, fast and easy to be implemented on hardware.

The Collision-Free certificates determine whether collisions occur. If test was satisfied, the certificates

guarantee a collision-free configuration for a sub-tree of the BVH, or between two sub-trees. Given two tree nodes n and m, we use the subspace presentation to update certificates $\|p\|_M$ which can guarantee that no triangle from $T_n$ collides with a non-neighboring triangle from $T_m$. Whereas, if false, it is inconclusive. Here, $p$ p are the current reduced coordinates. And we use the M-weighted 2-norm as

$$\|p\|_M = \sqrt{p^T M p} \quad \|p\|_M = \sqrt{p^T M p} \tag{8}$$

for some positive definite (typically diagonal) constant matrix M. So the collision-free certificate corresponds to an ellipsoidal region in the $p$ p space. By comparing $\|p\|_M$ with pre-computed bound $B_n$ and $B_{nm}$ $B_{nm}$, we could implement the culling.

## 6. RESULT AND DISSCUSSION

As a simple example to illustrate how our method works, we use a series of cones as training data, to predict a novel pose position. We have the triangle meshes as original training models to build data matrix T, as is shown in Fig. (**4**):
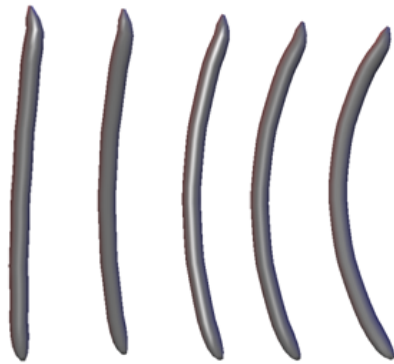


**Fig. (4).** Training models set: different poses of a same cylindrical object.

Through the training and regression process we could get proper Key Points set and Subspace mode. Then we use them to predict deformed position as shown in Fig. (**5**) and perform collision detection as shown in Fig. (**6**). And Fig. (**7**) shows another experiment using more complex models, a lion and hand which contain thousands and tens of thousands vertices. Fig. (8) shows the effect of self-collision denoted by red..



**Fig. (5).** Deformed model that is predicted by SA scheme. The points set (lower drew in red color) is Key Points selected. The prediction process is only at these points.
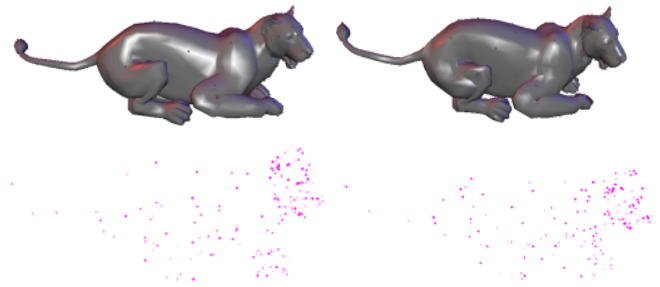


**Fig. (6).** Experiment using triangle mesh of a lion. On the left is the origin model. On the right is deformed model, right foreleg of which is bended inward.
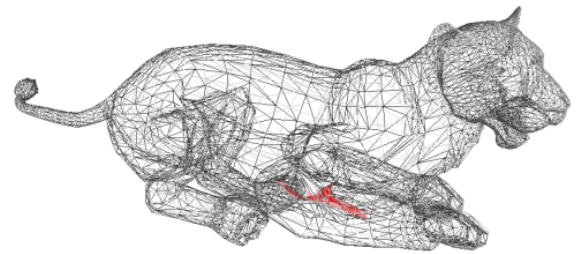


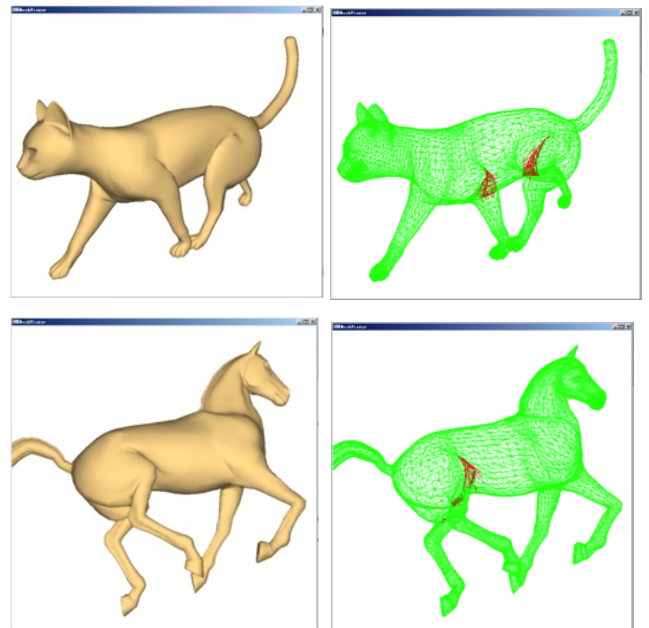**Fig. (7).** An experiment including intra-object collision using triangle mesh of a lion.



**Fig. (8).** The effect of self-collision.

## CONCLUSION

We have presented a novel collision detection method using the statistical method during a pre-processing stage. The mapping between key points and the collision detection threshold is constructed during this stage. In runtime, given a new set of key points, the corresponding threshold will be predicted, thus avoiding the extra computations in runtime. Furthermore, the collision detection criteria can be plug into the classic BVH traversal procedure, with only a comparison between two floats.

Like other example-based methods, there exist a few limitations in our work calling for further research. However, if examples are completely missing in the perpendicular direction, the predictor cannot perform correctly on the new key points. An easy way to improve this is to insert an extra training example in that direction. Second, as shown in Fig. (**9**), although our trained key points representation maintains most information of original model, error still exist in the region where triangle aspect ratio is large.
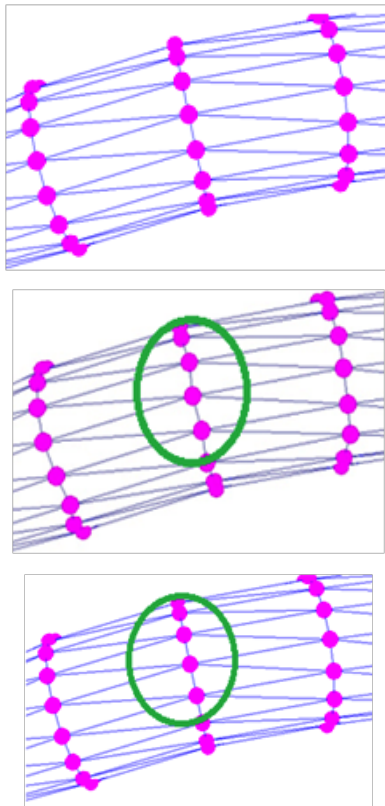


**Fig. (9).** Original model pose (top), novel pose predict using all points of model (middle), and the same pose predict using only key points (bottom). The middle one has little residual with the left while the right one's error is slightly bigger.

## CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

## ACKNOWLEDGEMENTS

## REFERENCES

[1]  M. Meyer, and J. Anderson, "Key point subspace acceleration and soft caching," *ACM Transactions on Graphics*, vol. 26, no. 3, pp. 74.1-8, 2007.

[2]  J. Barbic, and L. James, "Subspace self-collision Culling," *ACM Transactions on Graphics,* vol. 29, no. 4, pp.1-9, 2010.

[3]  H. Moon, and P. Phillips, "Computational and performance aspects of pca-based face-recognition algorithms," *Perception,* vol. 30, pp. 303-321, 2001.

[4]  P. Sloan, J. Hall, J. Hart, and J. Snyder, "Clustered principal components for precomputed radiance transfer," *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 382- 391, 2003.

[5]  T. Melzer, M. Reiter, and H. Bischof, "Appearance models based on kernel canonical correlation analysis," *Pattern Recognition*, vol. 36, no. 9, pp. 1961-1971, 2003.

[6]  R.Y. Wang, K. Pulli, and J. Popović, "Real-time enveloping with rotational regression," *ACM Transactions on Graphics,* vol. 26, pp. 1-10, 2007.

[7]  M. David, Y. Andrew, N. Jordan, and I. Michael, "Latent dirichlet allocation", *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, 2003.

[8]  W. Feng, B. Kim, and Y. Yu, "Real-time data-driven deformation using kernel canonical correlation analysis," *Transactions on Graphics*, vol. 27, no. 3, pp. 1-8, 2008.

[9]  D. James, and C. Twigg, "Skinning mesh animations", *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 399-407, 2005.

[10]  L. Kavan, S. Collins, J. Zara, and C. Osullivan, "Skinning with Dual Quaternions", *I3D '07: Proceedings of the Symposium on Interactive 3D Graphics and Games*, pp. 39-46, 2007.

[11]  M. Teschner, "Collision detection for deformable objects," *Computer Graphics Forum,* vol. 19, no. 1, pp. 61-81, 2005.

[12]  S. Redon, A. Kheddar, and S. Coquillart, "Fast continuous collision detection between rigid bodies," *Computer Graphics Forum*, vol. 21, no. 3, pp. 279-288, 2002.

[13]  X. Zhang, S. Redon, M. Lee, and Y. Kim, "Continuous collision detection for articulated models using taylor models and temporal culling," *ACM Transactions on Graphics*, vol. 26, no. 3, pp. 1-15, 2007.

[14]  N. Govindaraju, D. Knott, D. N. Jain N, I. Kabul, and R. Tamstorf, "Interactive collision detection between deformable models using chromatic decomposition," *ACM Transaction on Graphics*, vol. 24, no. 3, pp. 991-999, 2005.

[15]  M. Tang, S. Curtis, S. Yoon, and D. Manocha, "Interactive continuous collision detection between deformable models using connectivity-based culling," *IEEE Transactions on Visualization and Computer Graphics,* vol. 15, no. 4, pp. 544-557, 2009.

[16]  J. Heo, J. Seong, D. Kim, A. Otaduy, and J. Hong, "Fracturing-aware stable collision detection," In: *ACM SIGGRAPH/ Eurographics Symposium on Computer Animation*, Switzerland, pp. 149-158, 2010.

[17]  M. Tang M, Y.J. Kim, and D. Manocha, "Continuous Collision Detection for Non-rigid Contact Computations Using Local Advancement," In: *Proceedings of International Conference on Robotics and Automation*, Alaska, pp. 4016 - 4021, 2010.