3. Kondratov V.T. Teorija izbytochnykh i sverkhizbytochnykh izmerenij: formalizevannoe opisanie vtoroj grupy pravil vyvoda uravnenij izbytochnykh izmerenij krutizny preobrazovanij. Soobshhenie 6. Vymiruvalna ta obchysluvalna tekhnika v tekhnologichnykh protcesakh. – 2013. – № 4. – C. 134 – 147.
4. Kondratov V.T. Teorija izbytochnykh i sverkhizbytochnykh izmerenij: sushhnost sverkhizbytochnykh izmerenij. Soobshhenie 7.1. Visnyk Khmelniczkogo naczionalnogo universitetu. Tekhnichni nauky. – 2014.– № 1. – C. 233 – 242.
5. Kondratov V.T. Teorija izbytochnykh i sverkhizbytochnykh izmerenij: sushhnost sverkhizbytochnykh izmerenij. Soobshhenie 7.2. Visnyk Khmelniczkogo naczionalnogo universitetu. Tekhnichni nauky. – 2014. – № 2. – C. 223 – 236.
6. Aliev T.M., Ter-Khachaturov A.A. Izmeritelnaja tekhnika. M.: Vysshaja shkola. – 1991. – 384 s.
7. Zakon bolshikh chisel. Rezhim dostupa: http://www.exponenta.ru/educat/class/courses/tv/theme0/10.asp .

Рецензія/Peer review : 20.2.2014 р. Надрукована/Printed : 26.3.2014 р.

УДК 004.921

S.I. VYATKIN, S.A. ROMANYUK
Institute of Automation and Electrometry, SB RAS

FUNCTION REPRESENTATION IN GEOMETRIC MODELING

There had been considered the issue of synthesis by means of perturbation functions. The free forms based on the analytical perturbation functions have an advantage of spline representation of surfaces, that is, a high degree of smoothness, and an advantage of arbitrary form for a small number of perturbation functions.

Keywords: Geometric modeling, function-based surface representation, perturbation functions, patches of arbitrary forms.

С.И. ВЯТКИН, С.А. РОМАНЮК
Институт автоматики и электрометрии СО РАН

ФУНКЦИИ ПРЕДСТАВИТЛЕНИЯ В ГЕОМЕТРИЧЕСКОМ МОДЕЛИРОВАНИИ

Розглянуто питання синтезу з використанням функцій збурення. Вільні форми на основі аналітичних функцій збурення мають переваги сплайнового подання поверхонь, тобто характеризуються високою ступіню гладкості і забезпечують формування довільної форми при невеликій кількості функцій збурення.

Ключові слова: геометричне моделювання, функцій завдання поверхнонь, функцій збурення, патчі довільних форм.

Statement of the Problem

Several representations of geometric objects are currently used in computer graphics. Each of the objects, according to its properties, is used in different fields, beginning from 3D simulation and CAD systems up to real-time visualization systems.

The polygonal surface representation is a piecewise linear interpolation of a surface. Its merit is a simple representation and universal application because the piecewise linear representation exists for any surface. We should mention the insignificant computational expenses required for visualization and geometric transformations. The drawback is large database for storing the information on the surface geometry. Highly detailed models (multiresolution geometric models), e.g., antique sculptures, subjected to computer reconstruction have hundreds of millions of triangles.

The spline representation of surfaces [1], along with analytical representation, is used in AutoCAD and 3D Studio systems. It is characterized by a highly accurate representation of 2D and 3D objects.

The functional representation describes most accurately the object geometry and has the smallest size of the required data. Procedures of functional representation demonstrate compact and flexible representation of surfaces and objects that are results of logical operations on volumes. Its disadvantage is complicated geometrical processing and visualization in real time.

Analysis of Research and publications

Several kernels described in [2]:

- Gaussian function (Blinn, Bloomenthal, Shoemake)

$$h(r)=\exp(-a^2r^2)^2,\ \ r>0 \tag{1}$$

Note:

Produced point, line, plane and stopped for arc and triangle.

- Inverse function (Wyvill, van Overveld)

$$h(r)=1/r,\ r>0 \tag{2}$$

- Inverse squared function (Wyvill, van Overveld)

$$h(r)=1/r^2,\ r>0 \tag{3}$$

Note:

Yielded solutions for various primitives. For planes solution, however, did not converge and solution for arcs is expressed via elliptical integrals. It is too difficult.

- Metaballs (Nishimura)

$$h(r)=\begin{cases} 1-3r^2, & 0\le r\le 1/3, \\ 3/2(1-r)^2, & 1/3<r\le 1, \\ 0, & r>1. \end{cases} \tag{4}$$

- Soft objects (Wyvill)

$$h(r)=\begin{cases} 1-(1/9)r6+(17/9)r4-(23/9)r^2, & <1, \\ 0, & r>1. \end{cases} \tag{5}$$

- W-shaped quartic polynomial

$$h(r)=(1-r^2)^2,\ r<1. \tag{6}$$

Note:

Produced solution for point, line, plane, arc; unfortunately, a triangle primitive is not one of them.

- Cauchy function (Sherstyuk)

$$h(r)=1/(1+s^2r^2)^2,\ r>0, \tag{7}$$

where *r* is the distance from the point and coefficient s controls the width of the kernel.

Cauchy kernel covered the whole set of geometric primitives. This kernel allows direct analytical solutions of convolution integral.

An implicit surface S is defined as an isosurface at level T in some scalar field f(p):

$$S = \{p\in R^3 \mid f(p) - T = 0\} \tag{8}$$

A convolution surface is the implicit surface based on a function f(p), obtained via convolution

$$f(p) = g(p)^*\ h(p) = \int_{R^3} g(r)h(p-r)dr \tag{9}$$

The geometry function g(p) defines the shape of an object and its position in 3D space. The kernel function h(p) defines the description of some potential that is produced by each point on the object. Convolved together, these two functions produced a scalar function f(p) that defines the convolution surface.

Using the Cauchy convolution kernel the field functions for a five of primitives can be derived. These primitives are:

- Points
- Line segments
- Arcs
- Triangles
- Planes

Note:

Circles and polygons may be built by combining, respectively, arcs and triangles.

In principle, any geometric primitive may be used as a skeletal element for convolution surface model, but in practice, the choice of such primitive is often limited by technical difficulties of evaluating the convolution integral. All these modeling primitives are presented as closed-form functions, that return the amount of field generated by the primitive at an arbitrary point, calculated to a machine-size float precision. This makes it possible to visualize convolution surfaces using direct rendering algorithms. Several algorithms described in [3].

- Ray-marching (Tuy, Perlin, Hoffert)

This is a brute-force method that steps along the ray, evaluating the field function f(t) on each step. The surface is detected when the sign of f(t) first changes.

$$F(r) = f(t) = 0$$

- LG-surfaces (Karla, Barr)

Was developed an algorithm guaranteed to detect the surface, modeled by functions with computable L and G parameters that represent the Lipschitz constants for the function f and its derivative df/dt along the ray. For non-algebraic modeling functions f, computations of L and G may become prohibilitively difficult, even if L and G a derived in symbolic form.

- Sphere-tracing algorithm (Hart)
- Ray-tracing with interval analysis (Mitchell)

These are modifications of LG-surfaces. It is important to note that all algorithms (Karla, Barr, Hart) that bound the rate of change of the implicit functions f(t), either with a Lipschitz constant or with derivatives df/dt (Mitchell) work better when these bounds are as tight as possible. Therefore, the must be computed at run-time for each ray individually and for each interval along this ray, which may not be an easy task to accomplish for complex functions f. The use and ultimately will turn the root-isolating algorithms (Mitchell, Karla, Barr) into a simple bisection, and the sphere-tracing algorithm into ray marching.

Space mapping technique based on radial based functions (RBFs) is a powerful tool, which offers simple and quite general control of modeled shapes [4].

This paper describes free forms based on the analytical perturbation functions. It is shown that an adequate surface smoothness and a compact object description can be achieved using a limited number of base and perturbation functions.

For visualizing the multilevel recursive object space subdivision algorithm was used [5]. The general surface theory is described in [6].

FUNCTION-BASED OBJECTS CREATED USING THE ANALYTICAL PERTURBATION FUNCTIONS

We propose describing complex geometric objects by specifying the function of deviation (an implicit second-order function) from the base surfaces: triangles (Figs. 1, 2) and quadrics (Fig. 3) [5].

In Fig. 1 the triangle (v1, v2, v3) is represented in the functional space by intersection of five planes: three clipping planes p1, p2, and p3, which are perpendicular to the base planes of the triangle with the normal oriented inward the triangle, and two base planes p4, and p5 of the triangle with oppositely directed normal.

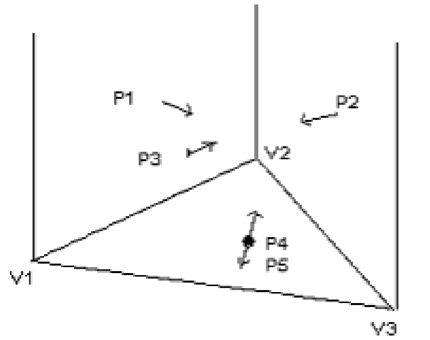

Fig. 1 The clipping and base planes of the triangle

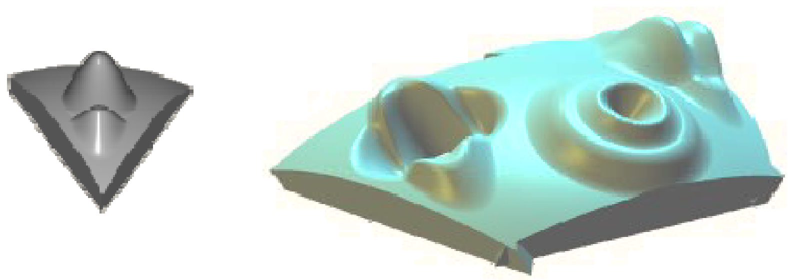

Рис. 2 A patch of arbitrary form (left), three smoothly matched patches of arbitrary form (right).

The freeform is a composition of the base surface and the perturbation $F'(x,y,z) = F(x,y,z) + \sum_{i=1}^{N} R_i(x,y,z)$,

where the perturbation function R(x, y, z) is found as follows:

$$R_i(x,y,z) = \begin{cases} Q_i^3(x,y,z), if\ Q_i(x,y,z) \geq 0 \\ 0, if\ Q_i(x,y,z) < 0 \end{cases} \qquad (10)$$

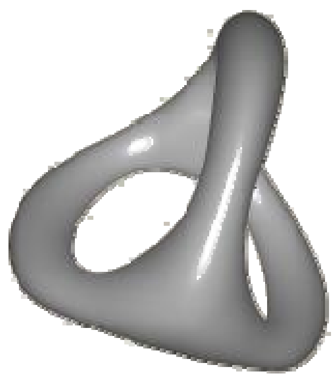

Fig. 3 The freeform based on a 1 quadric (synthesized by means of the 4 analytical perturbation functions of second degree)

Herein, Q(x, y, z) is the perturbing quadric.

Since max[Q + R] ≤max[Q] + max[R], for estimating the maximum Q on some interval we have to calculate the maximum perturbation function on the same interval. The obtained surface is smooth, and creation of complex surface forms requires few perturbation functions.

Papers proposed a multilevel ray-casting algorithm that carries out effective search for the surface point involved in the image formation.

GEOMETRIC OPERATIONS

We have considered geometric objects. Two major types of elements of the set of geometric objects are simple geometric objects and complex geometric objects. A complex geometric object is a result of operations on simple geometric objects.

The set of geometric operations Φ is expressed mathematically in the following form [7]:

$$\Phi_{¡}: M^1+M^2+\ldots+Mn \rightarrow M, \qquad (11)$$

where n is the number of operation operand.

Let the object G1 be defined as $f1(X) \geq 0$. The unary operation (n =1) (17) of the object G1 means operation G2=Φ¡(G1) with the definition

$$f2=\psi(f1(X)) \geq 0, \qquad (12)$$

where ψ is a continuous real function of one variable. Let us consider the following unary, binary operations and relations in more detail.

Projections

Projections of a solid onto three orthogonal planes yield us a representation of its different sides. The projection of 3D solid onto the coordinate plane is considered as a union of sections of the solid by planes parallel to the coordinate plane at a sufficiently small distance from each other. We will a mathematical description of the process for a space of arbitrary dimension.

Let the initial object $G1 \subset En$ be described by the function

$$f1(x1,x2,\ldots,xi,\ldots,xn) \geq 0 \quad (13)$$

and its projection $G2 \subset En\text{-}1$ be described by the function

$$f2(x1,x2,\ldots,xi\text{-}1,xi+1,\ldots,xn) \geq 0 \quad (14)$$

The object G2 can be defined as a union of sections of the object G1 by the hyperplane xi=Cj , where Cj+1 = Cj+Δxi, j=1, N and C1=ximin. Let be the function for the section. As a result, the function for the projection at $\Delta xi \rightarrow 0$ is a union of all functions $f1j$:

$$f2=f11 \vee f12 \vee \ldots \vee f1j \vee \ldots \vee f1N. \quad (15)$$

In the realization of this operation for the whole scene we fixed one coordinate, depending on what projection had to be obtained.

The results are recorded as a set of pictures of the initial 3D solid and its 2D projections onto three orthogonal planes.

Offsetting

The offsetting operation was implemented by transformation of perturbation function coefficients. Thus, one can created an enlarged or diminished copy of the initial object, i.e., make positive or negative offsetting, respectively. For example, solid beats can be simulated. Let the initial object be defined by the function $f(X)>0$, then in the case of this operation, the obtained solid will be described by the function $F=f(X)+C$, where $C<0$ determines the negative offsetting (compression) and $C>0$ determines the positive offsetting (extension). Otherwise, adding together the positive or negative constant and the free term of the perturbation function yields extension or compression of the whole object).

Set-theoretic operations

Let the objects G1 and G2 be defined as $f1(X) \geq 0$ and $f2(X) \geq 0$. The binary operation (n=2) (11) of the objects G1 and G2 means operation G3=Φ¡(G1,G2) with the definition

$$f3=\psi(f1(X),f2(X)) \geq 0, \quad (16)$$

where ψ is the continuous real function of two variables. Let us dwell on the binary operations: set-theoretic operations and 3D metamorphosis (morphing).

For function-based objects on the bases of perturbation functions we propose the following. To create a complex scene, one should describe in it a certain number of primitives necessary for a concrete task. The rendered object with which the rendering algorithm interacts by means of query represents the whole 3D scene. Hence, the geometric model should allow designing of objects and their compositions of infinite complexity. This is primarily achieved by means of Boolean operations of uniting and intersection.

3D Metamorphosis

This operation transforms the first defined object into second with obtaining multiple intermediate forms. The term originates from the word metamorphosis and refers to the animation technique in which one pattern is gradually transformed to another. During the metamorphosis (morphing), the initial pattern is gradually transformed to the final one.

A sequence of frames of transformation of one object to another is generated by means of the initial, final, and key intermediate models.

Let F1 and F2 be values of the perturbation functions of the first and second objects, respectively. Then the resulting perturbation function is calculated as follow:

$$F=\beta F1 + (1-\beta)F2, \quad (17)$$

where β is the positive continuous function.

For function-based objects with the use of perturbation functions, one can perform 3D morphing of nongomeomorphic objects.

Twisting

Twisting is a solid deformation being a particular case of bijective mapping which serves for defining deformations of initial objects. For twisting of the initial solid we found and transformed its coordinates x, y, z.

Global and local deformation

First thing that is necessary to state is that if we want to propagate the deformation it should be somehow added to all object that it affects. Actually the current scene-tree is organized so that it is no possibility to add object only by referencing i.e. without copying. This is done for avoiding situations when being changed somewhere the object unintentionally change the other part of the scene that referenced to it too. Thus the additional perturbations should have such parameters to assure the part-per-part connectivity for each pair of the object the perturbation affects. In this case it will be looked as one perturbation.

Sweeping

We consider the swept volume as a projection of a moving solid from the 4D(x,y,z,t) space to the 3D(x,y,z) space. Then we draw the solid each time new coordinates that were changed by the proper law. In so doing, the previous images are stored in the memory and used to obtain the result of swept volume. The newly formed figure is a union of images of the swept solid for different positions.

RELATIONS

A binary relation is a subset of the set M^2=MXM. It can be defined as

$$Si : MXM \rightarrow I \tag{18}$$

The examples of binary relations are inclusion, point membership, interference or collision.

Collision detection

Collision detection is a complicated problem solved in various computer programs. This means that for each animation frame, one should test whether any two or more objects collided.

The ideal case is collision detection of any complexity between two arbitrary objects in the minimal time. Since the control of collisions between all pairs of objects is a resource-consuming process, such tests are usually done only for part of objects. The detection algorithm can be simplified prior to testing the presence of the given point (belonging to one of the objects), e.g., inside the cube confining the second object. The problem of simulating the behavior of interacting bodies having irregular shape arises in some applications such as dynamics of body collisions and celestial mechanics, molecular dynamics, graphics simulations for the problem of nano-assembly automation and its application in medicine using collective robotics, computer games and haptic interactions.

Particularly in calculating motions of many objects that move under changing constraints and frequently make collisions, one of the key issues of dynamic simulation methods is calculation of collision impulse between rigid bodies. A fast algorithm for calculating contact force with friction by formulating the relation between force and relative acceleration as a linear complementary problem was equally demonstrated and this model was based on solving the linear complementary problem [8]. Baraff's algorithm has achieved great performance for real-time and interactive simulation of two-dimensional mechanisms with contact force, friction force and collision impulse, although friction impulse at collision was not completely covered in such a model. In geometric haptic rendering models, collision detection is not trivial to compute. One of the most popular collision detection algorithms in geometric haptic rendering is H-Collide [9]. It uses a hybrid hierarchy of spatial subdivision and OBB trees. The simplest algorithms for collision detection are based upon using bounding volumes and spatial decomposition techniques. Examples of bounding volumes include bounding spheres, bounding boxes, convex polyhedrons. Examples of bounding boxes include axis-aligned bounding boxes and oriented bounding boxes. In work [10] authors used a bounding sphere hierarchy to detect collisions. Spatial decomposition techniques based on subdivision are used to solve the interference problem. Recursive subdivision is robust but computationally expensive. In particular, Hahn [11] used a subdivision based collision detection algorithm.

For curved objects, Herzen and etc. [12] have described an algorithm based on subdivision technique. A similar method using interval arithmetic and subdivision has been presented for collision detection by Duff [13]. However, for commonly used spline patches computing and representing the implicit representations is computationally expensive [14]. In [15] Pentland and Williams used implicit functions to represent shape and the property of the "inside-outside" functions for collision detection. But this algorithm has a drawback in terms of robustness, as it uses point samples. Thus, the most popular collision detection algorithms are extremely distance and extremely points (Barraf: four nonlinear equations solving), testing sample points (Pentland, Gascuel: accuracy of sampling using huge memory), interval methods (Duff, Snyder: interval bounds on the output of functions with time-consuming). The detailed explanation of main problems is described in [13]. The main problems are procedurally defined functions, time-dependent surfaces and surfaces of high complexity. Surgery simulation and entertainment technology require fast deformable models and efficient collision handling techniques. Efficient collision detection algorithms are accelerated by spatial data structures, bounding volume hierarchies, distance fields and etc. Such data structures are commonly built in a pre-processing stage. But pre-processed data structures are less efficient for deforming objects. Collision detection for deformable objects introduces additional challenging problems.

As a result of work of the known algorithm of collision detection of functionally defined object, the collision is not always detected and, moreover, detection of different collisions requires a greatly different time.

We propose another way of collision detection without using any bounding volumes around each object and pre-processing stage. For objects based on perturbation functions the object collision is detected in a constant time for collisions of different complexity, and the detection of events is absolutely ensured. This way of collision detection is based on the relation of object intersections, function representation with perturbation functions and on the recursive object space subdivision for search the contact point of the objects.

CONCLUSIONS

The freeform representations created by means of the scalar and analytical perturbation functions have the following advantages: fewer surface for mapping curvilinear objects, short database description, fewer operations for geometric transformations and data transfer, simple animation and deformation of objects and surfaces, and a wide spectrum of applications (interactive graphics systems for visualizing function-based objects, CAD 3D simulation systems, 3D web visualization, etc.). We have investigated various geometric operations on functionally defined objects on the basis of the perturbation functions. We have analyzed the collision detection algorithm by means of recursive object space subdivision. We may conclude that in the proposed function-based object collision detection algorithm, the collision is always detected and does not depend on the relative position of collided objects and parts of their surfaces, i.e., such an algorithm guarantees detection of the event, which has been proved both experimentally and theoretically, it is required to have equal number of levels of object space subdivision and, therefore, equal computation time. The object collision was detected in a constant time for collisions of different complexity and the time spread in the tests was below 1% of the given time.

References

1. COMPUTER AIDED GEOMETRIC DESIGN/ Academic Press, edited by Robert E. Barnhill and Richard F. Riesenfeld, New York, San Francisco, London, 1974, Proceedings of a Conference Held at The University of Utah, Salt Lake City, Utah March 18-21, 1974.

2. Blinn J.F. A generation of algebraic surface drawing. ACM Transactions on Graphics, 1(3): 235-256, July 1982.

3. Sherstuyk A. Convolution Surfaces in Computer Graphics. Doctoral dissertation, Monash University, School of CSSE, 1998.

4. Matej S. “Practical considerations for 3-D image reconstruction using spherically symmetric volume elements,” IEEE Trans. On Medical Imaging, vol.15. 1996.

5. Vyatkin S.I., Dolgovesov B.S., Yesin A.V., Zhigach A.A., Chizhik S.E., and Shcherbakov R.A. Geometric Modeling And Visualization Of Functionally Defined Objects, Computer Graphics and Geometry, Vol. 3, No. 2, 2001 article http://www.cgg.ru/febr/vjat/pivweb.html#a

6. Poznyak E. and Shikin E., Differential Geometry MGU. Moscow, 1990.

7. Pasko A.A., Adzhiev V.D., Sourin, A.I. et al. Function representation in geometric modeling: concepts, implementation and applications // The Visual Computer.1995. 11(6). P. 429.

8. Baraff D., “Fast contact force computation for nonpenetrating rigid bodies”, in Computer Graphics Proceedings, Annual Conf. Series. ACM SIGGRAPH, pp. 23-34, 1994.

9. Gregory A., Lin M., et al. "H-Collide: A Framework for Fast and Accurate Collision Detection for Haptic Interaction". IEEE Virtual Reality, 1999.

10. Ruspini D.C., Kolarov K. and Knatib O. "The haptic display of complex graphical environment". Proceedings of SIGGRAPH 97, vol. 1, pp. 295-301, August 1997.

11. Hahn J.K. “Realistic animation of rigid bodies”. ACM Computer Graphics, 22(4):pp. 299-308, 1988.

12. Herzen B.V., Barr A.H., and Zatz H.R. “Geometric collisions for time-dependent parametric surfaces”. ACM Computer Graphics, 24(4), August 1990.

13. Duff T. “Interval arithmetic and recursive subdivision for implicit functions and constructive solid geometry”. ACM Computer Graphics, 26(2): pp. 131-139, 1992.

14. Hoffmann C. “Geometric and solid modeling”. Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1989.

15. Pentland A. and Williams J. “Good vibrations: modal dynamics for graphics and animation”. ACM Computer Graphics, 23(3): pp. 185-192, 1990.

Література

1. Комп'ютерний геометричний дизайн / Під редакцією Барнхіл Р.Е. і Різенфелд Р.Ф., Нью -Йорк, Сан-Франциско, Лондон, 1974. Праці конференції, що відбулася в Університеті штату Юта, Солт-Лейк-Сіті, штат Юта 18-21 березня, 1974.

2. Блін Дж.Ф. Генерація алгебраїчно заданих поверхонь. ACM. Графічні перетворення, 1 (3): 235-256, липень 1982 року.

3. Шерстюк А. Згортки поверхні в комп'ютерній графіці. Докторська дисертація, Університет Монаш, школа Котб 1998 року.

4. Матей С. Практичні аспекти 3-D реконструкції зображення з використанням сферично симетричних об'ємних елементів, IEEE Trans. Медична візуалізація, т. 15. 1996.

5. Вяткін С.І., Долговесов Б.С., Єсін А.В., Жігач А.А., Чижик С.Е., Щербаков Р.А. Геометричне моделювання та візуалізація функціонально заданих об'єктів. Комп'ютерна графіка та геометря, т. 3, № 2, 2001.

6. Позняк Е., Шикин Е.В. Дифференциальна геометрія: Видавництво МДУ, 1990. 384 с.

7. Пасько А.А., Аджиєв В.Д., Соурін А.І. та ін. Функції для геометричного моделювання: концепції, впровадження та застосування. Візуальний компьютер.1995. 11 (6). П. 429.

8. Барафф Д. Обчислення в комп'ютерній графіці швидкого контакту силою непроникних твердих тіл. Праці конференції. ACM SIGGRAPH. - С. 23-34, 1994.

9. Грегорі М. Лін та ін. "Н-коллайдер: основа для швидкого і точного визначення перетину для тактильної взаємодії». IEEE Віртуальна реальність, 1999.

10. Руспіні Д.С., Коларов К., і Хатіб О. Тактильна взаїмодія в складному графічному середовищі. Праці конференції SIGGRAPH 97, вип. 1, - С. 295-301, серпень 1997.

11. Хан Дж. К. "Реалістична анімація твердих тіл". ACM Комп'ютерна графіка, 22 (4): - С. 299-308, 1988.

12. Герцен Б.В., Хіджри Барр, і Затз Н.Р. Геометричні зіткнення для залежних від часу параметричних поверхонь. ACM Комп'ютерна графіка, 24 (4), серпень 1990.

13. Дафф Т. Інтервальна арифметика та рекурсивний поділ з використанням неявних функцій і конструктивної стереометрії". ACM Комп'ютерна графіка, 26 (2): - С. 131-139, 1992.

14. Хоффманн С. "Геометричне і твердотільне моделювання". Морган Кауфман Паблішерс, Inc, Сан -Матео , Каліфорнія, 1989.

15. Пентланд А. і Вільямс Дж. Повноцінні вібрації: модальна динаміка для графіки та анімації. ACM Комп'ютерна графіка, 23 (3). - С. 185-192, 1990.

Рецензія/Peer review : 5.1.2014 р. Надрукована/Printed : 26.3.2014 р.