

Interactive design and simulation of net sculptures

Grigore D. Pintilie¹, Peter Heppel², and Janet Echelman³

¹ Massachusetts Institute of Technology, Cambridge, MA

² Peter Heppel Associates

³ Echelman Inc.

Abstract. We present a graphical user interface that allows an artist to virtually design and visualize net sculptures. Net sculptures consist of net pieces that are seamlessly connected to each other and to fixed rails. They are flexible and hence dynamic under external forces such as gravity and wind. The interface that we describe allows an artist to create net sculptures made up of multiple net pieces. Simple operations such as clicking on points and click-and-drag gestures are used to create and modify individual net pieces, and drag-and-drop gestures are used to connect net pieces to multiple rails. The effect of gravity on the net sculpture is simulated, allowing the artist to simultaneously design and visualize net sculptures as they would appear once installed in a real setting.

Key words: 3D graphical user interfaces, net sculptures, mass-spring particle systems

1 Introduction

Net sculptures are composed of multiple net-pieces that are joined together and hung on suspended rails, producing aesthetic and dynamic 3D sculptures that move under the influence of wind and gravity [1]. The net pieces that make up a net sculpture must have dimensions that adhere to fabrication protocols, so that they can be produced by available equipment. Net pieces are connected to each other seamlessly, either by having the same number of knots along the connecting edge, or numbers of knots that are multiples of each other (which are known as gear changes). Creating and working with net pieces in the real world can be time-consuming and expensive, and thus experimenting with complex designs is prohibitive, motivating the need for computer-aided design methods which would allow an artist to virtually design and visualize net sculptures.

2 Overview and Related Work

Net sculptures can be modeled using 3-dimensional surfaces. Hence, an artist could use one of the many available computer-aided design tools to design a

net sculpture, for example using parametric surfaces [2]. However, such surfaces, which could have any shape imaginable, cannot easily be decomposed into individual net pieces that can be fabricated in order to produce the net sculpture. Moreover, gravity plays a very significant role in the final shape of the net sculpture, and so without simulation to take this effect into account, a parametric surface designed by the artist may not have the same shape when produced and hung. A more effective interface would thus have to combine the particular characteristics of net pieces and realistically simulate the net sculpture as a physical object.

Since net sculptures have 3D geometries, the interface must also display and allow the artist to interact with 3D objects. There are many challenges in the development of effective 3D user interfaces, such as ensuring proper spatial orientation and interaction with objects [3,4]. Commonly available 3D design tools give a large degree of freedom over the results; however they can have steep learning curves, and an artist can easily get lost in the complexity. By combining predictable physically-based object behaviour and simple gestures, easy to use and fun 3D interfaces have been achieved [5]. In the same spirit, we present an interface that allows an artist to create and modify net sculptures while at the same time simulating their physical behaviour under the effect of gravity.

The interface does not require the user to explicitly position objects in 3D, which can be a frustrating experience when using a 2D input device such as a mouse [6]. Instead, net pieces are automatically connected either to pre-defined rails or to other net pieces, and their geometry is automatically generated. For other operations, with which the user adjusts dimensions of a net piece for example, we use 2D mechanisms. More specifically, such a mechanism involves clicking on a point with the mouse, followed by a dragging gesture (moving the mouse while the button is still pressed). During the click-and-drag gesture, the movement of the mouse is not related to positioning in 3D space, but rather to increasing or decreasing a parameter value, the effect of which is visualized in real-time.

In connecting a free-hanging side of a net piece to a rail, the user does actually move a point through 3D space. To make this movement correspond to the movement of the mouse, the point moves on a plane that is perpendicular to the viewing direction [7]. However this plane on which the point is being moved does not actually matter. Instead, if the point being moved is dropped while the mouse pointer is on top of a rail to which the net can be attached, the attachment is made. *On top of* is taken loosely to mean that the 2D position of the mouse is roughly the same as the projected 2D position of a point on the rail. This operation thus follows the typical drag-and-drop gesture that many 2D interfaces already use. The only inherently 3D operation that the artist has to perform while using the interface is to position the camera in 3D space, so as to view the net sculpture from different angles.

In the rest of the paper, we first describe the geometry of rails that net pieces can be attached to, the operations that an artist can perform to create

and modify net sculptures, and finally the simulation of net sculptures to take into account the effects of gravity.

3 Rails

Rails can be either created by the artist, or pre-specified for a particular installation by the architectural team. There are no particular constraints in the creation of rails; they are commonly fabricated as metal tubes, and can take the form of any open or closed curve. The geometry of such rails can be created for example using B-splines by many CAD programs. These curves can be read into the net design interface either as B-splines or as continuous line segments. Two example rails that are imported into the interface are shown in Figure 1.

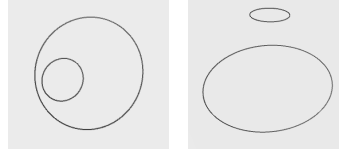


Fig. 1. Two rails are shown, each rail having the form of a closed curve. The two rails are shown from top view (left), and side view (right).

4 Net piece geometry

The geometry of a net piece is illustrated in Figure 2. It is made up of knots arranged in rows and columns, with each row staggered with respect to the previous row. Each row has the same number of knots as all other rows, and similarly, each column has the same number of knots as all other columns. As shown in Figure 2, each knot is connected by links to two knots in the row above (except for the knots in the first row), and to two knots in the row below (except for knots in the last row). These links are commonly called *bars*, and we will use this term through the rest of the paper.

Topologically a net piece is rectangular, with the same number of knots in each column, and the same number of knots in each row. However the bar lengths can be varied from row to row or from column to column (but not both), so that the net piece can also have a trapezoidal geometry. The number rows and columns in a net piece, along with the bar lengths, are the only parameters required to produce it using a net fabrication machine. Hence, these are also the parameters that an artist can control for each net piece when designing a net sculpture.

In determining knot positions, the Pythagorean relationship is used:

$$l^2 = (s/2)^2 + h^2 \quad (1)$$

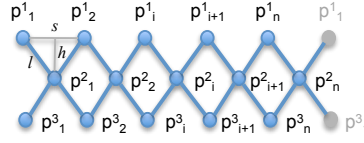


Fig. 2. A net piece is composed of knots connected by bars. The position of a knot is denoted by p^a_b , where a is the row-index and b is the column-index of the knot. The scalar s is the distance between two adjacent knots in the same row, h is the vertical distance between knots in adjacent rows, and l is the length of a bar attaching two knots. The right-most knots of this net piece link back to the left-most knots, forming a closed tube in this example.

In Equation (1), l is the bar length, s is the spacing between adjacent knots in a row and h is the vertical distance between knots in adjacent rows. The user will be allowed to adjust s and l , from which h will be calculated. While s will be kept the same when computing the initial geometry of a net piece, the bar length l can vary smoothly either row-wise or column-wise, but not both, and thus h may also vary. The artist will be allowed to control how the bar length l varies throughout a net piece, as will be described in Section 6.

5 Creation of new net pieces

New net pieces can be created in two ways: by directly attaching them to a rail, or by attaching them to an existing net piece. Both ways are illustrated in Figure 3. To create a new net piece, the user puts the interface in the *new net* mode. In this mode, spheres are placed along rails or sides of existing net pieces, where new net pieces can be attached. The user then simply clicks on one of these spheres to create a new net piece.

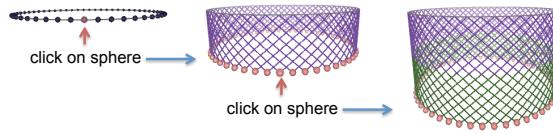


Fig. 3. Creation of net pieces by attachment to rails (left and middle images), or to existing net pieces (middle and right images). On the rail, the nodes on the top edge of the new net piece are attached to equally spaced points along the rail. On the existing net piece, the nodes on the top edge of the new net piece are attached to nodes on the bottom edge of the existing net piece.

5.1 Attaching a new net piece to a rail

In the *new net* mode, points on a rail are generated at equidistant intervals, with distance measured in parameter space along the curve. These represent points that the nodes in the first row of a new piece can be attached to. In Figure 3, in the left-most image, these points are shown with solid spheres. To create a new net piece that attaches to these points, the user simply moves the mouse over any one of the spheres, and clicks. As the user moves the mouse over a sphere, a transparent red sphere is drawn on top of the original sphere, to let the user know that that sphere is selected. If the user clicks while a sphere is selected, a new net piece is created and attached to the rail, as shown in the middle image in Figure 3. The new net piece has the name number of knots in the width direction as points generated on the rail. Because the rail is a closed curve in this case, the net piece itself is also closed, with the first and last column attached to each other.

When creating a net piece by attaching it to a rail, the positions of the knots in the first row are set to the positions of the points generated at equal intervals along the rail. The positions of the knots in subsequent rows are computed based on these positions and the scalars s , h , and l , which are related as given in Equation (1). In particular, we assume that s and l are given (and controlled by the user, as will be described in Section 6), and h will be calculated. Moreover, we assume that the rails are given on the x - y plane, and the net will hang down in the z direction. Thus, the position of a node in a net piece is:

$$\vec{p}_b^a = \begin{cases} \frac{\vec{p}_b^{a-1} + \vec{p}_{b+1}^{a-1}}{2} - h\hat{z}, & \text{if } b \text{ is odd} \\ \frac{\vec{p}_{b-1}^{a-1} + \vec{p}_b^{a-1}}{2} - h\hat{z}, & \text{if } b \text{ is even} \end{cases} \quad (2)$$

In Equation (2), a is the column-index and b is the row-index of the knot at position \vec{p}_b^a . (The indexes are automatically wrapped around the first and last point in each column).

The positions that are computed for the knots in a new net piece using Equation (2) are not necessarily the final positions of the knots after taking into account the force of gravity. The final hanging shape of all the net pieces in a net sculpture will be computed using the dynamical process described in Section 7. Equation (2) is however very useful in attaining a reasonable starting geometry for the net.

5.2 Attaching a new net piece to an existing net piece

In the *new net* mode, solid spheres are also drawn on the knots on the last row of a net piece to which a new net piece can be attached. For example, the middle image in Figure 3 shows such red spheres at the bottom of the net piece. Similar to the spheres shown on a rail, the user can click on these spheres to attach a new net piece to the last row of the existing net piece. When creating a new net piece by attaching it to an existing net piece, the positions of the knots in the first row of the new net piece are simply set to the positions of the knots in the

last row of the existing net piece. The positions of all the other knots are then calculated via Equation (2).

5.3 Splitting a net piece into multiple net pieces

A net piece can be divided into multiple net pieces, as shown in Figure 4. The resulting net pieces will have the same number of rows as the original net piece, but the number of columns in each net piece may be different. To divide a net piece, the user first puts the interface in the *divide net piece* mode. In this mode, spheres are placed on the top nodes of every net piece. When the user clicks on one of these spheres, the net is divided at that column. This operation is useful for various reasons. One reason is that the artist may want to use different bar lengths in each net piece, or may want to use net pieces with different colors, material properties, or bar thickness.

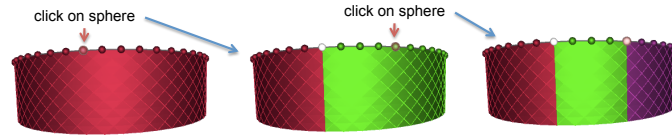


Fig. 4. Splitting a net piece into multiple net pieces. In this drawing, the net piece is drawn using both lines and triangles, for a more clear visualization of each net piece. In the image on the left, the user clicks on the sphere at the top of the column along which the net piece should be split, resulting in the net piece being divided into two at this column (middle image). The two resulting net pieces are now attached to each other column-wise, i.e. the knots on the last column of the net piece on the left are attached to the knots in the first column on the piece on the right. Another split is created by clicking on another sphere in the middle image (right-most image).

6 Changing net-piece dimensions

The interface allows the artist to modify the number of rows and columns of each net piece, as well as bar lengths, as described in the following sections.

6.1 Changing the number of rows in a net piece

To change the number of rows of knots in a net piece, the user first puts the interface in the *change number of rows* mode. In this mode, spheres are shown at knots in the last row of all net pieces, as shown in Figure 5. The user can then click on one of these spheres and drag it up to decrease the number of rows or down to increase the number of rows. The reason for this mapping is that when dragging up, the net piece appears to shorten, and thus the bottom row of the net seems to follow the direction of the mouse gesture.

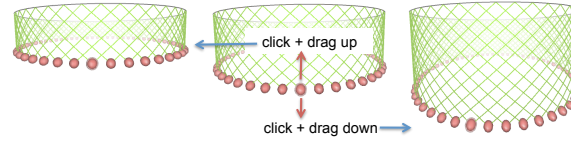


Fig. 5. The number of rows in the illustrated net piece is changed by clicking and dragging one of the spheres placed on the last row of the net piece. Dragging up decreases the number of rows, while dragging down increases the number of rows.

6.2 Changing the number of columns in a net piece

To change the number of columns in a net piece, the user first puts the interface in the *change number of columns* mode. In this mode, for a net piece attached to a rail, as shown in Figure 6, spheres are drawn at all the points where the knots in the first row of a net piece are attached to the rail. The user can then click on one of these spheres and drag up or down to change the number of columns in the net piece.

When the user drags up, the spacing between the attachment points is decreased, meaning that the number of columns increases, as more points are required to circumnavigate the entire rail. If the user drags down, the spacing between the attachment points is increased, meaning the number of columns decreases as fewer points are required to cover the entire rail. Note that the number of rows in the net piece stays constant during this operation; however, because the bar length also stays constant, the distance between knots in adjacent rows changes, and therefore the actual length of the net piece may appear to increase and decrease, as can be seen in Figure 6.

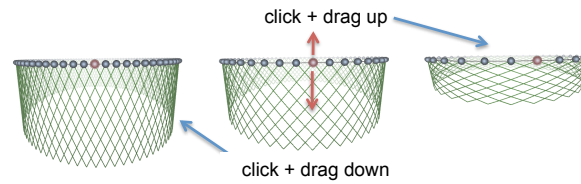


Fig. 6. Changing the number of columns in a net piece attached to a rail. The user clicks on a sphere and drags up to increase the spacing between attachment points, or drags down to decrease the spacing. Because the number of attachment points changes, the number of columns in the net piece also changes. Because the bar lengths and number of rows stay the same, the physical length of the net piece also increases or decreases as shown.

6.3 Gear changes

To change the number of columns in a net piece connected to another net piece, the same mode and gestures are used. However, the number of columns in such a net piece is constrained, because it must seamlessly connect to the net piece above. This operation is also commonly called *gear changing*. The number of columns in a net piece connected to another net piece can be either double (gear-up of 2), triple (gear-up of 3), half (gear-down of 2), a third (gear-down of 3) of the number of columns in the net piece it is connected to. These gear changes are illustrated in Figure 7.

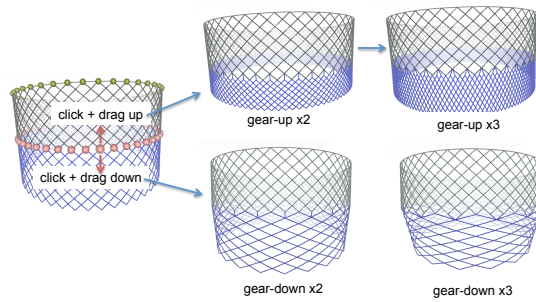


Fig. 7. Changing the number of columns in a net piece which is attached to another net piece, through gear changes. Gear-up changes are obtained by clicking on a sphere and dragging up, which doubles or triples the number of columns in the net piece (top row, right). Gear-down changes are obtained by clicking on a sphere and dragging down, which halves or thirds the number of columns in the net piece (bottom row, right).

6.4 Changing the bar length of a net piece

The bar length of a net piece can also be adjusted by the artist. To do so, the artist first puts the interface in *change bar length* mode. In this mode, spheres are drawn on the knots in the first column of a net piece, as shown in Figure 8. The artist can click on one of these spheres and drag up to decrease the bar length, or down to increase it. For a single net piece, the bar length is the same in all of its columns. Note that the number of rows and columns in the net pieces are not changing during this operation, even though the actual physical length is changing, due to smaller or larger bar lengths.

When two or more net pieces are attached at their first and last columns, the bar lengths are the same at each column where net pieces are connected. However, the bar lengths are linearly interpolated between the first and last column in the net-piece, as shown in Figure 9. A varying bar length results in a varying physical length of the net piece at each column, as shown in Figure 9.

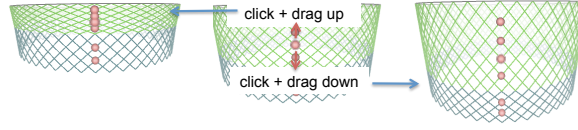


Fig. 8. Changing the bar length of a single net piece. The user clicks a sphere which is drawn at knots in the first column of a net piece, and drags up/down to decrease/increase the bar length. In this example, the net piece is not connected to other net pieces column-wise, so the bar length is the same in every column. A net piece attached to the last row of the net piece is updated as well to maintain that connection.

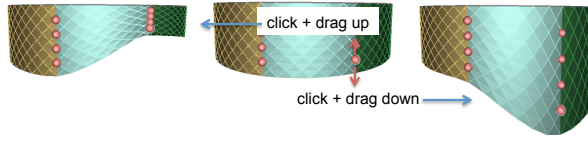


Fig. 9. The bar length at a column of a net piece is modified by clicking on a sphere drawn on this column, and dragging up to decrease the bar length or down to increase it. For each net piece, the bar length is linearly interpolated between its values at the first and last column. The bar lengths at the columns where two net pieces are attached are the same for both net pieces.

7 Dynamics of net sculptures

The geometry created for each net piece, as described in previous sections, serves only as an initial configuration for a net sculpture. It is also desirable to the artist to see how the entire net sculpture hangs under the pull of gravity. In order to simulate this hanging shape, we model each net piece in the sculpture as a mass-spring system, where the knots are masses and the bars between them are springs. Such dynamical model has been used in various applications, such as cloth modeling [9,10] and surgery simulation [11]. Cloth has different material properties, and here we try to adapt the force equations to better capture the properties of nets. The force acting on each knot can be written as:

$$\vec{F}_j = \begin{cases} \vec{g} + \sum_{i=1}^n k_{i,j}^s (l_{i,j} - l_{i,j}^{eq}) \frac{\vec{p}_i - \vec{p}_j}{|\vec{p}_i - \vec{p}_j|} & \text{if } l_{i,j} > l_{i,j}^{eq} \\ \vec{g} + \sum_{i=1}^n k_{i,j}^c (l_{i,j} - l_{i,j}^{eq}) \frac{\vec{p}_i - \vec{p}_j}{|\vec{p}_i - \vec{p}_j|} & \text{if } l_{i,j} \leq l_{i,j}^{eq} \end{cases} \quad (3)$$

Equation (3) expresses the force on a knot j , \vec{F}_j , as a combination of the force of gravity, \vec{g} , which points down, and the springs attaching it to n other knots. The scalar $k_{i,j}^s$ is the stretching spring constant for the bar attaching knots i and j , and $k_{i,j}^c$ is the compression spring constant for the same bar. A piece of rope is very resistant to stretching, but not very resistant to compression, since

in the latter case it simply bends; thus we set $k_{i,j}^s$ to be relatively high and $k_{i,j}^c$ to be relatively low.

In Eqn. 3, $l_{i,j}$ is the actual length of the bar attaching knots i and j during the simulation. The equilibrium length for the same bar is $l_{i,j}^{eq}$, which stays constant and is set when the positions of the knots are calculated as previously described. Lastly, \vec{p}_i and \vec{p}_j are the positions of the knots i and j , which are initialized using Eqn. (2), and which change during the simulation. To update the positions for each knot, Eqn. (2) is written into two ordinary differential equations, relating the acceleration, velocity, and position of every knot in the net sculpture:

$$\vec{a}_j = \frac{d\vec{v}_j}{dt} \quad (4)$$

$$\vec{v}_j = \frac{d\vec{x}_j}{dt} \quad (5)$$

In Equations 4 and 5, \vec{a}_j is the acceleration that knot j experiences due to the forces on it, \vec{v}_j is the velocity of a knot, and $d\vec{x}_j$ is the position of the knot. The acceleration of a knot is related to the force acting on it, as dictated by Newton's law:

$$\vec{a}_j = \frac{\vec{F}_j}{m_j} \quad (6)$$

In Eqn. 6, m_j is the mass of the knot j . For design purposes, we would simply like to find the equilibrated hanging shape of a net sculpture, ignoring any oscillatory motion that is typical of mass-spring systems. Hence we assume an infinitely damped system, free of inertia, in which the velocity on a knot is simply the force acting on it:

$$\frac{d\vec{x}_j}{dt} = \frac{\vec{F}_j}{m_j} \quad (7)$$

To find the equilibrium hanging shape of a net sculpture, Eqn. 7 is discretized and integrated using the Runge-Kutta method [12], with the initial positions of each knot being the positions computed as previously described. Net sculptures are inherently stiff systems, since the spring constants should be high and the masses relatively low, and thus stability is an issue. We use relatively low spring constants, large masses, and small time steps, making the simulation stable during the design process. Once design is complete, the spring constants are gradually increased to achieve a more accurate hanging shape. Figure 10 shows the initial and equilibrium hanging shape of a single net piece hung on a circular rail.

8 Attaching net pieces to multiple rails

To design a more complex net sculpture, an artist often uses net pieces that connect to each other and also to more than one rail. To facilitate this within

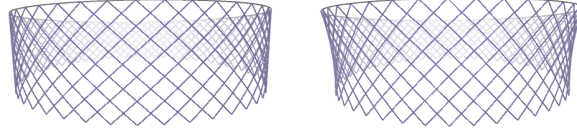


Fig. 10. A single net piece attached to a circular rail, with knots in the initial computed positions (left), and after the equilibrium hanging shape is reached (right).

the user interface, we employ a drag-and-drop mechanism, which is illustrated in Figure 11. To perform this operation, the user first puts the interface in the *connect net* mode. In this mode, spheres are drawn on the last rows of net pieces which can be attached to a rail. The user clicks on one of these spheres and starts dragging it. While dragging, spheres are drawn on rails to which the net piece can be attached. To complete the attachment, the user drops the sphere they are dragging on top of any of the spheres placed on the rails they wish to attach the net piece to. *On top of* means simply that on the 2D projected image, the user has moved the mouse so as to select any of the spheres drawn on the rail on which they are attaching the net piece to.

After this operation is performed, all the knots on the last row of the net piece which is being attached are automatically fixed to points on the rail to which the net piece is being attached. From this configuration, the mass-spring simulation method is used to bring the net to its hanging shape.

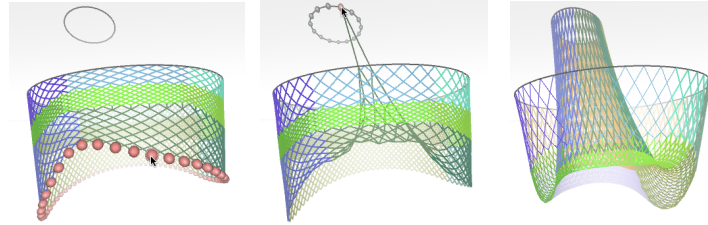


Fig. 11. The free-hanging side of a net sculpture (left), is attached to another rail (middle). A point on the free-hanging side is clicked, dragged, and dropped when the mouse pointer coincides with a point on the rail that it should be attached to. The final hanging net geometry is then shown (right).

9 Conclusions and future work

We presented a graphical user interface that allows an artist to virtually design net sculptures, which are composed of multiple net pieces that seamlessly connect to one another and to supporting rails. The artist can create and modify net

pieces by clicking on points and using simple click-and-drag gestures. Moreover net pieces can be connected to other rails by dragging and dropping a point on the end of a net to the rail that it should be attached to. In preliminary trials, it was apparent that such an interface has great potential in allowing an artist to virtually design realizable net sculptures. In the future, we aim to further explore its effectiveness in allowing an artist to fully exercise their artistic and creative visions.

10 Acknowledgments

We would like to thank Echelman Inc. for commissioning and funding this work. The pictures shown in the paper were created using the proprietary JNet software, ©Echelman Inc, which implements the methods presented.

References

1. <http://www.echelman.com/>
2. Thingvold, J.A., Cohen, E.: Physical modeling with B-spline surfaces for interactive design and animation. Proceedings of the 1990 symposium on Interactive 3D graphics. pp. 129-137 ACM, Snowbird, Utah, United States (1990).
3. Brooks, F.P.: Grasping reality through illusion; interactive graphics serving science. Proceedings of the SIGCHI conference on Human factors in computing systems. pp. 1-11 ACM, Washington, D.C., United States (1988).
4. Coquillart, S., Frhlich, B., Hirose, M., Kitamura, Y., Kiyokawa, K., Strzlinger, W., Bowman, D.: 3D User Interfaces: New Directions and Perspectives. IEEE Computer Graphics & Applications. 28, 20-36 (2008).
5. Agarawala, A., Balakrishnan, R.: Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. Proceedings of the SIGCHI conference on Human Factors in computing systems. pp. 1283-1292 ACM, Montral, Qubec, Canada (2006).
6. Hinckley, K., Pausch, R., Goble, J.C., Kassell, N.F.: A survey of design issues in spatial input. Proceedings of the 7th annual ACM symposium on User interface software and technology. pp. 213-222 ACM, Marina del Rey, California, United States (1994).
7. Strauss, P.S., Issacs, P., Shrag, J.: The design and implementation of direct manipulation in 3D. SIGGRAPH 2002 Course Notes. (2002).
8. Oh, J., Stuerzlinger, W.: Moving objects with 2D input devices in CAD systems and Desktop Virtual Environments. Proceedings of Graphics Interface 2005. pp. 195-202 Canadian Human-Computer Communications Society, Victoria, British Columbia (2005).
9. Ji, F., Li, R., Qiu, Y.: Three-dimensional Garment Simulation Based on a Mass-Spring System. Textile Research Journal. 76, 12-17 (2006).
10. Baraff, D., Witkin, A.: Large steps in cloth simulation. Proceedings of the 25th annual conference on Computer graphics and interactive techniques. pp. 43-54 ACM (1998).
11. Liu, A., Tendick, F., Cleary, K., Kaufmann, C.: A Survey of Surgical Simulation: Applications, Technology, and Education. Presence: Teleoperators & Virtual Environments. 12, 599-614 (2003).
12. Butcher, J.: Numerical Methods for Ordinary Differential Equations. John Wiley & Sons, Ltd, Chichester, UK (2003).