

# Practical Issues in a Virtual Sculptor System for Rapid Modeling

<sup>†</sup>*Wai-Seng Ng*, <sup>‡</sup>*Chien-Chang Ho*, <sup>‡</sup>*Ming-Yang Yu*

<sup>‡</sup>*Kai-Hsuan Chen*, <sup>‡</sup>*Bo-Ruei Chen*, <sup>‡</sup>*Kuan-Ting Liu*, <sup>‡</sup>*Ming Ouhyoung*

<sup>†</sup>Communication and Multimedia Laboratory, National Taiwan University

E-mail: markng@cmlab.csie.ntu.edu.tw

## ABSTRACT

We present a real-time interactive modeling system for users to create sculpture in a virtual world. We find several practical issues of virtual sculpting and propose solutions to address them in this paper. Our previous work, Cubical Marching Squares (CMS), provides a relatively accurate method to generate surface from volume data. Also, another previous work provides a volumetric data structure which preserves information of sharp features after boolean operation is applied. By combining the data structure for boolean operation and CMS to display the data, it is possible to create sculpture in a natural way in a virtual world. However, an artist friendly system for sculpting requires continuous operation to mimic the real world sculpting experiences. Also, the system must provide an intuitive way for artists to operate, and so these requirements raise four issues to be addressed. First, a motion graver is created to overcome the zigzag artifacts as shown in Figure 1, which is caused by discretely sampled sculpting movements. Second, to perform continuous boolean operation at interactive rate is another second issue. Third, a force feedback device is used to provide touchable hints during operations. Finally, some basic Computer Numerical Control (CNC) like operations are provided for more precise control over the sculpting operation.

## 1. Introduction

Before computer graphics was invented, people make "three-dimensional models" by sculpting without computers in ancient times. On the other hand, in modern times, to produce a three-dimensional model using computers, we need different kinds of specialties and professional software to make a model. The require-

ments are the same if we want to customize a three-dimensional model. Thus, creating or modifying a three-dimensional model using today's computer techniques is not an easy task for everyone, especially for rapid modeling or prototyping. Furthermore, even for a well trained modeler, sometimes they want to create a rough model in an easy way before editing it in professional three-dimensional modeling software. Thus, sculpting may be the right answer to create models or for modeler to take a shortcut.

To perform sculpting in a virtual world, volumetric techniques [8, 9, 11] are usually used to provide a smooth and efficient sculpting environment. Besides efficiency, the most important requirement of sculptures is sharp feature, which we need to preserve. Our virtual sculptor system is based on our previous work [6] that preserves sharp features by using the CMS algorithm proposed in [7]. Comparing to original marching cubes algorithm, sharp features preserving algorithms provide better appearances if the shapes of volume data contain edges or corners. To simulate the real world sculpting experiences successfully, there are several issues that need to be taken care of. The first issue is the motion artifacts. We assume our system is able to perform continuous sculpting operation. Although, CMS preserves sharp feature, however, sharp features cause another problem, motion artifacts. As shown in Figure 1(a-c), since the graver's motion is discretely sampled during operation, in combination of sharp features, the zigzag artifacts become more serious.

The second issue is performance. The extension of our work is to speed up our previous system [6]. One of the contributions of our system is to make it faster and the user can use the system interactively. The previous system is recalculated for all the cells each time when performing the Boolean operation. It is fast enough in

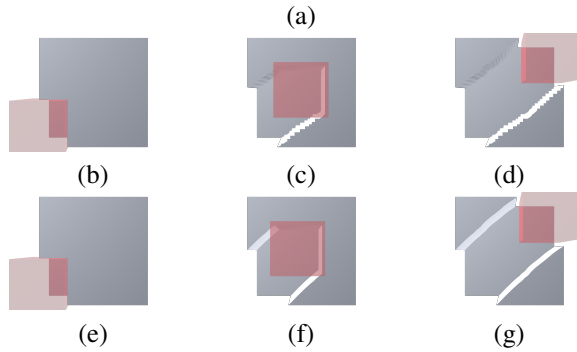


Figure 1: This figure shows the Virtual Sculptor System and the zigzag artifacts appeared while performing continuous carving. (a) shows the user uses a force feedback device to interact with the system. The top row (b-d) shows the result of the zigzag problem while sculpting. The second row (e-g) shows the result of the proposed method. The problem mainly happens when the graver is moving and the motion is discretely sampled. The artifact becomes more serious if the graver is not a smooth shape object, such as when it contains corners or edges. When the user uses a non-round shape graver for carving, the translation and rotation operation will be performed at the same time with different speed. Each carving operation contributes a sharp mark, part of the zigzag artifact, along the graver's moving path. This behavior creates sampling problem to generate "zigzag shape". The proposed method will use a motion graver, which reconstructs a new graver each time in response to each motion, and therefore can easily remove the zigzag problem and provides a smooth and nature result.

single shot basis. However, while performing continuous carving, it causes a delay time for the user. We also take advantage of multi-threads system on a multi-core PC to speed up the whole computation.

The third issue is a friendly user interface for sculpting. A primary input device, PHANToM is integrated into this system as our previous work [12]. The main feature of the controller is provided six degree of freedom haptic device, which allows user to move the graver quickly and accurately in three-dimensional space. Thus, sculpturing the model can be more easily through the haptic controller. Another feature of PHANToM is force reactions, and it provides a realistic and quick force feedback to the user.

The last issue is convenience and automation. Some basic Computer Numerical Control (CNC) like operation is provided to assist the user to perform sculpting operation more precisely. The assisted operations related to controlling the graver include "attach to surface" and "attach to axis". Attach operations help user to perform particular tasks, such as engraving character or drilling holes. Another assisted operation spins the object to create a pottery wheel like environment.

To sum up, although sharp feature preserving algorithms preserve the characteristic of sculptures, previous solutions still suffer from zigzag artifacts, and low performance while performing continuous operations. Also, force feedback and assisted operations help a user to perform sculpting more easily and more precisely. In this paper, we address these four practical issues and provide solutions to create a practical virtual sculptor system.

## 2. Related work

There are two main approaches for sculpturing. The first one is volumetric approach. Marching cubes algorithm [11] is first proposed to provide a volume data structure to represent object and Galyean et al. [5] applies it to be a volumetric modeling technique. Although Ferley et al. [3] proposed an adaptive volume sculpting system, the original marching cubes algorithm will not preserve sharp features and will not maintain topology consistency. By using more information for the volume data structure, extended marching cubes(EMC) [9] algorithm and Dual contouring(DC) [8] method is proposed to preserved more details. Perng et al. [13] proposed a real-time volume sculpting us-

ing EMC algorithm. However, topology consistency is an essential issue of a mesh. Ho et al. [7] proposed a cubical marching squares(CMS) algorithm to deal with topology consistency and a novel data structure which can greatly reduce memory consumption [6]. The second approach is adaptively sampled distance fields. Frisken et al. [4] propose a system with adaptively sampled distance fields which can preserved fine details and Perry et al. [14] proposed a system for sculpting digital characters.

There are several frameworks and application systems for sculpting that have been proposed in recent years [1, 2, 15]. They are integrated to numerous famous 3D software packages such as 3D Studio Max, Maya, and ZBrush. These professional packages have many powerful features. Professionals may feel intuitive to operate these professional software, but it may not easy for beginners. Our goal is to provide an easy to use modeling environment for everyone, especially for those who are not familiar with three-dimensional mesh modeling software.

### 3. The Virtual Sculptor System

Our proposed system pipeline is as follows. First of all, user can import his/her own model or creates model directly with objects provided from our system. Secondly, our system captures the information from the PHAN-ToM device, while a Wii remote controller is held in another hand for motion parallax display. After all, the system can create a motion graver and directly use it as an input to the Cubical Marching Squares framework. In this process, our system calculates the force at the same time. Finally, the output is shown in front of the user. The solutions of the four issues mentioned previously are described in the rest part of this section.

#### 3.1. Motion Artifacts

The artifacts created from traditional virtual sculptor is addressed in this section. The zigzag edges shows in Figure 1 and Figure 2(b) can be a big problem when an artist uses a box as the graver to perform carving. We restricted the graver to be convex in three dimensions. And this restriction is not a serious problem because most of the traditional gravers in real world are point, flat bottom, round, knife and liner shapes.

The solution we found is that the sampling problem

can be translated to a convex hull problem. The three-dimensional convex hull is a minimal convex set containing a set of points in three dimensions. We use two samples of the graver operation, carving or stuffing, while sample is a set of points that forms a set and two sets are on different position. Two sets of points are directly processed by the convex hull algorithm at the same time [16]. The output of the set of points is the convex set of the two samples. Then we use this convex set to perform graver operation. The temporal result seems to be smooth at the edge between the graver and the sculpted object since a smooth convex hull is created and the sampling problem thus has disappeared. The algorithm for creating a motion graver is given in Algorithm 3.1.

---

**Algorithm 1 CreateMotionGraver.** *This procedure creates a linear approximated motion model of two successive samples (model A and B) of a graver.*

---

```

1: CreateMotionGraver (Model A, Model B)
2: for vertex  $v$  in  $A.vertex$  do
3:    $vertexlist \leftarrow vertexlist \cup v$ ;
4: end for
5: for vertex  $v$  in  $B.vertex$  do
6:    $vertexlist \leftarrow vertexlist \cup v$ ;
7: end for
8:
9: return CONVEXHULL( $vertexlist$ );

```

---

#### 3.2. Performance issues

The user holds a virtual graver which has position and rotation. We created a bounding box that contains the graver and only recalculates those cells which are overlapping with the bounding box. The other cells outside the bounding box are in idle state. Finally, if there is no intersection between the graver and the object, no computation will be performed and this little modification increases the frame rate while the user is carving nothing in carving mode(here named **Efficient Rays**). The performance comparison of the previous work and the proposed system is listed in Table 1.

#### 3.3. Force feedback

In the data structure of our previous work [6], each ray has to spend lots of time to compute a boolean oper-

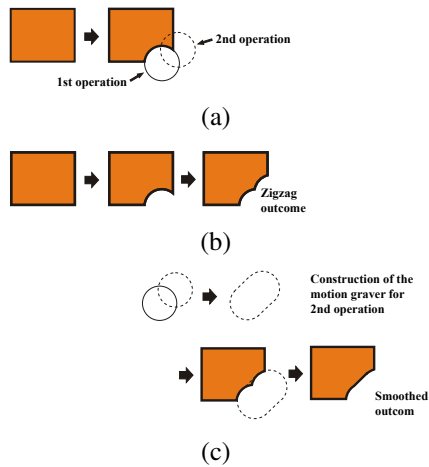


Figure 2: This figure demonstrates how zigzag artifact happens and the solution for it. (a) shows the two successive samples of a graver operation. (b) shows the result without any treatment and it causes a zigzag. Although it is theoretically correct, but it is visually wrong. (c) shows the concept of constructing a motion graver to replace the original model for carving. (d) shows the result of the carving using the constructed motion graver.

ation(union/substract), although there is no sample in a ray. This computation always used 30ms to compute one union operation of the data structure and it is a huge problem to build a real time system. Thus, we provide a new way to reduce the processing time. The system just easily skips the ray that has no sample in the graver or in the object, so there is no operation in the ray. That means there remains a few rays needed to compute and this little modification makes the processing time from 30ms to 5ms. Because of the faster ray operation, it also helps a lot when we provide a new approximate force model.

The force model we used is based on the ray data structure. It used arrays of rays to store a list of samples in the graver and the object. The union or subtract operation is used for stuffing or carving. A new operation is used to simulate simple force simulation. First, intersection operation is added in this system. The intersection is simply calculated the intersection ray of two rays and captures the intersection part into one ray. Second, the output ray of this operation is the intersection of the graver and the object. The intersection set stored arrays of rays which can be modeled as a volume to force

simulation. If the volume is larger than a user specified parameter, the carving operation is performed and the user can feel force feedback from the PHANToM devices, otherwise the volume is smaller than the parameter, and no operation will be performed and the user can still feel force feedback but nothing happens in the object. This is the situation when the graver is staying on the surface of the object but the user doesn't want to go deeper into the object.

### 3.4. Assisted Operation

There is a lot of ambiguity while the user can not judge whether the graver is carving on the surface of the object or not. Then we provide an attach mode function in this framework, just like a CNC operation for machine tools. The graver can easily attach to the object surface. After the graver has attached the surface of the object, the graver can move smoothly on the surface and performs carving or stuffing. This concept can eliminate a part of depth problem between the graver and the object.

First, there are many triangles in the isosurface and we can compute the distance between the graver and each triangle. Then, we choose the triangle which is the shortest distance as the attaching surface. Second, the graver can move directly to the target surface with the shortest distance. Finally, the graver is attached to the object and the user can perform further operation while continuing to move freely on the object.

## 4. Experiments and Results

The proposed system is performed on a PC with Intel Core Quad Q6600, 4GB RAM and an nVidia 8800GT graphics card for rendering. Our system is implemented by C++ and Direct3D library. The implementation of the system is mainly based on multi-thread architecture which can be used in Cubical Marching Squares because each cell is independent to the others. The multi-thread interface we used is OpenMP and this interface helps to build up a real-time sculpting for the user. Thus, the frame rate can be up to 30 frames per second and the user can use the system interactively. Figure 3 shows speed up factors for the three major techniques, bounding box, efficient rays, and multi-threading.

We have invited an artist to use our system to create a few three-dimensional models after a short training

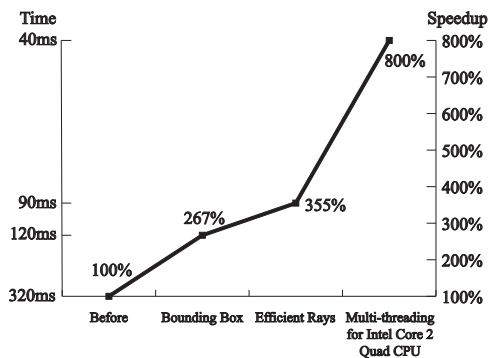


Figure 3: Speed up factors for the three major techniques, bounding box, efficient rays, and multi-threading.

Table 1: This table shows the time average of a boolean operation on the target object. Columns from left to right: graver shape, sculpted object, time average of the old system and time average of our proposed system.

Graver	Object	Average time (before acceleration)	Average time (after acceleration)
Cube	Cube	262.0ms	39.2ms
Sphere	Cube	298.8ms	39.5ms
Tetrahedron	Cube	281.5ms	42.1ms
Cylinder	Cube	318.9ms	59.8ms

period of 15 minutes. After that, the artist started to create her results given in Figure 4. It takes about a few minutes to finish a three-dimensional model in our system although it may take longer time to finish in professional software.

## 5. Conclusions and future work

In this paper, we present a real-time sculpting system with motion graver. We employed the convex hull idea for creating a smooth graver between two separate graver locations. A user can use the specified graver for smoothing the creations. The proposed method has eliminated lots of zigzag edges of the object but some of the results remain to have sharp edges at the boundary and the user may want a more smoothing result. One possible solution of the problem is to translate to a more complicated model of the motion graver such as smooth spline surfaces [10].

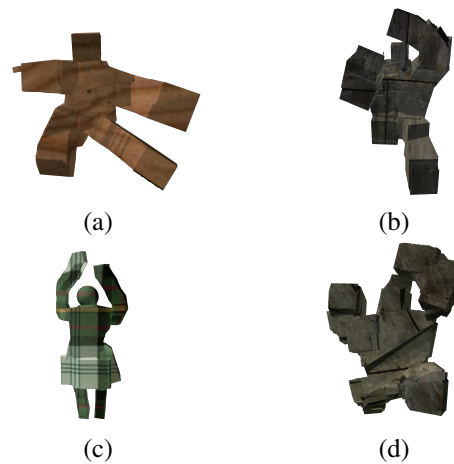


Figure 4: Some results created from our system.

The depth problem is mostly addressed in this system. Like many other system, our system provides an additional camera view port to help the artist to feel the space perception. Two views can provide an additional feature to an artist when the artist wants to get more depth information. Second issue the artist provides is that she wants a floor such as a ground attached to an object. Also, she wants a fixed rotation to the object and performs rotation along the normal of the ground, where we propose to use a Wii remote controller in a separate hand for motion parallax effect.

Although the sharp features are mostly preserved by adaptive resolution, the problem is that the depth of the octree is limited. Thus, the user cannot perform a detailed manipulation after generating a rough model at first. It is a dilemma between quality and processing time. On the other hand, our artist provided a useful comment that many artist want to have a rough model before editing in professional three-dimensional modeling software such as 3ds Max, Maya or ZBrush. Therefore, our system provides a convenient modeling interface with haptic interface and the artist can create a rough three-dimensional model in a few minutes.

## 6. Acknowledgments

This research is sponsored by the National Science Council, Taiwan (R.O.C.) under contract numbers NSC95-2221-E002-299, NSC96-2752-E002-006-PAE, and NSC96-2622-E002-002. We would like to thank our artist "Kelsy Tang" for her valuable feedback.

## REFERENCES

- [1] Carlos D. Correa. Interactive sculpting of meshes as implicit surfaces. <http://www.youtube.com/watch?v=vaX6GR9vC0Y>, 2007.
- [2] Carlos D. Correa and Deborah Silver. Visualizing collision effects between penetrating and non-penetrating objects. In *SIGGRAPH '07: ACM SIGGRAPH 2007 sketches*, page 47, New York, NY, USA, 2007. ACM.
- [3] Eric Ferley, Marie-Paule Cani, and Jean-Dominique Gascuel. Resolution adaptive volume sculpting. *Graphical Models*, 63(6):459–478, 2001.
- [4] Sarah F. Frisken, Ronald N. Perry, Alyn P. Rockwood, and Thouis R. Jones. Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of ACM SIGGRAPH*, pages 249–254, 2000.
- [5] Tinsley A. Galyean and John F. Hughes. Sculpting: an interactive volumetric modeling technique. In *Proceedings of ACM SIGGRAPH*, pages 267–274, 1991.
- [6] Chien-Chang Ho, Cheng-Han Tu, and Ming Ouhyoung. Detail sculpting using cubical marching squares. In *ICAT '05: Proceedings of the 2005 international conference on Augmented tele-existence*, pages 10–15, New York, NY, USA, 2005. ACM.
- [7] Chien-Chang Ho, Fu-Che Wu, Bing-Yu Chen, Yung-Yu Chuang, and Ming Ouhyoung. Cubical marching squares: Adaptive feature preserving surface extraction from volume data. *Computer Graphics Forum*, 24(2), 2005.
- [8] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of hermite data. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 339–346, New York, NY, USA, 2002. ACM.
- [9] Leif P. Kobbelt, Mario Botsch, Ulrich Schweincke, and Hans-Peter Seidel. Feature sensitive surface extraction from volume data. In *Proc. SIGGRAPH 2001*, pages 57–66, 2001.
- [10] Charles Loop. Smooth spline surfaces over irregular meshes. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 303–310, New York, NY, USA, 1994. ACM.
- [11] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proc. SIGGRAPH 1987*, pages 163–169, 1987.
- [12] Yan-Hong Lu, Wei-Teh Wang, Rung huei Liang, and Ming Ouhyoung. Virtual sculptor: A feature preserving haptic modeling system. In *Proc. of ACM International Workshop on Immersive Telepresence*, 2002.
- [13] K.-L. Perng, W.-T. Wang, M. Flanagan, and M. Ouhyoung. A real-time 3d virtual sculpting tool based on modified marching cubes. In *Proceedings of International Conference on Artificial Reality and Tele-existence*, pages 64–72, 2001.
- [14] Ronald N. Perry and Sarah F. Frisken. Kizamu: a system for sculpting digital characters. In *Proceedings of ACM SIGGRAPH*, pages 47–56, 2001.
- [15] Pixologic. Zbrush: Software from pixologic. <http://www.pixologic.com>, 2007.
- [16] F. P. Preparata and S. J. Hong. Convex hulls of finite sets of points in two and three dimensions. *Commun. ACM*, 20(2):87–93, 1977.