Trampoline: A Double-sided Elastic Touch Device for Repoussé and Chasing Techniques

Jaehyun Han
HCI Lab, KAIST.
291 Daehak-ro, Yuseong-gu
Daejeon, Rep. of Korea
jay.jaehyun@gmail.com

Seongkook Heo
HCI Lab, KAIST.
291 Daehak-ro, Yuseong-gu
Daejeon, Rep. of Korea
seongkook@kaist.ac.kr

Jiseong Gu
HCI Lab, KAIST.
291 Daehak-ro, Yuseong-gu
Daejeon, Rep. of Korea
jiseong.gu@kaist.ac.kr

Geehyuk Lee
HCI Lab, KAIST.
291 Daehak-ro, Yuseong-gu
Daejeon, Rep. of Korea
geehyuk@gmail.com

Figure 1: Products with relief models on a mug, a mobile phone case, a teacup, and a plate.

CHI 2014, April 26–May 1, 2014, Toronto, Ontario, Canada.
ACM 978-1-4503-2474-8/14/04.

Abstract

Relief is often used to add patterns to product surfaces, but interaction techniques for modeling relief on the surface of virtual objects have not received due attention. We adopt the repoussé and chasing artwork techniques in an alternative interaction technique for modeling relief on virtual surfaces. To support the interaction technique, we develop a double-sided touchpad called Trampoline that can detect the position and force of a finger touch on both sides. In addition, it provides an elastic feedback to users as the surface consists of an elastic fabric. With this device and the interaction technique developed, we implement a relief application and present modeling results that demonstrate the efficacy of our system.

Author Keywords

Repoussé and Chasing; Trampoline; Double-sided Touchpad; Elastic Touchpad; 3D Modeling

ACM Classification Keywords

H.5.2 [User Interfaces]: Input devices and strategies (e.g.,mouse, touchscreen).

Introduction

Relief is a traditional sculpting technique that has been applied to many artworks. In recent times, it has also been used to add patterns to product surfaces, as

Figure 2: Great Plate of Bacchus, made using the repoussé and chasing techniques (from Wikipedia).

shown in Figure 1. Automatic relief mapping using an image to produce a surface volume on the virtual model is one of the popular methods used to create a relief on the surface of a virtual object, but this method does not provide designers with the freedom to directly modify the virtual surface. In addition, interaction techniques from traditional computer-aided design (CAD) applications and studies [2, 4, 6, 7] primarily focus on modeling the 3D shapes of virtual objects and do not support intuitive interaction for creating reliefs. Therefore, we sought an interaction technique that is specialized for relief modeling on a virtual surface.

There are many relief modeling techniques, such as repoussé, casting, and carving. Among these techniques, we chose repoussé and chasing as metaphors for a new modeling technique. Using repoussé and chasing, artists can model a relief by working on both sides of a surface, as shown in Figure 2. We believe that users can learn the new modeling technique easily as many of them are already familiar with the concept of repoussé and chasing from their real world experiences.

In this paper, we introduce the repoussé and chasing techniques combined with the design rationales of a new touchpad. The new touchpad can estimate the position touched and the force applied on both sides of the touch surface in order to support the proposed interaction technique. Using an implemented relief application, we show the proper working of both the interaction design and the device. The discussion section explains the limitations of our current implementation and outlines future work.

Related Work

Many interaction techniques to support 3D modeling processes by non-CAD users have been proposed [2, 4]. Galyean and Hughes proposed a modeling system that uses a 3D isotonic device with elastic feedback [2]. McDonnell et al. proposed a system that uses a Phantom haptic device to track finger movements [4]. These proposals provide intuitive mapping among the input device, the interaction technique, and the users' experience from the real world. However, the primary focus of these studies is modeling the shapes of 3D virtual objects. In contrast to these studies, this paper focuses on an interaction technique for relief modeling, for which few research examples exist. Relievos [5] is a rare example that demonstrates an interaction technique for creating a relief on a touch phone.

Studies have also been conducted on 3D modeling using physical proxies to represent virtual objects [6, 7]. To implement such a system, Sheng et al. used a Vicon tracking system to capture the relative positions of fingers and a physical proxy [6], whereas Smith et al. used a custom sensor system comprising a matrix of resistive sensors [7]. Modeling 3D models with these systems can be applicable to relief modeling. However, prototypes from these studies could not sense the raising of the device surface. Thus, the embossing (or extrusion) operation in these studies requires an additional mode while our device supports concurrent embossing and engraving operations.

In order to concurrently support bi-directional deformation of a surface, an input device must sense the manipulation on both sides of the input surface. Combining two capacitive touch devices is a common approach used in double-sided touch interactions [9].

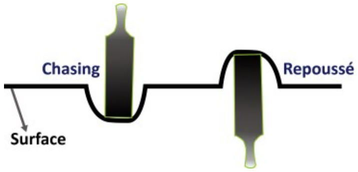

Figure 3: Concept of repoussé and chasing.

However, the rigid surfaces of such a combination cannot provide the appropriate elastic and compliant haptic feedback needed to support users' repoussé and chasing skills. Many techniques for sensing the deformation of a physical surface have been introduced [1, 8]. These techniques, however, use a camera to capture the surface deformation, and therefore cannot support interaction techniques on both sides of the surface.

The first contribution of this paper is adoption of repoussé and chasing in an interaction technique for relief modeling on virtual surfaces. The second contribution is a new input device that supports the proposed interaction technique by estimating the 2D position of a touch and the force applied on both sides of an elastic surface.

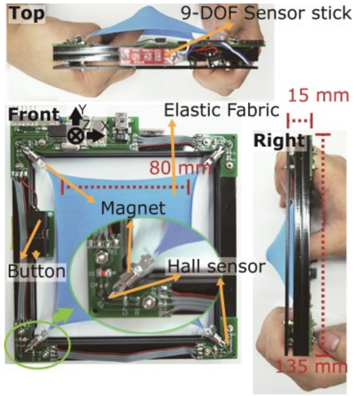

Figure 4: Trampoline prototype.

Repoussé and Chasing

Repoussé and chasing are metalworking techniques used for relief modeling. The concept of repoussé is easy and straightforward: the surface is raised, as shown in Figure 3, when a person presses on the back side of a surface. The surface is stretched, but it remains continuous and the amount of stretching is proportional to the applied force. Thus, the person will feel the resistance of the surface deformation proportional to the applied force. Hammering and pressing are two formative actions in repoussé. Hammering the surface causes it to stretch extensively in a small area. By pressing the surface, a person can control the amount of deformation and change the position of the deformation while they move their tool on the surface. Chasing is the opposite of repoussé, as shown in Figure 3. It has the same formative action, but the force is applied on the front side of the surface. Using these techniques together, a person can emboss and engrave the surface concurrently. The strength of the force on the surface determines the amount of surface deformation. The resistance to the deformation helps the person to control their hand to achieve the intended deformation.

Trampoline

In order to support these techniques, we set three design requirements on the input device. First, the device must be a two dimensional touchpad with the ability to sense the touch position on both the front and back sides of the device surface. Another basic requirement is the estimation of the applied forces on the touch surface. One-to-one mapping between the applied force and the amount of surface deformation can provide a realistic experience to users and does not require additional steps to select the deformation size. Lastly, the device should provide elastic feedback. In the real world, resistance increases with increases in deformation. We anticipated that the elastic feedback would provide a feeling of resistance caused by the surface deformation to the users.

We could not find any device satisfying our design requirements in existing studies. As a result, we decided to make a new touch device called Trampoline. Trampoline consists of four linear hall sensors, four permanent magnets, and an elastic fabric (spandex). One side of each magnet is attached to a corner of the elastic surface and the opposite side of the magnet is attached near the hall sensor, as shown in Figure 4. The size of the device is 135 x 135 mm and that of the elastic surface is 80 x 80 mm. The thickness of the current prototype device is 15 mm in order to provide a stable grip to users; however, this can be reduced to a few millimeters if necessary. In order to support additional interactions, we attached two buttons to the left side of the device and a 9-degree-of-freedom (9-DOF) sensor stick from Sparkfun to the top of the device. Four analog signals from the hall sensors are converted into digital values. In addition to these values, the states of two buttons and the outputs from the 9-DOF sensor stick are sent to tracking software every 15 ms.

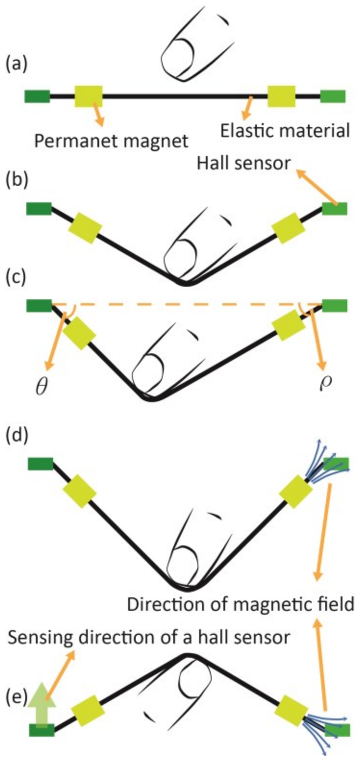

Figure 5: Working principle of Trampoline.

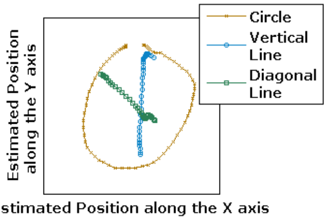

Figure 6: Trajectory of drawings on Trampoline.

Working principle

The device estimates the touch position and the force on the surface based on the output of the hall sensors. The strength of the magnetic field at each hall sensor is determined by the orientation (θ and ρ in Figure 5) of the magnet and the sensor when the distance between them is constant. Because of this, we can approximate the orientation of a magnet by measuring the sensor output. When two magnets are attached to the end of an elastic material, as illustrated in Figure 5, the ratio of sensor outputs from the two sensors represents the approximate location where the force is applied (see Figures 5(b) and 5(c)). The sum of two outputs can be used for pseudo force estimation, as shown in Figures 5(b) and 5(d), because the displacement of a finger is mapped to the applied force on the surface. As shown in Figures 5(b), 5(c), and 5(d), the sensor output is positive when the direction of the magnetic field and the sensing direction of a hall sensor are the same. The output becomes negative in the opposite case, as illustrated in Figure 5(e). Thus, we can predict the direction of a touch by observing the sign of a sensor's output.

Signal processing

The tracking software applies four signal processing steps to estimate the position of a touch, the side touched, and the displacement along the z-axis.

First, the software stabilizes the raw sensor values using a low-pass filter. Second, the stabilized values s_n (n is an index to the four sensors) from the previous step are calibrated using cubic functions, to cope with the individual difference of sensors, i.e., $\mathrm{c_n} = f_n(\mathrm{s_n})$. Third, the software estimates a touch position, a direction, and a displacement using the calibrated values, c_n. The horizontal and vertical positions, p_x and p_y, of a touch point is estimated by the following equations: $p_x = (c_1 + c_2)/c$ and $p_y = (c_1 + c_2)/c$, where $\mathrm{c} = \sum c_n$. The side that is touched is signified by the sign of the sum c. The displacement of the finger along the z-axis is estimated by the magnitude of the sum c. As the sum c always contains some amount of noise, we had to ignore it if it is below a certain threshold. This means that a very light touch is ignored in the current prototype. As the last step, a speed-dependent low-pass filter [3] is applied to the estimated touch position to reduce jitter without causing latency.

Figure 6 shows the trajectory of estimated touch positions for the drawing of a circle, a vertical line, and a diagonal line using an index finger. The overall movements are matched with the intended movements, except for the marginal position shift at the starting position of each line. This occurs when the applied force is increased in the fixed position. The displacement is mapped to the applied force on the surface by a one-to-one mapping function, as shown in Figure 7. Thus, we can approximate the force on the surface while receiving elastic feedback from the device surface.

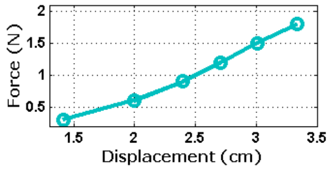

Figure 7: Applied force on the surface depending on the displacement.

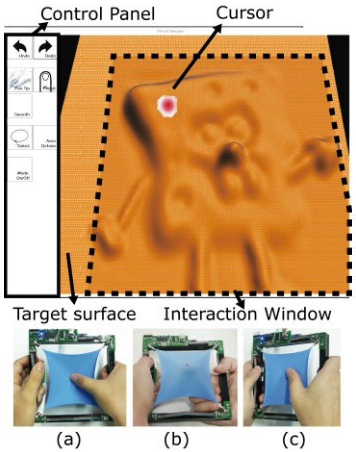

Figure 8: Application screen and interactions for: (a) engraving, (b) embossing, and (c) moving the interaction window by tilting.

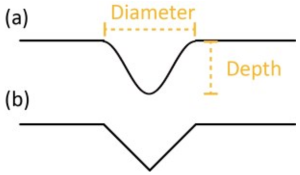

Figure 9: Shape of deformation with (a) a fingertip and (b) a pen-tip.

Relief Application

We implemented a relief modeling application to demonstrate the feasibility of our prototype and modeling techniques. The application was written in C# using the OpenTK library and the OpenGL shading language. The size of the application screen was approximately 1500 x 1000 pixels. We placed a control panel on the left side of the application screen, as shown in Figure 8. It has buttons for selecting the current operation mode and a deformation shape, as well as undo and redo buttons. The remaining screen area shows the surface where the deformation occurs. An interaction window is located on the surface, represented by a black wire square. The surface in the interaction window can only be deformed. Users can control the position of the window by tilting the device along the X and Y axes and zooming in/out by pushing two buttons on the left side of the device. This enables deformations at various resolutions without distracting the working context. There are two states, light-touch and heavy-touch, according to the strength of the touch. In the light-touch state, only the feedback of the current position is acquired, whereas, in the heavy-touch state, the current operation is performed. In the heavy touch state, the amount of deformation is proportional to the applied force on the device surface.

We designed two basic operations analogous to their physical counterpart operations in the physical world. The first operation is pressing. In this operational mode, users can choose one of two deformation shapes: a fingertip or a pen-tip. As shown in Figure 9, the surface is deformed smoothly with a fingertip and sharply with a pen-tip. The diameter and depth of the deformation is proportional to the force applied to the surface. The other operation is smoothing. The smoothing operation applies a bi-cubic interpolation to the vertices in the cursor area, thereby reducing discontinuity in the surface. Interpolation is applied to the surface repeatedly during a smoothing operation.

In addition to the two basic operations, we implemented additional operations such as area selection, area modification, and undo/redo operations to get the benefits of virtual object manipulation. Area selection is achieved by dragging. The selected area is colored yellow. After selection, we can modify the selected area by using the area manipulation operation. The depth of the modification of each vertex in the selected area is proportional to the distance between the vertex and the cursor. This operation can be used to create a continuous deformation over a selected area. For example, this operation is useful in making a face and a body of an animal. When the manipulation mode are turned off, all visual feedback like the cursor and the interaction window disappears. In this state, users can check a modeling result while changing the viewing angle using the keyboard. With this relief application and the Trampoline device, we created a number of relief models, as shown in Figure 10.

Discussion and Future Work

There are several usability issues in the current system. First, we used a mouse and graphical user interface (GUI) elements on the screen to support various functions in the relief application. Switching to other input devices definitely results in overhead. Using the spatial movement of the device as input commands could be a possible solution to this problem, as discussed by Smith et al. [7]. Second, a lack of enough feedback to distinguish the touch state causes unintended deformation of the surface. Even when the

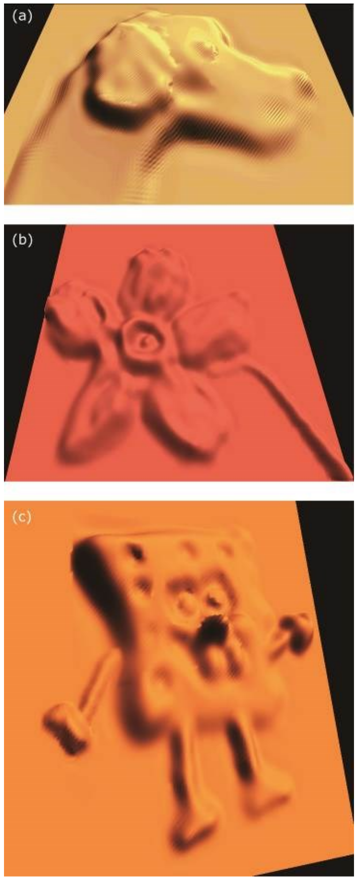

Figure 10: Relief modeling examples with the Trampoline: (a) a dog, (b) a flower, and (c) SpongeBob.

color of the cursor changed according to the touch state, it was hard to perceive the state change immediately. Adding haptic or sound feedback for this state change would help people to recognize the touch state better. Another issue is the fixed deformation direction. Surface deformation in the real world occurs in the direction of the applied force. However, the surface deformation in the current implementation always occurs in a fixed direction perpendicular to the initial state of the surface since our prototype cannot estimate the force direction. We are investigating ways to improve our prototype while keeping its structure simple.

Conclusion

We adopted actual artwork techniques, repoussé and chasing, to an interaction technique for modeling relief on virtual surfaces. In order to support the technique, we implemented a new prototype device called the Trampoline. This device can estimate the position of a 2D touch, sense force on both sides of the device surface, and provide force feedback on the applied force. We also implemented a relief modeling application to explore further possibilities and current limitations of the interaction technique and the Trampoline device. We believe that relief modeling on the surface of virtual objects could be made more practical by using spatial gestures, providing better feedback for touch states, and improving the prototype device.

ACKNOWLEDGMENTS

This work was supported by the IT R&D program of MKE/KEIT. [KI10041244, SmartTV 2.0 Software Platform]

References

[1] Follmer, S., Johnson, M., Adelson, E., and Ishii. H., deForm: An Interactive Malleable Surface For Capturing 2.5D Arbitrary Objects, Tools and Touch. In Proc. UIST, ACM Press (2011), 527-536.

[2] Galyean, T. A., and Hughes, J. F. Sculpting: An Interactive Volumetric Modeling Technique. In Proc. SIGGRAPH, ACM Press (1991), 267–274.

[3] Lee, G., Lee, S., Bang, W.-C., and Kim, Y. A TV Pointing Device Using LED Directivity. In Proc. ICCE, IEEE (2011), 619–620.

[4] McDonnell, K. T., Qin, H., and Wlodarczyk, R. A. Virtual Clay: A Real-time Sculpting System with Haptic Yoolkits. In Proc. Interactive 3D graphics, ACM Press (2001), 179–190.

[5] Relievos, Apple App Store, https://itunes.apple.com/app

[6] Sheng, J., Balakrishnan, R., and Singh, K. An Interface for Virtual 3D Sculpting via Physical Proxy. In Proc. GRAPHITE, ACM Press (2006), 213–220.

[7] Smith, R. T., Thomas, B. H., and Piekarski, W. Digital Foam Interaction Techniques for 3D Modeling. In Proc. VRST, ACM Press (2008), 61–68.

[8] Vlack, K., Mizota, T., Kawakami, N., Kamiyama, K., Kajimoto, H., and Tachi, S., GelForce: A Vision-based Traction Field Computer Interface. In Porc. CHI EA, ACM Press (2005), 1154-1155.

[9] Wigdor, D., Leigh, D., Forlines, C., Shipman, S., Barnwell, J., Balakrishnan, R., and Shen, C. Under the Table Interaction. In Proc. UIST, ACM Press (2006), 259–268.