

SandCanvas: A Multi-touch Art Medium Inspired by Sand Animation

Rubaiat Habib Kazi¹, Kien-Chuan Chua¹, Shengdong Zhao¹, Richard C. Davis², Kok-Lim Low¹

¹Computer Science, National University of Singapore
13 Computing Drive, Singapore 117417
{rubaiat, chkch, zhaosd, lowkl}@comp.nus.edu.sg

²SIS, Singapore Management University
80 Stamford Road, Singapore 178902
rkdavis@smu.edu.sg



Figure 1: A user interacting with *SandCanvas* (left), and images created with *SandCanvas* (right).

ABSTRACT

Sand animation is a performance art technique in which an artist tells stories by creating animated images with sand. Inspired by this medium, we have developed a new multi-touch digital artistic medium named *SandCanvas* that simplifies the creation of sand animations. *SandCanvas* also goes beyond traditional sand animation with tools for mixing sand animation with video and replicating recorded free-form hand gestures. In this paper, we analyze common sand animation hand gestures, present *SandCanvas*'s intuitive UI, and describe implementation challenges we encountered. We also present an evaluation with professional and novice artists that shows the importance and unique affordances of this new medium.

Author Keywords

Sand animation, multi-touch, tabletop computing, creativity

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms

Design, Experimentation, Human Factors

INTRODUCTION

Sand animation, also known as sand art, is a form of visual storytelling in which an artist dexterously manipulates fine granules of sand to produce images and animations. The process begins by applying sand to a lighted surface, after which images are rendered on the surface by drawing lines and figures with bare hands. It is an increasingly popular medium for performances and stop-motion animation [14].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.

Copyright 2011 ACM 978-1-4503-0267-8/11/05...\$10.00.

Two characteristics combine to make sand animation a unique art form. First, because it is a performance medium, its attraction and aesthetics are closely tied to the creation process as well as the finished artwork [16, 18]. The creation process in performance media is improvisational, fast, continuous, and often accompanied by other forms of performance art, like music, choreography, drama, and dance. The sand animator's task is to unfold a narrative through a progression of visual images produced with a seamless stream of physical gestures.

Second, sand animations are formed through a powerful and expressive vocabulary of physical interactions between artist's hands and small granules of sand. In contrast with sketches or paintings, which are produced with discrete pen or brush strokes, sand animation leverages the delicate structure of the artist's whole hand (often both hands). These hand gestures are easy to learn, quick to perform, and economical to correct, which makes this medium suitable for exploration and brainstorming in addition to storytelling through live performance.

Sand animation has increasingly attracted audiences and artists because of its innovative and expressive graphic style [14, 24]. However, sand-animation performance spaces are difficult to set up and maintain [1], which prevents many novices from getting started. This led us to create *SandCanvas*, a new digital artistic medium inspired by sand animation. *SandCanvas* adds undo and recording features that make sand animation easier to produce, it allows easy experimentation with colors and textured backgrounds, and it adds new capabilities that go beyond traditional sand animation, such as recorded gestures and video mixing.

The increased availability of multi-touch display surfaces has removed some of the technical obstacles to creating a digital sand animation medium, but we still faced significant challenges. Multi-touch UI toolkits do not currently capture all the richness in human hand gestures

[21]. In particular, they do not attempt to map touch regions in the current time step to touch regions in the previous time step. This required us to devise a new, fast approach to performing this mapping. Also, simulating the physical behavior of sand in real time is still a major challenge. We achieved real-time performance by optimizing an existing technique [28] and using graphics hardware acceleration.

In this paper, we make the following contributions:

- We introduce a new digital artistic medium that leverages the expressiveness of hand gestures on a multi-touch platform to provide a visual experience that goes beyond physical sand animation.
- We analyze the sand animation process, highlighting common pouring and manipulation techniques and developing a taxonomy of hand gestures.
- We present *SandCanvas*'s intuitive UI design and its enhanced digital capabilities.
- We describe our implementation, which employs new techniques for performing real time sand simulation in response to gestural input.
- We evaluate *SandCanvas* with 1 professional artist, 4 amateur artists and 2 novice users to gain insight into the importance and unique affordances of this medium.

RELATED WORK

SandCanvas is inspired by new media interactive installations and recent advances in interactive surfaces. We divide related work into three sections. First, we briefly summarize recent work in technologically enhanced static and performance art creation systems in general and for multi-touch surfaces. We then describe related systems and algorithms for sand motion simulation. Finally, we discuss systems closely related to *SandCanvas*.

Technologically Enhanced Art Creation Systems

Performance art has a rich history that spans hundreds of years [17]. Myron Krueger's *Videoplace*, developed between 1969 and 1975, was an early interactive artwork that incorporated computer vision [15]. Since *Videoplace*, numerous audio-visual performance systems have been driven by human gestures [18, 27]. Examining numerous audio-visual performance systems, Levin derived a set of design goals for new performance art systems [18]. According to him, successful systems should be predictable, instantly knowable and indefinitely masterable. We pursued similar goals when designing *SandCanvas*.

In recent years, researchers have produced 2D animation by demonstration systems that could be used for performance art. K-Sketch [9] is a general purpose and informal sketch based 2D animation tool that allows novices to create animation quickly and easily, but all interaction must be done through a single point. In Video Puppetry [4], artists record simultaneous manipulations of multiple physical puppets to create animation. Researchers have also explored

the use of multiple touch points to record real-time deformation of characters [13].

Researchers have also produced notable multi-touch painting systems. Project Gustav attempts to create a realistic painting experience [22], while I/O brush [25] allows artists to paint with patterns and movements “picked up” from everyday materials. Fluid Paint [30] and IntuPaint [31] use the entire region of contact between brush and surface to model brush strokes. *SandCanvas* also captures the entire region of contact between the surface and the artist's hand to model interactions with sand.

SandCanvas bears a resemblance to each of the art creation systems presented here. It is a medium for performance art where the final performance is a kind of 2D animation. Instead of animating a fixed set of characters, however, the artist creates characters in sand using rich gestures that cannot be represented adequately with a set of discreet points. *SandCanvas* also has unique creative tools like recorded gestures that aren't found in any of these systems.

Physical Sand Simulation

Since the sand particles used in sand animation are very fine, the number of sand particles is potentially huge. Hence, physically accurate interaction with sand is particularly challenging. Li and Morshell devised one simulation approach, but it assumes that sand is moved by convex objects only [19], which prevents touch regions of arbitrary shape from interacting with sand. Bell and colleagues devised a sand simulation method that handles arbitrary shapes [5], but it models each grain as a discrete element and will not produce real-time simulations on the scale needed for sand animation.

Summer and colleagues developed a faster technique that still falls short of real-time performance [28]. Onoue and colleagues sped it up by assuming that only rigid objects would interact with sand [23]. Our method is also based on Summer's, but we do not assume objects are rigid, because an artist's hand can change as it moves across the canvas.

Sand Art for Storytelling

We have found few sand art systems worth noting. Hancock and colleagues' sandtray therapy system allows storytelling on a sand background, but users manipulate figurines instead of sand [11]. Ura and colleagues developed a tool for painting with simulated sand, but it reduces input to discrete points [29]. iSand¹ is an iPhone application for sand art that shares this limitation, and its sand granules are much larger than those used in traditional sand animation.

In contrast, *SandCanvas* captures rich human hand gestures in multiple areas instead of multiple points. It also preserves the expressive and playful nature of sand animation and adds new capabilities that go beyond traditional sand animation, such as recorded gestures and video mixing.

¹ <http://itunes.apple.com/us/app/isand-sand-particle-painting/id346966446?mt=8>

SAND ANIMATION ANALYSIS

To better understand the special requirements of our new medium we sought to enumerate the gestures commonly used by sand animators. After a meticulous observation of 30 sand animation videos, we identified a set of sand animation techniques commonly employed by artists.

Common Sand Animation Techniques

Sand animation has pouring and manipulation techniques.

Pouring Techniques

Pouring is an additive technique that varies depending on how much of the canvas is affected. *Canvas pouring* is used to set the texture and initial context for painting (Fig. 2 left), or, to change context while storytelling. *Skinny pouring* is used to draw tiny details, lines, and shapes (Fig. 2 right).

Manipulation Techniques

Sand manipulation techniques move sand rather than adding it. We classified these techniques by how the artist's hand interacts with sand. *Fingertip drawing* traces out lines with the tip of one or more fingers (see Figure 3 left). While, *finger carving* (see Figure 3 right) uses the whole finger, typically the index finger, small finger, or the outside of the thumb, for drawing and fine tuning shapes.

Artists do not use their fingers exclusively. Palms are often used to create semi-elliptical, or spiral like patterns, such as clouds. We call this technique *palm rubbing* (see Figure 4 left). Whole hands are often used to make big sweeps to clear the canvas and set up a new context for the animation, which we termed *hand sweeping* (see Figure 4 right).

One final technique that bears mentioning is actually a special version of other techniques. Sand animators will sometimes use both hands simultaneously to quickly draw or pour *symmetrical* patterns in sand (see Figure 5). This technique, which is quite rare in other artistic media, is very common in sand animation.

These techniques can be combined to fluidly transform one image into another (see Figure 6), creating surprise and conjuring emotion. Here lies the beauty of sand animation.

Taxonomy of Sand Animation Gestures

After listing common sand animation techniques, we saw many similarities and differences between them. To better compare and contrast these techniques, we created the low-level taxonomy of gestures found in Table 2. While there are other gesture taxonomies in the literature [10, 33], we needed one that was created specifically for multi-touch art work like sand animation.

Pouring	Manipulation
Canvas	Fingertip draw
Skinny	Finger carve
	Palm rub
<i>Symmetrical</i>	Hand Sweep

Table 1: Common sand animation pouring and manipulation techniques. *Symmetrical* is a modifier that can apply to both pours and manipulations.

We manually classified gestures along five dimensions: *mode*, *form*, *precision*, *hands* and *actuation*. **Mode** separates *pouring* gestures from *manipulation* gestures. **Form** indicates any motion in the gesture. In *static* gestures, the hand is held in one position and one configuration, while dynamic gestures change the position or configuration of the hand. (This is similar to the *pose* and *path* concepts in Wobbrock et al.'s gesture taxonomy [33]) The **precision** of the gesture can be *coarse* or *fine*, and the **hands** dimension indicates the number of hands involved in a gesture: one (*uni-manual*) or two (*bi-manual*).

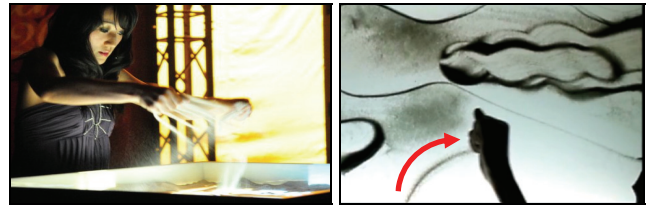


Figure 2: Canvas pouring (left) creates background textures, while skinny pouring (right) is for drawing lines.

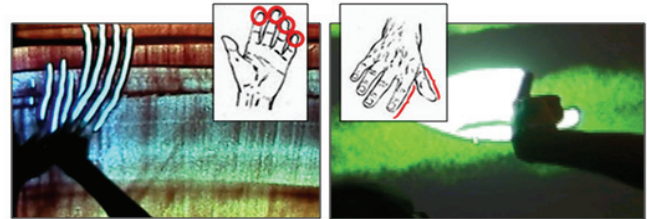


Figure 3: Fingertip drawing (left) and finger carving (right) to create and manipulate shapes.

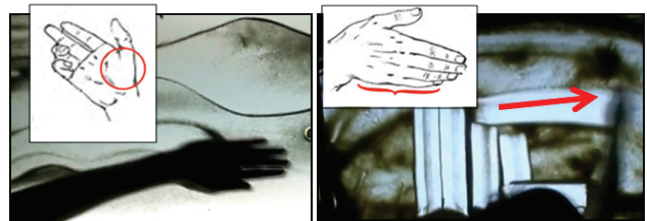


Figure 4: A palm rub (left) draws cloudy patterns, and a hand sweep (right) clears part of canvas.



Figure 5: Symmetrical hand sweep (left) & skinny pour (right).



Figure 6: Fluid transformation of images.

Finally, **actuation** indicates the portion of the artist’s hand that interacts with sand: a *single finger*, *multiple fingers*, the *palm* (without fingers), or the whole *hand* (both palm and fingers). When using one or more fingers, we distinguish between the finger *tips* and the *side* of the finger. We also note when artists use *tangible* objects to interact with sand.

We can classify common sand animation drawing techniques within our taxonomy to gain a more detailed understanding of these gestures (see Table 3). For example, we found the canvas pour technique uses the pouring mode (p); the form can be either static or dynamic (s/d); the precision is coarse (c); it can be done either by one hand (u) or two hands (b), and it leverages the whole hand (h) instead of the other parts to perform the sweep.

This analysis of sand animation gestures helped us to understand the range of interactions that sand animators

Dimension	Categories	Description
Mode	<u>P</u> ouring	Pouring on surface
	<u>M</u> anipulation	Manipulating on surface
Form	<u>S</u> tatic	Hand held in one position
	<u>D</u> ynamic	Hand moved
Precision	<u>C</u> oarse	Gesture has low precision
	<u>F</u> ine	Gesture has high precision
Hands	<u>U</u> ni-manual	Use one hand
	<u>B</u> i-manual	Use both hands
Actuation	<u>S</u> ingle fingertip	Use single fingertip
	<u>S</u> ingle finger <u>s</u> ide	Use side of a single finger
	<u>M</u> ulti-fingertip	Use multiple fingertips
	<u>M</u> ulti-finger <u>s</u> ide	Use side of multiple fingers
	<u>P</u> alm	Use palm without fingers
	<u>H</u> and	Use both palm and fingers
	<u>T</u> angibles	Use other objects

Table 2: Taxonomy of sand animation gestures. The underlined letter(s) in each category indicate the abbreviation used for that category in later tables.

	Canvas pour	Skinny pour	Fingertip draw	Finger carve	Palm rub	Hand sweep	*Symmetrical
Mode	p	p	m	m	m	m	p/m
Form	s/d	s/d	s/d	s/d	s/d	s/d	s/d
Precision	c	f	f	c/f	c/f	c	f/c
Hands	u/b	u/b	u/b	u/b	u/b	u/b	b
Actuation	h	h	st/mt	ss/ms	p	h	all

Table 3: Classification of common sand animation drawing techniques. Each column shows the categories of gestures that we observed for that technique. Category abbreviations are found in Table 2.

need in *SandCanvas*. The following section explains how we designed and developed a UI to support these gestures.

DESIGN AND IMPLEMENTATION OF SANDCANVAS

SandCanvas is designed to run on an interactive surface based on the principle of diffused illumination [2]. Our table uses a 29cm by 21cm white acrylic surface as a diffuser and projection screen for a rear-placed LCD-projector. An array of 140 infrared LEDs also shines on the surface from below, and objects touching the surface reflect this infrared light back on a 320 by 240 pixel infrared video camera. Our software was written with OpenFrameworks², a C++ toolkit for graphic applications with image processing tools. This software runs on a 3.0 GHz Intel Core2 Duo CPU E8400 running Windows Vista with 4GB RAM and a graphics card with an nVidia GeForce 9500 GT2 GPU.

Sand Simulation in *SandCanvas*

Real-time simulation of sand movement in response to rich hand gestures is a challenging problem. Here we describe all the steps in our simulation process.

Tracking and Modeling Contact Shapes

As our multi-touch platform was vision-based, we used standard image processing techniques for contact shape detection. We used dynamic background subtraction to remove the background from the current frame, thresholding to adjust the level of acceptable tracked pixels, Gaussian blur for smoothing and filtering out random noise, and highpass filter to amplify edges. The resulting contact shapes are represented as 2D polygons. We then use the Community Core Vision³ tools to correspond touch regions with one another across successive frames.

Sand Modeling

Although particle systems and voxels are commonly used for modeling the motion of granular materials, they are computationally expensive and cannot handle a very large number of sand granules. Instead, *SandCanvas* uses a discrete height field that is often used to model ground surfaces [23, 28]. The height field has a resolution equal to the screen resolution, thus, each pixel has a height value (16-bit float), which we call a *column*.

Given a set of hand contact shapes and a grid of sand columns, we compute the sand deformation in three steps:

- First, when a contact shape moves on the surface, we find the columns of sand that collide with the path of this contact shape.
- Second, sand within those columns is pushed outward towards the surrounding columns.
- Finally, by detecting steep slopes, sand is moved from higher columns to lower columns, producing realistic sand settling motion or *sand erosion*.

² <http://www.openframeworks.cc>

³ <http://ccv.nuigroup.com>

Computing the Collision Region

When a contact shape first touches the surface, the collision region is the polygon that represents the contact shape itself. However, when a contact shape moves on the surface (Figure 7(c)), the first part of our algorithm computes the region swept by the contact polygons across successive frames. Sand within this *collision region* need to be pushed on the surface. Figure 7(d) shows the contact polygons for the current frame (f_{t+1}) in orange and the previous frame (f_t) in grey. We must now compute the collision region from these two polygons.

Our first step is to compute the point-to-point correspondences between the two polygons. An iterative-closest-point method [6] could be used to compute these correspondences, by rotating the polygons until they are closely aligned. While this gives accurate results, the iterations take a long time. Instead, we simply align the centers of the two polygons and examine each point in the f_{t+1} polygon to find its closest neighbor in the f_t polygon (see Figure 7 (e–f)).

In our second step, we construct a graph, which consists of the polygon in f_{t+1} , the polygon in f_t , and the correspondence edges (see Figure 7(g)). We then compute all the line segment intersections and create a DCEL (doubly-connected edge-list) for this graph (see Figure 7(h)). Finally, we compute the outline of the graph by finding the lexicographically minimal point and walking along the outside edges until we reach our starting point (see Figure 7(i)).

Volume-Conserving Sand Displacement

Now that we have found the columns that collide with the user’s hand, we need to push the sand (as shown in Figure 7(j)) from the previous frontier (grey edges) towards the new frontier (black edges). To do this, we calculate the Euclidean distance transform [8] within the collision region from the center of the previous polygon f_t (this center is the “x” in Figure 7(j)). In the resulting distance map (illustrated in Figure 7(k)), for pixels surrounding the collision region, their distance values are set to a very high distance value (higher than any value computed in the distance transform). To propagate the sand, the sand in each column within the collision region is evenly distributed to its neighboring columns that have higher distance values. We use the algorithm in [26] to perform the distance transform and the propagation efficiently on the GPU, which we implemented using OpenGL Shading Language fragment shaders.

Sand Erosion

Our method to simulate the settling motion of the sand around the collision region is based on the algorithm described in [28]. Our method is implemented for the GPU using OpenGL Shading Language fragment shaders. In the first pass, the slopes of each pixel with the eight neighboring columns are examined. If a slope is larger than a threshold value, excess sand is distributed from the higher column to the lower column using a gathering approach in the second pass. The two passes are performed every

rendering frame, and the sand erosion takes many frames to complete, producing the realistic effect of sand rolling down the slope over time. To produce the asymmetric erosion caused by the temporary obstruction of the hand, unlike [28], we simply do not distribute sand into any collision region that might exist during a frame.

Performance

Our system runs at interactive rates (20–35fps). For a single finger, the average fps is 35, while for 7–8 fingers manipulating at the same time, the average fps is 20–25. With these frame rates, sand movement does not significantly lag behind hand movements, and users are able to feel immersed in the sand animation experience.

Exploring the design space for sand pouring

In addition to developing the algorithm to simulate sand manipulation, another essential component of Sand Animation is sand pouring. Most sand animators begin new scenes in their animations by quickly pouring sand on the canvas to set a background texture. We considered using computer vision techniques to distinguish pouring from manipulation gestures, but we quickly determined computer vision was not up to the task, given the variety of gestures and lighting conditions. Building a 3D deformable mesh model of the user’s hand would achieve the highest fidelity, but it is difficult to build this model in a robust way [32].

Instead, we designed a bi-manual touch interface for pouring. Users touch a pouring button with their non-

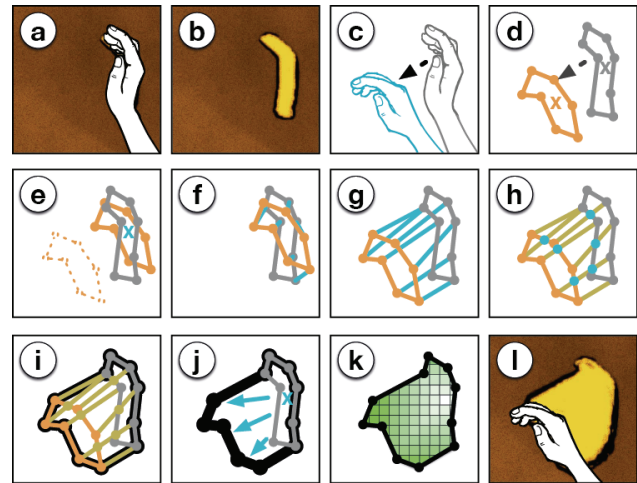


Figure 7: A step-by-step illustration of computing the collision region and pushing the sand. (a) The hand first touches the surface. (b) The sand is pushed to the boundary of the contact shape. (c) The positions of the hand in two successive frames. (d) Their contact polygons. (e) The two polygons are aligned at their centroids. (f) Point-to-point correspondences are computed between the two polygons. (g) A graph is constructed consisting of the polygons and the correspondence edges. (h) The intersections of the edges are computed. (i) The boundary of the whole graph is computed. (j) Sand is to be pushed from the grey frontier to the black outline. (k) A distant map is constructed and sand is pushed from the lighter pixels to the darker pixels until it reaches the boundary of the collision region. (l) Final rendering of the new sand height map (after some sand erosion).

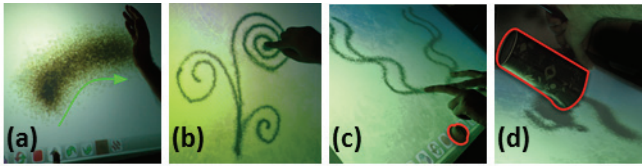


Figure 8: Pouring with touch and tangibles. (a) Canvas pour with whole hand. (b) Skinny pour with fingertips. (c) Symmetrical pour, tangible on button. (d) Pour with tangible.

dominant hand while specifying a pouring region with their dominant hand. This style of interaction avoids mode error. It has been shown to be effective [20] and has been used successfully in several systems [9, 12]. Users distinguish *canvas pouring* from *skinny pouring* by the size and shape of the pouring region. Sweeping through the canvas with the whole hand results in *canvas pouring* (Figure 8(a)), while pointing or tracing a path with a fingertip results in *skinny pouring* (Figure 8(b) and 8(c)). Users can place tangible objects on the pouring button if they wish to pour with both hands (Figure 8(c)).

We also allowed users to pour sand using tangible objects such as jars, thinking that this would provide a more natural feel (Figure 8(d)). In vision based multi-touch systems like ours, bright objects can be detected even when they do not touch the surface. We attached a piece of white paper to the face of a black jar so that it would pour sand when brought close to the surface in pouring mode.

SandCanvas User Interface

Before designing *SandCanvas*'s user interface, we interviewed two professional sand animators to learn how they would like to enhance sand animation in the digital form. We first interviewed Sheh Meng, a professional practitioner with 10 years of experience in performing and teaching sand animation. According to him, most sand animators record video clips of their animations and edit them in a post-production step. Post production also allows animators to play with colors, saturation, and contrast. Hence, a desirable system should provide these capabilities. Sheh Meng also asked for features that allow new types of expression. He suggested a tool for recording gestures and saving them for future reuse.

Second, we interviewed Erika Chen, the winner of "Impresario the Open Platform" 2010. Erika is the world's first singer sand animator, having unique, extensive collaborations with drama, dance and live musicians. She was mostly interested in mixing sand animation with other media, such as clip art or ink drawings.

Our final user interface for *SandCanvas* is the toolbar shown in Figure 9. This toolbar appears at the bottom of the canvas, and it can be reduced to include only the sand pouring and expand UI buttons if the artist desires more canvas area. Based on our interviews, we put the following tools into this toolbar.

Record Session. Users can record their animation as video for later editing.

Undo and Redo. Users can undo and redo up to five operations. This number can be increased at the cost of additional memory.

Change Texture. Users can start with an empty canvas or they can choose from a set of predefined sand textures. Some textures are computer generated, while others are images of real sand.

Reset. Change the surface to the initial state of the texture.

Record Gesture. Users can press a button to begin recording a sequence of pouring or manipulation gestures. Pressing the button a second time stops recording and places a new gesture button icon in the toolbar (see Figure 10). The recorded gesture can be played back by touching this gesture button with one hand and touching the canvas with the other hand. Each touch plays the gesture starting at that touch point. This enables users to play gestures in parallel in different parts of the canvas.

Capture Frame. Users can capture snapshots of the canvas to use as frames in a stop-motion animation (see Figure 11). After pressing the *capture frame* button, it changes to show a thumbnail of the image that was captured.

Change Color. In film and animation production, color is used to create specific moods [3]. For example, a love scene will need different colors than a suspense scene. In *SandCanvas*, users can create a sequence of color gradients before a performance and cycle through them by pressing the change color button.



Figure 9: The user interface panel. From left - hide UI panel, change texture, change color, reset, record session, pour, undo, redo, capture frame and record gesture.

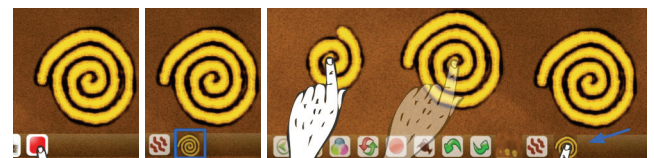


Figure 10: Steps of Gesture recording and playback after pressing the gesture record button (a) User draw a gesture and stop recording (b) An icon having the gesture appears in the UI panel (c) By pressing the recorded gesture button with non-dominant hand and touching by another hand initiates the gesture in different parts of the canvas.



Figure 11: Three key frames for stop-motion animation

Enabling Mixed Media. In film production and storytelling, *mixed media* refers to the mixing of images from separate sources [3, 24]. *SandCanvas* allows users to define a set of still images and video clips that will appear underneath sand during a performance. These images and clips are placed in sequence with color gradients and are also accessed through the *change color* button.

As soon as our system was implemented, we wanted to evaluate it to establish its usability and to understand the importance and the unique affordances of this artistic medium. The details of our evaluation are described below.

USER EVALUATION

Before our final study, we performed a pilot study with three users who gave us qualitative feedback which we report here with other data. Our final study with seven users was both qualitative and quantitative, using a formal protocol designed to answer the following questions:

- Q1. How do users evaluate the realism, fidelity and intuitiveness of Sand Canvas?*
- Q2. Do users find the novel features of SandCanvas useful, and can they apply them effectively?*
- Q3. Can we gain further insight into the expressiveness of this new medium by analyzing gestures that users employ?*
- Q4. Is there evidence that SandCanvas facilitates creativity?*

Participants and Environment

Our formal study had seven participants, all males ranging from 24 to 29 years old ($M=26$, $SD=1.63$). Among them, one is a professional artist, four are amateur artists, and two are novice users. 4 out of 7 users reported that they create artistic works once a week. Our pilot study participants were three females aged 26 to 29. One is a professional artist with prior sand animation experience and the other two are amateur artists. All evaluation sessions took place in a university laboratory using the tabletop system described previously in our design and implementation section. Lighting in the room was dim to give the surface maximal tracking accuracy. Each user received \$25 for their participation.

Method

The formal evaluation process was conducted in the following four steps.

- 1) *Exploration: 10 minutes.* In this step, users were given no explanation of the system, and were told to play with *SandCanvas* while thinking out loud. This step helped us gauge the initial learnability of the system and users' initial impressions.
- 2) *Training: 10–15 minutes.* In this step, users were given a brief description and demonstration of the features they didn't discover in step one. We asked users to recreate a sequence of five drawings, each designed to teach sand animation techniques (Figure 12). Users first recreated all five drawings in their own way. After this, a facilitator demonstrated an easy way to create each drawing, and asked the user to try again.

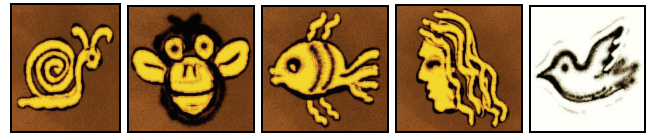


Figure 12: Pictures given to users in step 2 (training).

- 3) *Guided task: 10–15 minutes.* In this step, users were asked to create an animation sequence based on three key frames provided (see Figure 11). This step allowed us to compare user performance on a fixed task.

- 4) *Free task: Up to 30 minutes.* In the final step, users were asked to use their own imagination and create the best sand animation they could. This step helped us assess users' preferred techniques, and it allowed us to observe creative use of *SandCanvas*.

At the end of the study, users were given a questionnaire and interview. The entire study took about 90 minutes.

Results and Discussion

Users' overall reaction was very positive. They found *SandCanvas*'s UI intuitive and they were able to create meaningful artworks in the time they were given. The medium was a pleasure to use; as one user reported, "*The ability to play with sand itself is the most interesting part.*"

- Q1. How do users evaluate the realism, fidelity and intuitiveness of SandCanvas?*

Most of our users felt that the behavior of virtual sand in *SandCanvas* closely mimics the feeling of physical sand. They often perform gestures on *SandCanvas* as if they were playing with real sand: 3 users piled sand in the middle of the canvas and observed its spreading behavior. In the post-study questionnaire, users rated the realism of *SandCanvas* as 4.4 on a scale of 1 (not realistic) to 5 (very realistic). However, one user commented that it has yet to achieve the fluidity of real sand. We believe this is because our current implementation does not model sand grain momentum.

Users commented that they liked the look and feel of *SandCanvas* and indicated that it was very easy to learn: average rating 4.6 on a scale of 1 (extremely difficult) to 5 (extremely easy). In the initial exploration step, five functions (*undo*, *redo*, *reset*, *change texture*, and *change color*) were discovered by all users. All but one user discovered *capture frame* and all but three guessed the purpose of the *record session* button. No users discovered how to pour sand or record gestures, but we expected that these bi-manual functions would require training. After the training step, all users understood all features. One user commented, "*After going through the instructions once, the functionalities are quite obvious*".

- Q2. Do users find the novel features of SandCanvas useful, and can they apply them effectively?*

To better understand the relative merits of our novel features, we recorded the number of times each feature was used in the free task step. Our seven participants spent a total of 159 minutes on free tasks, during which we logged 380 feature usages (average 2.39 features per minute).

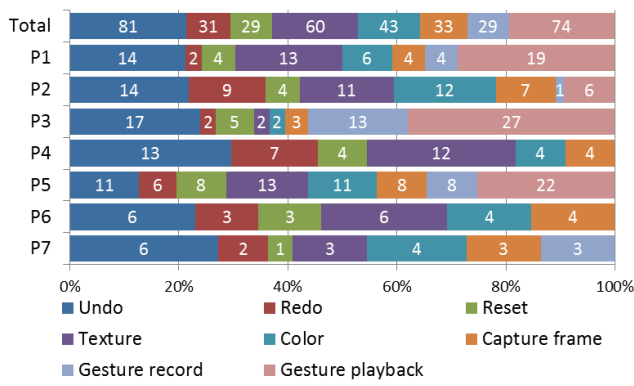


Figure 13: Feature usage counts among participants.

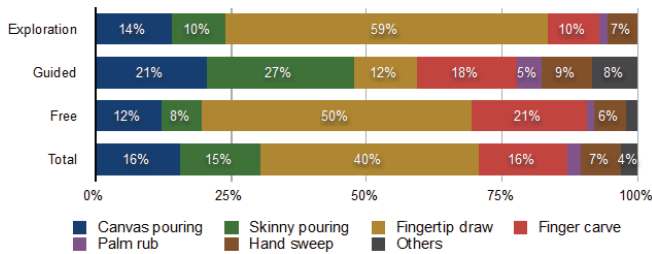


Figure 14: Overall usage of techniques in evaluation steps 1 (exploration), 3 (guided task), and 4 (free task).

Figure 13 summarizes feature usage. Each user made moderate use of most features, though *undo* and *change texture* stood out, accounting for 81 (21%) and 60 (16%) occurrences of all feature usages logged. Only four users took advantage of *gesture playback*, but they made heavy use of this feature (74 times or 29% of all feature usage). We also note that no users took advantage of the *record session* function, because post-production was outside the scope of this study.

The fact that almost all users used most of the features multiple times indicates that users found them useful in creating art works on *SandCanvas*. We are also encouraged to find out a number of users (3 of 7) embrace the more advanced gesture record functionalities and frequently used it in their art creation process.

Q3. Can we gain further insight into the expressiveness of this new medium by analyzing gestures that users employ?

In addition to our feature analysis, we wanted to enhance our understanding of the unique affordance of *SandCanvas* by analyzing the gestures employed by users. We analyzed the video tapes of the user evaluation and classified all 3580 gestures they performed in steps 1 (exploration), 3 (guided task), and 4 (free task). Note that step 2 was for training purpose only; therefore it is not included in the analysis. Step 1 is included because it demonstrates users' the initial reaction to *SandCanvas*, which can be contrasted with later stages when they become more experienced.

Figure 14 shows the breakdown of common sand animation techniques used in the three evaluation stages. Overall, pouring accounted for 31% of gestures and manipulating

accounted for 69%. The most common technique was *fingertip draw* (40% of gestures). *Finger carve*, *canvas pour*, and *skinny pour* are the next most popular techniques (each contains 15-16% of all gestures). *Hand sweep* (7%) and *palm rub* (2%), were used less frequently, but they did play a role in drawing. Finally, we note that tangibles were also used occasionally (4%).

We also observed two approaches to creating artworks in *SandCanvas* (Figure 15), each with a different distribution of drawing techniques. In the free task, five users took a subtractive approach, in which sand is manipulated to create shapes. Two users took an additive approach, in which shapes are made by pouring sand onto the canvas.

Different stages of our experiment also showed different distributions of drawing techniques. The guided task requires users to take an additive approach, in which shapes are made by pouring sand onto the canvas. In the free task, however, 5 of 7 users took a subtractive approach, in which shapes are created by drawing in sand. Because of this, pouring was used much more frequently in the guided task (48% of gestures) than in free tasks (18% of gestures). However, the beauty of *Sand Animation* is that users are free to switch between these approaches, making smooth and seamless transitions to create interesting and often surprising effects, such as Figure 6.

In addition to high level techniques, we analyzed gestures according to our low-level taxonomy to get a detailed sense of how users employed their hands. Almost all gestures (99%) were dynamic rather than static, which helps to justify our efforts to support dynamic gestures. Precision was more or less evenly split between coarse (42%) and fine (59%), indicating the variety of gestures performed. 7% of gestures were bimanual, and almost all of these were performed by three participants drawing symmetrical shapes. This confirms our intuition that bimanual interaction would be an essential part of this medium.

Figure 16 shows the hand actuation dimension of our taxonomy for gestures performed in the exploration, guided task, and free task steps of our experiment. This data shows that *SandCanvas* truly leverages many parts of the hand. The most common gestures were single fingertip (46%), followed by single finger side, multi-fingertip, and hand gestures (19%, 18%, and 11%, respectively).

Tangible gestures were less common (4%), but played a vital role in the artworks where they appeared. One user

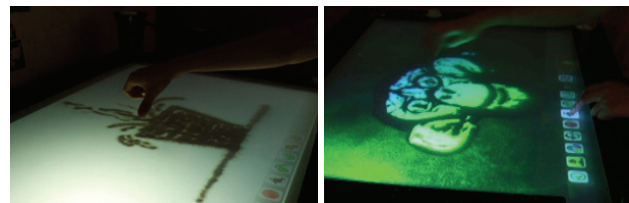


Figure 15: Drawing and animating with pouring (left) and manipulation (right).

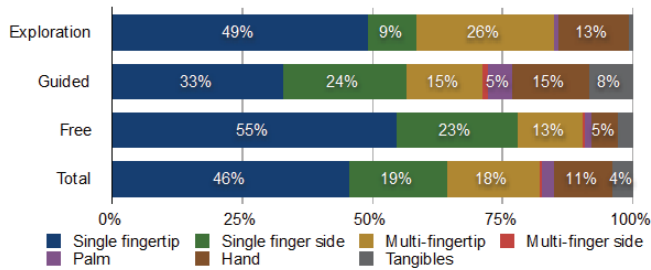


Figure 16: Another dimension of gestures we are particularly interested in is the distribution of different parts of hand (or actuation) in creating art works.

used a sheet of paper to pour over a very large area, and another used a sharp object to draw a star shape. Palm gestures were rare (2%) and served the same purpose as hand gestures. We have noted elsewhere, however, that palm gestures are useful for creating cloud-like shapes. Finally, this hand actuation data highlights the importance of capturing the full region of contact between the user's hand and the drawing surface. Single finger side and hand gestures together accounted for 30% of all gestures. None of these gestures could have been captured by a system that reduced users' input to a set of points.

The gesture analysis presented here demonstrates that *SandCanvas* truly capitalizes on the expressive vocabulary of hand gestures found in sand animation. This was possible only because our implementation captures the full area of contact with the surface, and because it carefully balances sand simulation speed and accuracy. All parts of the hand can be used to produce creative works of art.

Q4. Is there evidence that *SandCanvas* facilitates creativity?

We found four classes of evidence to support this. The richness of gestures captured by *SandCanvas* provides some initial evidence that it facilitates creativity. As one user put it, "[The] use of different parts of hand for direct manipulation inspires creativity." This rich input inspired users to envision radically new uses that go beyond performance art. As another user reported, "I like the instant gratification of it. I would like to use it for brainstorming and story prototyping, because it's so easy and quick to create." Because these gestures are more intuitive than the complex tools in conventional interfaces, *SandCanvas* may also reduce memory demands, making it easier for users to enter a state of creative flow [7].

Second, the variety of artworks produced by users during the free task is also evidence that *SandCanvas* facilitates creativity. In the 159 minutes our 7 users spent on this task, they were able to create 13 different artworks, with 161 distinct drawings (about 1 drawing per minute). Among the 13 artworks, eight of these were performance artworks, four were stop motion animations, and one was a static image. The subject of these artworks ranged from portraits to dynamic landscapes to action sequences with multiple characters. The ability of users to create such a collection of art work in a short time demonstrates the potential of *SandCanvas* as a creative medium.

Third, we observed users devising creative strategies for producing similar effects. For example, one training task asked users to draw a snail. Figure 17 shows two users' approaches to drawing the spiral shape: one used a finger carve gesture, while the other used his whole hand.

Finally, we found that *SandCanvas*'s novel gesture recording feature inspired particularly creative uses. Many users took advantage of gesture recording to clone objects on the canvas, (e.g., to quickly create a crowd of people). One user recorded several drawings of words and played them back all at once to give an impression of many simultaneous speakers. Another user interleaved playback of ring-shaped pouring and drawing gestures that produced a complex interplay between gestures. Finally, one user combined gesture playback with undo to produce stop-motion animation. He recorded the drawing of a spaceship and moved it across the canvas with a sequence of capture frame, undo, and play gesture operations. These unexpected and creative uses of *SandCanvas*'s gesture recording function show that *SandCanvas* is truly a creative medium that goes beyond traditional sand animation.

The evaluation we have presented here has demonstrated the intuitiveness of *SandCanvas* and the effectiveness of our approach to modeling real-time interactions between hand and sand. But it does much more. It has also given us a deeper understanding of the affordances of this new medium, particularly the variety of gestures at users' disposal. Finally, it has shown how *SandCanvas*'s modeling approach, novel creative tools, and intuitive UI combine to produce an important new creative medium.

LIMITATIONS AND FUTURE WORK

SandCanvas works well overall, but it has a few limitations. Like most other diffuse illumination multi-touch tables, sometimes it triggers just before the finger touches the canvas. The system is also dependent on surrounding ambient light. We hope to improve our hardware and eliminate these problems.

Our study users suggested adding a gesture library to *SandCanvas*, giving them access to a larger collection of their own gestures and others created by fellow users or artists. Users also suggested making gesture recording hierarchical, so that complex gestures can be composed from simple ones. One of our users, a professional artist, commented that such library would allow novice users to quickly learn expressive sand animation techniques.

In the future, we plan to integrate video editing and special effects capabilities into *SandCanvas* to enable complete

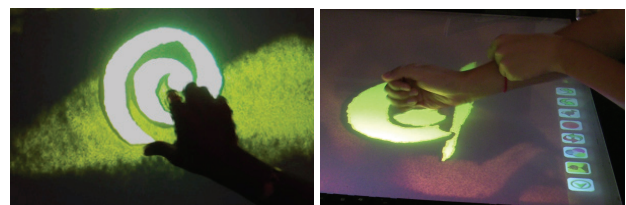


Figure 17: Two users' approaches to the same training task.

production work. Finally, with the increasing popularity and ubiquity of multi-touch devices (such as iPad), we plan to deploy *SandCanvas* to smaller multi-touch devices so that it can soon be used by more creative hands.

CONCLUSION

The elegance of sand animation lies in the seamless flow of expressive hand gestures that cause images to fluidly evolve, surprising and delighting audiences. While physical sand animation already possesses these properties, *SandCanvas* enhances them. *SandCanvas*'s color and texture features enable faster, more dramatic transitions, while its mixed media and gesture recording features make it possible to create entirely new experiences. Session recording and frame capture complement these capabilities by simplifying post-production of sand animation performances.

Producing this new artistic medium required us develop a new approach to real-time sand simulation that strikes a balance between speed and realism. It also required a simple and intuitive UI that would enable users to employ our new features effectively. Our evaluation of *SandCanvas* shows that we succeeded. When we analyzed it with respect to common sand animation techniques and our own taxonomy of gestures, we found it to be a genuinely rich artistic medium that enhances both professionals' and novices' opportunities for creative expression.

ACKNOWLEDGMENTS

We thank the professional sand animators who provided interviews, insightful comments, and permission to use sand animation images: Erika Chen (Figures 2, 3 left, 4 left, 5, and 6) and Sheh Meng (Figure 4 right). We thank Yi Bo, Haimo and other members in the NUS-HCI lab for their support. This research is supported by National University of Singapore Academic Research Fund R-252-000-375-133.

REFERENCES

1. Sand Animation: Interview with Cako. <http://www.sandanimation.com/page.php?2>.
2. (Ed.) CM-T, *Tabletops - Horizontal Interactive Displays*. Springer, 2010.
3. Bacher HP, *Dream Worlds: Production Design for Animation*. Focal Press, 2007.
4. Barnes C, et al. Video puppetry: a performative interface for cutout animation. In. *ACM SIGGRAPH Asia 2008 papers* 2008, ACM (2008), 1-9.
5. Bell N, Y.Yu, and P.J.Mucha. Particle Based Simulation of Granular Materials. *SIGGRAPH* 2005.
6. Besl PJ and McKay ND. A Method for Registration of 3-D Shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 1992; 14(2): 239-256.
7. Csikszentmihalyi M, *Flow: The Psychology of Optimal Experience*. (1st edition). New York: Harper and Row, 1990.
8. Danielson P-E. Euclidean distance mapping. *Computer Graphics and Image Processing* 1980; 14: 227-248.
9. Davis RC, Colwell B, and Landay JA. K-sketch: a 'kinetic' sketch pad for novice animators. In. *Proceeding of CHI* 2008, 413-422.
10. Efron D, *Gesture and Environment*. New York: King's Crown Press, 1941.
11. Hancock M, Cate T, Carpendale S, Isenberg T. Supporting Sandtray Therapy on an Interactive Tabletop. *Proceeding of CHI* 2010.
12. Hinckley K, et al. Design and Analysis of Delimiters for Selection-action Pen Gesture Phrases in Scriboli. *Proceeding of CHI* 2005, 451-460.
13. Igarashi T, Moscovich T, and Hughes JF. As-rigid-as-possible shape manipulation. In. *ACM SIGGRAPH 2005 Papers*, 1134-1141.
14. Kenyon H, *Animating under the Camera*, in *Animation Magazine*. 1998.
15. Krueger MW, Gionfriddo T, and Hinrichsen K. VideoPlace—an artificial reality. *SIGCHI Bulletin* 1985; 16(4): 35-40.
16. Lakin F. A performing medium for working group graphics. *Proceedings of the CSCW* 1986, 255-266.
17. Levin G, *Audiovisual Software Art: A Partial History*, 2009.
18. Levin G, *Painterly Interfaces for Audiovisual Performance*, in *Media Laboratory, School of Architecture and Planning*. 2000, Massachusetts Institute of Technology
19. Li X and J.M.Morshell. Modelling soil: Realtime dynamic models for oil slippage and manipulation. *SIGGRAPH* 1993.
20. Li Y, et al. Experimental Analysis of Mode Switching Techniques in Pen-based User Interfaces. *Proceedings of CHI* 2005, 461-470.
21. McCullough M, *Abstracting Craft: The practiced Digital Hand*. MIT Press, 1998.
22. Nelson Chu WB, Li-Yi Wei, and Naga Govindaraju. Detail-preserving paint modeling for 3D brushes. *Non-Photorealistic Animation and Rendering* 2010.
23. Onoue K and Nishita T. An Interactive Deformation System for Granular Material. *Computer Graphics Forum* 2005; 24(1): 51-60.
24. Richard T, *The encyclopedia of Animation Techniques*, 1999.
25. Ryokai K, Marti S, and Ishii H. I/O brush: drawing with everyday objects as ink. In. *Proc. of CHI* 2004, 303-310.
26. Schneider J, Kraus M, and Westermann Ru. GPU-Based Real Time Discrete Euclidean Distance Transforms with Precise Error Bounds. *VISAPP* 2009(2009).
27. Snibbe SS and Levin G. Interactive dynamic abstraction. In. *Proc. of the 1st international symp. on Non-photorealistic animation and rendering* 2000, ACM (2000), 21-29.
28. Summer RW, Brien JFO, and Hodgins JK. Animating Sand, Mud and Snow. *Eurographics* 1999(1999).
29. Ura M, et al. A Paint Tool for Image Generation of Sand Animation Style. *IEICE Tech* 2009(2009).
30. Vandoren P, et al. FluidPaint: an interactive digital painting system using real wet brushes. In. *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces* 2009, ACM (2009), 53-56.
31. Vandoren P, et al. IntuPaint: Bridging the Gap between Physical and Digital Painting. *IEEE TABLETOP* 2008(2008).
32. Wilson AD, et al. Bringing Physics to the Surface. *ACM UIST* 2008(2008), 67-76.
33. Wobbrock JO, Morris MR, and Wilson AD. User-defined Gestures for Surface Computing. *ACM CHI* 2009(2009), 1083-1092.