Gesture Play: Motivating Online Gesture Learning with Fun, Positive Reinforcement and Physical Metaphors

Andrew Bragdon[1], Arman Uguray[1], Daniel Wigdor[2], Stylianos Anagnostopoulos[1], Robert Zeleznik[1], and Rutledge Feman[1]

[1]Brown University
Providence, RI, USA
{acb, auguray, sanagnos, bcz, rchinfem}@cs.brown.edu

[2]Microsoft Research
Redmond, WA, USA
dwigdor@microsoft.com

ABSTRACT

Learning a set of gestures requires a non-trivial investment of time from novice users. We propose a novel approach based on positive reinforcement for motivating the online learning of multi-touch gestures: introducing simple, game-like elements to make gesture learning fun and enjoyable. We develop 3 metaphors, button widgets, animated spring widgets, and physical props, as primitives for simple, physically-based puzzles which afford the disclosure of static and dynamic hand gestures. Using these metaphors, we implemented a gesture set representing 14 of 16 gesture types in an established hand gesture taxonomy. We present the results of a quantitative and qualitative evaluation which indicate this approach motivates gesture rehearsal more so than video demonstrations, while memory recall was equivalent overall but improved in the short-term, for controlled tasks.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Human Factors

Keywords: Gestures, approachability, multi-touch, disclosure, learning

INTRODUCTION

Direct, multi-touch gestural UIs [31] offer a strong value proposition, including: gestural commands physically chunk command and operands into a single action [1]; bimanual interaction enables higher input bandwidth [2]; stroke-based gestures can be easier to learn and recall than keyboard-based ones [3]; and different commands can often be intermingled implicitly when gestures also specify command parameters (e.g., selection lassos and erasure scribbles while drawing). Gestures can also be committed to physical muscle memory which can help users focus on their task [4].

Despite these advantages, when given the choice to learn gestures/keyboard shortcuts or use a GUI alternative, users generally favor GUIs [5]. We believe that this stems from two factors: learning gestural interaction requires a significant investment of time and effort; and users perceive that this investment comes at too high of an immediate cost. Appert and Zhai found their participants required an average of 10 observations of a set of 14 pen gestures to achieve a recall rate[1] of approximately 80% [3]. Similarly, Freeman et al. saw a 67% recall rate[1] after 8 repetitions of a set of 16 touch gestures [6]. Even worse, a failed gesture attempt can discourage novices, making them less likely to continue to learn the system [7]. This leads to the conclusion that users who learn more than several application gestures and shortcuts are rare.

The goal of this work is not to reduce the investment of effort needed to master a gesture set (i.e., the number of repetitions needed), but rather to mask this up-front investment cost with an immediate, albeit ancillary, reward: fun − thus adapting Webster's hypothesis [31]. By altering the *perceived* cost structure of gesture learning, we expect more users will be motivated to learn more gestures even if they end up spending *more* time and effort learning.

Without such an immediate reward, we believe that users are daunted by the thought of having to repetitively practice a large set of unfamiliar gestures. However, by exposing gesture learning as a challenge consisting of a set of individually fun elements, we believe users may consider learning gestures as a welcome, engaging opportunity and ultimately more users will make the novice to expert transition.

Our approach is thus to motivate gesture learning through positive reinforcement. To begin to explore this space, we developed the Gesture Play system. Gesture Play is an online gesture learning system designed to be fun and engaging, but not overly distracting or addictive. To teach each application gesture, Gesture Play provides a simple, puzzle-like physical mechanism invoked from a toolbar similar to GestureBar [7]. Each puzzle utilizes simple physical metaphors – consisting of buttons, props and springs (Figure 1) – designed so that properly manipulating it equates to properly rehearsing its gesture. Physically simulating the components of each puzzle not only produces engaging behaviors, but also supports rapid puzzle mastery since users can apply intuitive reasoning. As further motivation to explore more gestures, we award collectible trophies when a user successfully completes a gesture, similar to web-based casual games [8].

Usage Scenario

Jane begins using an unfamiliar gesture-based application. She notices the Gesture Play toolbar and taps on a command name that is relevant to her task (see Fig. 1-1). Instead of performing the command, her tap opens a gesture puzzle inside a practice area that is safely isolated from her main work-area. She sees 5 metallic finger pads and 4 connecting springs beneath a semi-transparent hand overlay. Unfamiliar with multi-touch gestures, she pokes at the background of the puzzle with one finger. This causes the springs to "bounce"

[1] Values cited from graphs

ITS'10, November 7–10, 2010, Saarbrücken, Germany.

enough to focus her attention on the finger pads and hand overlay image (Fig. 1-2). Next, she tries using one finger to drag one finger pad; as it moves, the attached spring extends, turns red (Fig. 1-3) and eventually "slips off" her finger and bounces around losing momentum quickly (Fig. 1-4). Intrigued by this response, she tries placing her hand in the posture indicated by the hand overlay and notices it fade away (Fig. 1-5). Hesitantly, she starts crumpling her fingers and sees the springs turn green; reinforced, she continues crumpling until a "Nice Job!" tooltip animates onto the screen followed by a notification that she has won a trophy for completing the 5-finger crumple gesture (Fig. 1-6). She lifts her hand from the surface and the springs oscillate back to their original position. Enjoying this experience, she repeats it several times purely for her own amusement. She is then encouraged to attempt several other gestures which reward her with more trophies. Having developed a sense of mastery over the puzzles, she decides to complete her trophy collection by exploring all the system gestures.

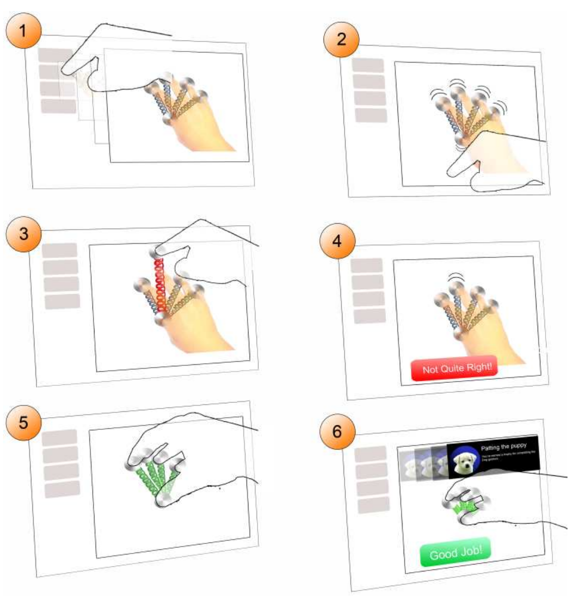

Figure 1. Gesture Play usage scenario, see above.

Contributions

We present the design of Gesture Play, a novel online gesture teaching tool that motivates learning through fun, engaging UI elements, along with two evaluations. A quantitative evaluation versus a control indicates that Gesture Play is as efficient as video demonstrations at teaching gestures while being engaging enough that users play with it more spontaneously, for controlled tasks. A qualitative evaluation indicates users prefer Gesture Play over video demonstrations.

BACKGROUND AND RELATED WORK

We review three areas of related work, including: gesture teaching, operant conditioning, and physics-based UIs:

Gesture & Accelerator Learning

Teaching pen gestures has an extensive background. Kurtenbach *et al*. [9] extended contextual crib sheets of available gestures with traceable *in situ* animated demonstrations. When a Help button is pressed, InkSeine [10] overlays gesture images on top of contextual UI targets. Marking Menus [4] teach a restricted gesture set as a by-product of interacting with a hierarchical, radial menu system. HoverWidgets depict hover path gestures emanating from a common starting point [11]. OctoPocus similarly uses colored feed-forward trails to show the next portion of available gestures, as unlikely candidates fade away [12]. GestureBar uses a toolbar to invoke a gesture explorer comprised of a practice area and annotated animations, further showing that purely gesture-based applications could be made approachable [7].

In the area of hand gestures, Charade conveyed gestures to users through a pictogram language [13]. Vogel *et al* used a video-based tutorial to teach in-air gestures [14]. Brandl *et al* used a crib sheet-style mechanism to reveal pen and multi-touch gestures [15]. Finally, ShadowGuides taught multi-touch gestures using a crib sheet in tandem with annotated feed-forward similar to OctoPocus [6] and demonstrated a performance improvement over video-only; however it requires up front training and thus is not walk-up-and-use.

ShadowGuides also proposed a taxonomy of surface gestures and a set of example gestures spanning that taxonomy. We adopted this gesture set and their evaluation protocol of comparing to video-based demonstration.

Each of these systems, except Marking Menus, was evaluated in contexts where tasks could only be performed gesturally. Our work differs by addressing the issue of motivating arbitrary multi-touch gesture learning when familiar and initially simpler non-gestural alternatives are available.

Physics-Based UIs

Several projects use physics-driven behaviors as UI elements. Early work includes physical treatment of windows [18] and a 3D paper flyer metaphor [19]. Recent examples, BumpTop [20] and "Bringing Physics to the Surface" [21], use a physics engine to create realistic effects. Magic Paper allows users to create and play with 2D physics scenes using pen input and gestures [22]. Jacobs *et al* hypothesize that the appeal of these UIs stems from their being consistent with the user's understanding of the natural world – a principle dubbed "Reality-Based Interfaces" [23].

Affective Computing

Work in affective computing has previously posited that fun learning experiences may lead students to spend more time in an experience and thus increase learning [24]. This notion is the foundation for Gesture Play, although to our knowledge, has not been applied to online gesture learning.

Operant Conditioning

Grossman et al. applied paired-associate learning to accelerating the online learning of hotkeys [5]. We instead approach the problem of motivating learning using operant conditioning theory concepts that we briefly outline [16]:

An operant response is a behavior that is modifiable (increased or decreased likelihood) by its consequences. There are typically four types of consequences:

Positive Punishment: an attempt to decrease the likelihood of a behavior recurrence by presenting an aversive stimulus after the behavior (operant response) occurs.

Negative Punishment: an attempt to decrease the likelihood of a behavior recurrence by removing an appetitive stimulus after the behavior (operant response) occurs.

Positive Reinforcement: an attempt to increase the likelihood of a behavior recurrence by presenting an appetitive stimulus after the behavior (operant response) occurs.

Negative Reinforcement: an attempt to increase the likelihood of a behavior recurrence by removing an aversive stimulus after the behavior (operant response) occurs.

Under operant conditioning theory, the traditional method of teaching hotkeys could be characterized as negative reinforcement: by learning the hotkeys (behavior), users can avoid using the menu (the aversive stimulus). Since many users learn very few or no hotkeys, Grossman *et al* explored accelerator techniques. The two best-performing techniques, audio shortcut reminders and disabled menu items in effect utilize negative reinforcement and positive punishment, respectively, to motivate users to learn the shortcuts. The audio reminder serves as a distracting reminder (aversive stimulus) that the user should learn shortcuts (but still allows them to use the menu) as well as a memory aid`, while the disabled menu presents the user with a dead-end when they attempt to execute a command without shortcuts (punishment).

We note that negative reinforcement and punishment are inherently unpleasant and possibly less motivating to the user in the short term. We therefore argue in favor of positive reinforcement because it has the potential to motivate learning through a pleasant user experience.

Typing tutors and Giraffe [17] are notable applications of positive reinforcement to UI learning. Typing tutors motivate touch typing with gaming elements that make repetitive exercises seem more fun than tedious. Giraffe applied similar techniques to teaching the Graffiti text input language. Both of these approaches require offline learning which we and previous authors believe has disadvantages. Online learning is lightweight and allows the user to learn as few or as many gestures as they want, when they want. This eliminates up-front training, a potential barrier to adoption, by enabling learning-while-doing. We are unaware of prior investigations into making online gesture learning fun or game-like.

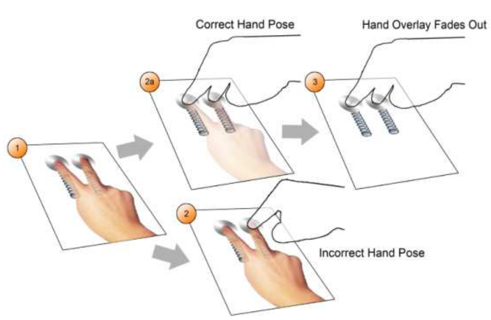

Figure 2. (1) Semi-transparent (80% opacity) Hand Overlay Images afford initial gesture hand pose. (2) Incorrect pose does not hide overlay to reinforce differences. (2a) Correct pose hides overlay

DESIGN OF GESTURE PLAY

Our goal is to harness the fun quality of games for a purpose. ESP [25] used this approach for human computation, while our goal is to motivate the user to learn gestures. Webster [32] argues that introducing play may lead to improvements in workers learning computer systems – we apply this notion to learning gestures.

Gesture Play attempts to find a “sweet spot” (based on the MDA framework for difficulty level [30]) in which gaming elements are neither too self-disclosing, in which case users might not feel challenged enough to perform any actions, nor so involved as to make the system become addictive, overly difficult, or off-putting. This sweet spot needs to encourage users to spend just enough time to become actively engaged and to experience amusement, but not more.

Design Principles

We identified four guiding principles for a fun, online gesture learning system:

- *Short-Term Fun (G1)*: individual gesture learning techniques should be fun and engaging
- *Long-Term Fun (G2)*: overall progress learning a gesture set should be rewarded
- *Casual (G3)*: system mechanics should be easy to approach and learn for a broad range of users
- *Minimize Distraction (G4)*: gesture learning should consist of brief, interruptible activities

Our strategy was to conceive of gesture learning as a system of mechanical puzzles, where each could be readily solved in very few attempts (G4). To facilitate approachability (G3), we use only two metaphors, based on “everyday physical objects” of which users possess *a priori* knowledge – springs (with physical props) and buttons. Such physically-based puzzles allow users to reason about them in a natural way [23] and effectively transform all gestures into physical manipulations [26]. In addition, to encourage repeat interactions (rehearsal), the physically-based puzzles produce engaging animated responses to user input, for both correct and incorrect gesture performances and employ game-like graphics and animations where needed (G1). To motivate longer-term exploration, we award collectible trophies for each learned gesture (G2). We do not use audio except for a short bell sound when a trophy is awarded (G4).

Gesture Play Primitives

In this section, we present the design elements which comprise Gesture Play, and how they are derived from our design principles. We also describe iterations that are the result of early pilot testing where illustrative.

Initial Posture Affordance: Hand Overlay Images

When a puzzle is first shown, semi-transparent hand overlay images indicate the initial hand pose (Figure 2). When the user assumes the correct initial pose, the hand overlay fades out. By extension, if a user assumes an incorrect initial hand pose, the correct hand pose will continue to be visible side-by-side with their actual hand pose, helping to illustrate the difference between what they are doing and what is correct.

In initial pilot testing without overlays, users sometimes did not assume the correct initial hand posture. This first led us to add non-realistic depictions via an overlay, similar to [6], and while this improved almost all of the problems, users did occasionally misinterpret the 3-D nuances of the pose, such as whether the palm should be flat on the surface or poised above it. This led us to choose photographic representations to indicate hand placement nuances, such as the 3-D pose of the hand, which addressed this issue.

Once the user's hand is in contact with the display in the proper posture, the system affords the next step of the gesture. Freeman *et al* classifies this next step as either dynamic or static [6]. In the dynamic case, the system must afford some change in the position and/or posture of the hand in contact with the display. In the static case, the system must afford removing the hand from the display.

Static Posture Affordance: Button Widgets

For gestures which require only a static pose, we use the button widget metaphor (Figure 3). Buttons should always be pressed and then released. When one button is pressed (or if the user "misses" and touches the background), any remaining buttons will flash red. Once all buttons are pressed and held, the buttons turn green, indicating the user can release.

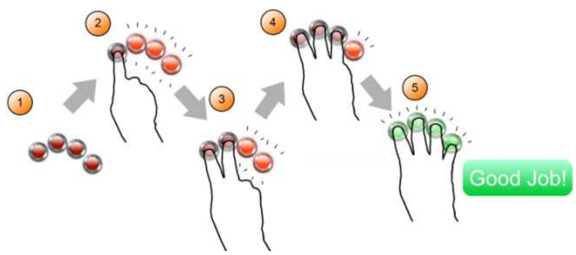

Figure 3. (1) Button widget affordance, (2) user places a single finger, causing remaining buttons to light up, (3) (4) adding additional fingers, (5) buttons turn green, success notification is shown.

For gestures involving a particular pose, such as corner hand side or 2-palms flat, a button of a corresponding characteristic shape are shown, e.g. an L-shaped button, or a "mitten"-shaped button (Figures 8-H and 8-G). This is similar in nature to that shown in [28]. Distinguishing between dynamic and static gestures at this stage is an innovation over prior work, where users were simply induced to place their hand in the appropriate posture and await further instructions.

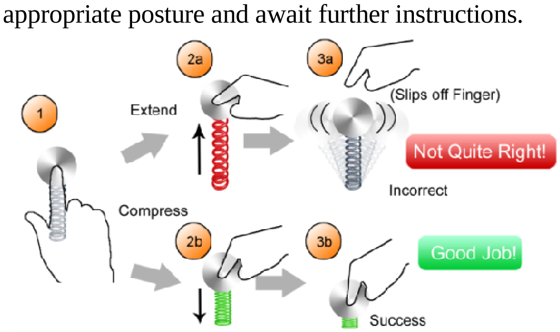

Figure 4. Spring widget states: (1) User assumes posture. User extends (2a) or compresses (2b) spring causing visualization to change. (3a) Spring overextends, slipping off user's grip and "bouncing." (3b) User compresses spring fully, and is notified of success completing gesture.

Dynamic Posture Affordance: Spring Widgets

For gestures which require movement following the initial contact, we use the spring widget metaphor (Figure 4). Springs can be connected at each end to an object, typically a contact pad, which can be manipulated by touch input. Springs are typically pinned to the screen on one end. We always associate desired movement with compressing a spring, in which case it turns green. If springs are extended, corresponding to a "wrong" movement, they turn red and eventually "slip off" the user's finger. Springs are procedurally animated using a 2-D physics model; dragging a spring's contact pad takes the spring out of equilibrium as it stretches or compresses to maintain attachment to the pad. When the user releases, the spring system is not in equilibrium and it "bounces around" for several seconds dissipating momentum until equilibrium is restored. Touching the background (missing a contact pad), stimulates an "earthquake" effect by briefly taking the spring system out of equilibrium to draw attention to the pads (we observed that novice multi-touch users often "poke" furtively at the display at first). The reactive, simulation nature of the spring widgets produces animated results unique to the user's input.

When a gesture is performed incorrectly, a tool-tip fly-in indicates the performance was "Not Quite Right," much like [7]. When performed correctly, a fly-in says "Good Job".

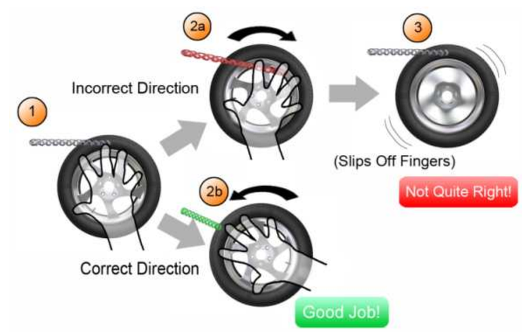

Figure 5. Physical props afford and constrain input. (1) User assumes posture, (2a) rotates right or (2b) rotates left, causing spring visualization to change. (3) Incorrect performance causes a notification and spring to bounce when contact is released.

Dynamic Posture Affordance: Physical Props

To clarify interactions and to augment the linear motions implied by springs, we utilize additional physical props for some gestures. Like springs, these props are based on "everyday things" to leverage *a priori* knowledge about how to interact with them [27]. Unlike springs, they are not interactive by themselves, but become interactive when attached to a spring(s). For instance, the 5-finger rotate gesture puzzle connects a wheel prop radially to a spring (Figure 5) to allow rotation about a pivot at its center. Thus, the wheel transduces linear spring motion into circular motion. The opening palm gesture, illustrated with a hand overlay gripping a block of wood which appears to be screwed to the background, indicates that the grip should be opened *without* moving the palm.

Props also add variety without requiring the user to learn an entirely new metaphor for each gesture. The finger pads mentioned above are essentially very simple props and so in essence, at least one prop accompanies every spring widget.

Additional Contact Affordances: Progressive Disclosure

For some gestures, such as *1-finger right, add 2nd finger pull down*, additional progressive disclosure is needed. For such gestures, we show additional affordances in a feedforward

[12] manner once they become needed (Figure 6). Once the first part of the gesture is complete, the second part appears with an attention-getting "shrink" animation, showing additional widgets, and a new hand overlay. Feed-forward can also be used with buttons (Figure 9-L). In this example, once the user depresses the button, a second button appears with an animation. A new hand overlay appears showing a second finger pressing the new button. The user then presses this second button, at which point both buttons turn green, indicating the user can release.

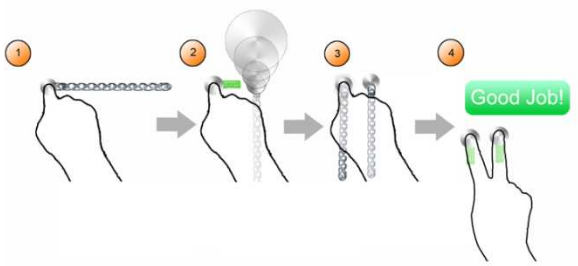

Figure 6. (1) User begins compressing spring. (2) Animation displays feed-forward of a spring and new hand overlay (not shown). (3) New spring discloses second gesture step. (4) User assumes new posture and completes gesture by compressing springs.

Long-Term Engagement: Trophies

To further motivate users to return to learn more gestures, we provide a collectible trophies system. When the user successfully completes a gesture, they are awarded a unique trophy (Figure 7), which includes an icon, name and text, similar to many video games [8]. The trophy eventually collapses into a box which shows the last three trophies earned, a counter indicating how many trophies have been earned, and the total number available. Touching this box opens a full list of all trophies, including grayed-out entries for ones that have not been awarded yet. Trophy names make use of word plays on the original command name, much like video games that award trophies [8].

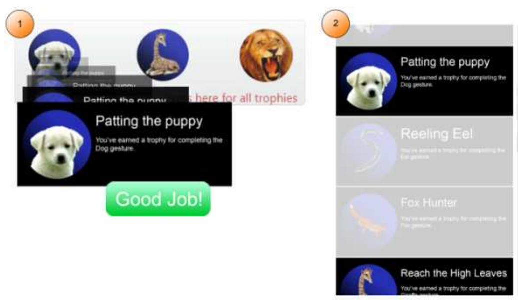

Figure 7. (1) User completes Dog gesture and gets a trophy. (2) User presses trophy button to reveal full trophy list, including ones not yet earned.

Gesture Set

We implemented 16 unique gestures (Figure 8), based on the taxonomy-spanning set developed in [6]. Notably, we excluded pure path gestures (e.g. pigtail) because they do not naturally fit the spring metaphor without further extensions or violations of how springs work in the real world (*e.g.*, springs as complex paths). We leave path gestures to future work, perhaps adapting an OctoPocus-like [12] approach. We also did not implement any gestures that utilize absolute timing information, as these are not covered in [6]. The 16 gestures implemented in Gesture Play represent 14 points in the taxonomy. These 14 gesture types are afforded using only spring and button widgets metaphors coupled with physical props where appropriate. Still, some gesture types may require additional metaphors, for instance above-the-surface interactions that involve 3-D information.

Similar to GestureBar, the help system is invoked by means of a persistent toolbar (Figure 1). When a button is clicked, the gesture puzzle for that command is displayed in a practice area; interactions there do not affect the application, so users can safely "play around" as they learn the gesture.

Gesture Recognizer Support

Gesture Play is best-compatible with gesture recognizers that can provide interactive recognition with Boolean output. If the recognizer identifies a partially complete gesture as being wrong, Gesture Play will respond by causing the contact pads to slip off and/or by notifying the user that the gesture was incorrect. Gesture Play provides the recognizer with the context of which gesture the user is learning.

IMPLEMENTATION

Gesture Play was implemented in Windows Presentation Foundation (WPF) using a Microsoft Surface, with IR-cameras for sensing, and a 1024x768 rear-projected screen. The physics setup uses the Farseer 2-D physics engine [29].

We wanted the user experience with the spring widgets to produce similar responses no matter which direction the user dragged the contact pads. Thus each spring that is visible to the user is implemented as four invisible 2-D spring models where one end of each spring is attached to a common point on the object in question (with the exception of rotational elements, which use two springs). The other end of each spring is attached along a fixed distance away from the common point along 4 cardinal directions (*i.e.*, the springs are arranged in a '+' shape centered on the object). While this is not a perfect simulation of reality, in pilot testing we found that this matched the user's anticipated response. Props are manipulated in the following manner: when the user touches the prop, miniature, invisible 2-D spring models attach from each finger contact point, to the touched point on the prop. The approach used in [21] could also be explored.

For simple gestures, the gesture recognition is fully implemented in software. More complex gestures are recognized using a Wizard-of-Oz approach based on that used in [6], in which a Wizard uses a screen (not visible to the user) to perform recognition. As in [6], the screen shows only sensor data from the Surface camera (the Wizard does not see anything a recognition algorithm would not see), using a pre-defined, consistent recipe for performing recognition. The same Wizard was used for all experiments.

EVALUATION

We hypothesize by breaking up the learning of each gesture into a separate, puzzle, and tying the learning of multiple gestures together through collectible trophies, users will find learning each gesture fun, and also be motivated to return to experience each puzzle and collect the trophies. We further hypothesize that users will perform more gesture rehearsals

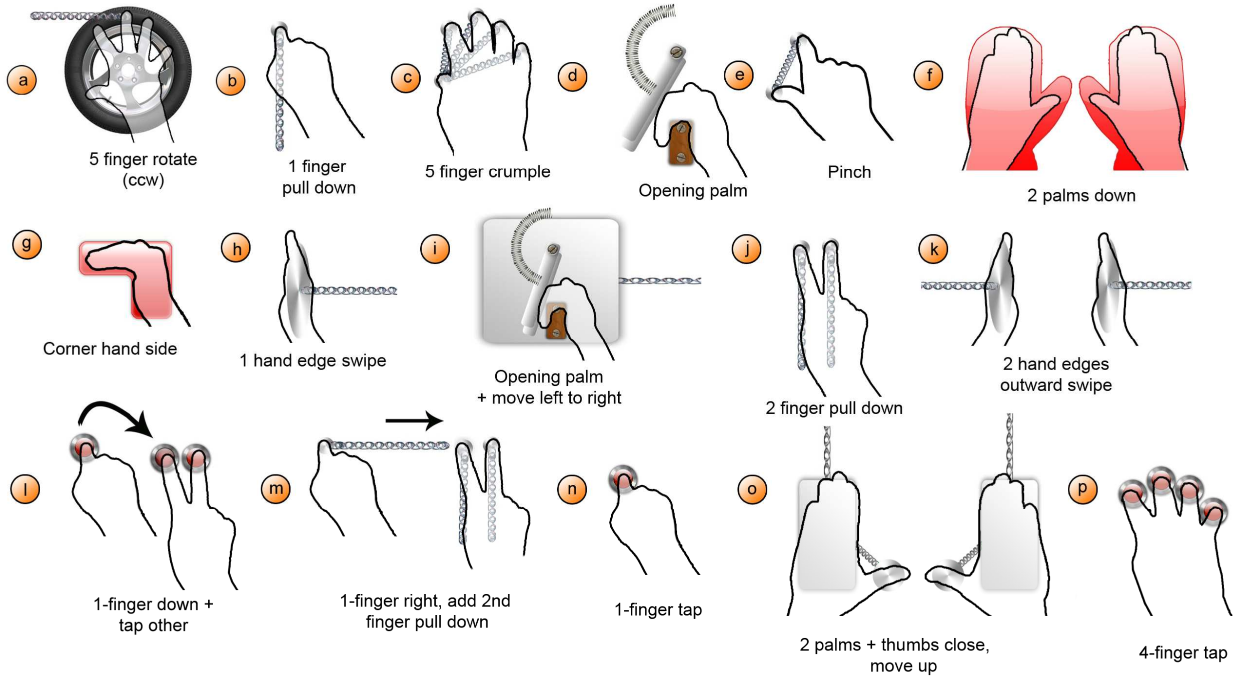

Figure 8. The 16 gestures implemented in Gesture Play, based on the taxonomy from [6]; widget designs are shown with initial postures/hand overlays.

using Gesture Play, as they interact and "play with" the gesture puzzles (thereby rehearsing the associated gesture), than when using video demonstrations (control). Moreover, we hypothesize that recall rates with gesture puzzles will be at least as high as with the control. We also hypothesize that, unlike ShadowGuides and other previous multi-touch gesture teaching methods, Gesture Play will be immediately approachable, not requiring a tutorial on the on-screen visuals before the user can begin to use it. In addition, we hypothesize that despite being slightly more effortful to learn with, that Gesture Play will be preferred by users.

Thus, the goal of our evaluation is to test the following:

H1: Gesture Play will be on-par with established techniques in terms of recall.

H2: Gesture Play will motivate users to rehearse a gesture

H3: Gesture Play will be immediately approachable: users will have few errors on the first attempt to perform a gesture, without being instructed.

H4: Gesture Play will motivate users to learn more gestures overall.

H5: Despite requiring greater effort in some cases, Gesture Play will be preferred due to its motivational advantages.

To evaluate these hypotheses we conducted two experiments, one quantitative, and one qualitative.

EXPERIMENT 1

Our goal with this experiment was to evaluate H1 (measure recall rates) and H2 (motivate gesture rehearsals). We modeled our methodology after that demonstrated in [6].

Participants and Equipment

We recruited 20 participants (compensated) from the undergraduate population of Brown University (13 female, aged 19-28) using Internet ads. We excluded students majoring in Computer Science and related fields, and advertised widely to get participants from a broad range of backgrounds. Participants had no experience with anything more advanced than an iPhone. Three participants played computer games regularly. Participants used a Microsoft Surface multi-touch computer (see Implementation, above).

Study Design

We used a between-subjects design in which participants were randomly assigned to use either Gesture Play or an alternative teaching tool consisting of video demonstrations, mimicking the baseline tool used in [6] (as in [6], no video was longer than 3 seconds). Videos are activated via an identical toolbar to Gesture Play, and have a replay button.

The current task was displayed at the bottom of the screen; once completed a Next button appeared allowing users to go on to the next task. The concept of a gestural command was explained to users. Unlike with previous multi-touch gesture-teaching systems, where participants were taught the annotations and affordances in advance, no introduction to each help system was given other than informing participants it was available and how it could be invoked. The trophy system was disabled in this experiment. As in [6], each gesture was given a neutral animal name, such as "horse" or "cat".

Part 1: Learning Performance & Approachability

The goal of this portion of the experiment was to test H1: that learning performance with Gesture Play will be at least equal to that of the video teaching tool.

After a pre-questionnaire and introductory statement, participants were randomly assigned to one of the two teaching techniques. They were then asked to complete a series of trials, comprising 12 blocks of 16 trials each. Each block contained each of the 16 gestures in a random order. Between blocks, we toggled the availability of the help toolbar, providing six blocks of learning trials, and six blocks of memory trials, allowing us to measure recall progressively. As in [6], participants were not asked to memorize the gestures. Since our study differed from [6] in that there were multiple memory trials, we instructed participants to successfully complete a gesture only once in a given trial (preventing uncontrolled rehearsal). During the memory trials, self-rated confidence on a 5-point Likert scale was recorded after each gesture but before the success/failure feedback was shown. Users were required to repeat a gesture until they performed it correctly, or gave up. In summary, our experimental design was as follows:

2 learning techniques
x 10 participants per technique
x 12 blocks of trials
x 16 gestures per block
= 3,840 correct gestures completed

Part 2: Propensity for Play
The goal of this portion of the experiment was to test H2: that users will be inclined to practice gestures more with Gesture Play than with the video teaching tool.

At the conclusion of the 12 blocks, participants were verbally asked to perform 6 gestures using their assigned help system. After each gesture, participants were asked two qualitative questions about their opinion for that gesture. The number of times the user performed the gesture was recorded as a measure of the participant's propensity for play with the tool. We believed that if such propensity existed after completing 192 trials (after which presumably any novelty would have worn off), that would demonstrate a strong effect.

Part 3: Questionnaire
Following Part 2 of the experiment, participants were asked to complete a questionnaire.

The goal of the questionnaire was to determine whether the system they used was difficult to learn, whether they felt it was helpful, and whether the tasks were easy to complete. We also asked them to rate how 'fun' the learning system was. This qualitative feedback can be taken-up by designers in considering the Gesture Play technique.

Results
As our data was non-parametric, discrete, and did not appear to fit a known distribution, we used a 2-tailed Mann-Whitney test to test for differences between conditions

Part 1: Learning Performance & Approachability
Looking at recall rates, there was no significant difference in recall from the memory trials between conditions for memory blocks 2-6 (Figure 9)[1], although the averages for Gesture Play were higher for all trials (an average of 5.02% higher, σ^2 = 4.39%). Interestingly, there was a significant difference for the first block, with Gesture Play showing 164.3% greater recall (Z=2.28, p<0.05); however, the overall number of gestures remembered was relatively low, just under 25% (not surprising, since they had seen and practiced each gesture just once at that point). Mirroring this, the user ratings of memory confidence on a 5-point Likert scale showed a similar trend, with a higher average rating for Gesture Play in all

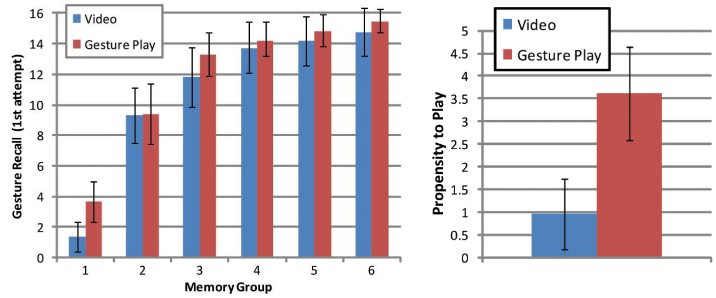

Figure 9. (Left) Learning performance; number of gestures recalled vs. block. (Right) Propensity to play: number of gestures performed unprompted in Part 2.

trials (an average of 9.81% higher, σ^2 = 6.42%), however these differences were not significant[2]. These results thus support H1: that Gesture Play is at least as effective as established techniques at teaching gestures.

We note that although ShadowGuides showed significantly better recall than the video technique, we observed much better recall for video tutorials than they reported [6]. We posit that this may be due to differences in experimental design or user populations. Gesture Play performance is on-par with ShadowGuides in the number of rehearsals.

We now examine approachability [7]. In particular, we look at the number of failed attempts, on average, it took to get the gesture right during the first trial, when the user was most inexperienced. There was no significant difference between video and Gesture Play (Z=1.98, p=0.05) (note that this was borderline, but did not satisfy the α=0.05 threshold). The number of failed attempts was on average quite low for Gesture Play, $\overline{x} = 0.29$ $(\sigma^2 = 0.33)$ and video $\overline{x} = 0.1$ $(\sigma^2 = 0.15)$. These results thus also support H3: Gesture Play is approachable without requiring *a priori* instruction on its disclosure mechanisms.

Part 2: Propensity for Play
As described above, after the 192 trials were complete, users were requested to perform 6 gestures and were asked two controlled questions after they performed each. Gesture Play users on average performed the gesture 3.6 times, or 273.7% more than the video control (Figure 9). This difference was statistically significant (Z=3.03, p<0.01). These results thus support H2: that Gesture Play encourages users to practice the gesture for the sake of interacting with the system.

Part 3: Questionnaire
There was no significant difference between subjects in terms of difficulty to learn the help system (Z=-0.92, p=0.36), the helpfulness/not helpfulness of the help system (Z=-0.86, p=0.39), and the ease of completing the learning task (Z=-0.47, p=0.64).

However, there was a significant difference in the responses to the question of how fun or not fun the system was, with Gesture Play participants reporting an average of 4.0 (somewhat fun) versus an average of 3.1 (neither fun nor not fun) for video (Z=-2.21, p=0.027). These results support the core fun aspect of Gesture Play in H4.

EXPERIMENT 2
Our goal with this experiment was to directly compare Gesture Play and video demonstrations to collect qualitative

[1] Significance values for recall rates for memory block 2: (Z=-0.038, p=0.97), block 3: (Z=0.66, p=0.50), block 4: (Z=-0.24, p=0.81), block 5: (Z=0.08, p=0.93), block 6: (Z=-0.44, p=0.65)]

[2] Significance values for confidence ratings for memory block 1: (Z=0.947, p=0.34), block 2: (Z=0.41, p=0.67), block 3: (Z=0.72, p=0.46), block 4: (Z=0.46, p=0.64), block 5: (Z=-0.64, p=0.52), block 6: (Z=.17, p=.86)

feedback as a test of H4 (motivate more learning) and H5 (prefer to previous techniques). Unlike Experiment 1, the Gesture Play Trophies function was enabled. The experiment was meant also to evaluate the difficulty of the Gesture Play system in the context of the MDA Framework's "sweet spot" for difficulty level [30], and provide additional insights into the Gesture Play technique for future implementations. This was accomplished through think-aloud responses, interview questions, and Likert-scale responses.

We recruited 9 participants from the undergraduate population of Brown University (5 female, aged 18-22) using Internet ads (same restrictions as Experiment 1). We used a within-subjects design in which participants executed 8 gestures with Gesture Play, and 8 gestures with the video method. Gestures were randomly assigned to each block for each participant. As with Experiment 1, participants were instructed on the nature of gestural commands but were not instructed on how to use either help system, beyond how to invoke it using the toolbar buttons. Unlike Experiment 1, participants were allowed to 'play' with the system throughout the experiment, and repeat gestures which had been completed correctly. Participants were asked to think-aloud. At the conclusion of the study, which took approximately 40 minutes, participants completed a questionnaire. Where necessary, Gesture Play was referred to by the name "springs help system" and videos as the "video help system." Equipment used was identical to Experiment 1.

Results and Observations

User Preference

8 of 9 participants described Gesture Play as "fun," "engaging," and "rewarding" to solve the "easy puzzles," and that solving puzzles produced a "sense of accomplishment". When asked which of the two help systems they preferred overall, 6 of 9 participants chose Gesture Play, supporting H5. The 3 participants who chose video thought that Gesture Play was "a little more difficult," since the puzzles needed to be solved, and they preferred the minimalist nature of videos.

Play as Motivation

7 of 9 participants stated that the fun and engagement they felt when using the springs would motivate them to try and play with more gestures, thereby learning more total, supporting H4. The 2 participants who chose video mentioned that although Gesture Play was fun, this was not a strong motivating factor for them. 4 of 9 participants felt that with Gesture Play, they felt "less penalized for being wrong".

Unprompted (and unguided), all but 1 participant described Gesture Play as "fun" to use. No participants described the video condition as fun. This gives confidence that the responses for fun were for the teaching system itself and not for gestural interaction in general. In addition, we observed the participants universally "played with" Gesture Play, often executing a gesture numerous times after completing it, unprompted, consistent with Experiment 1. In addition, beyond repeating the gesture successfully, participants often actively explored the range of possibilities for the gesture, and experimented with whether other related motions would be recognized as being part of that gesture.

In contrast, videos generated no repeat performances in virtually all cases. Participants had a very strong tendency to move on to the next gesture after completing each successfully. In the relatively rare cases where participants did repeat a gesture before moving on, in virtually all cases they performed it just 1-2 additional times. 3 of 9 participants mentioned that they did not like waiting for the videos to complete, despite the fact that no video was longer than 3 seconds, and went on to say that they preferred Gesture Play because they were actively engaged the whole time and "did not have to wait." These participants went on to say that it was "annoying" to watch "someone else do a really easy thing and then copy them." One participant stated the video technique "makes you feel like a child, in a bad way." This supports our goal of hitting a "sweet spot" of difficulty as advocated by the MDA framework.

Trophies as Motivation

Three of 9 participants felt the trophy system was interesting and motivating to them, while the remaining two thirds did not feel they were motivating. The participants who were interested in the trophies felt motivated to collect them, with one participant saying that although they had no "actual value" to her they were fun to collect and she felt compelled to do so even though this was somewhat "irrational." Two participants suggested receiving "awards" as part of the trophy system, such as downloadable backgrounds or skins for the application, or even social networking website integration.

All 7 of the users who stated that Gesture Play was more motivating than videos also stated that the puzzle/game-like aspect was the primary source of motivation, rather than the trophies. When asked if they felt trophies would add much to the video help system, participants felt it would add little since they had "no sense of accomplishment" after replicating a motion from video. Participants did feel that the trophies system gave a sense of progress, by which they could measure their own knowledge about a program based on how many trophies had been unlocked.

Difficulty and the MDA Sweet Spot

While 7 of 9 participants preferred Gesture Play, 4 of 9 stated that the videos technique made it easier to learn gestures. As explanation, participants stated that with videos, they were required to "imitate" the video, whereas for Gesture Play they need to "figure it out on your own." Several participants stated that this made Gesture Play more difficult, but later went on to say this made Gesture Play feel rewarding, as predicted by the MDA Framework [30]. Others felt that the Gesture Play puzzles were "easy" and "intuitive." When asked whether the puzzles were too hard, too easy or just about right, 1 of 9 participants felt they were too difficult, 1 of 9 too easy, and 7 of 9 "about right."

In terms of the potential for each help system to be distracting in the context of a real application, no participants felt either would be a serious distraction, while 3 participants felt Gesture Play would be more distracting than videos. These participants said they needed to think more to learn with Gesture Play than with videos.

DISCUSSION

We believe that the support for our hypotheses gathered in both experiments indicates that Gesture Play can help motivate users to learn gestures and represents a potential improvement over previous techniques.

Threats to Validity (Experiment 1)

The controlled nature of the study places obvious limitations on the generality of results. The fact that participants were tested multiple times in memory trials may have biased them to memorize the gestures more than they would in an actual application; however this effect was equal for both conditions. This effect was ameliorated by the fact that we did not allow users to rehearse or practice each gesture more than once per trial. The population used, university students, may not be representative of other populations. There was no significant difference in self-reported computer expertise between experimental and control groups.

Recall, Performance and Approachability

Gesture Play showed no significant difference in recall rates compared to video in each memory block, though on average the recall was 5.02% higher for blocks 2-6. The exception was the first memory block, in which the difference (164.3%) was significant. This is notable since it shows users can learn the gestures at least as well with Gesture Play as with videos, even though they may think about the gesture puzzle while performing the gesture. Given the improved performance after just one rehearsal, it is possible that with small number of gestures/short usage scenarios, users might learn gestures with fewer rehearsals given that greater levels of processing are required to solve the gesture puzzle. We leave this exploration for future work.

It is also notable that the average number of failed attempts for Gesture Play, on the first trial when the user was first exposed to the system, was so low, 0.29 failed attempts per gesture learned, given the lack of instruction, and the puzzle-like nature of Gesture Play. Indeed, the fact that there was no significant difference between Gesture Play and videos was quite surprising, and suggests that the physical metaphors of spring and button widgets, used in tandem with props, was effective in teaching the gestures, and specifically in communicating the requirements of the gesture, and could support a walk-up-and-use learning scenario, as described in [7].

The number of initial failed attempts differs from [7], however. We attribute this to the fact that we did not simulate a walk-up-and-use scenario on a full-fledged application, and also to the fact that the pen gestures used in [7] were more complex to perform, parameterized, and included a number of essential nuances in each gesture.

Perceived Cost Structure and Motivation

Gesture Play significantly outperformed videos in terms of the propensity to play, and to repeat gestures unprompted, by 273.7% more – even after 192 trials, certainly sufficient time for the novelty of a gestural user interface to wear off. Indeed, it is notable that the "effort" required to repeat a gesture is identical between the video and Gesture Play help systems, since the hand motion on the screen is the same. Thus, since the actual cost structure is identical, it is clear that Gesture Play lowers the *perceived* cost structure for the user, or conversely, motivates the user to a greater extent to repeat and explore the gesture. This was further borne out by the qualitative study, which showed that all but two participants felt Gesture Play was more motivating to learn gestures overall, and was further preferred as the learning system of choice despite requiring more physical effort.

When taken together, we believe these results strongly suggests that in ecologically valid settings, users would perform the number of rehearsals required to commit gestures to memory, sooner, and with much greater likelihood, using Gesture Play than video-based demonstrations. However, the fun or slightly increased difficulty of Gesture Play might also have distracting side-effects. We believe these initial results motivate future work to explore performance in ecologically valid settings, and with more diverse user populations.

Mnemonics & Learning Methods

Previous work has shown that users remember gestures via mnemonic stories [3]. However, the significant difference in recall rates after one trial may indicate that users initially rely on other methods. We hypothesize that Gesture Play creates stronger initial short-term memory impressions than video demonstrations since users need to think more about the gesture puzzles than the videos. However, we further hypothesize that when users become aware of their need to remember, they begin to create mnemonic stories and the method of presentation becomes far less important for recall. Thus, after the second memory trial, video and Gesture Play perform on par with each other. During interviews at the conclusion of the study, we found that participants from both the experimental and control groups relied primarily on mnemonic stories to remember gestures, and furthermore, that their stories were often very similar despite the major differences between the way the help systems teach gestures. For example, almost all users said they remembered the "horse" gesture because the L-shape hand posture closely resembled "the back of a horse."

User Preferences

Interestingly, despite the advantages that Gesture Play exhibits for motivating gesture rehearsals and providing a fun user experience, 3 of 9 participants still preferred video demonstrations overall. We hypothesize the preference for video was driven by short-term cost structure: videos may require less conscious thought to interpret gestures. This leads us to believe that there are at least two groups of users: those who prefer a fun and thus motivating process, and those who like the "easiest" possible option in the short-term. Further work is needed to confirm this hypothesis.

Fun as a Design Principle

The quantitative study indicated that, between subjects, Gesture Play was significantly more fun (rated 4.0, "somewhat fun") than videos, which were not fun for users (rated 3.1, "neither fun nor not fun"). Our studies also indicate that participants were strongly motivated to perform additional repetitions and explorations of each gesture to experience the fun, amusement, and "sense of accomplishment" produced by

Gesture Play's physical metaphors and trophy system. Thus, we propose the following design principle, extending [32]:

Motivation through Scoped Fun User Interfaces Principle: users will be more motivated to explore, play with, and learn unfamiliar user interfaces if such interfaces use the positive reinforcement of a fun, engaging, and rewarding experience that is of sufficiently short duration to not distract from the primary tasks, but long enough to be engaging.

We did not observe anything to suggest that this principle would be restricted to the domain of learning gestures; we therefore hypothesize that this principle applies broadly, beyond gesture learning, to user interfaces in general.

We propose that this generalized fun principle could become valuable in UI design and could be applied broadly to post-WIMP and WIMP user interfaces alike. Applying design techniques used to construct games, such as the MDA Framework [30], may represent a fundamental HCI advance.

CONCLUSION

We have presented the design of Gesture Play, which uses the positive reinforcement of fun, physical metaphors, including spring widgets, button widgets, and physical props, to teach gestures. Our quantitative evaluation indicates that users have a significantly greater propensity to play with and rehearse gestures using Gesture Play than with video demonstrations, and further that memory recall is equivalent to video demonstrations, while short term recall after a single rehearsal is improved. Our qualitative evaluation indicates that users felt Gesture Play was more motivating for learning gestures because of its "fun" nature. Finally, we propose a design principle for the creation of fun user interfaces.

ACKNOWLEDGEMENTS

We thank Andries van Dam, Dennis Wixon, Kay Hofmeester, Hrvoje Benko, and Dustin Freeman for their assistance. This material is based upon work supported under a National Science Foundation Graduate Research Fellowship, by a Google Research Award and NSF Grant IIS-0812382.

REFERENCES

1. Buxton, W. Chunking and phrasing and the design of human-computer dialogues. In *IFIP WCC '86*.
2. Forlines, C., Wigdor, D., et al. Direct-Touch vs. Mouse Input for Tabletop Displays. In *Proc. of CHI'07*.
3. Appert, C. and Zhai, S. Using strokes as command shortcuts: cognitive benefits and toolkit support. In *CHI'09*.
4. Kurtenbach, G. *The Design and Evaluation of Marking Menus*. University of Toronto. 1993.
5. Grossman, T., et al. Strategies for accelerating on-line learning of hotkeys. In *Proc. of CHI'07*.
6. Freeman, D., Benko, H., Ringel Morris, M., and Wigdor, D. ShadowGuides: Visualizations for In-Situ Learning of Multi-Touch and Whole-Hand Gestures. In *ITS'09*, 165-172.
7. Bragdon, A., Zeleznik, R., Williamson, B., Miller, T., and LaViola, J. GestureBar: improving the approachability of gesture-based interfaces. In *Proc. of CHI'09*.
8. *MSN Games*. http://zone.msn.com.
9. Kurtenbach, G. and Moran, T. Contextual Animation of Gestural Commands. In *Eurographics CG Forum* (1994).
10. Hinckley, K., et al. InkSeine: In Situ search for active note taking. In *Proc. of CHI'07*, 251-260.
11. Grossman, T., Hinckley, K., Baudisch, P., Agrawala, M., and Balakrishnan, R. Hover widgets: using the tracking state to extend the capabilities of pen-operated devices. In *CHI'06*.
12. Bau, O. and Mackay, W. OctoPocus: A Dynamic Guide for Learning Gesture-Based Command Sets. In *UIST'08*.
13. Baudel, T. and Beaudouin-Lafon, M. Charade: remote control of objects using free-hand gestures. *Comm. of the ACM*, 36, 7 (1993).
14. Vogel, D. and Balakrishnan, R. Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. In *Proc. of UIST'04*, 137-146.
15. Brandl, P., Forlines, C., Wigdor, D., et al. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. In *Proc. of AVI'08*.
16. Domjan, M., ed. *The Principles of Learning and Behavior*. Thomson/Wadsworth, 2003.
17. *Giraffe*. Palm, Inc.
18. Beaudouin-Lafon, M. Novel Interaction Techniques for Overlapping Windows. In *Proceedings of USIT'01*.
19. Denoue, L., et al. A fast, interactive 3D paper-flier metaphor for digital bulletin boards. In *Proc. of UIST'03*.
20. Agarawala, A. and Balakrishnan, R. Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. In *Proc. CHI'06*.
21. Wilson, A., Izadi, S., Hilliges, O., Garcia-Mendoza, A., and Kirk, D. Bringing physics to the surface. In *UIST'08*.
22. David, R. Magic Paper: Sketch-Understanding Research. *Computer*, 40, 9 (2007).
23. Jacob, R., Girouard, A., Hirshfield, L., Horn, M., Shaer, O., Solovey, E., and Zigelbaum, J. Reality-based interaction: a framework for post-WIMP interfaces. In *Proc. of CHI'08*.
24. Elliott, C., et al. Lifelike pedagogical agents and affective computing: an exploratory synthesis. In *Artificial intelligence today: recent trends and developments*. 1999.
25. von Ahn, L. and Dabbish, Laura. Labeling images with a computer game. In *Proceedings of CHI'04*, 319-326.
26. Wobbrock, J., Morris, M.R., and Wilson, A.D. User-defined gestures for surface computing. In *Proc. of CHI'09*.
27. Gibson, J. The Theory of Affordances. In Shaw, R. and Bransford, J., eds., *Perceiving, Acting and Knowing*. 1977.
28. Cao, X., Wilson, A., Balakrishnan, R., Hinckley, K., and Hudson, S. ShapeTouch: Leveraging Contact Shape on Interactive Surfaces. In *Proceedings of Tabletop'08*.
29. *Farseer Physics Engine*. farseerphysics.codeplex.com.
30. LeBlanc, M. *Mechanics, Dynamics, Aesthetics: A Formal Approach to Game Design*. 2004. http://algorithmancy.8kindsoffun.com/MDAnwu.ppt.
31. Saffer, D. *Designing Gestural User Interfaces*. O'Reilly, 2009.
32. Webster, J. Making Computer Tasks at Work More Playful: Implications for Systems Analysts and Designers. In *SIGCPR'88*.