An Interaction Model for Grasp-Aware Tangibles on Interactive Surfaces

Simon Voelker[1] Christian Corsten[1] Nur Al-huda Hamdan[1] Kjell Ivar Øvergård[2] Jan Borchers[1]

[1]RWTH Aachen University; Aachen, Germany; {voelker, corsten, hamdan, borchers}@cs.rwth-aachen.de

[2]Buskerud and Vestfold University College; Tønsberg, Norway; koe@hbv.no

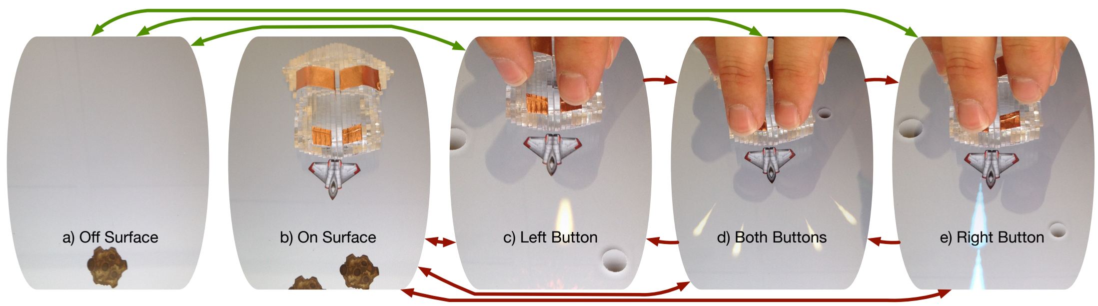

Figure 1. Interaction stages for a grasp-aware tangible space ship for an iPad game: The space ship is off surface (a) and not detected until the user places it on surface (b). When touching the left "button" (c) the ship shoots from the left, whereas adding the right finger fires from both sides (d). When releasing the left finger, only the right "button" is touched, thus shooting from the right. Releasing the finger results in (b) again.

ABSTRACT

Tangibles on interactive surfaces enable users to physically manipulate digital content by placing, manipulating, or removing a tangible object. However, the information *whether* and *how* a user grasps these objects has not been mapped out for tangibles on interactive surfaces so far. Based on Buxton's Three-State Model for graphical input, we present an interaction model that describes input on tangibles that are aware of the user's grasp. We present two examples showing how the user benefits from this extended interaction model. Furthermore, we show how the interaction with other existing tangibles for interactive tabletops can be modeled.

Author Keywords

Tangibles; Interaction Model; Interactive Surfaces

ACM Classification Keywords

H5.2 [Information interfaces and presentation]: User Interfaces. - Input Devices and Strategies.

INTRODUCTION

Tangibles on interactive surfaces [5] bring physical interaction to the user by addressing the haptic sense, exploiting physical affordances, and enabling eyes-free interaction [17]. They can serve as direct physical representations of digital objects [14], or as generic tools to physically manipulate any content [17]. When interacting with a tangible, the user necessarily grasps it—when placing the tangible object on the surface or right before and while manipulating it. Research on tangibles that are *not* used on interactive surfaces shows that the information whether and how a tangible is grasped can be used to enrich the expressiveness of tangible interaction: Touché [13], for example, augments a door knob with a grasp password by analyzing how the user grasps the knob, and HandSense [19] uses capacitive sensors to recognize whether an object is being held in the left or right hand. However, for tangibles on interactive surfaces, the design space for grasp has not been mapped out yet.

Our contribution is an extension of Buxton's interaction model [3] to characterize the state of grasp-aware tangibles on interactive surfaces. This model benefits designers to design, describe, and compare grasp-aware tangibles. We show how our model can be used to capture the interaction with existing tangibles and how grasp information can enrich the expressiveness of tangible interaction, e.g., as shown in Fig. 1.

RELATED WORK

In [3], Buxton presents a Three-State Model for input devices, such as mouse, touch screen, and puck or stylus on a tablet. The model represents all possible interaction states of each device: an input device that has no effect on the system, such as a puck that is not in contact with the tablet surface yet, is considered in State 0 (*Out of Range*). The system starts detecting the puck in State 1 (*Tracking*) when the puck touches

ITS'14, November 16–19, 2014, Dresden, Germany.

http://dx.doi.org/10.1145/2669485.2669494

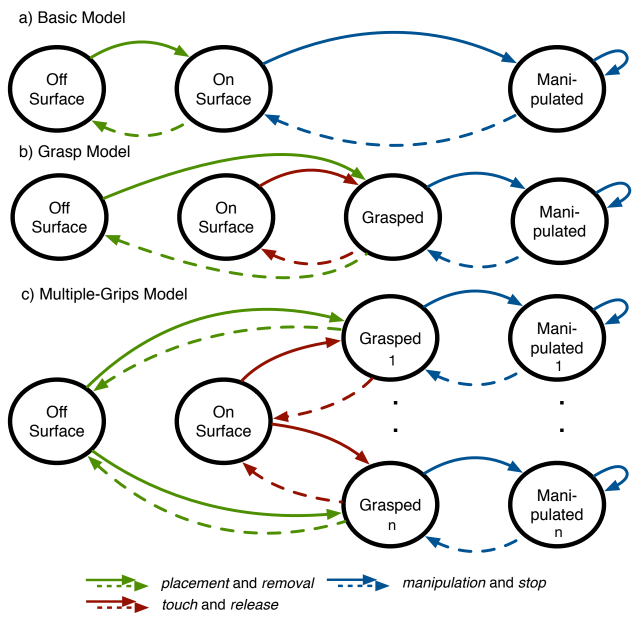

Figure 2. Consecutive tangible interaction models. (a) generalization of Buxton's Three-State Model to tangibles on interactive surfaces, (b) including *whether* and (c) *how* it is grasped and manipulated.

the tablet and is moved across the surface. When physical manipulation is involved, such as pushing a button on the puck, the device enters State 2 (*Dragging*).

This model enables designers to choose a suitable input device for an application or task, and to compare different input states. For example, Patten and Ishii [9] used Buxton's model to compare mouse input in a graphical user interface (GUI) with non-manipulable tangibles in a tangible user interface (TUI). Similarly, Bricks [5] and Urp [14] follow this model.

Buxton's model has been modified and extended to capture new strategies for touch input on interactive surfaces. Benko et al. [1] distinguished State 1 and 2 for finger input on touch screens by taking the size of a touch into consideration. Richter et al. [12] added a fourth state to Buxton's model to capture pressure input on pressure-sensitive touch screens.

INTERACTION MODEL

Fig. 2.a shows our basic model for tangibles on interactive surfaces, an adaptation of Buxton's Three-State Model for a puck on a tablet. When the tangible is *Off Surface*, it has no effect on the system until the user places it *On Surface*. As long as the tangible rests on the surface, the *On Surface* state is retained. When the user physically manipulates the tangible, it reaches the third state, *Manipulated*.

A manipulation can be translating the tangible [14], translating parts of the tangible [17], or transforming the tangible. Each of these manipulations is modeled as an individual *Manipulated* state. For example, the SLAP knob [17] has two *Manipulated* states, one for moving the tangible and one for rotating the knob. When manipulation stops, the tangible returns to the *On Surface* state until the user removes the tangible from the surface or manipulates it again.

However, before the user can manipulate a tangible, she needs to grasp it with her hands. Wimmer [19] has shown that such grasp information can be used as additional input for grasp-aware tangibles. Yet, this information has not been modeled for tangibles on interactive surfaces so far. By adding a separate *Grasped* state to our model, we will explain how grasp information can extend the interaction of tangibles on interactive surfaces. For simplicity and to avoid confusion, in this paper we refer to any direct contact between a user's fingers and a tangible as a grasp.

Grasping the Tangible

Using a basic model (Fig. 2.a) as a starting point for tangible interaction on interactive surfaces, we add a separate *Grasped* state that is reached when the user grasps the tangible—prior to placing, manipulating, or removing it from the surface. By adding this state, we derive two consecutive models which we refer to as *Grasp Model* and *Multi-Grip-Model*.

Grasp Model

The first extension is a single *Grasped* state that models whether the user is grasping a tangible or not (Fig. 2.b). This binary distinction can be used as a quasimode [10] or to trigger a transcendent action that stops as soon as the tangible is not grasped anymore. One example application that adheres to this model is to display additional tooltip information around a tangible while it is being grasped, as we show in our sample applications.

We include this *Grasped* state in the model based on the observation that each action in the basic model requires the user to grasp the tangible. While *placing* the tangible on the surface, the user stays in contact with it. Therefore, *placing* results in the *Grasped* state. When *releasing* the tangible, the state changes to *On Surface* until the user grasps the tangible, which necessarily involves *grasping* it, therefore leading back to the *Grasped* state. The tangible stays in this *Grasped* state until the user *manipulates* it. When manipulation stops, the user is still in contact with the tangible, therefore the *Grasped* state is re-entered. As *removing* the tangible from the surface is inverse to *placing* it, the *Off Surface* state is only reached via the preceding *Grasped* state.

In our model, any grasp interaction on the tangible's surface, such as tapping or sliding with a finger, is considered a sequence of grasps and therefore modeled by one or more *Grasped* states. Any physical manipulation on or with the tangible, such as exerting pressure or moving the object, however, is defined as manipulation, therefore, modeled by a *Manipulated* state(s).

However, some manipulations do not require the user to keep grasping the tangible to manipulate it, e.g., when flicking a tangible. If these manipulations stop without the user touching the tangible, it changes its state directly from *Manipulated* to *On Surface*. Furthermore, actuated tangibles such as Madgets [16] can be manipulated without the user touching them. These tangibles can transition from the *On Surface* state directly to the *Manipulated* state, that describes the actuated manipulation, and back. This can be achieved by simply adding the corresponding transitions.

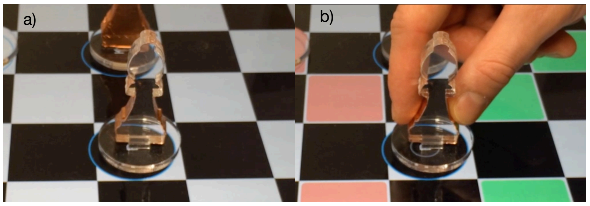

Figure 3. ***Grasp-Model*** **example:(a) A chess piece placed on a multi-touch surface. (b) By grasping a chess piece, the user is shown where he can move the playing piece (green squares) and where the piece can be hit by his opponent (red squares).**

Multi-Grip Model

So far, our model is limited to the binary information of *whether* a tangible is grasped or not. To include information about *how* the user is grasping the tangible, we extend the *Grasp* Model by expanding the *Grasped* state to several distinct states (Fig. 2.c).

Knowing how a tangible is grasped can be used to trigger different actions, such as shown in Fig. 1, where different grips are used to fire different projectiles. Leitner et al. [8] suggested that light pressure (acting as a simple grasp) on pressure sensitive tangible buttons, e.g. Geckos, could be reacted upon by showing a tooltip.

In our *Multi-Grip* Model, each different way of how a user could grasp a tangible is represented by its own $Grasped_i$ state (Fig. 2.c). For example, if the Touché [13] door knob can identify ten different grasps, the model would have ten different $Grasped_i$ states. As in the Grasp model, each $Grasped_i$ state is connected to the *Off Surface*, *On Surface*, and *Manipulated* states by the corresponding actions.

In addition, each $Grasped_i$ state can also directly influence the events that are triggered by the *manipulation* of the tangible. To combine the information of grasp and manipulation, each $Grasped_i$ state needs a *manipulation* transition to its own *Manipulated* state (Fig. 2.c). For example, the RGB color picker knob changes each color channel selectively depending on *where* the user grasps the knob while rotating (Video Figure).

In this model, transitions between the *Grasped* or *Manipulated* states are possible if the user is able to change *how* she grasps the tangible without releasing it. However, for visual simplicity, we omitted these transitions in Fig. 2.c.

APPLYING THE MODEL

In this section we explore the applicability of our model by providing three interaction design examples and an overview how other existing tangibles can be modeled. In both *Grasp Model* and (*Multi-Grip Model*) examples we used a combination of the PUCs marker setup by Voelker et al. [15] and a technique introduced by Rekimoto [11] to detect *whether* and *how* a tangible is touched on a capacitive touch display.

Grasp Model: Tutorial on Demand

In GUIs, tooltips provide help on demand without cluttering the screen. Using the new *Grasped* state, we can bring this principle to TUIs.

Tangible	Off-Surface	On-Surface	Grasped	Manipulated
SLAP Knob	1	1	–	2
Madgets Knob	1	1	–	4
CapWidgets	–	–	1	1
TUIC	–	–	1	1
Geckos	1	1	many	1
CapStones	–	–	1	1
Papillion	1	1	many	1

Table 1. Number of states of existing tangibles

Novice chess players are often unsure what moves a particular playing piece allows. To support the player, a tangible chess game on an interactive surface could offer a tutorial mode in which, whenever the player grasps a piece, all possible moves are highlighted on the board. This on-demand display reduces clutter and occlusion problems (Fig. 3.b).

In general, having the tangible react to being grasped can be exploited for quasimodes [10] that are activated prior to when the user manipulates a tangible. The user can safely explore the TUI before actually manipulating objects.

Multi-Grip model: Coarse-to-Fine Video Navigation

For demonstration purposes, we built a multi-granularity video navigation knob whose grasp zones semantically match different user grips (video Fig.). Turning the knob from the top with thumb and index finger (precision grip) results in fine motor adjustments and therefore could map to fine granularity, whereas grasping the tangible with the entire palm would map to coarser control.

This design does not require adding hinged mechanical parts to the tangible, and instead of using one tangible per granularity, it allows to select different granularities using just one physical controller. This simplifies tangible construction, saves space on the interactive surface, and allows the user to switch granularity levels without looking at the knob, e.g., by sliding the hand up and down along the grasp rings (Video Fig.).

Modeling Existing Tangibles

In this section we demonstrate the descriptive and comparative power of our interaction model (Table 1). For example, CapWidgets [7], TUIC [20], Geckos [8], CapStones [4], and Papillion [2] can detect the *Grasped* state, while SLAP [17] tangibles can only detect tangible manipulation. CapWidgets and CapStones cannot detect when a tangible is *On Surface* or *Off Surface* without user grasp and manipulation.

Hennecke et al. [6] designed pressure-sensitive tangibles (NoCs) with optical markers that behave differently when pressure, e.g., by a touch, is exerted or released. Using our model, each NoC marker can be modeled with a single *Grasped* state. When an NoC_i marker is grasped, the tangible enters a $Grasped_i$ state; when the grasp is released, the tangible goes back to the *On Surface* state. Multiple NoC markers on a single tangible can be modeled with an equivalent number of $Grasped_i$ states.

DISCUSSION AND FUTURE WORK

For tangibles whose surface can continuously detect *any* grasp, such as Geckos [8], our model would result in a large set of $Grasped_i$ states. However, many of those graps would be meaningless or are very similar to each other. For example, a knob that is rotated from the top with three fingers, the different angles between the fingers does not matter, all would be the same kind of rotating grasp. Hence, such grasps can be clustered into a single $Grasped_i$ state, therefore reducing the number of states for the corresponding model. This decision has to be made by the designer of the tangible.

As a next step, we could go beyond mere grasp-aware tangibles and expand our model to also include hover-aware tangibles, such as FlyEye [18], that exploit the proximity of a user's finger to the tangible as additional input. A chess piece, e.g., could already inform the user that no movement is possible before the user actually has to grasp it. This, by the way, would help the player follow the World Chess Federation rules, which require the player to move a chess piece once it is grasped.

CONCLUSION

Detecting whether tangibles on interactive surfaces are being grasped or not, and how they are being grasped, expands the design space of possible interactions with this class of user interfaces. We adapted and extended Buxton's well-established Three-State Model to represent these additional states, leading to our *Grasp* and *Multi-Grip* Models. Besides applying our model to existing tangibles for interactive tabletops, we implemented two sample applications to demonstrate the increased richness of interaction when applying our different models. We hope that our model facilitate future research and design in the new space of grasp-aware tangibles.

ACKNOWLEDGMENTS

This work was funded by the German B-IT Foundation, the Norwegian Research Council (project number 217788), and the German Research Foundation (DFG).

REFERENCES

1. Benko, H., Wilson, A. D., and Baudisch, P. Precise Selection Techniques for Multi-touch Screens. In *Proc. CHI '06*, 1263–1272.
2. Brockmeyer, E., Poupyrev, I., and Hudson, S. Papillon: designing curved display surfaces with printed optics. In *Proc. UIST '13*, 457–462.
3. Buxton, W. A Three-State Model of Graphical Input. In *Proc. INTERACT '90*, 449–456.
4. Chan, L., Müller, S., Roudaut, A., and Baudisch, P. CapStones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens. In *Proc. CHI '12*, 2189–2192.
5. Fitzmaurice, G. W., Ishii, H., and Buxton, W. A. S. Bricks: Laying the Foundations for Graspable User Interfaces. In *Proc. CHI '95*, 72–79.
6. Hennecke, F., Berwein, F., and Butz, A. Optical pressure sensing for tangible user interfaces. In *Proc. TEI '11*, 45–48.
7. Kratz, S., Westermann, T., Rohs, M., and Essl, G. CapWidgets: Tangible Widgets versus Multi-touch Controls on Mobile Devices. In *Proc. CHI EA '11*, 1351–1356.
8. Leitner, J., and Haller, M. Geckos: Combining Magnets and Pressure Images to Enable New Tangible-Object Design and Interaction. In *Proc. CHI '11*, 2985–2994.
9. Patten, J., and Ishii, H. A Comparison of Spatial Organization Strategies in Graphical and Tangible User Interfaces. In *Proc. DARE '00*, 41–50.
10. Raskin, J. *The Humane Interface: New Directions for Designing Interactive Systems*. Addison-Wesley Professional, 2000.
11. Rekimoto, J. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. In *Proc. CHI '02*, 113–120.
12. Richter, H., Ecker, R., Deisler, C., and Butz, A. HapTouch and the 2+1 State Model: Potentials of Haptic Feedback on Touch Based In-vehicle Information Systems. In *Proc. AutomotiveUI '10*, 72–79.
13. Sato, M., Poupyrev, I., and Harrison, C. Touché: Enhancing Touch Interaction on Humans, Screens, Liquids, and Everyday Objects. In *Proc. CHI '12*, 483–492.
14. Underkoffler, J., and Ishii, H. Urp: a Luminous-Tangible Workbench for Urban Planning and Design. In *Proc. CHI '99*, 386–393.
15. Voelker, S., Nakajima, K., Thoresen, C., Itoh, Y., Øvergård, K. I., and Borchers, J. PUCs: Detecting Transparent, Passive Untouched Capacitive Widgets on Unmodified Multi-touch Displays. In *Proc. ITS '13*, 101–104.
16. Weiss, M., Schwarz, F., Jakubowski, S., and Borchers, J. Madgets: Actuating Widgets on Interactive Tabletops. In *Proc. UIST '10*, 293–302.
17. Weiss, M., Wagner, J., Jansen, Y., Jennings, R., Khoshabeh, R., Hollan, J. D., and Borchers, J. SLAP Widgets: Bridging the Gap Between Virtual and Physical Controls on Tabletops. In *Proc. CHI '09*, 481–490.
18. Wimmer, R. FlyEye: Grasp-Sensitive Surfaces using Optical Fiber. In *Proc. TEI '10*, 245–248.
19. Wimmer, R., and Boring, S. HandSense: Discriminating Different Ways of Grasping and Holding a Tangible User Interface. In *Proc. TEI '09*, 359–362.
20. Yu, N.-H., Chan, L.-W., Lau, S. Y., Tsai, S.-S., Hsiao, I.-C., Tsai, D.-J., Hsiao, F.-I., Cheng, L.-P., Chen, M., Huang, P., and Hung, Y.-P. TUIC: Enabling Tangible Interaction on Capacitive Multi-touch Display. In *Proc. CHI '11*, 2995–3004.