My Tablet Is Moving Around, Can I Touch It?

Alejandro Catala
Human Media Interaction
University of Twente, The Netherlands
a.catala@utwente.nl

Mariët Theune
Human Media Interaction
University of Twente, The Netherlands
m.theune@utwente.nl

Dirk Heylen
Human Media Interaction
University of Twente, The Netherlands
d.k.j.heylen@utwente.nl

TEI '17, March 20-23, 2017, Yokohama, Japan
ACM 978-1-4503-4676-4/17/03.
http://dx.doi.org/10.1145/3024969.3025082

Abstract
Touch displays moving autonomously, i.e., being *self-actuated*, are starting to be part of system prototypes reported in the literature. In the future, user interactions with such prototypes may take place while the display is moving. However, since current prototypes do not yet require this type of interaction, there is a lack of empirical evidence reporting issues related to touch input in such conditions. This leads us to propose two basic questions: Can we request users to deliver touch gestures during the actuation? Which aspects should we take into account when having touch input in these moving devices? In order to start to answer these questions, we report in this paper a first study to get an insight into how people perform and feel it when they have to carry out touch input on a self-actuated tablet. The preliminary results show that the self-actuated tablet does not necessarily need to be still during touch interaction, and that single-touch gestures based on drag or tap are preferable over others. Furthermore, the results also revealed an issue with tap gestures because of the movement of the tablet.

Author Keywords
Self-actuated tablet; Touch input; Gesture; Movement.

ACM Classification Keywords
H.5.2 User Interfaces: Input devices and strategies, Interaction styles, Evaluation/methodology; H.1.2 User/Machine Systems: Human factors.

Introduction
Some emergent research is starting to incorporate touch-enabled displays which can move autonomously. For example, there are tangible devices to enable layered visual interaction by varying the elevation of the tablet display [12], a tangible game with a tablet that moves between landmarks on a regular tabletop [3], self-actuated vertical displays to draw formulas on a whiteboard [1], or even quadcopters with flexible displays to bring videoconferencing functions to the space surrounding the users [4]. In all these samples of technological contributions, users may need to deliver touch commands or select targets on the screen while the display could be moving. However, although the authors do show possible uses of self-actuated displays, they do not consider touch interaction to be carried out while the display is moving. Consequently, they do not discuss the touch input issues that can arise by having such actuation.

In our own ongoing research we are developing artificial agents embodied in self-actuated tablets, to be used in tabletop games, which can require input while the agents are moving around to deliver more engaging and appealing interactive experiences to the users. To support this type of interaction, we need to find out more about the limitations of touch input in self-actuated tablets. Beyond the well-known issues in touch input, such as the fat finger problem [7], exit errors [13], occlusion or the positioning of the interactive widget and device [2], or biomechanical constraints [5][1615], there is a lack of evidence on the difficulties related to touch input specifically affected by the movement of the displays. Hence, we present an exploratory study that is the first of this kind to provide empirical results on which interaction issues users have to face if requested to deliver touch input in self-actuated tablets. It will help to characterize this sort of interactions, and will therefore support the future design of interactive systems involving such devices by allowing us to take more informed design decisions. With this paper, we want to provoke discussion on to which extent touch input can be considered during actuation and, hopefully, our observations will open new opportunities by inspiring the research community to bring new scenarios and use case prototypes that include touch input in moving tablets.

Related Work
The concept of actuation on tabletops is broad and diverse in purpose (e.g. [11], [6], [15]). When considering related work on user interaction with self-actuated tangible objects, we must distinguish self-actuation of tangibles that do not have touch displays from those that do. For example, Vonach et al. implemented modular actuated tangible objects [14], whose position and rotation can be controlled but without any capability to receive touch input yet. Pedersen and Hornbæk presented a set of interactions using their *tangible bots*, motorized tangibles capable of moving on an interactive tabletop [9]. Visualization and touch input happen around the device, on the tabletop interface. The previous examples cannot render advanced visual content in the device itself, or receive more integrated and direct touch input, which would expand the possibilities of tangible interfaces. An

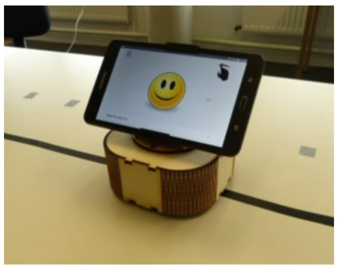

Figure 1. Self-actuated tablet moving sideways

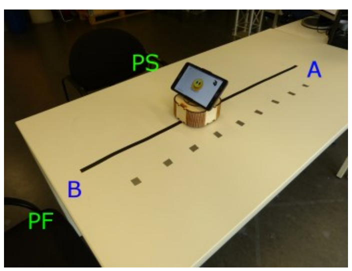

Figure 2. Overview of the experimental setting

example of a more advanced graphics integration is the case of the Sifteo Cubes [8][10]. Its creators included a small graphical interface as part of the tangible, although without any kind of actuation.

Some works consider self-actuated displays to provide advanced visual feedback and touch input capabilities. The work by Sim et al. on G-raffe presents a tangible block supporting the elevation of a display to enable 2.5d visual interaction [12], so that the tangible can show different information depending on both the location and the height of the display. Bader et al. [1] discuss some use cases for self-actuated displays, such as guiding a person in using a coffee machine, or guiding a user through an exhibition. It also reports a system for a self-actuated vertical surface capable of drawing formulas on the whiteboard. The work presented in [3] reports the design of a tangible game that uses an actuated tablet to augment digitally a regular tabletop. Gomes et al. [4] presents a toolbox for exploring interaction with tangible and displays in the mid-air, based on quadcopters in the space surrounding the user. Among the interactions and potential scenarios, it presents DisplayDrones, in the form of quadcopter with a touchscreen display to bring picture and video capabilities.

Despite the effort to develop new technology and explore use case scenarios with self-actuated displays, as shown in related work, touch input is being relegated to interactions in which the device remains still. We consider that touch input in moving displays may open new possibilities for interaction and new scenarios. As a first step forward to better understand the issues of touch input when the tablet is moving around, we carried out the following exploratory study.

Study Design

In this exploratory study we focused on some of the typical touch gestures (tap, drag, rotate, scale) that can be carried out in a regular tablet, and looked for issues that may arise if people are requested to interact while the surface is moving.

Apparatus

We prepared a tabletop (150x75cm) setting with a self-actuated tablet: a Samsung Tab A 7” (model T280) with a screen size of 150x94mm. It was mounted with a 45° angle on top of a plywood case of 13.6x13.6x5.7cm, containing a Zumo Robot by Pololu[1] (see Figure 1 and Figure 2). We chose this configuration because we are interested in applications in which the user is seated and the tablet is partially angled to facilitate the visualization. To facilitate the implementation and reproducibility of interactions between trials, we restricted the movement of the robot to follow a 1-meter-long line as shown in Figure 2 by implementing a line-follower program.

Gestures

Touch gestures had to be performed on a smiley representing a virtual character on screen (see Figure 1). The smiley was 4.3 cm in diameter and located in the center of the display. It was able to recognize all gestures (tap, drag, rotate, zoom) at all times, giving immediate visual feedback by applying the corresponding translation, rotation and scale transforms and showing a thumb up icon when the expected gesture was completed. Any touch event outside of the smiley was ignored. For a trial to be

[1] Zumo Robot: https://www.pololu.com/product/2510

successful, the user had to carry out the intended gesture and achieve the required completion conditions. In particular, **Tap** was implemented with the facility class G*estureDetector. SimpleOnGestureListener* provided by the Android SDK, and therefore the tap gesture succeeded as soon as it was notified by the corresponding listener. For the **Drag**, the user had to drag the smiley at least 6 cm. The **Rotate** gesture required users to rotate the smiley at least 90 degrees in any direction. The **Scale** gesture required users to pinch until the smiley was half size, or zoom until it was at least double size. With this range of gestures, we covered instant stationary gestures as well as different dynamic touch input with one or two fingers.

Method and Procedure

Sixteen healthy adults ranging from 23 to 41 years old (m=29.8, sd=4.5) participated in the study (4 women). They were all users of smartphones with multi-touch capacitive screen. Before starting the test, each participant performed the gestures without actuation to get acquainted with the task and to ensure he/she was able to carry out the gestures. Then, each participant proceeded with 24 trials, which corresponded to a single combination of trajectory, speed and gesture (as explained below). Before each trial, the graphical interface displayed an icon representing the gesture to be carried out. The user had to respond by saying aloud the expected gesture, to confirm he/she was ready. The experimenter could then activate the trial remotely to allow the robot to go from point A to B (see Figure 2). The user had to complete the requested gesture before the robot reached the end of the line. If the trial was completed successfully, the robot made a beep, and stopped, and the thumb up icon was displayed. The trials were grouped in blocks of the four gestures, being administered randomly. The blocks were controlled by Trajectory first, and then by Speed, which were also administered randomly. Repetitions were not considered because gestures were basic well-known touch gestures and users were adults, and, therefore, we wanted to observe the interaction issues rather than learnability of gestures. After each single block, participants were requested to answer the question: *To carry out the touch gestures, the robot speed was: __ (0=Too slow…10=Too fast)*. After testing the three different speeds for each trajectory, the participants were asked to express their preferences regarding the speeds and gestures.

In this study, we considered two different relative movements of the self-actuated tablet. The Trajectory could be either *Forward*, with the user seated at place PF, or *Sideways*, when seated at place PS (see Figure 2). We decided to limit the trajectory to these two relative movements in order to keep the number of trials manageable, and because they account for two main situations of the robot crossing an interactive space defined by the tabletop: when the tablet is approaching the user and when the tablet is crossing the interactive space in front of the user. Other trajectories such as moving away or any complex combination of trajectories were not explored in this study.

The speed at which the tablet moved varied at three different levels of *Speed.* The related work involving touch displays does not report speeds. Hence, we considered speeds similar to the ones reported in related works involving actuated tangibles. In particular, we established *Speed 3* (S3), the highest

Table 1. Error counts

	Forward			Sideways			
	S1	**S2**	**S3**	**S1**	**S2**	**S3**	All
TAP	0	2	3	0	0	5	10
DRAG	0	0	1	0	0	0	1
ROTATE	1	2	4	0	3	5	15
SCALE	0	2	3	0	0	1	6
All	1	6	11	0	3	11	32

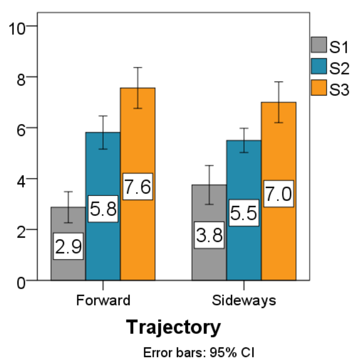

Figure 3. Perceived speed appropriateness

speed, to be 36 cm/s, which is just a few centimeters per second below the speed reported in [9], and *Speed 2* (S2) to be 26 cm/s, which is close to the one reported in [14]. With these two speeds, we can get a deeper insight on what would happen if we embedded a touch display on a self-actuated tangible moving at similar speeds to those in two different tangible settings. Finally, the slower speed tested, *Speed 1* (S1), was set at 13 cm/s. In this way, with a range of speeds, we can observe to which extent delivering touch input is affected by the motion.

Results and Discussion

Table 1 shows the error counts. Overall, 91.6% of the trials were successful (=(384-32)/384, where 384 = 16 users x 24 trials). The results suggest that it would be feasible to consider users delivering simple touch gestures on a tablet while being actuated. The trajectory of the tablet with respect to the user does not seem to be an important factor, of course, provided that the user is facing the tablet in a similar way to the configurations tested. However, the speed matters when touch input is involved. The results reveal some important aspects that must be taken into account if interactions really need to include touch input during actuation. The main observations and remarks are summarized as follows:

Touch input requires slower speed than with tangibles
We have tested a range of speeds, where S2 and S3 were selected to be similar to the speeds reported related work. As suggested by the counts in Table 1, in order to avoid interaction errors, slower speeds are recommended. S1 is more suitable for touch input. Figure 3 shows the perceived appropriateness of the speeds tested. Thus, we should avoid disproportionate speed when touch input is involved. Some users expressed in their comments that Speed 1 "is fine and convenient even for more complex gestures as *Rotate"*, and that Speed 2 would still be "a bit too fast as to carry out the interactions with comfort".

Multi-point or stationary gestures can be problematic
Rotate and Scale are typical gestures requiring, in the implementation used in the study, two fingers. They involved participants doing turning or pinching/zooming trajectories with their fingers, and therefore when the tablet is moving, it is expected that interaction issues are more frequent than in simpler gestures. Indeed, most of the failures were concentrated on those multi-touch gestures, in particular the Rotate gesture at speed S3. Sometimes, *Tap* also resulted in a problematic gesture at speeds S2 and S3. The point is that above a certain speed, the users experienced issues to perform a single tap because tablet motion caused the finger to drag slightly, preventing the tap from being correctly detected. Moreover, in 13 tap trials, the users needed to tap several times until the gesture was recognized. This issue would require to implement a suitable tap detector for self-actuated tablets, different from the one provided by the Android SDK. For instance, by considering single Down or Up events when the target does not really expect dragging, or by increasing the thresholds to filter accidental or unintended Move events that are triggering the drag instead of a tap gesture.

Single-point gestures are preferred and better performed
Figure 4 and Figure 5 depict the user preferences for speeds and gestures respectively. Participants showed their preference for S1 and S2 in similar proportion.

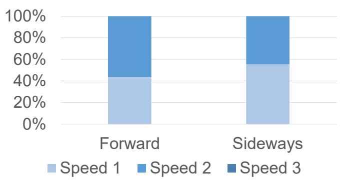

Figure 4. Speed preference

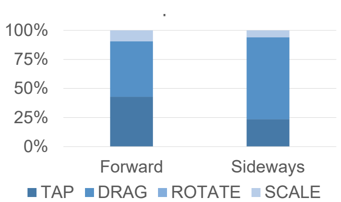

Figure 5. Gesture preference

However, in their comments they pointed out that a suitable speed would be between S1 and S2, which would explain the mixed preference. Single-touch gestures like *Tap* and *Drag* are the most preferred ones. The top gesture is Drag, in terms of both success and preference. Despite having to drag a long distance to traverse (at least 6 cm), it seems to be more convenient than *Tap* as it does not have the detection issue. Although all gestures can be performed at low speeds, issuing taps or drags would be recommended.

Limitations and Future Work

Among all the different issues that could be interesting to explore when interacting with self-actuated tablets, we started investigating only a subset of operational characteristics and touch gestures to better understand the most basic and likely issues present. Regarding gestures, we chose a diverse subset of typical gestures so that single-touch, multi-touch, dynamic and stationary were represented. Even in this case, the results and conclusions on trajectory and speed must be interpreted in terms of the four gestures considered and the related experimental conditions.

Taking into account the lack of previous reports about touch input issues on self-actuated tablets, we focused the study on basic interactions to better understand and determine potential issues. We chose quite demanding completion conditions for the gestures (i.e. rotation angles, dragging distances and scale factors) so that users were actually challenged. However, future work should focus on studying the target size and/or the precision to carry out gestures, as well as exploring more interactions involving movement in games and specific real applications.

In some applications, touch input may be less intensive, or other input modalities can be more relevant in a given context. We could need to combine gesture input, for example, with grasping and pushing, or fiduciary cards in the mid-air, etc. This means that the higher speeds tested could be re-enabled when combining several modalities. Exploring other more complex interactions, not only touch, would also be interesting. Nevertheless, we believe that we must first contribute with empirical evidence on which issues touch input introduces on self-actuated tablets. After that, we can explore the combination of different input modalities as part of our future work, with the aim of providing a comprehensive framework to support interactions on self-actuated tablets.

We aim to develop playful artificial agents embodied in self-actuated tablets, or surface-bots, with similar hardware to the one implemented for this study. For this kind of application, we need to allow users to provide commands and select targets on the screen while agents are moving. Our future work will focus on the design of menus and other widgets with appropriate target selection techniques for these devices. In this context, we will explore alternative implementations to tap detectors to overcome the identified issues and we will study drag based gestures in depth to be both effective and accurate. In this way, we will complete the study of issues with self-actuated tablets by testing possible widget solutions.

Acknowledgements

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 701991.

References

1. Patrick Bader, Valentin Schwind, Norman Pohl, Niels Henze, Katrin Wolf, Stefan Schneegass, Albrecht Schmidt. 2015. Self-Actuated Displays for Vertical Surfaces. Human-Computer Interaction – INTERACT 2015 Volume 9299 of the series Lecture Notes in Computer Science pp 282-299.
2. Peter Brandl, Jakob Leitner, Thomas Seifried, Michael Haller, Bernard Doray, and Paul To. 2009. Occlusion-aware menu design for digital tabletops. In CHI '09 Extended Abstracts on Human Factors in Computing Systems (CHI EA '09). ACM, New York, NY, USA, 3223-3228.
3. Fernando Garcia-Sanjuan, Javier Jaen, Alejandro Catala. 2015. Augmented Tangible Surfaces to Support Cognitive Games for Ageing People. In Ambient Intelligence - Software and Applications, Volume 376 of the series Advances in Intelligent Systems and Computing pp 263-271.
4. Antonio Gomes, Calvin Rubens, Sean Braley, and Roel Vertegaal. 2016. BitDrones: Towards Using 3D Nanocopter Displays as Interactive Self-Levitating Programmable Matter. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 770-780.
5. Eve Hoggan, John Williamson, Antti Oulasvirta, Miguel Nacenta, Per Ola Kristensson, and Anu Lehtiö. 2013. Multi-touch rotation gestures: performance and ergonomics. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 3047-3050.
6. Masahiko Inami, Maki Sugimoto, Bruce H. Thomas, and Jan Richter. 2010. Active Tangible Interactions. Tabletops-Horizontal Interactive Displays, 171-187
7. Aurélien Lucchi, Patrick Jermann, Guillaume Zufferey, and Pierre Dillenbourg. 2010. An empirical evaluation of touch and tangible interfaces for tabletop displays. In Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction (TEI '10). ACM, New York, NY, USA, 177-184.
8. David Merrill, Emily Sun, and Jeevan Kalanithi. 2012. Sifteo cubes. In CHI '12 Extended Abstracts on Human Factors in Computing Systems (CHI EA '12). ACM, New York, NY, USA, 1015-1018.
9. Esben W. Pedersen and Kasper Hornbæk. 2011. Tangible bots: interaction with active tangibles in tabletop interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 2975-2984.
10. Clément Pillias, Raphaël Robert-Bouchard, and Guillaume Levieux. 2014. Designing tangible video games: lessons learned from the sifteo cubes. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 3163-3166.
11. Eckard Riedenklau, Thomas Hermann, and Helge Ritter. 2012. An integrated multi-modal actuated tangible user interface for distributed collaborative planning. In Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12), Stephen N. Spencer (Ed.). ACM, New York, NY, USA, 169-174
12. Jungu Sim, Chang-Min Kim, Seung-Woo Nam, and Tek-Jin Nam. 2014. G-raffe: an elevating tangible block supporting 2.5D interaction in a tabletop computing environment. In Proceedings of the adjunct publication of the 27th annual ACM symposium on User interface software and technology (UIST'14 Adjunct). ACM, New York, NY, USA, 87-88.
13. Philip Tuddenham, David Kirk, and Shahram Izadi. 2010. Graspables revisited: multi-touch vs. tangible input for tabletop displays in acquisition and manipulation tasks. In Proceedings of the

SIGCHI Conference on Human Factors in Computing Systems (CHI '10). ACM, New York, NY, USA, 2223-2232.

14. Emanuel Vonach, Georg Gerstweiler, and Hannes Kaufmann. 2014. ACTO: A Modular Actuated Tangible User Interface Object. In Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14). ACM, New York, NY, USA, 259-268.
15. Malte Weiss, Florian Schwarz, Simon Jakubowski, and Jan Borchers. 2010. Madgets: actuating widgets on interactive tabletops. In Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 293-302.
16. Katrin Wolf, Christian Müller-Tomfelde, Kelvin Cheng, and Ina Wechsung. 2012. PinchPad: performance of touch-based gestures while grasping devices. In Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12), Stephen N. Spencer (Ed.). ACM, New York, NY, USA, 103-110.