

Cell Phone Puppets: Turning Mobile Phones into Performing Objects

Michael Nitsche and Sanjeev Nayak

Georgia Institute of Technology/ Digital Media
85 Fifth Str. NW TSRB
Atlanta, GA 30332-0165, USA
{michael.nitsche, snayak8}@gatech.edu

Abstract. We introduce *Puppettime*, a digital puppetry project that uses mobile phones as interfaces to control virtual puppetry via motion gestures. The goal of the project is to explore cell phones as performative objects in novel interaction designs. Combining the evolution of mobile devices into tangible interfaces with traditional puppetry, the paper discusses the connections between these domains. It describes the design rationale behind the *Puppettime* project as well as its implementation and first feedback, focusing on the core thesis that puppetry provides a valuable and underused metaphor for interface design that supports digital entertainment between co-present players.

Keywords: Puppet, tangible user interface, virtual character.

1 Introduction

The potential of gestures as intuitive input feature was recognized early on [1] but only the recent inclusion of advanced sensors in smart phones has made motion gesture recognition available to the larger audience of phone users. The inclusion of numerous sensors to smart phones allowed them to evolve into tangible controllers (as defined by [2]). But even though today's devices are stacked with accelerometers, compass, gyro, touch screens, and camera(s), all of them valuable features to support tangible interaction, the dominant interaction design for cell phones remains focused on the use of screens, buttons, microphones, and speakers.

The potential of mobile devices clearly encourages increased motion-based interaction but it remains woefully underused. The problem is not one of technology but one of design metaphors. We argue that in order to embrace the new opportunities a fundamental change of our understanding of mobile phones as media devices is needed. Through motion-based design, cell phones turn from audio-visual media devices into performing objects. They evolve from input/ output devices to objects, whose physical condition becomes prevalent not only to the player but every bystander. Designing for this form of mobile play cannot exclusively rely on audio-visual representation rendered by the phone, but it has to include the object itself in the design. This mirrors main arguments of earlier research on embodiment and

situatedness of ubiquitous interaction [3]. Within the larger field, the goal of this paper is to address particularly the recognition of the device itself as performing object and explore the opening design space through an example project.

Parallel to our past work with digital puppetry [4] we have developed *Puppettime*, a prototype for a cell phone based digital puppetry system. It addresses a number of prevalent questions in the development of mobile interaction design for co-located digital entertainment, including design for multiple devices and for multi-user conditions, as well as testing of the underlying technology for fine grain control mechanisms. In this paper we will present the background, design, implementation, and some initial responses to the system.

2 Background

2.1 Mobile Phones and Tangible Interaction Design

A number of innovative projects have started to use phones as tangible objects in themselves. Phones have been used as controllers for Wii-like interaction with large screens [5], as well as controllers for 3D mobile games [6], co-located social games based on motion gestures [7], and multi-modal approaches that combine touch and motion [8]. First frameworks for the design of tangible interaction with phones are emerging [9] but the question remains how the interaction design and user experience will change as interactions with phones evolve further into a shared, movement-based, and co-located user experience. Does the design follow generic tasks, such as text input supported by accelerometers [10] or develop particularly phone-related gestures, like the DoubleFlip [11]?

This mirrors the challenge seen earlier in the transition of other interfaces to increasingly motion-based interaction design, particularly the focus on motion gestures in game consoles (Microsoft's Kinect, Sony's Move, Nintendo's Wii). Video games have discovered the social space in front of the console and designed particular interaction conditions for this space. But a key difference between gaming and general mobile phone usage is the different social context for gesture based interaction [12].

Motion gestures are performative. Unlike a screen-based gesture system, a motion input is clearly visible to others and constitutes an expressive action in itself. The Kinect transforms the living room into a performance stage for gaming and Sall/Grinter have found that "physical gaming often brought an awareness of other householders" [13]. Likewise, motion gestures for mobile devices add a performative element to the interaction. Instead of a hidden input, as seen with keyboards or touch screens, bystanders are able to read and interpret the gestures. But because the surroundings and bystanders are often less familiar in the public use of smart phones, the resulting awareness of others is not always wanted. Technically, we might be able to use our smart phones like Wii controllers, but conceptually we have to revisit the motion gestures when we apply them to public use. How, then, can we develop performative gestures for innovative cell phone interaction design?

2.2 Value of Puppetry

To address this question we turned to puppetry. Puppetry presents a highly developed form of expressive interaction and over the course of its 4000-year old history it has developed rich and culturally complex performative practices [14]. It depends on motion gestures – what we would call “input” – to control a wide range of expressions. Thanks to its long tradition of outstanding artistry, the granularity and efficiency of this “input” is remarkably higher in traditional puppetry than it is in most current digital technology. The most delicate movement can have the power to evoke highly effective expressions in traditional puppetry formats. Thus, it provides a valid testing goal post for new interface technology and builds on a critical overlap between entertainment technology as a form of creative expression.

Puppetry has been identified as a metaphor for interaction with digital worlds before. For example, it has been claimed that puppetry supports storytelling, improvisation, and public engagement in digital media [15]. The relationship between puppeteer and puppet is used to describe the relationship between user and avatar, starting in early HCI concepts [16], to more current educational projects [17], and tangible interfaces [18]. At the same time, traditional puppetry has started to explore and theorize its relation to the digital, gradually building frameworks to include it better [19].

In commercial entertainment, digital puppetry is at work in a range of projects such as Disney’s *Turtle Talk with Crush* exhibit at their Epcot theme park or the Henson studio’s puppeteering set up for the TV show *Sid, the Science Kid* [20]. The problem is that these installations and other related work, like *ShadowStory* [21] use custom-built hardware controllers or expensive high-end commercial controllers [22]. This limits their accessibility to consumers and instead focuses on expert users. These high end installations do not allow visiting players to easily turn into puppeteers or set up their own puppet shows but instead keep most visitors in the role of traditional audiences that can only contribute at a certain location and with the help of certain technology. *Puppettime* aims to explore and open up the performative range of digital puppetry to these large groups, test how well modern phones can deliver on the fine granularity of puppet controls, and provide the basis for a discussion of new design concepts for phones as performing objects in mobile entertainment. To do so, it set out to provide a simple, and highly accessible mobile-phone-based puppet interaction system.

3 Puppettime

3.1 Designing a Puppet Interface for Mobile Phones

Puppettime is a digital puppetry project that uses multiple mobile phones as objects to control puppet performances on a projected 3D virtual stage. It is multi-player, allowing different players to join at any time and participate locally in a shared performance. Once the players have logged on to the central virtual stage, each one can customize their puppets on the mobile devices before they launch them into the

performance. During such a performance, each player can either control their puppet with a single device – much like a simple rod puppet – or with two devices that allow for more detailed animation control of body and a puppet limbs.



Fig. 1. Stage application at work; lower menu presents stage director options; both puppets are controlled by players

The core principle is to use mobile devices as performative objects much like puppeteers use rods or strings. Puppets are controlled by spatial movement of the devices and players can directly engage with the virtual performance on stage “through” the device and not “on” the device. The goal was to direct attention away from the phone as a traditional focal point of the interaction and toward the performance of the puppets and their puppeteers themselves. Because the interaction is spatial and shared, players realize that they are collaborating with other players during the puppet play. Their collaboration happens not only on the virtual stage but also in the shared physical location. Other players are not hidden behind some interface but their physical behavior – how they move their hands to control the puppet – is obvious and the co-located collaboration spreads between virtual stage and physical control space.

3.2 Implementation

Puppettime functions across two platforms. Android smart phones are utilized as tangible motion controllers, and a laptop running *Unity3d* serves as a display device that receives the phone input and renders the virtual puppet show. The Android phones communicate with the Unity application over a local Wi-Fi network.

The Unity application serves as a digital counterpart to the physical stage. A 3D stage environment is rendered in real-time, giving a backdrop for the virtual puppet performance. A set stage, complete with props, lighting, and curtains gives the digital puppets a place to act out their stories. Unity’s built in physics engine adds a sense of realism to the environment as puppets collide with stage geometry, as they would in the real world. Using a cloth simulator, the curtain geometry was made to flow

realistically and could be manipulated by the puppet bodies as they moved through the cloth. The puppets themselves were built as rag dolls with their bodies anchored to an invisible rod. The puppets' arms, legs, and heads, were free to move, bounce, and flail along with the simulated gravity depending on how the puppeteer directed the puppet's body. The basic physics simulation helped to help make a tangible connection between the real world mobile device and the digital puppet.

The Android application (Android 2.1 and higher) allows players to log on to the Unity application via the IP address of the host computer, enter a user name to identify their phone, and select between two control schemes: single and dual control (fig. 2 left). Once logged on, players can customize their virtual puppets from a given selection (fig. 2 right). We provided puppets for the classic fairytale setting of Little Red Riding Hood: a wolf, a hunter, a grandmother, and a puppet for Little Red Riding Hood. To support the aforementioned playful engagement and customization, *Puppettime* allows participants to create their own puppet versions from these basic figures. Players can select a certain body part, and then swipe over the screen to flip between different elements (head, chest, arms, legs) from varying puppets to form their own character. This interaction is not unlike using a children's playbook with split pages that can be turned independently to create new combinations of shapes.

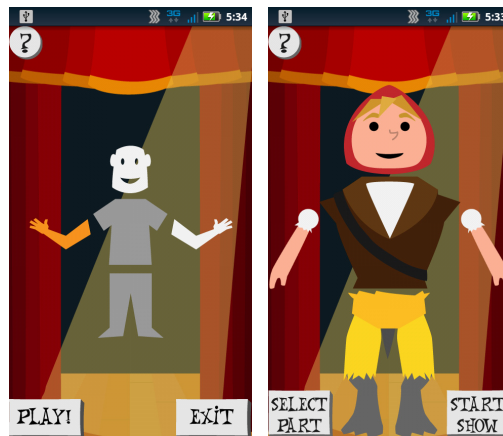


Fig. 2. Android app: selecting a control scheme (left) and puppet customization (right)

Once the player finishes the puppet assembly and activates the puppet, the phone turns into a tangible motion controller that allows the user to control their digital puppet in real-time. Motion data is gathered via the phone's accelerometer and magnetic sensors. The sensor data is turned into orientation values and is then sent via UDP packets over a local wireless network to the Unity application. The UDP protocol was chosen for its speed advantage over TCP. Fast reaction times are necessary to generate the impression of real-time control.

In the single phone scheme, the puppet's orientation will match the phone's orientation, and rotating the phone away from the vertical position will move the

puppet in that direction (fig. 3). A quick vertical jerk makes the puppet jump. This is the most basic form of interaction with the system, and gives a similar style of control to that of a single rod puppet.

The dual phone control adds a second phone that controls either the arms or the head in addition to the main body. The secondary phone provides an interface to allow the user to select which additional limb to control: head, any single arm, or both arms (fig. 2 left). Rotations of this second phone will rotate whichever limb is selected. The access in this case is more complex, since the limbs are restricted to only rotate around a fixed joint, and the arms and head cannot match the orientation of the phone directly because they remain attached to the virtual avatar's main body.

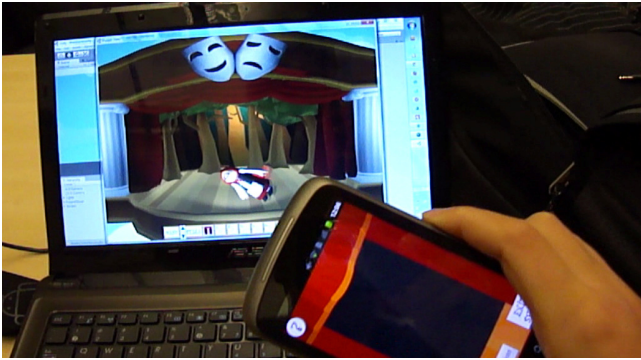


Fig. 3. Single phone control scheme; movements are directly projected onto the puppet body

A challenge of digital technology is the lack of any direct connection to the controlled avatar. Puppeteers interact not only with the main control device but might operate individual strings on a marionette or even more directly, the expressions of a hand puppet with their own hand movements. The mapping applied in *Puppettime*'s single rod control scheme is a direct translation of the phone's orientation onto the body of the virtual puppet. Players do not control a rod above or underneath the puppet but instead holds the puppet in their hands. This allows for a direct mapping of the phone as object to the virtual avatar as the virtual puppet.

All 3D objects were modeled in Maya and imported into *Unity3d*. *Unity3d* is a widespread platform for development of 3D applications and we used it here to code the stage for the puppet show. One laudable feature of Unity is its accessible multiplayer option. It allowed us to implement multiplayer performances relatively easily. The sessions are designed to be free form and users can log on and out as they please. They can also rebuild their puppets as they wish.

The virtual stage application also provides icons that control aspects of the performance that are not related to the puppets themselves, such as lights, sound, and the opening and closing of the curtains. These element control specific stage effects and can be activated on the laptop, which turns into a form of a digital stage manager.

4 Discussion

The project has been demonstrated at a number of occasions and different participants from minors to professional puppeteers have played with it. Users immediately understand the single phone mapping. They recognize their customized puppet on the stage and instantaneously map the phone as control device on the puppet's movements. The phone was readily accepted as a performing object and the responsiveness of the system was sufficient.

However, we also recognized emerging complex conditions that were initially not anticipated. Technically, the bi-manual control that allows players to control body and any selected limb with two phones at a time, works as smooth as the single phone control. However, it took players longer to adjust to the dual device control and to figure out how the rotations of the secondary phone affected the movements of the body parts. To improve performance, we limited the rotation range to avoid unnatural clipping. Still, operating the second phone in synch with the primary controller can be challenging for first time users. This corresponds with earlier work on bi-manual interaction, particularly when the actions of both hands are not mirrored but instead asymmetric and often more complex in their relation to each other. Guiard's model of the "kinematic chain" [23] indicates a higher level of gestural manipulation and expression through a hierarchical relationship between the hands' performances. This model has been discussed in interface design for entertainment technology but his concept of the non-dominant hand (usually left) to guide the general reference frame and the dominant hand to affect smaller scale effects within this frame changes in animation. Animation easily shifts the frame – e.g. from a full body movement to a detailed hand gesture – depending on the motion to be enacted. This challenge has already been noted by 2D puppetry approaches [24]. In our case, the asymmetric condition of one phone controlling the body and the other controlling a single limb allowed for a hierarchical and differently scaled interaction, as mentioned by Guiard, but players had to adjust to that condition, which itself was fixed once the devices were launched.

What this seems to call for is a bridging between different states of mapping. For example, one player commented that the initial positioning of the phones in relation to the virtual puppet is not intuitive. Because the virtual puppet remains frozen upright in the middle of the virtual stage until the phone controls are activated, recognizing the initial mapping before this launch is impossible. Consequently, the performing object of the phone did not match the representation on the virtual stage at the moment of activation. Control started with a kind of shock effect.

Another comment pointed to the device and its physical affordances. An expert puppeteer had no problems with the orientation of the devices but mentioned a lack of weight for the digital puppet. For example, the center of gravity is an important element in physical puppet controls and it is not developed in the current version of *Puppettime*. This mismatch distanced the virtual puppet from the expert puppeteer. He expected certain affordances that the hardware could not provide. One goal of the project was to test available cell phone hardware for the particular puppetry metaphor. Except for this expert statement, we encountered little to no problems with the role of

the cell phone hardware itself. However, we added a “freeze in location” feature that allowed players to freeze the virtual puppet’s position in space by pressing the touch screen of the phone. This allowed for a gradual exploration of the puppet’s animation and the controls. While it is not possible to adjust the weight of the device to that of the virtual puppet, this kind of multimodal interaction design might offer a possible solution for more nuanced control.

Our system can track movement along two axes, which meant that we had to include limitations in the animation control on the 3D stage. All puppets perform on a 3D stage but their movements upward or downward are limited to jumping. It is worth noting, that only the expert puppeteer commented on this limitation and suggested changes to the puppet designs and not necessarily to the control schemes. No other player noticed this seemingly obvious restriction and commented on it negatively. This indicates a possibly useful restriction for future designs for mobile phones as performative objects and indicates that the consistency and simplicity of the mapping is more important than the sheer range of animation effects called by it.

5 Conclusion and Future Work

We propose that puppetry provides a useful metaphor to explore mobile phones as performing objects using motion gestures. But the here introduced example project, *Puppettime*, also highlights limitations of the technology and challenges for future design. The system currently uses accelerometers and magnetic sensors to generate motion data. Most recent phones also include gyroscopes, capable of providing better orientation data. As they become more commonly available in smart phones, *Puppettime* and other gesture-based input concepts might very well implement the here explored interaction methods in their design. This does not necessarily mean more gestural input but instead better-designed transitions between possible states during this input.

The current interaction design for *Puppettime* showed a number of friction points for such a design. These include the initial orientation of phones in relation to a virtual character and the bi-manual control scheme. The transformation of the mobile device from a traditional cell phone to a performative object seems to cross a threshold moment that needs to be clearly understood by the puppeteer/ player. Future design iterations should improve this transition to support the mental projection by the player and lead into a less rapid feeling of sudden control (or loss of control). Although mobile devices are turning increasingly into tangible objects with new interaction design, they are likely to keep some of their more traditional interface structures. Thus, these thresholds between different interaction paradigms will remain in the case of the mobile phone and lead to a kind of double identity.

Acknowledgements. *Puppettime* was initially implemented by Sanjeev Nayak, Austin Denoble, and Vincent Zunga. We are thankful for in kind support by AT&T and assistance by Mark Guzdial.

References

1. Wexelblat, A.: Research Challenges in Gesture: Open Issues and Unsolved Problems. In: Wachsmuth, I., Fröhlich, M. (eds.) *GW 1997. LNCS (LNAI)*, vol. 1371, pp. 1–11. Springer, Heidelberg (1998)
2. Ullmer, B., Ishii, H.: Emerging frameworks for tangible user interfaces. *IBM Syst. J.* 39(3-4), 915–931 (2000), doi:10.1147/sj.393.0915
3. Dourish, P.: What we talk about when we talk about context. *Personal Ubiquitous Computing* 8(1), 19–30 (2004), doi:10.1007/s00779-003-0253-8
4. Mazalek, A., Chandrasekharan, S., Nitsche, M., Welsh, T., Clifton, P., Quitmeyer, A., Peer, F., Kirschner, F., Athreya, D.: I’m in the Game: Embodied Puppet Interface Improves Avatar Control. In: *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*, pp. 129–136. ACM, New York (2011)
5. Vajk, T., Coulton, P., Bamford, W., Edwards, R.: Using a Mobile Phone as a “Wii-like” Controller for Playing Games on a Large Public Display. *International Journal of Computer Games Technology* (2), 1–7 (2008)
6. Chehimi, F., Coulton, P.: Motion controlled mobile 3D multiplayer gaming. In: *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, pp. 267–270. ACM, New York (2008)
7. Fan, M., Li, X., Zhong, Y., Tian, L., Shi, Y., Wang, H.: Surprise Grabber: a co-located tangible social game using phone hand gesture. In: *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work*, pp. 625–628. ACM, New York (2011)
8. Hinckley, K., Song, H.: Sensor synaesthesia: touch in motion, and motion in touch. In: *Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems*, pp. 801–810. ACM, New York (2011)
9. Edge, D., Blackwell, A.F.: Bimanual tangible interaction with mobile phones. In: *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, pp. 131–136. ACM, New York (2009)
10. Jones, E., Alexander, J., Andreou, A., Irani, P., Subramanian, S.: GesText: Accelerometer-Based Gestural Text-Entry Systems. In: *Proceedings of CHI 2010*, pp. 2173–2182. ACM, New York (2010)
11. Ruiz, J., Li, Y.: DoubleFlip: A Motion Gesture Delimiter for Mobile Interaction. In: *CHI 2011: ACM Conference on Human Factors in Computing Systems*, pp. 2717–2720. ACM, New York (2011)
12. Rico, J., Brewster, S.: Usable Gestures for Mobile Interfaces: Evaluating Social Acceptability. In: *Proceedings of CHI 2010*, pp. 887–896. ACM, New York (2010)
13. Sall, A., Grinter, B.: Let’s Get Physical! In, Out and Around the Gaming Circle of Physical Gaming at Home. *Computer Supported Cooperative Work (CSCW)* 16(1-2), 199–229 (2007)
14. Blumenthal, E.: *Puppetry: A World History*. Harry N. Abrams, New York (2005)
15. Bottoni, P., Faralli, S., Labella, A., Malizia, A., Pierro, M., Ryu, S.: CoPuppet: Collaborative Interaction in Virtual Puppetry. In: Adams, R., Gibson, S., Arisona, S.M. (eds.) *Transdisciplinary Digital Art. Sound, Vision and the New Screen. CCIS*, vol. 7, pp. 326–341. Springer, Heidelberg (2008), doi:10.1007/978-3-540-79486-8_27
16. Walser, R.: *Elements of a Cyberspace Playhouse*. Paper Presented at the Proceedings of the National Computer Graphics Association, Anaheim, CA, March 19-22 (1990)

17. Marshall, P., Rogers, Y., Scaife, M.: PUPPET: Playing and Learning in a Virtual World. *International Journal of Continuing Engineering Education and Life-long Learning* 14(6), 519–531 (2004)
18. Mayora, O., Costa, C., Papiatseyeu, A.: iTheater Puppets Tangible Interactions for Storytelling. Paper Presented at the Intelligent Technologies for Interactive Entertainment, pp. 110–118. Springer, Heidelberg (2009)
19. Tillis, S.: The Art of Puppetry in the Age of Media Production. *TDR* 43(3), 182–195 (1999)
20. Eide, P.: Digital Puppetry. *The Puppetry Journal* 60(1), 12–13 (2008)
21. Lu, F., Tian, F., Jiang, Y., Cao, X., Luo, W., Li, G., Zhang, X., Dai, G., Wang, H.: ShadowStory: Creative and Collaborative Digital Storytelling Inspired by Cultural Heritage. In: *Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems*, pp. 1919–1928. ACM, New York (2011)
22. Kim, S., Zhang, X., Kim, Y.J.: Haptic Puppetry for Interactive Games. In: Pan, Z., Aylett, R.S., Diener, H., Jin, X., Göbel, S., Li, L. (eds.) *Edutainment 2006*. LNCS, vol. 3942, pp. 1292–1302. Springer, Heidelberg (2006)
23. Guiard, Y.: Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. *Journal of Motor Behavior* 19, 486–517 (1987)
24. Kipp, M., Nguyen, Q.: Multitouch puppetry: creating coordinated 3D motion for an articulated arm. In: *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, pp. 147–156. ACM, New York (2010)