# Co-locating Haptic and Graphic Feedback in Manual Controls

**Colin Swindells, Mario J. Enriquez, Karon E. MacLean, Kellogg S. Booth**
Computer Science, University of British Columbia
201 – 2366 Main Mall, Vancouver, BC V6T 1Z4
e-mail address {swindell, enriquez, maclean, ksbooth}@cs.ubc.ca

**ABSTRACT**

Based on data showing performance benefits separately for programmable versus static feedback in manual controls and for co-location of dynamic haptic/graphic controls, we hypothesized that combining these benefits would further improve context- and task-specific feedback. Two application prototypes created to explore this premise for (a) streaming media navigation and (b) information flow demonstrate the potential affordances and performance benefits for integration in manual controls. Finally, we describe two practical fabrication methods: embedding a haptic controller into an active graphic display panel, and rear projection onto a passive surface with an embedded haptic control.

**Author Keywords**

Multimodal user interface, manual controls, co-location, haptics, graphics, display with hole, LCD.

**ACM Classification Keywords**

H5.2 [Information interfaces and presentation (e.g., HCI)]: User Interfaces – Haptic I/O, Graphical user interfaces (GUI), Input devices and strategies, Interaction styles (e.g., commands, menus, forms, direct manipulation), Prototyping. I3.6 [Methodology and Techniques]: Interaction techniques. B4.2 [Input/output devices]: Image display.

## MOTIVATION

Dynamic, programmable haptic controls, such as knobs, sliders, and buttons, are increasingly used for industrial controls, games, automobiles, and ubiquitous computing applications [8]. These manual controls typically replace passive haptic controls with static labels. For example, a radio's volume knob will typically have a static label indicating the knob positions corresponding to *minimum* and *maximum* volume levels. Conversely, current implementations of programmable haptic controls have tended to decouple the haptic and graphic components [2, 8]. In other words, the control feedback is no longer co-located with the graphical feedback. Such decoupling often *reduces* affordances and task performance – as predicted by previous integrality and separability user studies [3, 6, 16]. For example, Jacob et al. [6] demonstrated performance improvements when the structure of the perceptual space of a graphical interaction task mirrored the physical control space of the input device. Ware & Rose [16] extended this research in a direction directly related to our prototypes. They showed that co-located haptic and graphic handles can improve task performance, and that these performance benefits can be obtained even when the haptic and graphic handle shapes do not match. Performance improvements have also been shown for spring-mass haptic feedback models compared to graphic-only feedback [5]. Thus, our prototypes combine the benefits of these prior co-location and haptic research contributions.

**Figure 1: Haptic knob embedded into an acrylic graphical rear-projected display surface**

In this paper, we demonstrate two physical prototypes in order to show how designers can easily integrate programmable haptic and graphic controls. We hope that the techniques used in our prototypes will provide interface designers with a new prototyping method to potentially improve affordances and task performance. These same techniques are also appropriate for incorporation into commercial products using currently available technologies.

## EXAMPLE PROTOTYPING SCENARIOS

To illustrate the potential of co-located manual control, we prototyped two application scenarios:

- Media editing with an embedded haptic knob
- Information flow with an embedded haptic slider

These examples were chosen to be common, easily understood tasks. They were also chosen to intuitively illustrate improved affordances and task performance compared to current methods without co-located haptic and graphic feedback. An accompanying video demonstrates users interacting with each of these prototypes.

### Manipulation of Streaming Media

Graphical user interface (GUI) media editors often contain graphical jog dial widgets for navigating audio / video clips. Controlling these GUI widgets with a mouse or other pointing device is slow and cumbersome; the handle mismatch decreases the coupling between the user and the task. Conversely, many physical media players (e.g. video player consoles and audio players), have physical jog dials for frequently used functions such as *play*, *stop*, *fast forward*, and *rewind.* However, such a jog dial still suffers from the problem of disjoint haptic and graphic feedback. A previous application exploring the coupling of static knobs and graphical displays has been performed by Kobayashi & Akamatsu [7]. Our media editing prototype extends this approach by using a co-located *dynamic* haptic handle.

Using our physical knob setup (shown in Figure 1), we illustrate the potential of integrating dynamic graphical components with the force-feedback possibilities created by embedding the physical controller directly into the display. Embedding provides the force-feedback device (e.g., the knob motor) with a grounding frame of reference. Consequently, a wide variety of dynamic haptic effects can be fully integrated within the dynamic graphical display. For example, our knob can provide up to 180 mNm of continuous torque with an update time of 100 μs. Without such physical embedding, this quality of haptic feedback is extremely difficult to obtain.

Figure 2 shows our prototype of a haptic knob coupled to a graphical filmstrip, representing a media stream being manipulated. It couples a haptic knob with a visual representation of the physical metaphor described by Snibbe et al. [13]. By placing the top of the physical knob beneath the graphical filmstrip, a strong visual and haptic coupling affordance is achieved. To enhance the effect, graphical notches were added to the bottom of the filmstrip so they visually interlock with notches on the knob's perimeter. As the user rotates the knob, haptically rendered detents (i.e., 'clicking' sensations) are provided in synchrony with the graphical notches and film strip boundaries. Additional effects, such as torque ramps at the start and end of film sections, and torque textures that represent content of the graphical media near the knob, could be easily added.

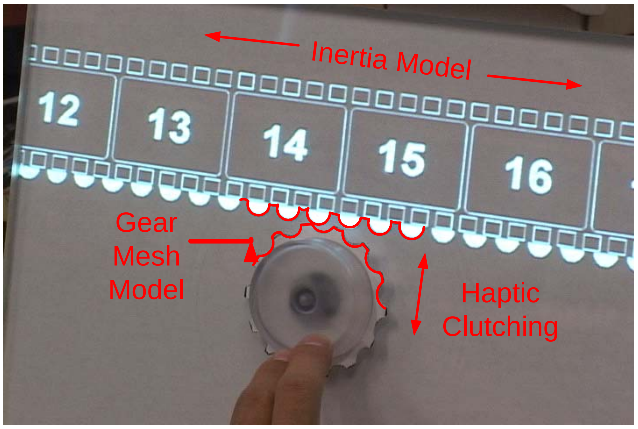


**Figure 2: Media editing with a haptic knob coupled to a graphical filmstrip**

To experiment with an interaction not possible with a static knob, we explored clutching effects, again as described by Snibbe et al. [13]. Clutching is achieved by rendering a virtual heavy mass, initially stationary, that can be engaged by pushing down on the knob to come in contact and then 'spinning it up'. The graphical film starts to rotate in synchrony with the virtual mass. When the user releases downward pressure on the knob, the clutch is disengaged; but the simulated inertia keeps the virtual mass rotating, and the film rolling across the screen. Haptic effects of the media can be felt on the knob as the film visually scrolls. Third, the user can stop the film scrolling by pushing the knob down – thereby applying a 'brake' and grounding friction to the simulated inertia in the film strip. In contrast to the distributed positions in the initial implementation of this metaphor by Snibbe et al. [13], the current implementation has the knob "in contact" with the graphical film, providing a stronger sense of direct interaction.

### Information Flow

Our second example involves information flow through a metaphorical pipe, with the prototype illustrated in Figure 3. Flow through the pipe is indicated graphically by small

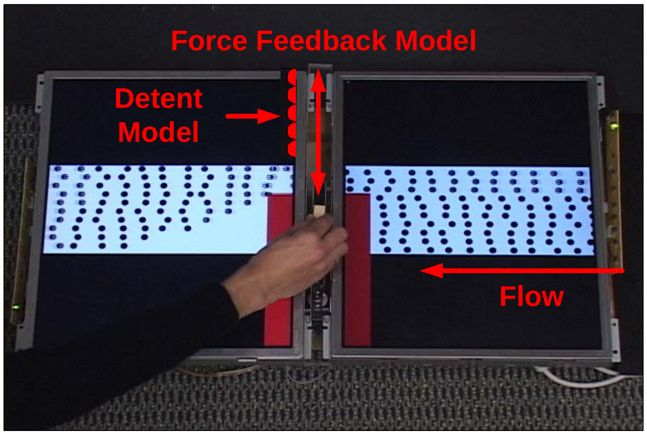


**Figure 3: Information flow through a virtual pipe using haptic and graphic valve widgets**

white circles, which could represent information such as network traffic, electricity usage, or water supply. To create a valve in the pipe, we built a force feedback slider and embedded it between two networked graphical liquid crystal displays (LCDs). To further improve the conceptual coupling between the haptic and graphic feedback controls, a graphical rectangle is fixed to the physical slider handle.

If used in a control room, collections of valve interfaces such as the prototype in Figure 3 could be connected into a network of graphical pipes on a wall or table. Thus, a tangible interface similar to those explored by Ullmer et al. [15] and MacLean et al. [9] can be realized. More flexible information flow surfaces could be obtained by embedding dynamic force-feedback sliders, knobs, and buttons into the DataTiles introduced by Rekimoto et al. [11]. In other words, a physical manual control could be embedded into a graphical tile. Tiles could then be mounted and re-arranged into sophisticated information topologies.

Like the previous knob example, the dynamic haptic slider provides many useful interactions not possible with static or purely graphical sliders. For example, as the user moves the valve handle (see Figure 3), s/he feels different haptic feedback related to the flow of particles against the slider. Reduced or increased flow through the pipe produces more or less haptic resistance via the slider. The user can feel a difference between fully closed and 99% closed. Because the haptic and graphic slider handle is mounted directly over the graphical pipe, the user gets a much stronger conceptual connection to the 'fully closed' and 'mostly closed' system states compared to a typical GUI widget controlled with a mouse. Because the haptics are completely simulated, a variety of haptic effects ranging from physically-based models to completely abstract models can be developed depending on the user's preferred metaphorical context. Additional effects (not implemented) could include appropriate graphic and haptic icons for different pipes and/or different data 'flow' within the pipes. This would especially help in time-and-safety-critical applications because users would learn to associate combined visual/haptic icons with particular system states; multimodal reinforcement has been shown to provide significant performance benefits over single modality cues [14].

## PROTOTYPING TECHNIQUES

The following sections describe the two rapid physical prototyping techniques that we used to realize the scenarios described above in terms of actual integration of the haptic and graphical displays. The separate graphic and haptic implementation details are fairly standard: we programmed the prototypes with a combination of C++ / OpenGL and Visual Basic code, and used our Twiddler and RTPM tools to set up the haptic control architecture [10, 12]. We custom built the physical knobs and sliders.

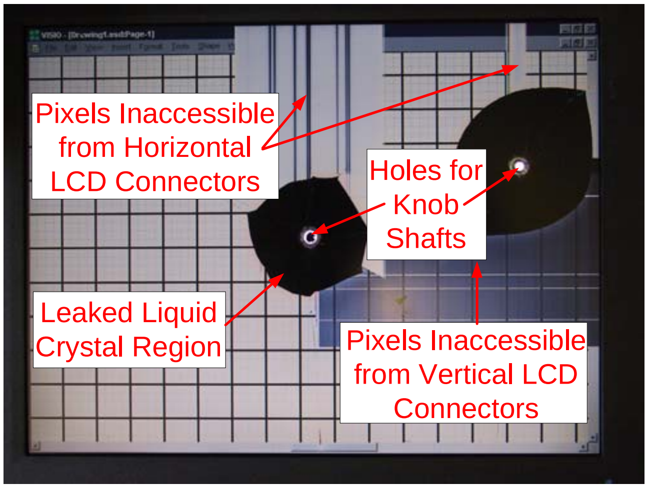


**Figure 4: LCD pixels damaged by physical drilling of two holes for knob shafts (following drilling of two experimental holes and subsequent enlarging of initial holes)**

### Integrating into an Active Display Surface

We first explored physical milling of a display surface (see details in our accompanying video). Specifically, we drilled a hole in an LCD through which we placed a knob shaft. Most pixels remained operational after drilling (see Figure 4). A fully functional LCD with a hole, or milled region, could be obtained by adding additional horizontal and vertical connectors to the pixels.

As illustrated in our slider example in Figure 3, multiple LCD surfaces can also be 'wrapped' around a physical control using current LCD technology. This may be cheaper and less destructive than drilling or milling an LCD: we estimate that approximately one LCD will be destroyed for every successful drilling or milling effort because of difficulties in preventing liquid crystal leakage as shown in Figure 4. Nevertheless, milling an LCD is a practical prototyping technique when imperfections in the graphical display surface may be acceptable.

A related display option could be based on E Ink [4], which consists of thin paper-like sheets with actively refreshable pixels. Their thinness and general manufacturing would ease the creation of active surfaces with holes, compared to LCD technology.

The main advantage of active surfaces over projection techniques is the reduced physical volume required; in our design the additional space behind or in front of the graphical surface is limited to the depth of the actuator, which is much less than the throw distance of a projector.

### Rear Projection

We ended up using a rear projection technique for our knob prototype (media manipulation, Figures 1 & 2) because it was quicker and cheaper to obtain a fully functional display surface (no missing pixels), and the extra space required was acceptable. As illustrated in Figure 5, a haptic knob can be mounted into a transparent support surface such as acrylic. The surface can be coated with a semi-transparent medium, such as paper, and rear projected using a standard data projector. This prototype required 1 m of space behind the screen; however, future versions could be more compact. For example, mini-projectors (as envisioned in [1]) could be used in the near future, by embedding multiple very small, low-cost projectors closely behind the display surface. Front projection techniques would rarely be a viable option because they would almost always produce disruptive shadows on the surface as the user interacts with the physical control.

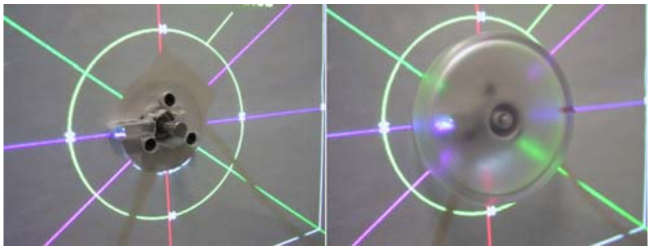

**Figure 5: Example mounting of a haptic component into a graphical screen**

## SUMMARY & FUTURE WORK

Previous research has shown both the performance benefits of adding programmable haptic feedback models to graphical systems, and of co-locating haptic and graphic feedback. In this paper, we demonstrate a system that combines these benefits by co-locating programmable haptic controls with dynamic visual feedback. We have presented two application prototypes, media navigation and information flow, as compelling examples of the possible affordances and task performance benefits for future manual controls. We have also illustrated two practical methods for implementing this type of integrated configuration: integrating programmable manual controls into an active media surface, and rear projection on to a passive surface with embedded controls.

Future work will include user studies to confirm our inferences from individual co-located and programmable haptic feedback research. We will compare the performance differences between [co-located and disjoint] and [haptics and/or graphics] conditions for both the presented and future prototypes. Specifically, we hope to devise more sophisticated co-located haptic-graphic coupling, to allow interactions not permitted by static or distributed controls.

## ACKNOWLEDGMENTS

We thank Ron Fussell, Jocelyn Smith, Jérôme Pasquero, Melanie Tory and the SPIN Lab at UBC for their constructive comments and suggestions.

## REFERENCES

1. Beardsley, P., Van Baar, J., Raskar, R., & Forlines, C. Interaction Using a Handheld Projector. *Computer Graphics and Applications, 25*, 1, IEEE Press (2005), 39 - 43.
2. BMW Group, iDrive. http://www.bmw.com/com/en/index_narrowband.html
3. Buxton, W. There's more to interaction than meets the eye: Some issues in manual input. In *User Centered System Design: New Perspectives on Human-Computer Interaction*. Lawrence Erlbaum, Hillsdale, NJ, USA, 1986, 319-337.
4. E Ink Corporation. http://eink.com
5. Huang, F., Gillespie, R.B., & Kuo, A. Haptic feedback improves manual excitation of a sprung mass. In *Proc. HAPTICS 2004*, IEEE Press (2004), 200 – 207.
6. Jacob, R. J. K., Sibert, L. E., McFarlane, D. C., & Mullen, M. P. Jr.. Integrality and separability of input devices. *Transactions on Computer-Human Interaction, 1*, 1, ACM Press (1994), 3 - 26.
7. Kobayashi, S., & Akamatsu, M. Spinner: A Simple Approach to Reconfigurable User Interfaces. In *Proc. International Conference on New Interfaces for Musical Expression (NIME)*, (2005), 208 - 211.
8. Immersion Corporation. http://www.immersion.com
9. MacLean, K. E., Snibbe, S. S., & Levin, G. *Tagged Handles: Merging Discrete and Continuous Control*, In *Proc. CHI 2000*, ACM Press (2000), 225 - 232.
10. Pava, G., MacLean, K. *Real Time Platform Middleware for Transparent Prototyping of Haptic Applications.* In *Proc. HAPTICS 2004*, IEEE Press (2004), 383 - 390.
11. Rekimoto, J. Ullmer, B. & Oba, H. *DataTiles.* In *Proc. CHI 2001*, ACM Press (2001), 269 - 276.
12. Shaver, M. J. & MacLean, K. *The Twiddler: A Haptic Teaching Tool: Low-Cost Communication and Mechanical Design*. Univ. of British Columbia, Dept. of Computer Science, TR-2005-09, 2003. http://www.cs.ubc.ca/cgi-bin/tr
13. Snibbe, S.S., MacLean, K.E., Shaw, R., Roderick, J., Verplank, W.L., & Scheeff, M. Haptic techniques for media control. In Proc. UIST 2001, ACM Press (2001), 199 - 208.
14. Spence, C. & Driver, J. (Eds.). *Crossmodal Space and Crossmodal Attention*, Oxford University Press (2004).
15. Ullmer, B., Ishii, H., & Jacob, R. *Tangible Query Interfaces.* In *Proc. INTERACT 2003*, IFIP Press (2003).
16. Ware, C., & Rose, J. Rotating virtual objects with real handles. Transactions on Computer-Human Interaction, 6, 2, ACM Press (1999), 162 - 180.