Hidden Interfaces for Ambient Computing: Interacting with High-brightness Visuals through Everyday Materials

Alex Olwal
olwal@acm.org
Google, Mountain View, CA

Artem Dementyev
artemd@google.com
Google, Mountain View, CA

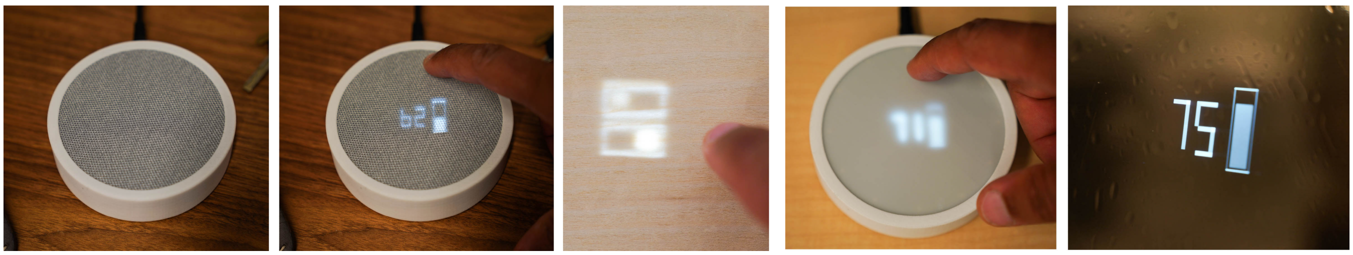

Figure 1: Our displays can be embedded underneath various transmissive materials, such as textile, veneer, acrylic or one-way mirrors, and appear on-demand for touch-based interaction. Parallel rendering of rectilinear elements achieves ultra-bright graphics on low-cost passive OLED displays. Our approach enables low-complexity interaction technology that can co-exist with traditional materials and aesthetics. ***From left to right: Display hidden under textile; Touch-activated controls; Switches under veneer; Diffuse controls under frosted acrylic; Sharp graphics behind a one-way mirror.***

ABSTRACT

Consumer electronics are increasingly using traditional materials to allow technology to better blend into everyday environments. Specifically, transmissive materials enable emissive displays to disappear when turned off, and appear when turned on. However, covering displays with textile meshes, veneer, one-way mirrors, or translucent plastics greatly limit the display brightness of typical graphical displays.

In this work, we leverage a parallel rendering technique to enable ultrabright graphics that can pass through transmissive materials. While previous work has shown interactive hidden displays, our approach unlocks expressive interfaces with practical end-to-end software and low-cost hardware implementation on mass-produced passive OLED displays.

We developed a set of interactive prototypes with touch-sensing that can blend into traditional aesthetics due to the ability to provide user interfaces through wood, textile, plastic and mirrored surfaces.

CCS CONCEPTS

• **Human-centered computing Ubiquitous and mobile computing systems and tools**;

KEYWORDS

hidden displays, ambient computing, ubiquitous computing, high brightness, rectilinear, passive OLED

CHI '21 Extended Abstracts, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-XXXX-X/18/06.
https://doi.org/10.1145/3411763.3451558

ACM Reference Format:
Alex Olwal and Artem Dementyev. 2021. Hidden Interfaces for Ambient Computing: Interacting with High-brightness Visuals through Everyday Materials. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '21 Extended Abstracts), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3411763.3451558

1 INTRODUCTION

With the proliferation of consumer electronics and smart appliances, homes are beginning to embrace the vision of ubiquitous computing. A graceful integration of this technology requires that it can respect existing aesthetics, social forms, and adaptation to user's styles. There is an increasing desire to create smart objects and appliances, which can preserve the industrial design and aesthetics of traditional materials, while providing access to interaction and digital display on demand.

Today's interactive devices tend to fall into two categories. There are devices with no or minimal displays, where the main interaction is performed through an app on a mobile device, or using speech. On the other hand, there are more high-fidelity displays with touch input. The disadvantage of the latter is that digital screens create tensions with environments that emphasize traditional materials, and when turned off, their black screens become "black holes" in the physical space.

To enable expressive interactions with graphical displays without these tradeoffs, strategies have been developed to realize displays that can disappear. "Smart mirrors" use partially reflective material to hide a display underneath, while making it possible to display graphics when turned on. Today's brightest displays are expensive as they are designed for high quality, and it would be paradoxical to cover them. The typical rendering strategies for low-cost displays on the other hand, are not sufficiently effective in generating the amount of light that is needed to pass through the transmissive materials that could hide them.

To address this challenge, smart speakers are increasingly using textile coverings with special-purpose illumination. They employ high-brightness LEDs underneath textiles to provide abstract indicators (e.g., Google Home Mini), 7-segment displays to show time (e.g., Sony LF0S50G Wireless Speaker) or to illuminate fixed icons (e.g., Withings Aura). These special-purpose displays are, however, not sufficiently expressive to enable reconfigurable user interfaces with graphical elements, scalable vector graphics and typography.

Hidden displays are often explored in the context of textiles so that they can be discretely embedded in clothing. Researchers have demonstrated interesting approaches to combining display capabilities with textiles. It has been shown how braiding [9], knitting [10], and weaving [2] can enable the integration of flexible, emissive optical fibers with textile fibers, using manual techniques or through programmable industrial machines. Discrete LEDs have also been manually sewn into the textiles with conductive thread, such as demonstrated with LilyPad Arduino [1]. To add graphical display capabilities, other approaches include liquid crystal ink [12], thermochromic ink [3, 11], and electroluminescent materials [5]. Living Wood [6] embeds LEDs inside wood panels, but acknowledges limited scalability of individual wiring to each LED.

Ideas and research around ambient displays have been popular in ubiquitous computing [7]. E-ink has been one of the most interesting technologies for ambient displays. Some researchers have explored how E-ink could be integrated into wearables, such as clothing [4]. Ambient displays also take many other non-graphical forms. One approach has been to add custom-designed displays to everyday objects, such as paper or walls, using screen printing of electroluminescent materials [8].

In this work, we propose a new approach to allow emissive graphical displays to be hidden under traditional materials with transmissive properties, enabling them to appear only for on-demand interaction (See Figure 1). We enable an increase in brightness by using efficient rendering of user interface primitives with rectilinear geometries – straight, axis-aligned lines or rectangles. We can also render circles and diagonal lines, but with lesser brightness improvements. In contrast, axis-aligned rectangles allow us to leverage optimized display driver mechanisms to parallelize the rendering of multiple columns and rows, which greatly improves light output. With our hardware prototype, we demonstrate user interfaces, which can appear through transmissive materials, using low-cost displays. We couple our visual output with capacitive sensing underneath the surface to build user interfaces with touch input.

2 EFFICIENT RENDERING FOR HIGH-BRIGHTNESS GRAPHICS ON PASSIVE OLEDS

Today's consumer devices mostly employ different types of active-matrix organic light-emitting diode (AMOLED) displays. They provide high-quality, flicker-free graphics due to all pixels having dedicated memory, which allows them to maintain per-pixel state. AMOLEDs, however have a significantly more complicated manufacturing and integration process making them considerably more expensive than their predecessor, passive-matrix OLEDs (PMOLEDs). Given their significantly lower cost (e.g., 2.50 USD PMOLED[1] vs. 25 USD AMOLED[2]) and ease of integration, PMOLEDs continue to be produced for simple devices where AMOLED is overkill.

PMOLEDs are based on a simpler design without per-pixel memory. While this greatly reduces manufacturing costs and complexity, it does require active display driver circuitry. The display driver runs an update loop to select columns and power rows, which controls how current flows through and illuminates specific pixels. To display general-purpose content, current display drivers inspect their framebuffer using the scanline technique (row-by-row), and select which columns to activate for the row. For each frame update, they thus need to iterate through all rows for the height of the display. This also means that only one row has pixels turned on at any given time, which limits the instantaneous display brightness to the luminance of 1 row of pixels (width × per-pixel luminance).

However, similarly to how multiple columns can be selected, nothing prevents us from simultaneously enabling multiple rows. This would result in the same pattern on each row. This does not provide as much advantage to diagonal lines or circles, given that the contents has little redundancy across rows. We would need a unique row/column combination for each pixel in a diagonal line across the width/height of the display. The parallel powering of multiple rows can, however, be very effectively applied to rectilinear, axis-aligned geometry, such as rectangles, where content is repeated across rows. Figure 2 shows how a rectangle outline can be rendered with just two operations, when parallel rendering can activate all similar rows. The corresponding scanline version will need a frame for each row, resulting in as many operations as there are rows (*display height* operations).

This architecture also supports unique brightness of each pixel, through variation of the voltage differential between rows and columns. Figure 1 shows how slide switches under veneer are rendered dim when off (top) and brighter when on (bottom).

There are both temporal and spatial advantages to this approach of parallel rendering (e.g., the speed-up can eliminate flicker). However, in this work we are particularly interested in leveraging the increased brightness, given that more areas of the display can be illuminated in each frame, compared to only a single row for scanline rendering.

3 RECTILINEAR USER INTERFACE PRIMITIVES AND CONTROLS

To fully take advantage of the parallel rendering of multiple rows, we designed a system around rectilinear graphics that can be rendered with significant efficiency given the redundancy across rows and columns.

(1) **Lines and rectangles**. Basic elements can be render effectively with 1 or 2 operations.

[1]PMOLED, 2.50 USD https://www.alibaba.com/product-detail/1-32-inch-OLED-display-screen_62585496231.html
[2] AMOLED, 25 USD, https://www.alibaba.com/product-detail/454-454-1-39-inch-Round_62369474552.html

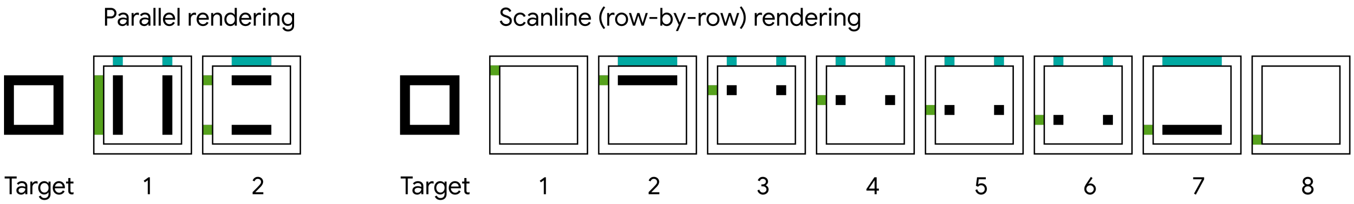

Figure 2: Parallel rendering activates multiple rows simultaneously to take advantage of redundancies in rectilinear graphics, which can greatly reduce the number of frames needed to render contents. The number of frames needed depend on the content. The Scanline content-agnostic rendering processes each row, which limits the brightness given that at most one row can be illuminated at any given time.

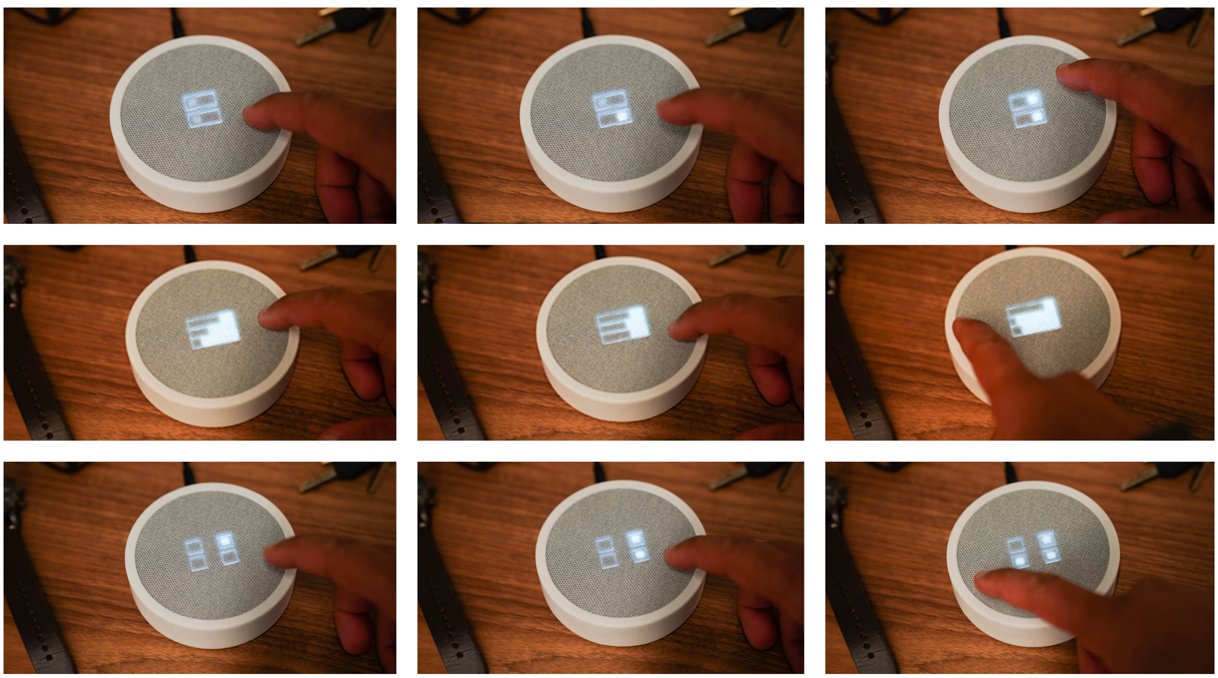

Figure 3: *Top: Switch toggle UI element on textile.* The switch will toggle on/off when touched nearby. It renders dark when turned off and bright when turned on. *Center: Slider UI element on textile.* The sliders move left when touching the capacitive touch sensor on the left and vice versa. The sliders are composed of a dark hollow rectangle and a bright solid rectangle. *Bottom: Toggle checkbox UI element on textile.* Checkboxes will toggle between filled/hollow (on/off) when the closest electrode is tapped.

(2) **Sliders: continuous parameters**. The vertical/horizontal slider is a common UI element for controlling and visualizing continuous parameters (e.g., audio volume). In 3 operations, we combine a hollow rectangle (outline), partially filled with a solid rectangle. See Fig. 3, top.
(3) **Selection controls: Toggle, radio buttons, and switches**. The toggle can be used to implement both checkboxes (multi-choice) and radio buttons (exclusive selection). We can render it with a dark hollow rectangle when not selected (2 operations), and add a bright filled rectangle when selected (3 operations). For switches, we use a dark outline that we fill with a dark rectangle to the left when turned off, and with a bright rectangle to the right, when turned on (3 operations). See Fig. 3, center and bottom.
(4) **Cursor: 2D selection.** A 2D cursor could be useful for certain interactions. For example, for controlling the color of smart lights, we could use the Y-axis for brightness, and X-axis for hue. To support such interactions, we use a crosshair, which can be rendered in 2 operations.
(5) **Typography: segmented characters**. To display texts and digits, we implemented optimized rendering of 7-segment characters. Individual characters require at most 3 operations, whereas a string of any length requires at most 5 operations due to the redundancies across rows and columns.

4 IMPLEMENTATION

We leverage a passive OLED display (Truly) with 128×96 resolution where all all row/column lines are broken out to a connector. We use a custom PCB with fourteen 16-channel Digital-to-Analog Converters (DACs) to directly interface the 224 lines (14 DACs × 16 channels = 224 channels = 128 columns + 96 rows) from a Raspberry Pi 3 A+ over SPI. The software creates each frame by writing row and column voltages to a framebuffer, which are then clocked out to the DACs (Analog Devices LTC2668).

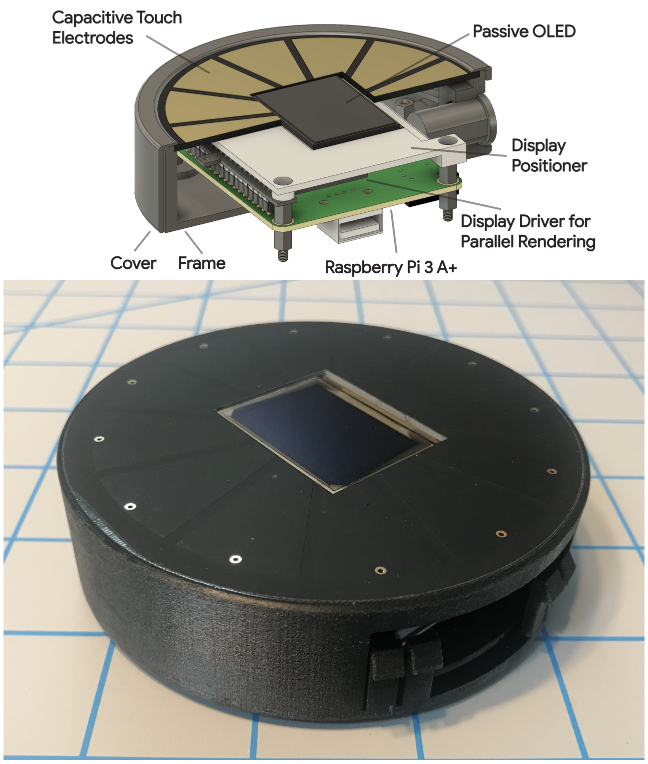

Figure 4: Our physical prototype is designed as a compact stack consisting of a Raspberry PI 3 A+ single-board computer, custom PCB with analog DACs, custom passive OLED, and our ring-shaped PCB with electrodes for capacitive touch sensing.

For touch interaction, we use a ring-shaped PCB surrounding the display with 12 electrodes arranged in arc segments. The electrodes are connected to a proximity capacitive touch sensor controller (NXP MPR121), which is interfaced over I2C from the Raspberry PI 3 A+. Our software is written in Python. A 3D-printed cylindrical enclosure encapsulates all components and is designed to hold in place different materials. See Figure 4.

5 LIMITATIONS AND FUTURE WORK

The use of rectilinear primitives constraints which content we can use and limits the UI to a specific design aesthetic. In the future, we want to investigate how we can represent more general text, fonts, and images.

Additionally, we observe that increased UI complexity adds operations, which reduces the overall brightness. For a user interface where different screens have different complexity, we will need to limit the brightness to the screen with the most operations for consistent application brightness.

In future work, the technique could be extended on larger displays and with capacitive touch sensing overlaid on display instead of around it.

While the proof-of-concept implementation that we use is based on a large array of DACs, more efficient designs could be engineered based on FPGA (field-programmable gate array) chips or ASICs (application-specific integrated circuit).

6 CONCLUSIONS

We introduce opportunities to enable interfaces that can be embedded in traditional materials and appear on demand. We leverage parallel rendering of rectilinear, axis-aligned interface elements. This high-speed rendering technique enables the significant brightness that is necessary to project through the materials that hide the display when not in use. We use an interactive hardware system to implement the approach in different materials, such as veneers, plastics, textiles, and one-way mirrors. The optimization of widely available passive OLED displays makes our approach particularly suitable for ubiquitous and ambient computing, where scalability and cost are critical considerations.

ACKNOWLEDGMENTS

We thank Ali Rahimi, Roman Lewkow, Olivier Bau, Aaron Soloway, Mayur Panchal, Sukhraj Hothi and the Google Interaction Lab.

REFERENCES

[1] Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. The LilyPad Arduino: Using Computational Textiles to Investigate Engagement, Aesthetics, and Diversity in Computer Science Education. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 423–432. https://doi.org/10.1145/1357054.1357123

[2] Kunigunde Cherenack, Christoph Zysset, Thomas Kinkeldei, Niko Münzenrieder, and Gerhard Tröster. 2010. Woven electronic fibers with sensing and display functions for smart textiles. *Advanced materials* 22, 45 (2010), 5178–5182.

[3] Laura Devendorf, Joanne Lo, Noura Howell, Jung Lin Lee, Nan-Wei Gong, M. Emre Karagozler, Shiho Fukuhara, Ivan Poupyrev, Eric Paulos, and Kimiko Ryokai. 2016. "I Don't Want to Wear a Screen": Probing Perceptions of and Possibilities for Dynamic Displays on Clothing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 6028–6039. https://doi.org/10.1145/2858036.2858192

[4] Christine Dierk, Molly Jane Pearce Nicholas, and Eric Paulos. 2018. AlterWear: Battery-Free Wearable Displays for Opportunistic Interactions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3173574.3173794

[5] Bin Hu, Dapeng Li, Okan Ala, Prakash Manandhar, Qinguo Fan, Dayalan Kasilingam, and Paul D Calvert. 2011. Textile-Based Flexible Electroluminescent Devices. *Advanced Functional Materials* 21, 2 (2011), 305–311.

[6] Lasse Kaila, Henrik Raula, Miika Valtonen, and Karri Palovuori. 2012. Living Wood: A Self-Hiding Calm User Interface. In *Proceeding of the 16th International Academic MindTrek Conference* (Tampere, Finland) *(MindTrek '12)*. Association for Computing Machinery, New York, NY, USA, 267–274. https://doi.org/10.1145/2393132.2393191

[7] Jennifer Mankoff, Anind K. Dey, Gary Hsieh, Julie Kientz, Scott Lederer, and Morgan Ames. 2003. Heuristic Evaluation of Ambient Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Ft. Lauderdale, Florida, USA) *(CHI '03)*. Association for Computing Machinery, New York, NY, USA, 169–176. https://doi.org/10.1145/642611.642642

[8] Simon Olberding, Michael Wessely, and Jürgen Steimle. 2014. PrintScreen: Fabricating Highly Customizable Thin-Film Touch-Displays. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 281–290. https://doi.org/10.1145/2642918.2647413

[9] Alex Olwal, Jon Moeller, Greg Priest-Dorman, Thad Starner, and Ben Carroll. 2018. I/O Braid: Scalable Touch-Sensitive Lighted Cords Using Spiraling, Repeating Sensing Textiles and Fiber Optics. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 485–497. https://doi.org/10.1145/3242587.3242638

[10] Linda Oscarsson, Elisabeth Jacobsen Heimdahl, Torbjörn Lundell, and Joel Peterson. 2009. Flat knitting of a light emitting textile with optical fibres. *Autex Research Journal* 9, 2 (2009), 61–65.

[11] Roshan Lalintha Peiris, Owen Noel Newton Fernando, and Adrian David Cheok. 2011. Flexible, non-emissive textile display. In *International Joint Conference on Ambient Intelligence*. Springer, 167–171.

[12] Akira Wakita and Midori Shibutani. 2006. Mosaic Textile: Wearable Ambient Display with Non-Emissive Color-Changing Modules. In *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology* (Hollywood, California, USA) *(ACE '06)*. Association for Computing Machinery, New York, NY, USA, 48–es. https://doi.org/10.1145/1178823.1178880