OUTRUN

BY GARNET HERTZ

BUILDING THE UN-SIMULATION OF A DRIVING VIDEO GAME.

I consumed many hours as a teenager in arcades playing classic video games, and spent a small fortune in quarters mastering a few of them. One of those games was *OutRun*, a driving game released by Sega in 1986, which featured a red Ferrari Testarossa racing down a freeway that snaked through a variety of landscapes.

At the time of its release, this game had a number of interesting features: a steering wheel controller that actually shook when you hit the ditch, user-selectable "chiptune"-ish soundtracks, multiple different endings, and a graphics processor that created an immersive sense of speed — or at least immersive enough to convince a 14-year-old to spend his paper-route money.

Twenty years later, I came across this game at the Beach Boardwalk arcade in Santa Cruz, Calif. It wasn't in the standard upright "fridge format"; this was an 800-pound, car-shaped, sit-down cabinet with fiberglass wheels, working taillights, dashboard, and a powerful hydraulic system that shook the entire cabinet from side to side when you veered into the ditch.

The cabinet was as big as a golf cart, and it got me thinking: "What would it be like if this could actually roll down the street?"

Photography by Four Eyes Photography, Art Center College of Design

WORLDS COLLIDE: (Opposite) Brody Condon drives the *OutRun* vehicle at the Zer01 Biennial in San Jose, Calif., in September 2010. The *OutRun* project explores the overlap between the physical world and game environments by combining a real-world vehicle and *OutRun*, an 8-bit arcade driving game released by Sega in 1986.

Fig. A: Disassembling the original OutRun arcade game cabinet (background) and three-wheeled 1959 Turf Rider Mark IV golf cart (foreground).

In my mind, this would be something like people driving while blindly following their GPS vehicle navigation systems and getting into accidents — or some form of an augmented reality video game concept car. Either way, I liked the idea. It reminded me of the type of dream cars that Ed "Big Daddy" Roth had built: souped-up hearses, hot-rodded kids' Radio Flyer wagons, and flying-saucer cars.

SKETCHING IT OUT

I thought the idea over and decided to go ahead with it in fall 2008. The design process began with mocking up a general sketch in Photoshop. Using some photos I'd taken of the game in Santa Cruz, I put together an orthographic view of a three-wheeled cabinet and placed it on a road scene.

In order for this project to move beyond an extreme case mod, I'd need to display something more than the original arcade game on the screen in front of the driver. Ideally, I envisioned that the screen would transform the real world into an 8-bit video game: in other words, I thought the system should make the entire world as its playing field. I sketched out the idea of making an 8-bit-looking "skin" for a GPS navigation system.

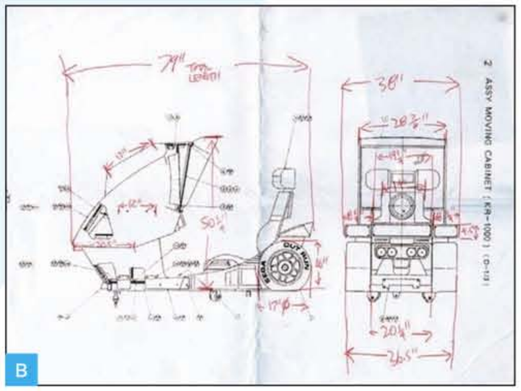
I wasn't sure how the system would work, but I located a nonfunctioning *OutRun* cabinet in Maine through an online discussion forum and had it shipped to me in California.

Armed with my Photoshopped images and a PayPal receipt from a stranger, I pitched the idea on the last day of class before summer break to my first-year undergrad video game development class at UC Irvine. I said that if anybody wanted to help me out with the project over the summer for free, I'd be happy to exploit their energy and talent. To my surprise, five students volunteered: Chris Guevara, David Dinh, Matt Wong, Erik Olson, and Richard Vu. We met and figured out tasks: studying the original game, looking into GPS software, locating a suitable drivetrain.

PUTTING MY CART BEFORE MY CART

The arcade cabinet arrived and only had minor damage. I found and bought an inexpensive three-wheeled Turf Rider golf cart from 1959 on Craigslist that looked like it came from *The Jetsons* and had similar measurements to the cabinet (Figure A).

After disassembling the cabinet and cart, we came to the conclusion that our golf cart wouldn't work for a number of reasons: its single front wheel wouldn't provide enough stability to prevent the car from tipping over, and its treadle steering would be difficult to mechanically couple to the faux-Ferrari arcade steering wheel. In order to maintain a low seat height and the original look of the arcade game, we'd have to reposition all six batteries from their original location under the golf cart seat: three would fit behind the rear axle, while three would need to be put in a top-heavy



B


C

D

Fig. B: Proposed component layout sketch for the system. We later figured out that half of the golf cart batteries would need to be positioned behind the rear axle.

Fig. C: Welding the custom frame, which combined components of the golf cart and arcade frames.

Fig. D: Two-dimensional, 8-bit-style media assets assembled using Google SketchUp to re-create the start sequence of the original *OutRun* game on Park Avenue, Balboa Island, Calif. This scene was tested in real time with a GPS receiver and Google Earth.

Fig. E: (Opposite) Alex Szeto works on the integration of hardware, software, sensors, and physical components.

position above the steering column (Figure B).

The project caught the interest of the Center for Computer Games and Virtual Worlds at UC Irvine. They asked me to work full time on it. They also provided staff support and a modern golf cart drivetrain with rims and tires.

In a grueling push over the summer of 2010, the physical drivetrain was pieced together. Although it ended up looking similar to how it started, it was custom-built from the inside out: frame, steering column, wiring, power management, sound system, sensors, brake lines, and throttle pedal. The original mahogany cabinet and the golf cart were Sawzalled, welded, and drilled into submission in more than 1,000 hours of sweat and bloody knuckles (Figure C).

RETHINKING THE DRIVER INTERFACE

The software development of the system was not straightforward. We thought that a GPS system would be the best solution, but we soon learned that many locations where we wanted to drive didn't have street data, the cart's driving speed was painfully slow when rendered in the style of a video game, and the resolution of most consumer-grade GPS systems was too low for what we needed. We came to these conclusions after building and testing a prototype system that featured 8-bit game sprites and roads positioned into Google Earth (Figure D).

Using a camera-based computer vision system seemed like a better approach. I contacted my friend Jeremy Bailey, who provided a quick mockup in Max/MSP/Jitter to give me an idea of how this could be accomplished.

My student Chris Guevara figured out how to visually detect real-world features through a single camera system. Our first main task was to find the vanishing point of real-world roads, since they were the most important feature from the original video game.

We used real-time edge detection and custom filters to identify what we thought were road-like curb lines. We then calculated the average vanishing point of these lines, and as long as we had a clear and open road with a painted curb and few shadows, the road detection system worked well. The system also worked while looking down hallways, and in unexpected situations, like making a pyramid shape with your arms above your head.

Although the vision system had several inaccuracies, it was considerably more flexible and playful than the location-based GPS system. We built our own version of the original arcade driving game in Flash, and used our calculated computer-vision road endpoint to change the shape of our Flash-based game road (Figure E).

VIRTUAL REALITY ON WHEELS

Making physical things is often a battle with tools and materials, and this project was complicated by trying to integrate computer vision, physical computing, custom software, and other systems — like a golf cart.

E

However, actually making a working real-world system has a significant power of legibility and reality over just mocking up something in Photoshop. Rolling down the street, the arcade cabinet car is understandable and real to a diverse audience — grandmothers and infants, for example — in a way that my Photoshopped image wasn't. And the process of wrestling between the materials and the concept had its own imperfect, *wabi-sabi* beauty to it.

The *OutRun* project was never intended to seamlessly turn a video game into real life and real life into a video game. Instead, it was intentionally built as a type of *chindogu* or critical design — a system that uses paradox, irony, and physical prototyping to raise a series of questions in a provocative way.

In the case of *OutRun*, when the game is extended beyond its normal constraints as a video game, its playfulness malfunctions: it's like a fantasy taken too far, and it results in confusion, nervous humor, and a questionable sense of over-reliance on technology.

To build in this way is to slip into the role of a trickster: using humor and paradox to challenge, bend, and break common assumptions about our everyday lives with technology.

Videos and build notes: conceptlab.com/outrun

Thanks to Walt Scacchi, Chris Guevara, Alex Szeto, Paul Dourish, Gillian Hayes, Jong Weon Lee, Eric Mesple, Yuzo Kanomata, Jeremy Bailey, David Dinh, Matt Shigekawa, Jesse Joseph, Mike Tang, Matt Wong, Erik Olson, Richard Vu, Kari Nies, and Craig Brown for their development efforts and assistance.

Support for this project was provided by the following organizations at the University of California Irvine: the Laboratory for Ubiquitous Computing and Interaction, the Arts Computation Engineering Program, the Center for Computer Games and Virtual Worlds, the Institute for Software Research, and the California Institute for Telecommunications and Information Technology. Support for this research is also provided through grants from the National Science Foundation (#0808783) and the Digital Industry Promotion Agency, Global Research and Development Collaboration Center, Daegu, South Korea. No review, approval, or endorsement implied.

Garnet Hertz is a research scientist and artist-in-residence in the Department of Informatics at the University of California Irvine and is adjunct faculty in the Media Design Program at the Art Center College of Design in Pasadena, Calif. conceptlab.com