Monologue on Navigation

or, why we are solving navigation all wrong and we really need a cane

Illah R. Nourbakhsh

A conversation between Clay Kunz, Thomas Willeke and me during ICRA '97 in Albuquerque, New Mexico led us through an examination of the basic goals and approaches used for achieving trash-can-sized mobile robot navigation. Here is a monologue that presents our conclusions.

What is the right "Grand Challenge" for robot navigation? We want a robot to perform navigation as well as a human navigator. The goal is human-level competence, and we can easily list the requirements:

- Robot functions with zero human supervision
- Robot avoids hitting and falling into objects as well as a human
- Introducing the robot to a new environment requires minimal human effort
- Robot can go anywhere it is physically capable of going
- Robot can go anywhere and know its location as well as a human
- Robot can purposefully go to a goal location as reliably as a human

Few would argue with these goals. In this monologue, we concentrate on the last two goals, which proclaim passive and active *navigation competence*. We want human-level performance. But this is even harder than you can imagine.

Consider the human navigation system. You have a visual recognition system that works very well. Every instant that you see your surroundings, your visual system lets you know, with some bounded uncertainty, where exactly you are in the local scene. You have a Local Positioning System, and it has a name: *vision*. Of course, Global Positioning can be much harder for humans, since Vision stops helping when you move beyond the visual horizon.

Now consider the robot navigator. Human level navigation competence means virtually instantaneous local positioning, and this has proven to be a challenge none too trivial for roboticists. Probabilistic progression, reasoning with uncertainty, POMDP's—all of these computer-intensive methods are having a tough time for two reasons: (1) the robot's sensors are so poor that the local scene wanes as soon as the robot moves, meaning that it is essentially always solving a "semi-global positioning" problem; (2) the robot has a vision system, but doing localization using vision as well as a human is akin to solving the Recognition Problem.

The take-home is that we are trying to solve the Recognition Problem whether we mean to or not. We are not even attempting this head-on, using vision techniques. In many cases, we are attempting to imbue the robot with Recognition-level local scene navigation using computationally hard AI-level techniques—all to solve a problem that is trivial (at least on this side of evolution) for us humans.

Recall the range of robot navigator failures that you see at your average robot contest—the navigation failures are errors that look miserable to human onlookers, because the onlookers can *see* very silly local mistakes being made. Robots mistake a wall for a doorway; robots become disoriented in a room; etc. This is not how humans err. Humans become globally lost (but with a perfectly good idea of where locally they are in the hallway or highway). The good news is, robots won't become globally lost because they can count and follow maps exceedingly well. The bad news is that robots are quite poor at a skill that humans find trivial.

And so we come to the first theme of this monologue—we're going about solving navigation the wrong way. We call it "the wrong way" because we are pouring computation into a problem that is visual in

humans. There are two possible alternatives—solve the problem visually on robots, or choose another human model.

We opt for the latter alternative: *choose another human model*. Turn to the blind human navigator as your standard-setter, and now you can sidestep vision altogether. A blind navigator does not do local scene positioning using vision, so we can ignore that challenge and solve a somewhat different navigation problem. Does this limit the capabilities of the resulting robot? Not very much, as blind navigators can navigate a very large part of the structured human world.

Does this make the problem easier? Not really, because blind navigators depend to a large extent on auditory signals, both for stereo localization and for detection of navigation "events" (such as walking past the Coke machine). Once again, auditory signal processing turns out to be difficult for computers (Audio Recognition is just as difficult as Visual Recognition) and easy for humans. Strike two.

Suppose we turn next to blind-deaf navigators. This is a class of navigators who use a cane and do not take advantage of other sensor inputs. Now *this* is interesting because we're down to one sensor type, and that one sensor's interactions with the world are highly limited. Does this limit our resulting robot? Yes, to some degree. But the range of environments a deaf-blind navigator can handle is still acceptably large.

Now let us examine what is easy and what is hard for human blind-deaf navigators. Detecting a curb is easy—the cane works well. Navigating is hard because one must use much counting, timing and reasoning. Aha! These are all skills that computers are well-endowed to handle! Rather than solving humanity's easy problems on a computer, we've limited ourselves to human navigation strategies that require skills that are hard for humans and easy for computers.

Figure 1: A full-screen picture of the sidewalk robot's path: too many pixels

But back to the Recognition Problem: what about detecting physical objects using the cane? This is a problem still. Recognizing curbs and ledges is hard, but it is far easier than recognizing an office hallway, a building, or a Coke machine's sound signature. In fact, depth from stereo has progressed far enough today that we can quite easily recover depth at the ground level sufficiently well to detect curbs, walls, ledges and shoes.

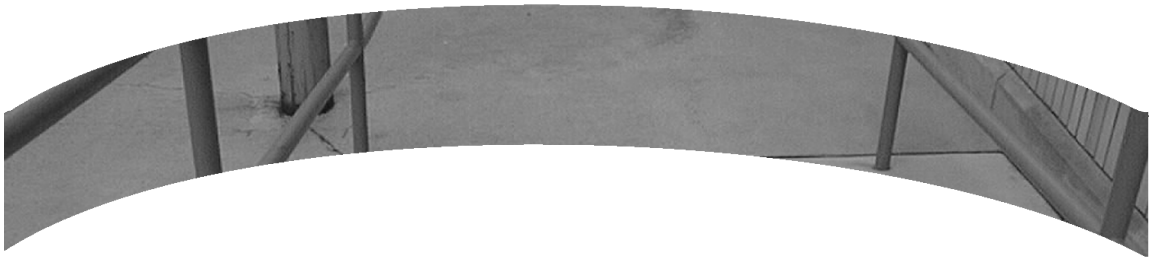

Figure 2: The cane-scan subset of Figure 1—now*this* is tractable

Of course, the catch with depth-from-stereo is speed. But here, we can bench on the fact that a deaf-blind navigator is using a cane, which provides (1) a very slow update rate, and (2) only depth information for a limited portion of the scene.

Figure 1 is a picture of a sidewalk. There is much information in this picture, and depth recovery would take some computation. Figure 2 is the same picture, with attention given only to the areas visited by a scanning cane. Far less information resides here—in fact, so little information that we can run your favorite depth recovery algorithm at, say, 3 Hertz on a cheap processor. And 3 Hz is already much faster than the cane sweep rate of a blind-deaf navigator.

So, we have our conclusion. Start with a vision sensor to recover the cane-sweep ground profile. Solve the deaf-blind human navigation problem. The goal: a robot that navigates as well as a deaf-blind navigator.