Aaron Zinman

Ambient Intelligence Final Project Proposal
What is it & why is it interesting?

I am proposing to create a "kitchen director". The director would take in the current context of the kitchen, map that to a set of recipes that it coordinates, and update in real-time pacing for the meal potentially doing error-correction. Its main function is to take the complicated task of managing multiple recipes that are separate into a single manageable recipe that is context-aware. It also supports the user by allowing them to access cooking resources such as technique video clips when they are unfamiliar with a necessary step or what the results might look like. Thus the device provides a cognitive support in terms of attention, memory, planning/reasoning, and knowledge.

It can be linked into other functions such as notifying guests when the meal might be later than a specified threshold, or to request guests pick up items that might have gone afoul or missing. Since the system would monitor progress through a recipe and its subsequent sub-processes, a user profile would be built and maintained showing cooking abilities. If the system were to posses a large and rich (breadth and depth) database of recipes, it could identify the complexity preferences and abilities of the user and suggest recipes or possible substitutions accordingly. Finally, if connected into a larger context-aware home, by detecting the presence of guests, their location, and the cook's habits, the laggerdliness of the meal preparation could be dynamically adjusted to allow for more social time during the act of cooking.

Usage Scenario

It's Thursday night and Joe has decided to through a dinner party for himself and his friends on Friday. Joe, a typical graduate student, is used to making meals that are quick and simple for just himself. However, this party is to be special, and Joe is planning on going all out. He wants to prepare five different dishes with his limited resources and for them all to come out at the same time. He also wants to be able to talk to his friends while he cooks. Joe launches the kitchen director and selects some individual courses. He filters them according to complexity and desired preparation time (he still has research to do during the day). The kitchen director informs him that for his meal, he should ideally start now by marinating his chicken tonight. This works well for spreading out the workload when serial time is precious.

After coming home from a long and successful day, he beings cooking. The kitchen director starts him off by preparing the ingredients into the first timestep, "mis en place". As he uses resources, the kitchen director is aware of the placement of his vegetables and the remaining instruments. It might suggest which tools he can wash later and which he should wash now for reuse. Even though he was given an initial estimate, Joe is taking longer today and accordingly the estimate continues to move up. After a threshold, the kitchen director suggests the use of lemon to prevent the potatoes from oxidizing and consequently turning brown, and offers to let his friends know that dinner might be a bit late. Since Joe doesn't really care, he declines both of these. Later when his friends show, the director adjusts the timing as Joe is paying more attention to his friends than his vegetables. Furthermore, since Joe is paying to much attention to his friends, the kitchen director suggest that he might turn down the temperature on the stove to cook his meal more slowly in order to finish with everything else. Finally, it suggests times when he is free to converse and how long before attention is needed.

How will it be implemented?

Sensors distributed throughout the kitchen and home provide constant feedback of the state into the system. Unique user and environmental profiling combine with a generalized knowledgebase to transform mappings of recipes, environmental state, and user preferences into the appropriately needed functions and adjustments to the timeline. Essentially, the situational data maps onto the dimensions of time, space (where in the kitchen), resources (which pans), complication (of the recipe, i.e. number of steps), laggerdliness (how much time you want between steps), and health (how much fat). Complication would be calculated by the linear equation of c1 * tdoing + c2 * num_steps, where c1 and c2 are constants and tdoing is amount of time actually preparing or observing versus passive waiting. The actual planning might use existing algorithms and programs such as the University of Washington's Sensory Graphplan to do partial order planning. There are many available academic systems as well as commercial (and particularly military) applications. Depending on the capabilities of the existing planning software, it might be necessary to do a custom algorithm, most likely implemented using a graph-based data structure and propagation.

The actual interface might involve several types of input and output. Input sensors monitor the environmental state, as well as user queries and direct manipulation. Since the kitchen involves so much with hands (including dirtiness), speech recognition might prove to be one of the more useful methods. This could be augmented with systems that detect gaze or facial direction. Likewise, the output should definitely involve audio, but might also include visual information (like CounterActive?), simple lights highlighting the featured item, or ambient displays in other rooms to give feedback on the status and its potential urgency. As a user study done by the Counter Intelligence group pointed out, care should be taken in the interface to provide high-level goals to the users rather than overload with small steps which only isolate and confuse.

What parts will you complete for this class

For simplicity sake, I will use a laptop as the heart and brains of the system. This might also be quite desirable in real products as it allows for a smaller investment and easy testing. Using Apple's speech-to-text and speech recognition software, the interface will use voice as the primary means of interaction. The software will be written such that it can take in optional sensor readings with specific mappings, but will only be worked on after the truly ubiquitous setup of voice-only is working first. This provides the classical Wizard-of-Oz ability to test the functionality of the system without requiring an expensive and more potential error. The recipes will be hand crafted for the system as I have no intentions of using natural language processing and common sense reasoning to fill in standardized recipes to suit my a priori unknown needs. The system should have user profiling including the ability to adjust its settings over time. Hopefully it will also allow the user to alter the profile by desired laggerdliness, and ideally would detect this on its own. In order to do anything useful with off-timing detection, the system should be able to do basic error correction given a knowledgebase (last resort: call dominos). One of the most difficult parts to this is going to be the interface design. I believe that it is only possible to do this through iterative design and testing. Since I love to cook, and my basic requirements do not need a prototypical kitchen, this shouldn't be a problem :)

What do you hope to learn?

I hope to gain an understanding of what is necessary for a program to be able to switch between the periphery and central as necessary. Besides the programming exercise, this is more an exercise in design. I've never had to design anything that required supporting a user doing a task that wasn't directly manipulated on the screen, particularly in this "agent" or "director" style. It should give me insights into what is necessary for intelligence to be put into our environment and how it might interact on a more appropriate level.