

Designing Systems that Direct Human Action

Ana Ramírez Chang^{1,2}, Marc Davis²

¹Group for User Interface Research
Computer Science Division
University of California, Berkeley
anar@cs.berkeley.edu

²Garage Cinema Research
School of Information Management and Systems
University of California, Berkeley
marc@sims.berkeley.edu

ABSTRACT

In this paper we present a user-centered design process for Active Capture systems. These systems bring together techniques from human-human direction practice, multimedia signal processing, and human-computer interaction to form computational systems that automatically analyze and direct human action. The interdependence between the design of multimedia signal parsers and the user interaction script presents a unique challenge in the design process. We have developed an iterative user-centered design process for Active Capture systems that incorporates bodystorming, wizard-of-oz user studies, iterative parser design, and traditional user studies, based on our experience designing a portrait camera system that works with the user to record her name and take her picture. Based on our experiences, we lay out a set of recommendations for future tools to support such a design process.

Author Keywords

Direction, recognition, error, mediation, active capture, error-prone systems, multimedia systems design.

ACM Classification Keywords

H.5.1 [Multimedia Information Systems]: Audio input/output, Video; D.1.7 [Programming Techniques]: Visual Programming

INTRODUCTION

Traditional human-computer interfaces bring interaction and processing together. Active Capture [1] uses multimedia capture in addition to interaction and processing to interactively direct, analyze, and capture human action (See Figure 1). In this paper we present a user-centered design process for Active Capture systems. We incorporate body storming, wizard-of-oz user studies, iterative parser design and traditional user studies in an iterative user-centered design process.

The design process includes designing the design of multimedia signal parsers together with a user interaction

script to develop “human-in the-loop” algorithms that direct and describe human action. We base the design of the interaction script on our previous theoretical work on strategies and a corresponding design space for systems that direct human actions [4].

In Active Capture applications, the user works with the system to reach a common goal; from creating a personalized movie trailer or commercial starring the user, to improving the user’s golf swing. In this paper we use the SIMS Faces system (name changed for blind review) as an example. The SIMS Faces application records the student saying her name and takes her picture for inclusion on the SIMS Faces page; a web page with students’ pictures and name pronunciations used by the SIMS community. The system positions the student in front of the camera and asks her to say her name, ensuring the response is a feasible length for a name. Next the system asks the student to look at the camera and smile, ensuring the student is well framed and smiling. See Table 1 for a detailed description of the system interaction script and parsers.

The design process is derived from our experience designing the SIMS Faces system. Based on our experiences, we lay out a set of recommendations for future tools to support such a design process.

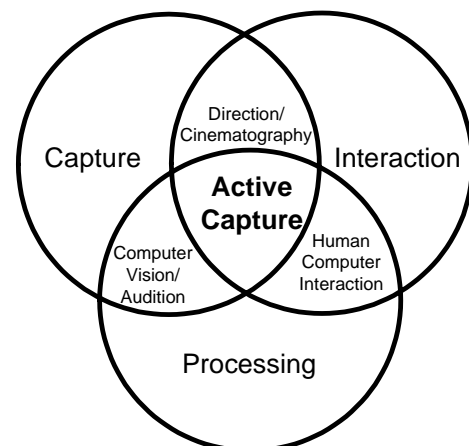


Figure 1. Active Capture.

| | | |
|---|--|---|
| Welcome | “Welcome to SIMS.” | |
| Position | “Please stand on the white marks on the floor and look at the camera” | |
| | Not Framed | “Hmm, I can't see all of you, please be sure you are standing on the white marks on the floor and looking at the camera.” |
| | No Eyes | “I can't see your eyes, please be sure you are facing the camera so your name will be recorded clearly.” |
| “That's great! Next we are going to record your name so people will know how to pronounce it.” | | |
| Name | “Please look at the camera and state your full name.” | |
| | Too Short | “Wow, that's pretty short for a name, just in case let's rerecord your name. Please be sure to clearly state your first and last name. Go ahead.” |
| | Too Long | “I heard you say something, but it sounded too long for a name, let's try again. Please say your full name, that is your first and last name, now.” |
| | “Thanks for saying your name, now we are going to take your picture.” | |
| Picture | “Please stand on the white marks on the floor and look at the camera. Smile.” | |
| | Moving | “Please stand still while I take your picture. Ok, smile.” |
| | No Eyes | “I can't see your eyes, perhaps you are wearing glasses or a hat, please remove them and look at the camera. Now smile” |
| | No Smile | “Your picture will be nicer if you smile, please look at the camera and smile.” |
| | “That was really great” | |
| Thanks | “Thanks for using the SIMS Faces System” | |

Table 1. SIMS Faces system interaction script.

COMPONENTS OF ACTIVE CAPTURE APPLICATIONS

Active Capture applications are made of two interdependent components – the interaction script and the action recognizers. The interaction script together with input from the simple parsers allows the computer and user to work together to achieve the desired action. In the SIMS Faces system, the computer and user work together to record the user saying her name and take her smiling picture. The simple parsers used include a motion detector, eye finder, sound detector, mouth motion detector.

Interaction Script

The interaction script describes how to work with the user to achieve the desired action. It also describes what to do when something goes wrong. For example, when the user is getting her picture taken, she may be partially out of frame. The application asks her to move so she is in the frame: “I don't entirely see you, perhaps sitting down or standing on a stool might help. Now smile.”

Action Recognizers

The multimedia parsers give the application an idea of what the user is doing. In the context of the interaction, the input from the parsers can be interpreted much differently than if used alone. For example, the probability motion in the user's mouth after she is asked to smile is a smile is very high. A recognizer for the desired action, the user smiling, can be defined as mouth motion after the smile trigger.

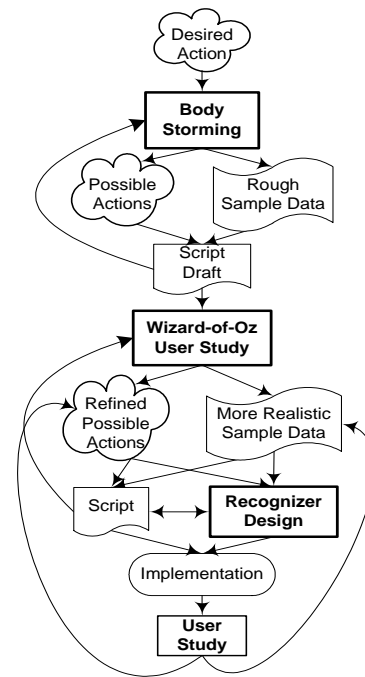


Figure 2. Active Capture design process.

DESIGN OF ACTIVE CAPTURE APPLICATIONS

The design of Active Capture applications follows traditional user-centered design with the use of bodystorming [5], wizard-of-oz user studies, traditional user studies, and an iterative design cycle (See Figure 2). The design of the interaction script and the action recognizers are interleaved in the design process as they are interdependent in the application.

Bodystorming

The design process begins with the desired action in mind and a bodystorming session to inform the first draft of the interaction script. Body storming is similar to brain storming, but the participants brainstorm with their bodies. With the desired scenario in mind, the design team acts out different variations of the interaction script and various reactions to the script (what could go wrong in the interaction). In addition, the sample interactions are recorded, providing sample data for use in the design of the parsers and interaction script. In the case of the SIMS Faces system, the bodystorming session raised and attempted to answer the following questions: Suppose we want to take the user's picture, how will we get her to stand in front of the camera? What if she is moving too much to take her picture? What if she is not framed properly? What if her eyes are closed? What if she doesn't smile? The interaction script should address all of these questions and more. The design of the interaction script is based on the strategies and associated design space for these systems presented in [4].

Wizard-of-Oz User Study

With a draft of the interaction script and digital clips for each command, instruction, or trigger, the wizard-of-oz study comes next. In the wizard-of-oz study, the computer plays

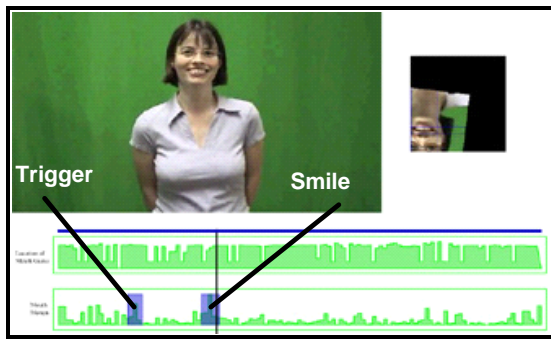


Figure 3. Recognizer design support tool prototype.

the clips and records the data, but the human decides when to play each clip. Since humans react differently to computers than they do to other humans, the wizard-of-oz study is important. It simulates the human-computer interaction bodystorming cannot simulate because in bodystorming the interaction is between humans.

In addition to testing the flow of the interaction, the wizard-of-oz study tests and reveals the triggers in the interaction script (the words or phrases that make the user react). The interaction script is designed with triggers, some may work well, others may not result in the desired reaction and there may be still others that weren't intended as triggers. For example, in the SIMS Faces application, the system offers to tell the user a joke to get her to smile after a few failed attempts to smile. "Let me tell you a joke. A guy walked into a bar, ow!" We expected the "ow" to be a trigger for a smile, but it turns out "Let me tell you a joke" also turned out to trigger a smile.

The data collected in the wizard-of-oz study provides realistic examples with close to realistic timing details of the interaction and resulting actions. This data is crucial for the design of the recognizer. The refined set of possible actions and realistic sample data allow the designers to iterate on the script and design the recognizer for the desired action. With these components in place, the application is ready to be implemented and evaluated with a traditional user study.

Designing Action Recognizers

An action recognizer defines the desired human response in terms of the multimedia parsers in the context of the interaction script. The sample data from the wizard-of-oz study contains useful examples of the action in terms of the multimedia parsers and their relation to the triggers. The designer then looks for reliable patterns in the data to form a new action recognizer.

Sifting through all the sample data from the wizard-of-oz study to find a pattern in the multimedia parsers with respect to the triggers can be a labor intensive task if not supported with good visualization of the data and easy navigation through the data. We have designed an interface that shows all of the data streams together and allows the user to annotate where the triggers are in each sample data as well as which parts of the sample data are important for the desired action (See Figure 3). The user can play back the segment of

video she has annotated in the corresponding data. As the designer finds a pattern in the data, she should be able to generalize it on a timeline with a track for each stream of data she is interested in. As she modifies the pattern on the timeline, the system should check to see which of her examples follow the pattern and which do not. The tool will allow the user to keep track of all of her example data and present it to her in a variety of different configurations to aid in her pattern discovery.

RELATED WORK

Our recommendations for tool-level support for action recognizer design draws on a variety of previous systems.

a CAPpella [3] supports the design of recognizers for use in context-aware systems. The system allows the user to collect and annotate data of the situation she wants the system to recognize. The user is presented with the data in streams, one stream for each type of data. The user then selects or annotates which sections of the different data streams are important. The system uses machine learning techniques to build a recognizer based on the data collected. *a CAPpella* does not require the user to find the pattern in the data, but it does require many examples (90 in their study). While machine learning may be very useful in extracting a pattern from the data, an Active Capture interaction designer may want a pattern that is more flexible and interpretable, or allows a larger space of variations on the desired action than demonstrated in the wizard-of-oz data. In addition, to support the design of action recognizers with machine learning, the algorithm may need much more data than can be easily acquired in the wizard-of-oz study.

MediaCalc [2] allows the user to experiment with different parsers as well as composition of parsers on rich input streams such as video or audio. While its interface allows both visualization of the data flow of the inputs and outputs of media analysis and synthesis algorithms linked to a timeline visualizing each step of the processing results, the challenge in ACAL [6] is to also visualize and interactively edit the control flow of the computer-human interaction.

The SIMS Faces application takes the interface of the FX Palo Alto Laboratory Video Guestbook [7] to the next level. The Video Guestbook takes the guest's picture, records her saying her name and scans her business card. The Video Guestbook does not focus on the interaction with the guest or ensure the quality of the data.

EVALUATION

As part of the design process of the SIMS Faces system, we ran a wizard-of-oz user study with 10 participants and a traditional user study with 8 participants.

In our wizard-of-oz study, participants were lead into a room divided by a green curtain. The mock application was set up on one side and the "wizard" on the other. The participant was lead to believe the room was divided so the study would not disturb other people working in the room.

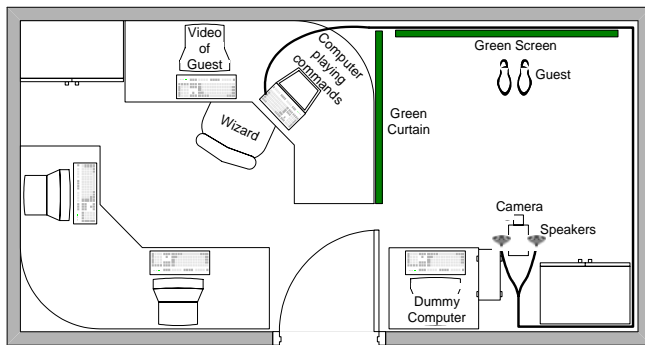


Figure 4. Lab setup for Wizard-of-Oz user study.

The “wizard” monitored the participant’s actions via a wireless camera and selected the clips to play on a computer behind the curtain. The computer played the clips through speakers situated next to the computer believed to be running the SIMS Faces system. (See Figure 4).

Each of the 8 participants in our traditional user study has pictures posted on a department webpage taken by a human. In the study with our implemented SIMS Faces system, we asked the 8 students to compare their picture taken by the SIMS Faces system with their picture on the department webpage. Both pictures were the same resolution and cropped similarly, although unlike the photos on the departmental web page, the SIMS Faces photos were cropped automatically by the system. Seven of the students preferred the picture taken by the SIMS Faces system and one student said both pictures were about the same. In addition to a picture, each student successfully recorded her name and selected to keep her recorded name after hearing it. These results demonstrate the SIMS Faces system successfully records the participant saying her name and takes a portrait photo she is happy with. These are the desired actions we set out to elicit and record with the design and implementation of the SIMS Faces system.

While a working Active Capture system is not an evaluation of the proposed design process, it does serve as a proof of concept. The proposed design process has led to a working Active Capture system and encapsulates the lessons learned from the process of designing the SIMS Faces System.

CONCLUSION

The interdependence between the interaction script and the action recognizers present unique challenges in designing Active Capture applications. We presented a user-centered iterative design process that leverages the benefits of bodystorming, wizard-of-oz user studies and traditional user studies. Each step in the process provides important data and examples for the next step of the interaction script and action recognizers. As the interaction script unfolds, the action recognizers start to take form. As we describe the action in terms of the multimedia parsers, we understand how to work with the user to reach our desired action.

The bodystorming session and the strategies and design space in lay the foundation for the interaction script. The wizard-of-oz study refines the interaction script and provides valuable

sample data for the design of the action recognizer. In addition, the wizard-of-oz study is important in the design process for two main reasons. 1) Humans act differently around other humans than they do with computers. The wizard-of-oz study provides realistic data for how the user will react to the script when it is coming from the computer. 2) It is important to test the triggers in the commands. Some triggers may not work as expected and others may appear in unexpected places. This iterative design process is based on the design of a working example Active Capture application, the SIMS Faces system.

FUTURE WORK

In order to explore the space of Active Capture systems more easily, we plan to further develop tool level support for the design and implementation of these systems. In addition to tool level support, we plan to integrate our previous theoretical work on strategies and a corresponding design space for systems that direct human actions as design patterns.

ACKNOWLEDGMENTS

We thank Pauline Jojo Chang, Leo Choi, William Tran and Madhu Prabaker for their work on the SIMS Faces system and tool prototypes. We also thank the participants of both studies. The first author is supported by an NSF fellowship.

REFERENCES

1. Davis, M. Active capture: Integrating human-computer interaction and computer vision/audition to automate media capture. In *Proc. ICME 2003*. IEEE Computer Society Press (2003).
2. Davis, M., et al. Time-Based Media Processing System. US Patent 5,969,716. Filed: August 6, 1996. Issued: October 19, 1999.
3. Dey, A.K., Hamid, R., Beckmann, C., Li, I., Hsu, D., a CAPpella: Programming by Demonstration of Context-Aware Applications. In *Proc. CHI 2004*, ACM Press (2004).
4. Heer, J., Good, N., Ramirez, A., Mankoff, J., and Davis, M., “Presiding Over Accidents: System Mediation of Human Action,” In *Proc. CHI 2004*, ACM Press (2004).
5. Oulasvirta, A., Kurvinen, E. and Kankainen, T., Understanding contexts by being there: case studies in bodystorming. In *Personal and Ubiquitous Computing*. Volume 7, Issue 2 (July 2003).
6. Ramirez, A. and Davis, M. Active Capture and Folk Computing. In *Proc. ICME 2004*, IEEE Computer Society Press (2004).
7. Trevor, J., Hilbert, D., Billsus, D., Vaughan, J., and Tran, Q., Contextual Contact Retrieval. In *Proc. International Conference on Intelligent User Interfaces (IUI 2004)*.

