Touchless Interfaces for PDF Reading Software

Eric Kidder
ekidder3@gatech.edu

Dingtian Zhang
alandtzhang@gmail.com

Gaurav Kanitkar
gaurav.kanitkar@gmail.com

INTRODUCTION

There are situations where it is difficult to interact with a touch-based device for any number of reasons. During these situations, additional steps must be taken to perform tasks because the device provides no alternative form of input, making the tasks more complex and time consuming. We propose that for certain tasks, the addition of a touchless interface will aid the user in being more effective.

Touch-based interfaces, such as keyboards, mice, and touchscreens, are the most common way of interacting with computing devices. They allow for agility, precision, and tactile feedback. However, there are situations which may make it difficult or impossible to make use of these interfaces, such as dirty hands, having both hands occupied, or distance between the user and device. One example of this is cooking, with a recipe displayed on a laptop or tablet. If the user has been handling food, his or her hands may be dirty and unable to interact with the device. In these situations, the user must take extra steps – such as washing his or her hands or moving closer – before interacting with the device. Steps like this could be avoided if the device also provided alternate, touchless interfaces that provided some or all of the functionality of the touch-based interface.

This project will investigate alternative, touchless interfaces for page-oriented software. Touchless interaction can be multimodal, including the visual modality (color, form, or position change), auditory modality (speech, sounds) or olfactory modality (odors) [1]. In our project, we will be concentrating on form and position changes of the users hands as relevant events in touchless interaction. The goal of our project is to design and implement a gesture-based touchless interface for a PDF reader software. We will then investigate how this interface handles user input, provides feedback, and its speed and efficiency. User input will be implemented using a combination of traditional and gesture-driven interfaces. The traditional interface will act as both a bridge – connecting the touch-based interface of the operating system and the touchless interface of the reader – and as a fallback for cases when the gesture input has difficulty or the user prefers a more traditional interface. The interface will provide appropriate guidance about the touchless interface as well as appropriate feedback in response to user events.

RELATED WORK

Touchless gestural input has received considerable attention in the past few years because of such benefits as removing the burden of physical contact with an interactive system and making the interaction pleasurable [1]. There are two types of hand gestural inputs: One is static hand postures, which the users keep their hand steady with poses; The other is hand movement, which users wave their hand in certain direction or trajectory [1], [2]. In real life, such as American Sign Language, these two types are combined to form a gesture [2]. However, they are mostly researched separately.

Generally there are two types of inputs. One is using data gloves by employing sensors (mechanical or optical) attached to a glove that transduces finger flexions into electrical signals for determining the hand posture. Another is vision based by using computer vision techniques combined with visual capturing devices, such as traditional monocular camera, multi-camera, depth sensor, etc [3]. Considering our major application is using PDF reader while cooking, we will focus on vision-based inputs, specifically monocular cameras which are most accessible for ordinary users.

Regarding vision-based hand gesture recognition, there are concerns and challenges, particularly for achieving the robustness necessary for user-interface acceptability: robustness for camera sensor and lens characteristics, scene and background details, lighting conditions, and user differences [4]. Many methods are proposed to meet these challenges by utilizing these information:

- Motion. Frame-to-frame comparison against a learned background model is an effective and computationally efficient method which requires several assumptions (such as a stationary camera or image pre-processing to stabilize the video) and a static background [5], [6].
- Color. Skin color occupies a rather well-defined area in color spaces so can be used for segmentation [7], [8], [9]. Combined histogram-matching and blob-tracking with Camshift [10] or the Viterbi algorithm [11] is a popular approach due to its speed, ease of implementation, and performance. Shortcomings stem from confusion with similar-colored objects in the background and limitations with respect to posture recognition.
- Shape. Shape is available if the object is clearly segmented from the background scenery, often for stationary-camera systems (using a background model) and a bit less reliably with a good hand-color model [13]. Popular methods include statistical moments [14], rule-based methods [15], active shape models [16], and shape context [17].
- Appearance. Methods that consider the intensity and/or color values across a region of interest are more powerful and robust than methods that consider shape alone. The

drawback is increased computational cost during training and recognition. Examples of appearance-based methods [18] have been employed for various vision-based interfaces, including those reported by Hasanuzzaman [9].

The extracted features are then subjected to various classifiers, from generic support vector machines to highly customized shape classifiers, as in Yin and Zhu [13]. Classification is sometimes externally combined with feature extraction, as in the boosting approach involving a combination of weak detectors. Other methods involve a distinct translation step into feature space and subsequent classification; for example, consider the motion track of a hand gesture, with its spatial location over time serving as feature vector and a hidden Markov model classifying hand trajectory into various temporal/ dynamic gestures [19].

PROJECT PLAN

To accomplish these goals, we have a number of tasks:

- Research existing gesture-based input designs
- Research libraries for accessing interface hardware using Java
- Research libraries for reading and displaying PDF files using Java
- Investigate the different algorithms and designs that are presented and how to use them in the application's gesture recognition system
- Implement and develop hand gesture recognition algorithms for touchless control
- Develop PDF reading software which can work well with touchless control inputs
- Design and implement a system for integrating the touchless interfaces with the PDF reading software
- Test how well the touchless interfaces respond to user actions and compare this with equivalent touch-based actions

DESIGN

The application will be split into three primary modules: the display module, the traditional interface, and the touchless interface:

- The display module will provide the primary output interface, including a bookshelf display which displays all the PDFs in a bookshelf manner, and a PDF display working similar to a PDF reader.
- The traditional interface and the touchless interface will serve as input handling modules processing commands within the two displays. The traditional interface will handle touch-based events, such as opening file.
- The touchless interface will work with the device's camera and provide the gesture controls over events. "Next Page", "Bookmark This Page", "Open Table of Contents", "Close Current Document", etc., are events within the PDF display; "Next Document", "Open Document", etc., are events within the bookshelf display.
- An event system will be used to tie these components together. This will allow the traditional and touchless interfaces to share code by invoking events instead of placing similar code in different modules. The display module will listen for events and update its display accordingly.

The user will be made aware of the touchless interface in two ways. First, a message that the application supports gestures will be displayed when the user starts the application. Second, an appropriate icon for gesture inputs (for example, an open hand) will be displayed and different sound effects will serve as notifications for different gesture inputs. When the user clicks or selects this icon using the traditional interface, it will give the user instructions on how the gesture system works. When the system detects that the user wishes to perform a gesture command, the icon will update itself to indicate the application is in "gesture command" mode. As gestures are performed, the icon will change to give the user feedback on what gesture is being detected and if there are any errors.

The gesture inputs, depending on a camera attached to the device, will be processed by local gesture recognition engines. We plan to include both hand motion and static hand posture in our gesture commands. For instance, waving the hand from left to right will be treated as the "Next Page" and "Next Document" command; holding a thumbs-up will be treated as "Bookmark This Page" command. The gesture recognition algorithms will be chosen mainly based on the performances. A hybrid approach combining motion, skin color and hand shape model is expected to be implemented along with feature extraction and classification techniques. User scenarios will be considered in algorithm development such as the lighting in the kitchen and that the users hand may get dirty when cooking, etc. The target environment is a PC (Windows, Mac, or Linux operating system) with a camera and microphone attached. The application will be written in Java and may make use of native libraries – such as OpenCV – to provide access to the hardware devices.

TESTING

Because we anticipate the touchless interface will take substantially more time, testing will begin after the display and traditional interfaces have been completed. This will allow us to test the event system and our event coverage, as well as determine which events are appropriate for touchless controls. As work on the touchless interface comes to a close, we will integrate it into the test suite. After project completion, we will begin usability testing. In this phase, we will use the traditional interface to provide baseline scores and then use these scores to measure the efficiency and easy-of-use of the touchless interface. We will also keep measure of the touchless system's errors in object and gesture tracking and different techniques to improve these scores.

REFERENCES

1. de la Barr, R., Chojecki, P., Leiner, U., Mhlbach, L., and Ruschin, D. (2009). Touchless interaction-novel chances and challenges. In *Human-Computer Interaction. Novel Interaction Methods and Techniques* (pp. 161-169). Springer Berlin Heidelberg.

2. Kulkarni, A. H., and Urabinahatti, S. A. Performance Comparison of Three Different Classifiers for Hci Using Hand Gestures.

3. Murthy, G. R. S., and Jadon, R. S. (2009). A review of vision based hand gestures recognition. *International Journal of Information Technology and Knowledge Management*, 2(2), 405-410.

4. Wachs, J. P., Klsch, M., Stern, H., and Edan, Y. (2011). Vision-based hand-gesture applications. *Communications of the ACM*, 54(2), 60-71.

5. Kang, H., Lee, C., and Jung, K. Recognition-based gesture spotting in video games. *Pattern Recognition Letters 25*, 15 (Nov. 2004), 17011714.

6. Lucas, B.D. and Kanade, T. An iterative imageregistration technique with an application to stereo vision. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence.* Morgan Kaufmann Publishers, Inc., San Francisco, 1981, 674679.

7. Klsch, M., Turk, M., and Hllerer, T. Vision-based interfaces for mobility. In *Proceedings of the International Conference on Mobile and Ubiquitous Systems* (Boston, Aug. 2226, 2004), 8694.

8. Rauschert, I., Agrawal, P., Sharma, R., Fuhrmann, S., Brewer, I., and MacEachren, A.M. Designing a human-centered, multimodal GIS interface to support emergency management. In *Proceedings of the 10th ACM International Symposium on Advances in Geographic Information Systems* (McLean, VA, Nov. 89). ACM Press, New York, 2002, 119124.

9. Hasanuzzaman, M., Ampornaramveth, V., Zhang, T., Bhuiyan, M.A., Shirai, Y., and Ueno, H. Real-time vision-based gesture recognition for human-robot interaction. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics* (Shenyang, China, Aug. 2226, 2004), 413418.

10. Bradski, G.R. Computer-vision face tracking for use in a perceptual user interface. *Intel Technology Journal*(Q2 1998).

11. Yuan, Q., Sclaroff, S., and Athitsos, V. Automatic 2D Hand tracking in video sequences. In *Proceedings of the IEEE Workshop on Applications of Computer Vision* (Breckenridge, CO, Jan. 57). IEEE Computer Society Press, 2005, 250256.

12. Starner, T., Auxier, J., Ashbrook, D., and Gandy, M. The gesture pendant: A self-illuminating, wearable, infrared computer-vision system for home-automation control and medical monitoring. In *Proceedings of the Fourth International Symposium on Wearable Computers* (Atlanta, Oct. 2000), 8794.

13. Yin, X. and Zhu, X. Hand-posture recognition in gesture-based human-robot interaction. In *Proceedings of the IEEE Conference on Industrial Electronics and Applications* (Singapore, May 2426, 2006), 16.

14. Freeman, W. and Roth, M. Orientation histograms for hand-gesture recognition. In *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition* (Zurich, June 1995).

15. Kawarazaki, N. Hoya, I., Nishihara, K., and Yoshidome, T. Cooperative welfare robot system using handgesture instructions. *Lecture Notes in Control and Information Sciences 306*, Springer, Berlin, 2004,

16. Cootes, T.F. and Taylor, C.J. Active shape models: 'smart snakes.' In *Proceedings of the British Machine-Vision Conference* (Leeds, Sept. 2224). Springer, Berlin, 1992, 266275.

17. Belongie, S., Malik, J., and Puzicha, J. Shape matching and object recognition using shape contexts.Transactions on Pattern Analysis and Machine Intelligence 24, 24 (Apr. 2002), 509522.

18. Viola, P. and Jones, M. Robust real-time object detection. International Journal of Computer Vision 57, 2 (May 2004), 137154.

19. Klsch, M., Turk, M., and Hllerer, T. Vision-based interfaces for mobility. In *Proceedings of the International Conference on Mobile and Ubiquitous Systems* (Boston, Aug. 2226, 2004), 8694.

20. Chen, Y.T. and Tseng, K.T. Developing a multiple-angle hand-gesture-recognition system for human-machine interactions. In *Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society* (Taipei, Nov. 58, 2007), 489492.

21. Bradski, G.R. Computer-vision face tracking for use in a perceptual user interface. *Intel Technology Journal*(Q2 1998).

22. Spruyt, V., Ledda, A., and Geerts, S. (2010, September). Real-time multi-colourspace hand segmentation. In *Image processing (ICIP), 2010 17th IEEE international conference on* (pp. 3117-3120). IEEE.

23. Fang, Y., Wang, K., Cheng, J., and Lu, H. (2007, July). A real-time hand gesture recognition method. In *Multimedia and Expo, 2007 IEEE International Conference on* (pp. 995-998). IEEE.