

Camera Canvas: Photo Editing and Sharing App for People with Disabilities

Trung Ngo¹, Christopher Kwan², and John Magee¹(✉)

¹ Department of Math and Computer Science, Clark University, 950 Main Street, Worcester, MA 01610, USA

{trngo, jmagee}@clarku.edu,

² Department of Computer Science, Boston University, 111 Cummington Mall, Boston, MA 02215, USA

ckwan@cs.bu.edu

Abstract. We are motivated to make the interaction between a computer and people with disabilities better with more flexible interactions in their activities. Camera Canvas, the existing photo drawing and editing desktop software for motor-impaired individuals to operate a computer mouse, has been integrated well in the mouse-replacement interface, Camera Mouse. Therefore, our approach is to optimize Camera Canvas for the sake of helping gain a better experience in tailoring local photos/images. With more image filtering designed in computer vision techniques, Camera Canvas enhances better user experience. We conducted user studies with a few users without disabilities, who didn't operate or operated Camera Canvas with Camera Mouse. We have initiated features of social media sharing in the software, and evaluation of social media integration is planned for future work. Our ongoing work includes conducting additional user studies and optimizing the software based on feedback or survey.

Keywords: Assistive technology · Mouse-replacement interfaces · Accessible interfaces · Photo editing · Image filtering · Social media sharing

1 Introduction

The individuals who cannot speak or use their hands to operate a computer mouse are extremely limited in their means of communication. Our goal is to design the image-editing software that contributes to this group of users for their sakes of communication and emotional expression. The process of developing Camera Canvas helps us gain knowledge about user interface design for the people with motor impairments [1, 2].

Integrating Camera Canvas with the camera-based mouse-replacement interfaces, especially the Camera Mouse input system [3], would gain a significant impact for the people with disabilities, by enabling them to control the mouse pointer by other facial movements such as moving their eyes, nose in front of a camera. A click is registered when the user keeps the mouse pointer within a small radius of the item to be selected, for a certain amount of time, one second. It's known that Camera Canvas can be also used with other camera-based mouse-replacement interfaces [4, 5]. There have been

other kinds of apps designed for use with Camera Mouse, where apps have been developed to enable the motor-impaired to create drawings: EaglePaint [1] is a program designed to draw freeform lines, EyeDraw is a one designed with an infrared eye tracker, and VoiceDraw is a drawing program, which makes drawing lines created from different sounds of users. Customizable [6] and automated [7] user interfaces have also been made for people with motor impairments.

Our approach was to develop a program that gives users photo-editing, filtering and sharing capabilities and that an interface would be customized extensively for further advanced tasks [1]. In fact, some users have better control of their movements along the axes [8], but some users only maneuver their movement well within a certain range. Therefore, we took into account that our program needs to be highly configured for a wide range of user groups.

The first version of Camera Canvas was designed to such a level that many users could participate in various tasks—drawing, editing, configuring the toolbar, and even playing simple games that could be highly interesting when being integrated with other mouse-replacement interfaces, especially Camera Mouse. However, we found the photo-editing task more potentially extensible. In fact, we proposed more image-filtering tasks that could be functioned for editing the local or self-drawing photos. And, we introduced color inversion, color rotation, corner detection and a few other techniques involved in this second version. Furthermore, given that Camera Canvas is a configurable desktop app, which could utilize the built-in computer system device like a webcam, we implemented the camera capture that runs a facial detection after a capturing the users’ face.

2 Software Overview and Methods

2.1 Configuration and Toolbar

Camera Canvas has three configurable settings in the Settings menu: toolbar placement, button size, and toolbar sliding speed [1]. An example main menu is shown in Fig. 1. The placement and orientation of the toolbar are adjusted from horizontally top to

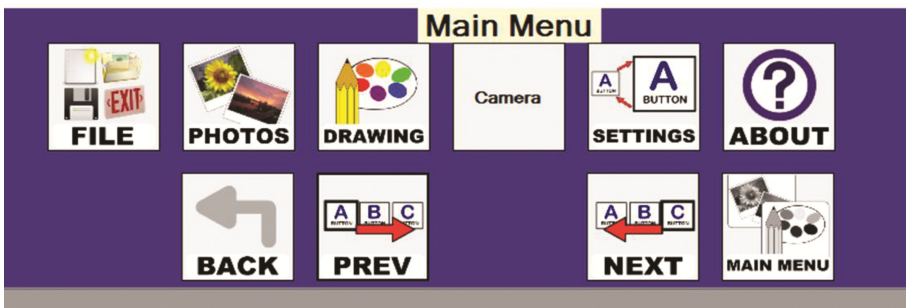


Fig. 1. Camera Canvas in default settings main menu with a horizontal top layout and smaller buttons. The below set of prev and next buttons indicates that there are more buttons off-screen.

vertically right (Fig. 2). Each setting constrains the mouse movement primarily along a single axis and in a single area of the screen so that users could gain more flexibility in controlling the setting toolbar in different areas of the screen [1]. Furthermore, we have added the Camera button as the main additional feature in the default Settings toolbar.



Fig. 2. User interface of Camera Canvas in photo-editing mode with the vertical left-sided photo settings toolbar.

2.2 Photo-Editing Tools

The originating version of Camera Canvas makes the drawing straight lines and geometric shapes plausible and creative for the people with disabilities thanks to the inspiration of drawing in EyeDraw [9]. For the Photo-editing mode, the originating version of Camera Canvas features several interaction techniques to make common photo-editing tasks possible. In specific, the Rotate tool uses a customized user interface component in Visual Studio, the IDE for our software suite, called a Choice Form, and it is equivalent to components such as sliders or small increment arrows, which are usually challenging for users controlling the mouse pointer.

There were several interaction techniques available in the first version of Camera Canvas, which common photo-editing tasks such as the rotation. For example, the Rotate tool uses a custom user interface component called a Choice Form that is easier for the motor-impaired having difficulties controlling the traditional mouse pointer. The middle of the Choice Form contains a preview of the titrated image, so the user could preview its effect before commenting the task.

2.3 Image Filtering

The updated version of Camera Canvas focuses more on introducing new features related to the image-filtering, photo-effect techniques (Fig. 3). Ranged from the simple task to more complex, intriguing ones, the new features include grayscale conversion, color inversion, color shift, and corner detection. Here, each technique was implemented in the same user interface design choice as those in the original version. In specific, each of these image filtering features has its Choice Form under the Effects button's directory with the preview mode of displaying the preliminary result of change.



Fig. 3. A grayscale-transformed image shown within our editing interface. The previous version of our interface also provided an interactive contrast control mechanism.

The first feature is grayscale transform, which turns the color image into a light intensity image. The human visual system is more attuned to intensity changes than color changes, therefore this mode may be beneficial for individuals with minor visual impairments in addition to possible motor impairments of our intended users. The ability to interactively adjust the contrast allows the user to manipulate the image in a way that they can explore details that might not be readily visible to them in the original color images. Figure 3 shows the straightforward result for the imported color image of a flower after using the grayscale feature combined with the contrast adjustment in the Preview mode.

The second feature, color inversion, is the task to transform each color components of the RGB (red-green-blue) values in the edited photo to its inverted values correspondingly. This technique creates a negative image, like what one would see on a physical film negative. In each color spectrum, dark areas become light, and light areas become dark. The negative technique may further be used to explore areas of the photograph that are not readily visible to the user in the original image. This approach may also be used by users for artistic effect. Figure 4 displays an inverted image alongside its original.



Fig. 4. The picture of a cat hanging on the tree branch in the original view (right) and after the color inversion process (left). (Color figure online)

The color shift also manipulates the RGB color values in the current. For example, Fig. 4 has many areas with a high amount of green color, which could be transformed more blueish after being processed by the blue color shift. We implemented three subtasks for the color rotation task since we want to examine all three color shift operations: red, green and blue. Like the previous features, this technique could be used for visual effect or to allow a user to explore details within the image. Figure 5 shows an image that has been manipulated by color shift and rotated.

Corner detection is also designed as a photo-effect feature (Fig. 6). This effect uses Canny Edge Detector to find the gradient edges and threshold these gradient values at zero. So, any pixel points which are not actually on the object edges would be set to zero, as to detect the outer line in the output using the Canny operator. Therefore, the effect creates a “sketch” of user’s physical outer edges in white pixels on a black background. This effect was included to demonstrate a possible dramatic artistic style element as well as provide a way to explore the important edges in an image.



Fig. 5. The result of a self-made picture after being processed by color shift integrated with rotation of 180°.

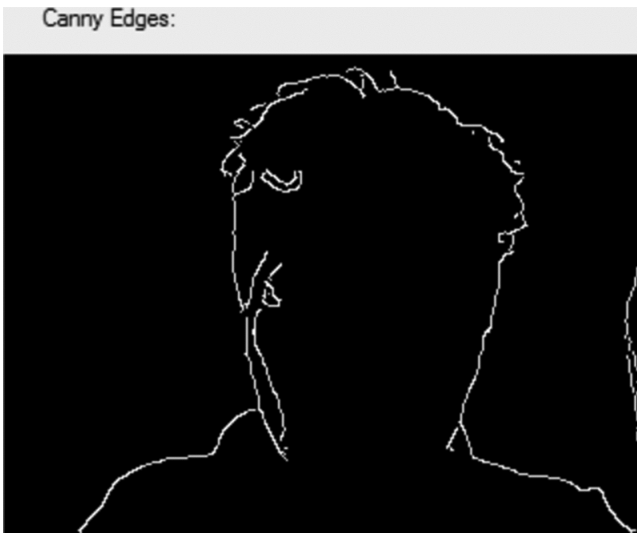


Fig. 6. The screenshot of our face run through the Camera Capture window application with the selected Canny Edge Detector

2.4 Camera Capture

We thought about the idea of getting interactions between the native software like Camera Canvas, and any mouse-replacement systems such as the compatible Camera Mouse. Their relation could produce the preliminary implementation of the additional feature called Camera Capture, which makes the use of the built-in webcam in the computer system.

Camera Capture starts with a task button in the Photo-editing setting toolbar with its similar name. When the user selected the button, he/she allows the computer to authorize the use of a webcam in the system. Camera Capture's user interface is designed more as a standalone program, which has its own Windows application, and Camera Capture consists of a few different Windows forms, most importantly the video capture form built in by the open-source library of C# for computer vision—Emgu. Camera Capture applies the famous feature classifier for object detection, the HAAR Cascading, which is used to distinguish the positive and foreground images with the negative ones by integrating it with Canny Edge detection in the given input. Canny Edge detection is also an alternative technique of corner detection, and basically it detects the outer edges of the foreground image alone and blacken the rest of the portrait.

2.5 Social Media Sharing

After completing a few important feature for enhancing more interactions in the photo-editing toolkit, we aimed at adding an advanced feature involving the tasks of social media sharing. In fact, the implementation for social media sharing feature is still under the construction for a few tasks for different social media platforms. We already registered the application on Facebook and Twitter platforms, so that Camera Canvas would hopefully connect more participants to each other, especially the group of people with disability but needs some attachment or connection under certain means. We are working to finish the photo-uploading and sharing tasks and test them on these platforms. Therefore, Camera Canvas featuring the social media interaction will be introduced soon.

3 Experimental Evaluation

We performed an informal observational evaluation of our new features in the Camera Canvas with seven participants.

The participants of our experiment in general found the program easy to use without prior experience using the Camera Mouse with other apps nor their background in understanding these image filtering techniques. The users expressed their interests in seeing how the edge background or give the benign feedback on the possible improvement for the visual look of the user interface.

We divided the subject roughly into two groups for software testing: a group for direct use of Camera Canvas and another one for integration of other mouse-replacement systems like Camera Mouse. For the tracking configuration in Camera Mouse, we set the feature tracking on each user's nose because it is large enough to stabilize the tracking movement throughout the testing.

The first group which consists of 4 users was assigned just to evaluate the Camera Canvas tended to feel a little easier to maneuver these tasks than the latter thanks to the traditional use of a mouse pointer when the first can operate the mouse manually. However, the latter, which consists of 3 users, found integrating with Camera Mouse more interacting and more flexible to complete some tasks in Camera Canvas. They all were allowed to run different tasks in Camera Canvas. Most of them liked the image filtering set of tasks because of its variability. The first two subjects enjoyed playing around with the feature of corner detection, where they were able to see through their facial shape in the interesting background of black and white theme. However, the last subject found the Color Rotation feature insignificant because the output image wasn't lucid enough in some cases for he to grasp the difference. But he tried playing around a lot on the feature of Camera Capture, and he enjoys adjusting different variables appeared in the separate Window form.

4 Ongoing Work

Our added features have been performed well for most of the users since the tasks are straightforward and quite simple. However, these, especially Camera Captures needs to get additional user studies of the group of people with disabilities as well as more amount of tests. We also plan to evaluate Camera Canvas with additional mouse-replacement interfaces. With more involvement of people with disabilities, we can modify or optimize the software to extend its accessibility to a higher, larger group of people.

We are also taking the designation of the operating system into consideration. Since Camera Canvas have only available in Windows and only as the desktop app, our motivation is to refactor the program, so it would be compatible, ready for installation, and deployed in Mac OS, Linux as well as in potentially mobile phones.

Acknowledgements. We would like to thank our participants for their valuable feedback and Wenxin Feng for her assistance during user studies.

References

1. Kwan, C., Betke, M.: Camera canvas: image editing software for people with disabilities. In: Proceedings of the 14th International Conference on Human-Computer Interaction (HCI International 2011), Orlando, Florida (2011)
2. Kwan, C., Paquette, I., Magee, J., Betke, M.: Adaptive sliding menubars make existing software more accessible to people with severe motion impairments. *Univ. Access Inf. Soc.* **13**(1), 5–22 (2014)
3. Betke, M., Gips, J., Fleming, P.: The camera mouse: visual tracking of body features to provide computer access for people with severe disabilities. *IEEE Trans. Neural Syst. Rehabil. Eng.* **10**(1), 1–10 (2002)
4. Gorodnichy, D., Dubrosky, E., Ali, M.: Working with computer hands-free using Nouse perceptual vision interface. In: Proceedings of the International CRB Workshop on Video Processing and Recognition, (VideoRec 2007). NRC, Montreal (2007)

5. Manresa-Yee, C., Varona, J., Perales, F.J., Negre, F., Muntaner, J.J.: Experiences hands-free interface. In: Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility, pp. 261–262. ACM, New York (2008)
6. Gajos, K.Z., Weld, D.S., Wobbrock, J.O.: Automatically generating personalized user interfaces with supple. *Artif. Intell.* **174**(12–13), 910–950 (2010)
7. Magee, J., Betke, M.: HAIL: hierarchical adaptive interface layout. In: Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A. (eds.) ICCHP 2010. LNCS, vol. 6179, pp. 139–146. Springer, Heidelberg (2010). doi:[10.1007/978-3-642-14097-6_24](https://doi.org/10.1007/978-3-642-14097-6_24)
8. Connor, C., Yu, E., Magee, J., Cansizoglu, E., Epstein, S., Betke, M.: Movement and recovery analysis of a mouse-replacement interface for users with severe disabilities. In: 13th International Conference on Human-Computer Interaction (HCI International 2009), San Diego, CA (2009)
9. Hornof, A.J., Cavender, A.: EyeDraw: enabling children with severe motor impairments to draw with their eyes. In: Proceedings of ACM CHI 2005: Conference on Human Factors in Computing System, pp. 161–170. ACM, New York (2005)