

CS459/559/559-D: Human-Computer Interaction

UG/G: Mini Project 1: Camera applications for visually impaired individuals

BACKGROUND:

When we think of cameras and photography, we falsely believe that only visually able individuals should be able to take photographs. In HCI we talk about making interfaces that are accessible to all users. Companies have recognized this issue and released tools for visually impaired users, for example Google released Guided Frame that uses voice commands and haptic feedback to help visually impaired users take photos.

YOUR TASK:

You will create two desktop applications that enable visually impaired users to take selfie images and images of a static scene. The applications can be created using any programming language.

SELFIE APPLICATION:

Create a desktop application that allows the user to take a selfie using their computer's webcam based on a set of user provided inputs. To make the task simpler, I have outlined potential steps needed to complete the assignment below. You do not need to follow the steps below and can have your own workflow.

- 1) Ask the user to specify positions as "top left", "top right", "bottom left", "bottom right", and "center" using speech inputs. These phrases represent the four quadrants and the center in the image where the person may want their face to appear.
- 2) Capture an image using your computer's webcam and determine if there is a face located in the image.
- 3) Now, figure out whether the bounding box represented by the face lies in the location specified by the user in Step 1 above. If it does not, provide the user with instructions on which direction to move their face. Since the user is blind, you will need to use audio-based instructions.

STATIC SCENE APPLICATION:

Create a desktop application that allows the user to take an image of a static scene, e.g. objects on a table, using an external camera attached to their computer based on a set of user provided inputs. To make the task simpler, I have outlined potential steps needed to complete the assignment below. You do not need to follow the steps below and can have your own workflow.

- 1) Capture an image using the user's external camera attached to their computer. If you don't have an external camera, then flip your computer around and use the computer's webcam.
- 2) Once the user takes the image use an object detector, for example YOLOv8, to label the objects in the scene. You are highly recommended to read the manual for YOLOv8 to understand what objects it can detect and set your scene so only those objects are present.
- 3) Use a text to speech algorithm to inform the user what the objects are and allow the user to select a specific object using speech. Once the user specifies the object, inform the user where the object is. You can specify positions as "top left", "top right", "bottom left", "bottom right", and "center". These phrases represent the four quadrants and the center in the image where the person may want their face to appear.
- 4) If the user is happy with the location of the object, then capture an image. Otherwise, ask the user to move their camera until the object of choice is in the location of interest.

TEST SCENARIOS:

Example test scenarios are provided below:

SELFIE APPLICATION:

Test case 1: the user specifies “top left” but is outside the field of view of the camera. Your program should try asking to make best guesses to try and get the person’s face into the view. For example, you can try having the person move left a few times and if they still do not appear then try moving right and so on.

Test case 2: the user specifies “top left”, but their face is in the bottom right. Your program should provide instructions on how to move until they reach the top left.

STATIC SCENE APPLICATION:

Test case 1: the user specifies “center”, but the object of interest is at the top left. Your program should provide instructions on how to move until they reach the center.

PROJECT SUBMISSION:

Your project submission should include the following:

- 1) Commented source code for your two applications. If you are using code from an online source, you must cite all sources. If you are re-using code from the web, ensure that you have permission to re-use it and cite it. Failure to do so will result in a 0. You should collect data from multiple users for the test scenarios.
- 2) A 6-8 paper written in IEEE Two Column, i.e. IEEEtran, format using LaTeX. Papers written in some other format will not be accepted. Your paper should analyze timing data to determine how quickly your interface allowed the user to take a photo. Include a section on similar applications (either research prototypes or existing real-world examples) and how your solution is different. Discuss how well you were able to perform using the test scenarios listed above. Finally, discuss potential improvements for your solution. All teams must submit a single team report, and individual contribution reports outlining the work done by each team member.

DEADLINES:

January 30 and February 1: team presentations during class. Must include a demo. The second scenario is unlikely to work in real-time, so you may want to pre-record a video.

February 2: team report and individual reports due