User Guide

Introduction

SeeForMe is an Android application that provides object, text, logo, and color recognition for visually impaired users. SeeForMe gives spoken feedback about their surroundings using their Android devices. This user guide provides a description of the system functions and capabilities, step-by-step procedures for system access and use as well as instruction on handling error and failures. If SeeForMe has not been installed on your phone, please refer to the Installation Guide.

System Overview

This system allows three different modes of photo recognition to be used in integration with TalkBack to allow the user to use their Android device's camera to take photos and have a description of the photo's contents read back to them.

1. Object recognition
   a. Limited accuracy, but can return up to 10 objects found
   b. Has difficulty with ambiguous objects, should not be used in high risk situations (such as crossing a street by reading traffic light)
2. Text recognition (with logo recognition)
   a. Automatically adjusts orientation of a photo taken
   b. Is able to recognize small and large sections of text (a business card and a page from a book)
   c. Has difficulty with text laid out in nonlinear layouts, so rows of text are easier than multiple blocks, such as in some infographics
   d. Has difficulty with handwriting
3. Color recognition
   a. Capture image and identify up to 9 unique colors from random sampling, other sampling technique need to be researched
   b. In dark lighting, camera may flash to allow true color to be identified
   c. Has difficulty in low lighting and up close pictures due to flash saturating object of interest

Using the System

1) **App Navigation (double tap to select button when using talkback)**
   a) See enlarged reference images of application interface in appendix.
   b) Open the SeeForMe application.
   c) Navigate the three buttons at the bottom of the screen that correspond to Object, Text, and Color recognition as desired
   d) Tap to select the desired mode
   e) Point camera at desired region of interest

f) Navigate to picture button at the bottom of the screen and tap to take a picture

g) Wait as the picture is processed

h) Results found will appear on the screen which is read back to the user through TalkBack

i) To navigate to a different recognition type requires tapping the back button on the top left of the screen in the Action Bar, which returns you to the initial screen from which you can choose your recognition type (changing this interface to a more user-friendly version is in progress)

2) Optimizing Specific Use Cases

a) Vending Machine
   - i) Select text recognition mode
   - ii) Hold phone up to glass, using edges of glass as reference to isolate picture of one item
   - iii) If item described is not the desired item, or no item is detected, shift phone approximately 6 inches to isolate new item
   - iv) Listen for an item such as “Doritos”, and a code such as “102”
   - v) Sometimes excess text will be output to the user (we are working on this)

b) Reading document or text
   - i) Select text recognition mode
   - ii) For typical document hold phone 12 inches away parallel to the document of interest
   - iii) Capture image and text is spoken aloud
   - iv) SeeForMe reads all text in image so if undesired neighboring text is read retake photo with camera closer to text

c) Color of clothing
   - i) Select color recognition mode
   - ii) Hold phone 6-18 inches from clothing in an ideally well lit environment
   - iii) Capture image and colors recognized will be spoken aloud
   - iv) Recognizing patterns is in progress

d) Object recognition when entering a new room
   - i) Select object recognition mode
   - ii) Capture image and 10 different object recognized will be spoken aloud

Error handling

1. Color detection failure

The color detection failure could due to the following reasons:

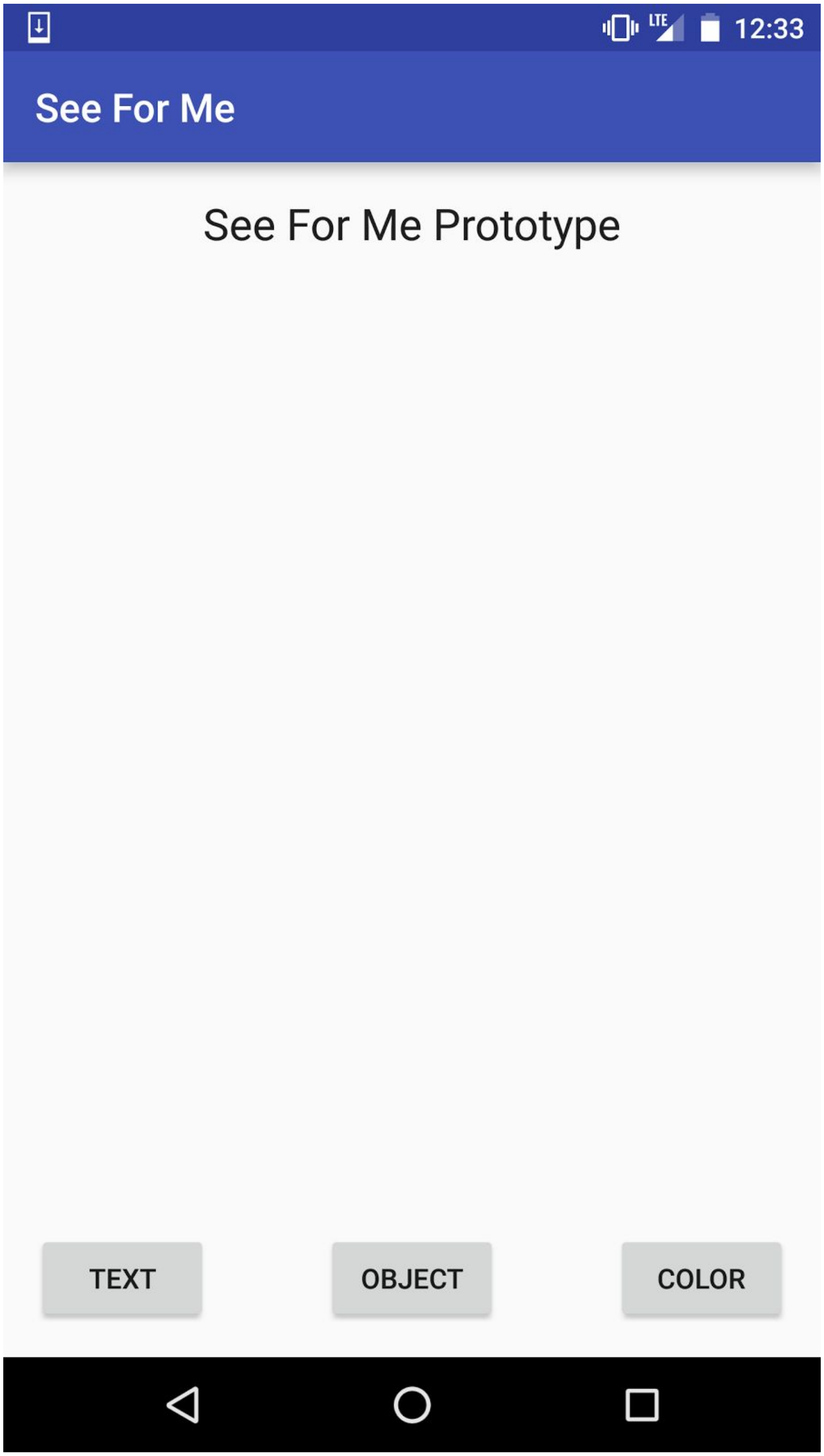
a) Lighting condition.

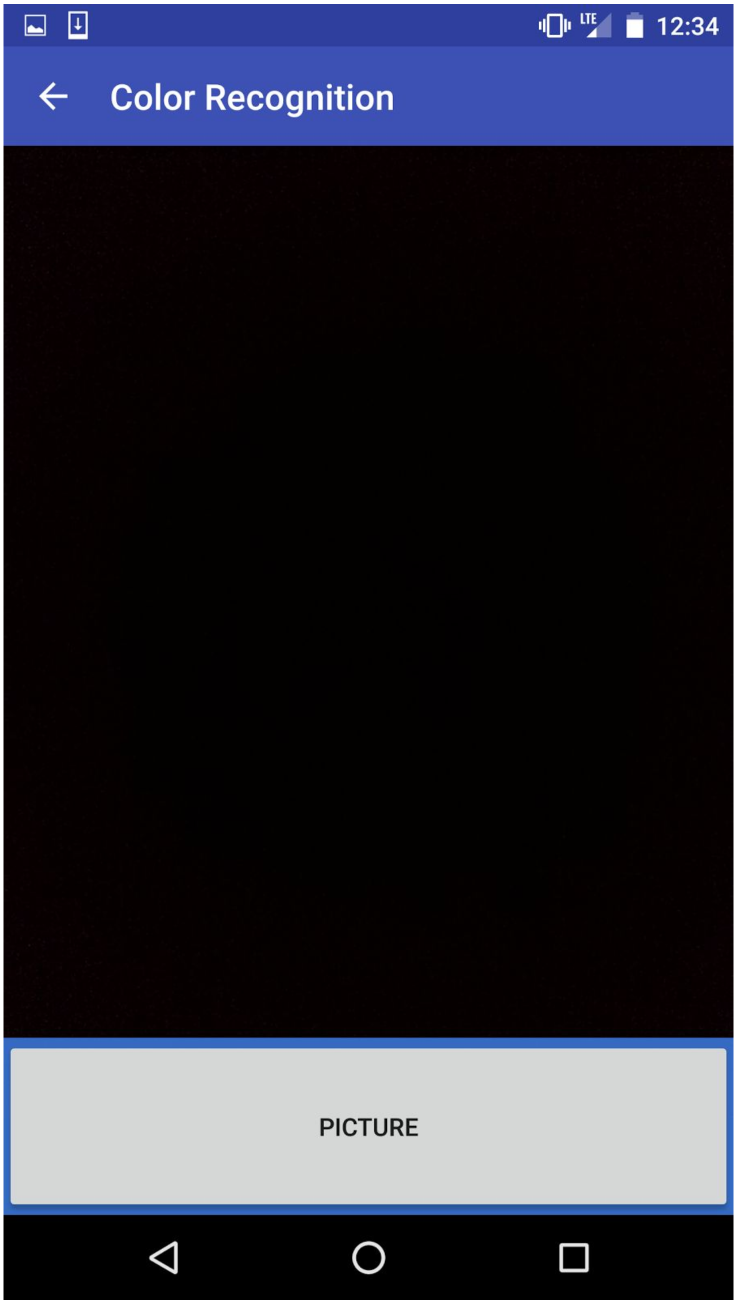
Dim lighting condition affect the quality of color recognition. We recommend retake the picture under normal lighting condition. Avoid harsh, direct lighting since it left reflection on glossy surfaces.

2. **Object detection failure**
   a) Unable to detect anything
   If the “processing image” message did not show up after pressing the picture button, the picture has not been taken yet, try pressing it until the message shows up.
   Photo processing time depends on factors outside our control as well as the user’s internet speed. However photos are typically processed within 15 seconds.
   If no response is sent back and read by Talkback after the “processing image” message pops up. Please check your Internet connection.

3. **Text detection failure**
   If you believe the detection is incomplete or inaccurate, the failure could due to the following reasons:
   a) The missing text is not captured in the picture.
   We recommend retake the picture by holding your camera away from the text of interest until the entire page is in the image.
   b) The surface that the text printed on is wrinkled which blocks part of the text.
   c) The surface that the text printed on is glossy and the reflection block part of the text.
   d) Text is distorted since the camera is not parallel to the text.
   We recommend adjust the position of document so it is orthogonal to the user and parallel to the camera, then retake the picture.
   e) Text printed vertically
   The engine does not support recognize text printed vertically. It tends to interpret words as individual characters.
   f) Text captured in the picture is printed in multiple orientation
   The engine can only process the image at one direction. We recommend take the picture both in landscape and portrait orientation if you believe there are text printed in different orientations.
   g) Multilingual
   The engine does language detection automatically. If you think the language is not the desired one, we recommend you retake the picture clearly following the instruction above.

Appendix





Project References

1. This sample demonstrates how to use basic functionalities of Camera2 API. You can learn how to iterate through characteristics of all the cameras attached to the device, display a camera preview, and take pictures. https://github.com/googlesamples/android-Camera2Basic
2. Looking up a color name from the rgb values is edited from https://gist.github.com/nightlark/6482130#file-gistfile1-java

3. Android google sample codes, to see how to activate and use google API
   https://github.com/GoogleCloudPlatform/cloud-vision
4. Manage google API cloud developer api key and signing
   https://developers.google.com/maps/documentation/android-api/signup
5. The Google APIs Client Libraries
   https://developers.google.com/api-client-library/java/apis/vision/v1