

Project 2

Start Assignment

Due Oct 16 by 6:01pm **Points** 100 **Submitting** a file upload
Available Sep 21 at 12am - Dec 18 at 6:01pm

CS 528, Fall Semester 2023 Project 2

Exploring the Android Camera, Images and ML Kit Vision API
(12/100 of course grade)

Due date: Monday, October 16, 2023 by 6pm. ~~Thursday, October 12, 2023 by class time!!!~~

Overview

The aim of this project is to get you familiar with Android apps that use the Android Camera, Images and ML Kit Vision API. You should complete this project in your GROUPS. The project covers a lot of ground because it is done by a GROUP of students. The GROUP will submit one project with all team members listed. You may discuss the projects with other classmates or on Canvas, but each group will submit their own code for the project.

Project Preparation

The following preparatory steps might be useful.

Step 1: Learn how to run your apps on a real phone Thus far you have run all your programs on the Android Studio Emulator. The emulator has limitations when running programs that require the phone's camera. So you will need to run this project on a real phone. A good video to learn how to connect and use a real smartphone with Android Studio is located

[[HERE](https://www.youtube.com/watch?v=Wp6KbJcnxGU)] [↪ \(https://www.youtube.com/watch?v=Wp6KbJcnxGU\)](https://www.youtube.com/watch?v=Wp6KbJcnxGU)



<https://www.youtube.com/watch?v=Wp6KbJcnxGU>

In order to run code on a real phone, you may need to install USB drivers for your phone's model on the debugging PC. To learn how to install USB drivers on your home machine and how to run the examples in the textbook(s), go through the following tutorials. Note that on Nexus phones, you just need to install USB drivers. Other smartphone models may have to get their phone drivers from their manufacturer's website. To allow Android Studio to run apps on your phone, you will need to turn on USB debugging mode

- **[How to enable USB Debugging mode](#)**

- (https://web.cs.wpi.edu/~emmanuel/courses/cs528/F23/enabling_USB_debugging_mode_2020.pdf)

- **[How to run code samples from textbook](#)**

- (https://web.cs.wpi.edu/~emmanuel/courses/cs528/F23/Run_Textbook_New.pdf)

Step 2: Understand fragments, camera and databases: This project will explore fragments, taking pictures with the smartphone camera and saving them on the smartphone's storage, as well as storing information in a database. You will need to make sure you understand these concepts before starting coding. First review the slides for lectures 3, 4 and 5. You should also read through the following Google tutorials to ensure you understand these concepts:

- **Fragments:**

- **[\[Creating a Fragment and using Fragments to build UI \]](#)** 

- (<https://developer.android.com/training/basics/fragments/creating.html>)

- **Camera:**

- **[\[Choosing a Camera Library \]](#)**  (<https://developer.android.com/training/camera/choose-camera-library>)

- **[\[Taking a Picture using Camera Intents \]](#)** 

- (<https://developer.android.com/training/camera/camera-intents>)

- **[\[Saving files \(e.g. Images\) to smartphone storage \]](#)** 

- (<https://developer.android.com/training/data-storage/app-specific>)

- **Database:**

- **[\[Saving Data in a Local Database using Room \]](#)** 

- (<https://developer.android.com/training/data-storage/room>)



Step 3: Download code for Chapter 19 (CriminalIntent) Download, and unzip the code for Chapter 19 of Android Nerd Ranch (5th Edition) **[\[Here \]](#)**

(https://web.cs.wpi.edu/~emmanuel/courses/cs528/F23/code/ANR_19_CriminalIntent.zip).

Step 4: Study the code for Chapter 19 of Android Nerd Ranch (5th edition) You will be required to extend and modify the code for chapter 17 for this project. Read chapters 7, 9, 11, 12, 16 and 17 of Android Nerd Ranch (5th edition) and also study the code for Chapter 19. Note that while the code contains material from chapters 18 (localization) and 19 (accessibility), you do not have to pay close

attention to material from those chapters. Run the code for chapter 19 on your phone and make sure you understand the code.

NOTE: Depending on what version of Android studio you are running, the CriminalIntent Android project may use an older gradle version.

Step 5: Get Face Detection including face contour detection, face mesh detection and selfie segmentation working: You can start out by getting the [\[ML Kit Vision Quickstart demo app \]](https://github.com/googleamples/mlkit/tree/master/android/vision-quickstart)  (<https://github.com/googleamples/mlkit/tree/master/android/vision-quickstart>) working and testing it out, just to be sure you understand how each required feature should work. You should get the source code on Google's [\[ML Kit Samples Github site \]](https://github.com/googleamples/mlkit).  (<https://github.com/googleamples/mlkit>) Note that the Android Vision-related ML Kit code is in the android/vision-quickstart directory. Download the code, compile and run the examples on a real phone and study them. Make sure you understand the code.

Project Requirements

Step 6: Make the following Changes to the code:

1. Currently, the app can only store one image in the top left corner. Taking a new image replaces the existing image. Make it possible to store more images below the "SEND CRIME REPORT" button in the designated positions shown below. Taking images 2,3 and 4 should store those images in the positions below the "SEND CRIME REPORT" button as shown without replacing the first image.
2. If 4 images are already displayed, taking a 5th image should replace image 1 (in top left corner). Taking a 6th image replaces image 2 (leftmost image below the "SEND CRIME REPORT" button) and so on.

Step 7: Add face detection:

Add face detection for your project. Add a checkbox such that when it is checked, face detection is enabled and when unchecked, face detection is off. When Face detection is enabled, the rectangles for face detection are overlayed around each face in the preview of the picture. When the picture is taken (with face detection enabled), the number of faces found in that picture is reported in the

bottom right corner of the screen as shown. So, for example, if the user tries to take a picture with 2 faces, two rectangles would appear around those faces. The corresponding text displaying how many faces were detected (e.g. "2 Faces detected") will be displayed in the the position shown on the app screen below. If face detection is not enabled, the number of faces detected will not be displayed (blank, no text displayed for number of faces detected).

Step 8: Add contour detection to face detection:

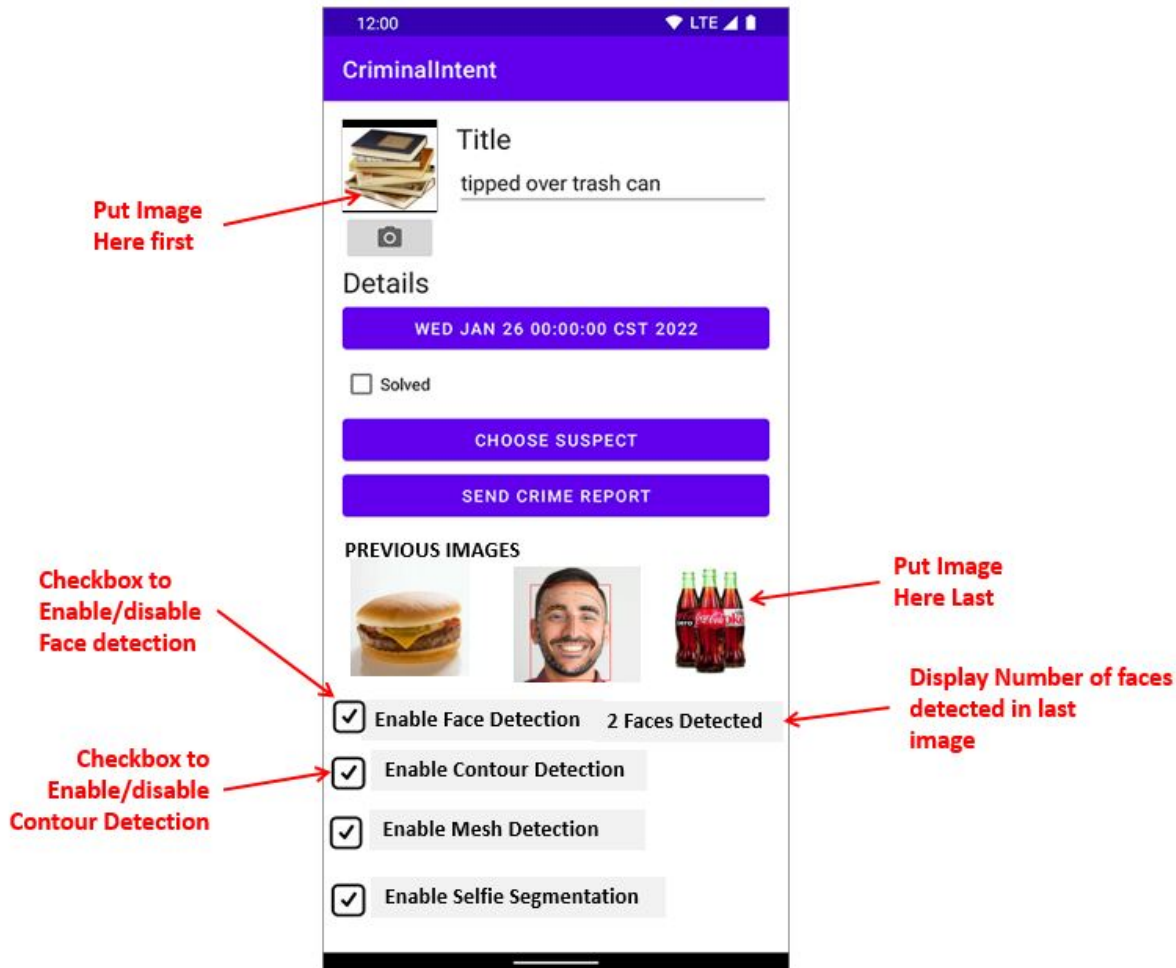
Add contour detection capability with a checkbox to turn it on/off. Turning on contour detection automatically turns on face detection as well.

Step 9: Add mesh detection to face detection:

Add mesh detection capability to the app with a checkbox to turn it on/off. When mesh detection is checked (enabled), meshes are generated and shown over each detected face. When unchecked, mesh detection is not enabled, no mesh is shown on detected faces.

Step 10: Add selfie segmentation:

Add the selfie segmentation capability to the app with a checkbox to turn it on/off. When selfie segmentation is checked (enabled), each selfie image is segmented. When unchecked, selfie segmentation is not enabled and selfies are not segmented.



Notes:

- Only check a single box at a time and apply one ML Kit feature at a time since applying all features to a single image makes the app appear too busy. Specifically, ensure that turning on one feature automatically turns off all other features.
- Checking a box will apply the corresponding feature to the next taken picture.

Step 9: Record session of running code on real phone You will submit both your Android Studio code including the APK (compiled Android program) and a video of you running the app on your phone. You will need to learn how to record a session of you running the app on your phone. Here's a good video on how to do Android screen capture

[[Here](https://www.youtube.com/watch?v=JUXz_eMS9Mg)]  (https://www.youtube.com/watch?v=JUXz_eMS9Mg)



(https://www.youtube.com/watch?v=JUXz_eMS9Mg)

Submitting Your Work

Make sure to double-check that everything works before submitting. Create a zip file containing your Android studio folder AND MP4 Video (Captured session) files. Submit your zip file on Canvas. Do not email me your program or submit it via dropbox.

Before submitting **MAKE SURE YOUR PROJECT'S APK FILE RUNS ON YOUR ANDROID PHONE** Name your submitted zip file for your group as follows. List all last names in the submission using the convention *LastName1_LastName2_LastName3_LastName4.zip*. Only one team member should submit each groups work.