Exercise 4 – May 11th, 2018
OpenCV

Exercise 4: Android & OpenCV

Fork the Github repository at https://github.com/mmbuw/mis-2018-exercise-4-opencv and use it to submit your work as a pull request afterwards.

Note: for OpenCV to work, you will also need to install the "OpenCV Manager" app. Follow the instructions at https://github.com/floe/opencv-android/tree/master/manager to install the APK (do *not* install the outdated version from the Play Store).

Note: in your own app, you cannot call most OpenCV methods immediately (e.g. in `onCreate`) – you need to wait for the `onManagerConnected` callback before all functions are available.

4: "Red Nose Day"

The example app already uses an OpenCV function (Canny edge detector) to process the incoming data by finding edges in the image. Next, look at the Python example at https://docs.opencv.org/3.4.1/d7/d8b/tutorial_py_face_detection.html and use the equivalent Java functions in your app to detect a person's face in the image. Then, draw a suitably sized red circle over the person's nose to create a "clown nose effect".

Note: put the XML files for the classifier into `app/src/main/assets/`, and use the provided `initAssetFile()` method to retrieve the correct filename. See https://github.com/opencv/opencv/tree/3.4.1/data/haarcascades for a collection of classifiers.

Add a README in which you briefly describe how your app determines the correct size for the red circle.

Deliverable

Submit your pull request on Github by Friday, May 25th, 9:00am. Take care to include a compiled APK file, which we will test the in the Android Emulator. Include a text file or comment with the first name, last name and student number of each team member (up to 2 persons).