

UNIVERSITY OF HAWAI'I AT MĀNOA
DEPARTMENT OF ELECTRICAL ENGINEERING
EE 693 B SMARTPHONE VISION

Handed out: 2017 March 1

Due on: 2017 March 8

Android Time To Contact (TTC) — Basic Project Setup:

This exercise sets up the basic template for the term project, which is an implementation of time-to-contact (TTC) on Android. Start by creating a new project in AndroidStudio, call it TTCCalculator. The steps for doing this are the same as they were for setting up the ViewFinder project (in fact you should be able to recognize a lot of code from ViewFinder). Refer to the handout for that project if needed. As indicated there, some changes to the `AndroidManifest.xml` are also needed. Add the lines

```
<uses-permission android:name="android.permission.CAMERA"/>
and
<uses-feature android:name="android.hardware.camera" />
before the line starting with <application. Also add the line
    android:screenOrientation="landscape"
before the line starting with android:label=...
```

Then replace the `MainActivity.java` file with the project template file at <http://people.csail.mit.edu/bkph/courses/EE693B/TTC/MainActivity.java>

Finally, add the other two class files you can find in that directory:

`ExtraInfo.java`; and

`Solve33.java`

to the project (in a similar fashion to how you added `ExtraInfo.java` to the ViewFinder project). As before, `ExtraInfo`, dumps out potentially useful information about the display and the camera in the log. `Solve33` is a simple linear equation solver. Together, these three files constitute the code that purports to implement the time-to-contact algorithm.

Take a look at the java code. Note how the inner loop of the TTC algorithm (stuff that gets run every time a new image comes in) avoids new allocation in order to limit garbage collection while running (similar to what we did in ViewFinder).

The implementation is incomplete, and less than optimal in places. Your task will be to fix the omissions and improve the code (search for `TODO:` comments) So, be warned, it won't work correctly when first installed!

The first task after creating the project is described on the next page...

Problem 1; A big omission in the code as supplied is the implementation of the actual algorithm in the paper

“Time to Contact Relative to a Planar Surface,”.

Search for the “TTCcalculation” method. There are some bits and pieces already there — fill in the rest. Your task is to fill in the missing parts of the method that accumulates the integrals/totals of products of brightness derivatives.

The comments describing what the method is supposed to do, and what its inputs and outputs are, should be helpful. The code at the end of the method sets up the 3×3 coefficient matrix, the right hand side vector of the equations, and then uses “Solve33” to find the least squares solution (with A , B , and C in the vector xx).

After you do write the code to accumulate the totals, the code should basically work, although not optimally. You may need to make adjustments to fit things on your screen or to place them where they will not be cut off.

That is this weeks task. When you get it running, send a screen shot.

We’ll tackle other aspects of the code next week. By the way, while all attempts were made not to have bugs in the code, software is rarely free of defects. Extra credit may be awarded if you find some, document them, and provide a fix!