

Assignment 1 - Height Measurement App

ENG1003, Semester 2, 2019

Deadline: Monday 2nd Sept 2019 20:00 (local time)

Weight: 10%

Aim

Ever wondered how tall, or how far away an object is from you? This is the sort of question surveyors answer all the time. In this assignment you will work with your team to develop a mobile app that uses the phone's gyroscope or accelerometer to estimate the height and distance from you to an object using simple trigonometry.

Background

A construction company, **DemolisBuild Pty Ltd**, has contacted your team to work on a height estimate app which they can use for their daily construction work. They intend to incorporate the app you produce into their existing systems. They will be using their own in-house IT team to handle interfacing the systems with your app. As such, it is very important that your team aims for stability and clarity in your code-base and provide an appropriate handover for **DemolisBuild's** IT team to allow them to integrate the app into existing systems.

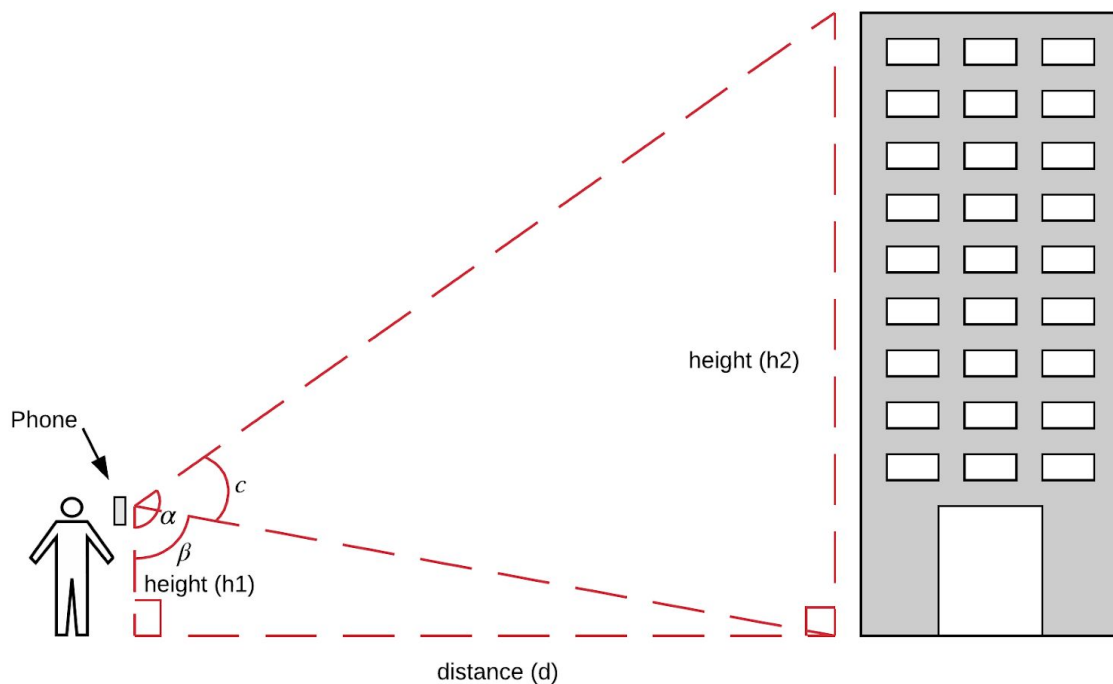
Calculating height and distance of objects

The basis for this app is the idea that by using the height of the phone off the ground (or the user's height as a proxy for this) and the angle (beta) of the phone to either the base or apex (alpha) of an object, we should be able to compute the height of (and distance to) the object in question.

You can assume the person and the object being measured are at right-angles to the ground, and the person and the object are on flat level ground.

A simple application of trigonometry (refer to the resources section) can be used to determine the distance to the object (D) using the angle to the base of the object (B) along with the height of the camera (H_1).

The distance to the building (d) and the angle to the apex of the person object can be used, along with the camera height (h_1) to determine the height of the building being measured (h_2).

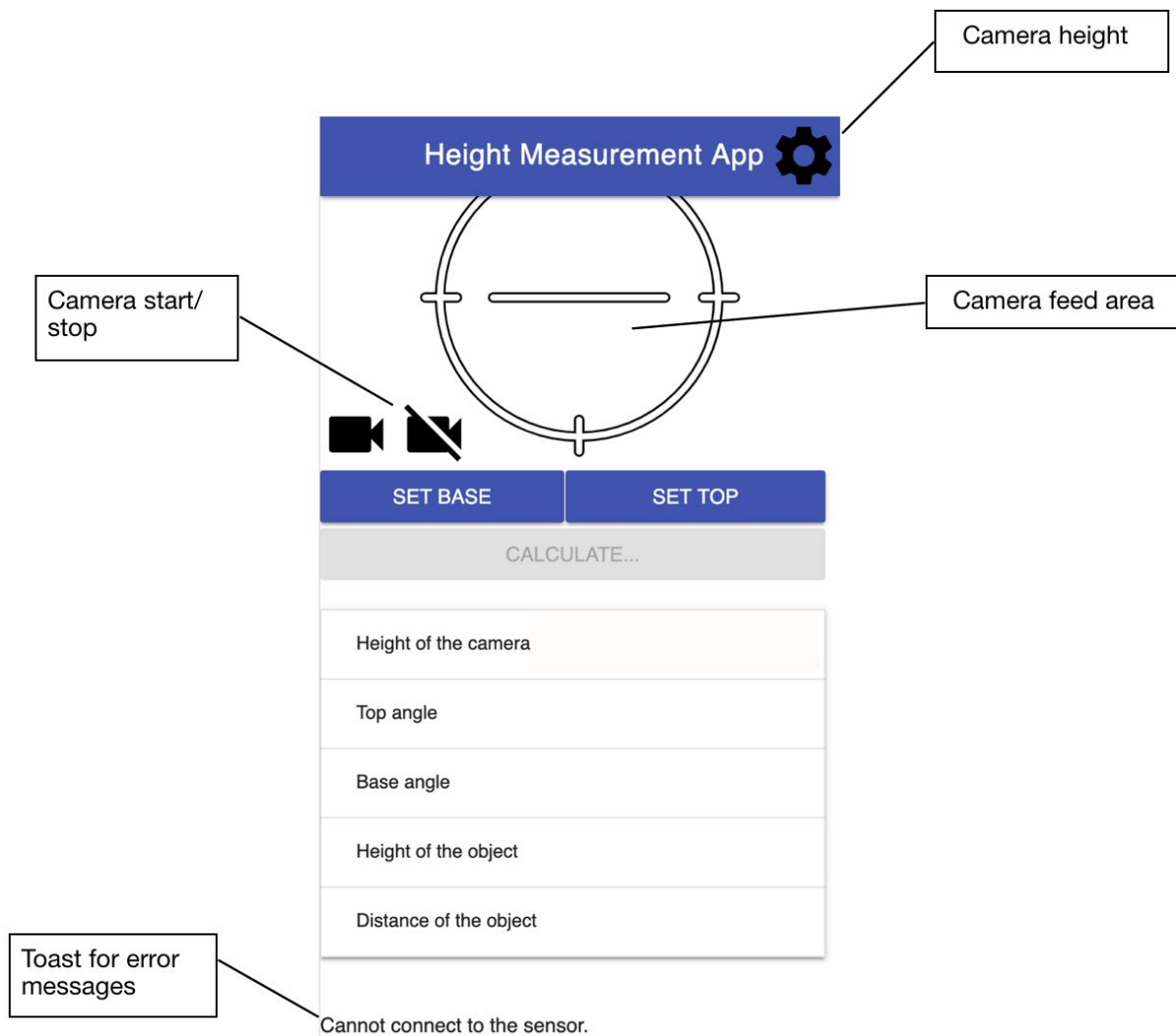


Angles obtained from the phone using device orientation

When using absolute device orientation (AbsoluteOrientationSensor) there are four values that you have access to: x , y , z and w ; where x value represents the forwards-backwards tilt as rotation quaternions of the phone. We've done the math for you in the sensor test app, so you have the euler values (in degrees) as alpha, beta, and gamma. Feel free to use the calculation or to improve on it for accuracy. You can observe these values on the team phone using the ENG1003 Sensor Test app. For this assignment you should use the code provided through the sensor test app: <https://eng1003.monash/apps/sensortest/>

What you are provided with?

The app skeleton can be downloaded from the ENG1003 Moodle site. The app skeleton is titled *assignment1.zip*. The code displays a page as shown below:



In the script folder, you will find camera.js file has the function `initialiseCamera()` to start the camera and `stopCamera()` to stop the camera. You will need to call these functions from their respective buttons. You do not need to understand the code written in these functions.

You will write JavaScript code in HeightEstimate.js file and will need to make changes to the index.html and style.css.

What you need to do

Getting started

1. As a team you should download a copy of the project skeleton and unzip this.
2. The 'Skeleton.zip' file contains the code of the web app.
3. You should open the entire skeleton folder in the Atom to understand the workings of the files for the assignment.
4. You should discuss within your team on the breakdown of coding features and responsibilities. It is recommended that you practice pair programming.
5. Plan to have steady progress rather than complete the assignment at one go, or to do it at the last minute.

Programming tasks

The Programming component of this assignment is worth 7% of your overall unit mark.

For the programming tasks, we suggest you begin by discussing how do all the features come together for the app to work **before** you begin coding.

Feature 1: Sensing device orientation

Write code to capture DeviceOrientation events and show these on the screen for the user. The important orientation component for us is the one representing tilt from front to back.

Feature 2: Smoothing sensor data

You may notice that the values you get from the sensors are very “jumpy”. It is a good idea to report to the user the value in two decimal places, or the whole degree value. A simple smoothing strategy is to take the current value as the average of the last N sensor values. You should implement this so that you display a more stable tilt value. It is up to you to decide how big N should be.

Feature 3: Set camera height

When the settings button is clicked, the user should enter the height of the smartphone camera from the ground in metres. They should estimate this based on their height. The camera height will be used later. The settings button is required to be set to prompt the user for this information and then store it. You will need to update the onclick attribute of the cameraHeight image in HTML to link a function to it.

Feature 4: Record tilt angles

You will need to use the set base button to allow the user to record the device angle when the screen crosshairs point at the ‘base’ of an object. Set top button would record the ‘apex’ of the object. The user should be able to tilt the device and tap these buttons at the appropriate point. You will need to use the camera to point the crosshair to the base/apex of the object. You will need to update the onclick attribute of the videoOn and videoOff images in HTML to link a function to it so that you can turn on and off the camera.

Feature 5: Calculate the distance to the object

Given the height of the smartphone from the ground, and the angle to the base of an object also on the same level, calculate with trigonometry the distance to the object. The JavaScript Math object provides trigonometry methods that may be useful to you. See the documentation in the References section below.

The calculate button should be disabled until all the required input are not entered.

Feature 6: Calculate the height of the object

Given the estimated distance to object and the angle to the base and apex, calculate with trigonometry the height of the object and display this for the user. This should be calculated and displayed once both angles have been updated and subsequently whenever the camera height, or angles, are updated.

Your app need only work in portrait mode. You can assume the phone is locked in portrait mode and the app will never be used in landscape mode.

You will use the same calculate button from feature 5 and feature 6

Feature 7: Display information

Display the current tilt, base angle, top angle, height of the object and the distance to the object in the MDL table provided. You can show appropriate error messages using the label provided at the end of the HTML file.

Testing the app

Upload your code to the ENG1003 server using the assignment upload page. The uploader can be found here:

<https://eng1003.monash/uploader/>

Once you have uploaded the code, you can view the uploaded assignment by selecting the appropriate assignment from the dropdown list on the assignment viewer. The viewer can be found here:

<https://eng1003.monash/view>

Milestones Checkpoint

It is expected that you have already read the assignment one instructions by your practical class. There are two milestones (checkpoints) that is worth 0.5% each in the last hour of your Week 4 and 5 practical classes. Your team will need to have planned out the work to be done in assignment one, and have a clear deadline for each part.

Your demonstrator will come by while you are working on the coding exercises to discuss the work that needs to be done by each of the milestones.

Presentation

This assignment includes a presentation component worth 3% of your overall unit mark.

In your week 6 prac class you will give a handover presentation, **DemolisBuild's** IT team on the current state of the app. Your team should present an overview of the functionality, the design of the code, and any specific hardware required for its use. You should warn the new team about any current issues in your app, as well as pass on any suggestions you have for improvements.

Format

Each team will deliver a 10 minute oral presentation (in prac class) describing and demonstrating their app and detailing any issues they encountered. Every member of the team should present for 2-3 minutes.

- The target audience for this presentation is another team who will be extending the project further.
- This presentation would be delivered in a formal business setting and so all team members should be dressed appropriately.
- This presentation must discuss the structure and functionality of the application as well as any design decisions made.
- The presentation should **NOT** contain a detailed description of specific lines of code.

As with any good presentation, it should be prepared well in advance of the due date (including any visual aids) and it should be well rehearsed as a group and individually.

Resources

- <https://eng1003.monash/materials/walkthrough/sensors.html>
ENG1003 Sensors Walkthrough
- https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Global_Objects/Math
(Mozilla Developer Network: Global Objects - Math)
- <https://www.mathsisfun.com/algebra/trigonometry.html> (Trigonometry)

Submission

Your team should submit their final version of the application online via Moodle. Your final submission will have the following requirements:

- A zip file named based on your team (e.g., "Team014.zip").
 - This should be a ZIP file and *not any other kind of compressed folder* (e.g. *.rar, .7zip, .tar*).

The submission should be uploaded by the team leader and must be finalised by Monday 2nd Sept 2019 20:00 (local time).

Please note: Your entire team needs to accept the assignment submission statement individually on Moodle.

You also need to individually complete the following tasks (described below)

- CATME peer assessment survey (open for a week after submission)
- Assignment code interview

Your presentation and interview will be done during your practical classes in Week 6.

Marking criteria

Programming tasks

Your assignment will be assessed based on the version of 'HeightEstimate.js' file you submit via Moodle. Before submission check your code still works with the original app skeleton, in case you have modified your copy of any of the other files. We will run it with the original app skeleton and test it on your team smartphone. We will use the same phones when marking your assignments.

Assessment criteria:

- Whether the app functionality satisfies the assignment specification
- Quality of app source code, including structure and documentation

You will be marked as a group, however your individual marks will be subject to peer review moderation based on CATME feedback and your assignment interview.

A detailed marking rubric will be available on the unit Moodle page.

CATME Peer Assessment

You are expected to work together as a team on this assignment and contribute roughly equal amounts of work. Peer assessment will be conducted via the CATME online system. You will receive email reminders at the appropriate time.

Not completing the CATME peer assessment component may result in a score of zero for the assignment.

Do:

- Give your teammates accurate and honest feedback for improvement
- Leave a short comment at the end of the survey to justify your rating
- If there are issues/problems, raise them with your team early
- Contact your demonstrators if the problems cannot be solved amongst yourselves

Do NOT:

- Opt out of this process or give each person the same rating
- Make an agreement amongst your team to give the same range of mark

Assignment code interview

During your week 6 prac class your demonstrator will spend a few minutes interviewing each team member to individually gauge the student's personal understanding of your Assignment 1 code. The purpose of this is to ensure that each member of a team has contributed to the assignment and understands the code submitted by the team in their name.

You will be assigned a score based on your interview, and your code mark will be penalised if you are unable to explain your team's submission:

Category	Description	Penalty
No understanding	The student has not prepared, cannot answer even the most basic questions and likely has not even seen the code before.	100%
Trivial understanding	The student may have seen the code before and can answer something partially relevant or correct to a question but they clearly can't engage in a serious discussion of the code.	30%
Selective understanding	The student gives answers that are partially correct or can answer questions about one area correctly but another not at all. The student has not prepared sufficiently.	20%
Tolerable understanding	The student is reasonably well prepared and can consistently provide answers that are mostly correct, possibly with some prompting. The student may lack confidence or speed in answering.	10%
Complete understanding	The student has clearly prepared and understands the code. They can answer the questions correctly and concisely with little to no prompting.	0%

Presentation

Students are marked individually for this assignment on their presentation skills Assessment criteria:

- Voice is of appropriate volume, speed and enthusiasm
- Language is appropriate for a formal context and jargon is only used where necessary (and explained if used)
- Eye contact is consistent and covers most of the audience
- Body language complements the presentation
- Explanations are clear and visual aids used appropriately

A detailed marking rubric will be available on the unit Moodle page.

Other information

Where to get help

There will be a FAQ posted in Moodle and updated periodically. You can also ask questions about the assignment on the General Discussion Forum on the unit's Moodle page. This is the preferred venue for assignment clarification-type questions. You should check this forum (and the News forum) regularly, as the responses of the teaching staff are "official" and can constitute amendments or additions to the assignment specification. Before asking for a clarification, please look at the FAQ and forum.

We will run additional assignment help-desks during weeks 4 and 5 after hours. Times and locations will be posted on Moodle.

Plagiarism and collusion

Plagiarism and collusion are serious academic offenses at Monash University. Students must not share their team's work with any student outside of their team. Students should consult the policy linked below for more information.

- <https://www.monash.edu/students/academic/policies/academic-integrity>
- <https://www.monash.edu/engineering/current-students/enrolment-and-re-enrolment/course-information/assessment-and-examinations/academic-integrity-and-plagiarism>

See also the video linked on the Moodle page under the Assignment block.

Students involved in collusion or plagiarism will be subject to disciplinary penalties, which can include:

- The work not being assessed
- A zero grade for the unit
- Suspension from the University
- Exclusion from the University

You are required to reference code that has been obtained or provided by other sources (i.e online), including formulas for calculating. This should be done within a comment above the code.

Late submissions

We do not accept late submissions without special consideration. Such special consideration applications should be made to the unit email address with a completed form and supporting documentation within two business days of the assignment deadline.

<http://www.monash.edu/exams/changes/special-consideration>

Unavailable team members

If team members are missing on the day of the presentation, the remaining members should proceed without them. Missing team members will receive a mark of zero unless they are granted special consideration. Such special consideration applications should be made to the unit email address with a completed form and supporting documentation within two business days of the presentation date.

<http://www.monash.edu/exams/changes/special-consideration>

You must also inform your team members if you will be absent on the day of the presentation.