Javascript Assignment 2017 - Augmented Reality Track Visualization

Instructions - Read this first!

Thanks for applying for a job at MARSS! We hope you will enjoy working on this assignment.

We would like you to spend about 4, max 5 hours on this assignment. Using your time appropriately is part of the test. The objective of this test is not only the destination, but the journey too.

Although we would like a finished product in the allotted time, we realise this may not always be possible. Please take notes and explain your thoughts as you go along and how you would improve if given more time.

Please upload a zipped archive with the result as indicated in the email.

The archive must include all (commented) code, but not necessarily any additional notes or documents you've made, though you can. They will be discussed during the following interview in the interview, where you can bring additional material.

If you so wish, and for any reason, you can do further modifications to the code itself after sending the zip file; just bring all material for the interview.

Context - Product X

Imagine you are being asked to implement some features of a larger project.

The final product is an **Augmented Reality**, **browser based, web application** for (potentially mobile) devices that can overlay "track" information onto a live camera video feed.

Tracking is, roughly, the process of relating different observations about the same object (*target*) building an history of its position over time (a *track*); it's the job of a Tracker (eg. a Radar Tracker), over successive scans, to analyse the data and determine whether there is none, one, or more distinct objects, and to associate subsequent "returns" to the same target.

In a more generic sense, tracking information (that could be related to boats, people, cars, flying objects, etc.) can come from diverse sources, including not only positional data but also other information such as a boat name, destination, size (see AIS if curious about it), etc.

Data from those multiple sensors is processed by our server, cleaned up, merged, augmented, and finally distributed to the different applications (including the AR tablet web application), using standard web technologies.

Result

We expect a full project structure according to your common best practices, including at least a stub of testing framework.

It's your choice if the page can be simply accessed locally, or if a static web server should be used, and whether a build process is required or not (you do not need to provide the setup for that, just briefly indicate how we are supposed to use it).

You are free to use any framework/library you see fit.

Assignment

For the product X described above in the Context section, you have been assigned the task of:

- data visualization of track(s) information over the video

Imagine different teams are taking care of other tasks, but you are building the main project structure as well.

- the data format for the tracks is described below
- additional samples are provided as json files (thou beware the exact data values are not meaningful wrt the provided video)
- you can use the attached video in a loop as a background
- if useful, assume a field of view of 50 degrees for the camera

Data Format

The data format for the tracks is as following.

Track updates can come one by one, or many at a time.

track information is indexed by trackID, and contain the following information:

```
b: bearing from tablet position to track, relative to north
d: distance in meters
s: (square) size of the target (approximate)
v: track speed in m/s
a: track altitude (m)
t: type of track (see below for examples)
x: X position compared to image center,
   -0.5 represents left border, 0.5 is right border
y: Y position to compared to image center,
   -0.5 represents bottom border, 0.5 is top border
r: risk level [optional]: 1 to 5 (where 1 is minimal and 5 is maximum)
D: track Details [optional].
   Could be coming as part of a "positional" update or as a separate
one
```

Track Types

```
H = Human,
V = Surface Vehicle (eg. car, scooter),
M = marine watercraft (eg. ship),
U = underwater (eg. submarine, diver),
A = airplane,
D = drone/UAV,
C = heliCopter,
X = Unknown
```

Data Format

```
{
  "cd8855": {
    "b":-77.13,
    "d":141.65,
    "a":0,
    "s":3.79,
 "v": 3.4,
    "t":"H",
 "r": 5,
    "y":0.015,
    "x":-0.043
  },
  "2cc615": {
    "b":74.25,
    "d":124.25,
    "a": 0,
    "s":1.48,
 "v": 1.1,
    "t":"H",
 "r": 1,
    "y":0.016,
    "x":0.099
  },
  "akb234": {
    "D": {
      "name": "Queen Victoria",
      "desc": "Cruise Ship"
    }
  }
}
```