Create an Augmented Reality application showing list of nearby places in a camera in smart phones.

This document is to understand how to create applications that implement augmented reality showing the list of nearby places such as ATMs, Hospitals in the real time showing it in the camera.

**Idea:**

Idea is to point the desired places in the appropriate angle in the mobile phone. Opens the camera and shows the nearby places in the appropriate direction. For example, if an ATM is in the north direction, whenever the phone is pointing to the direction covering the north, the image indicating the ATM is placed in the appropriate angle in the camera.

**Getting the places nearby:**

To get the places nearby, Google provides a web-service:

<https://maps.googleapis.com/maps/api/place/nearbysearch/json>?

This web-service may provide both XML as well as JSON data. However, the URL mentioned above is to get the JSON web-service. The GET request includes the location’s latitude and longitude values along with the radius to with the nearest places are needed to look upon. You may also want to specify the type of places needed, like Hospital, ATM. Read the documentation for more details.

So we need to specify our current location to get this web-service along with various other parameters to get the appropriate places nearby.

**Parsing the JSON response:**

We parse the response that we get from the web-service. Here, the place names, its latitude and longitude are of utmost importance for us.

**Working:**

We first find where the appropriate location is available with respect to the current location in terms of angle. Then if the camera covers that angle then, it should represent the place as an icon.

So, we do the following:

* We find the angle of the places we intent to show in the map
* We find the angle of the device that it is currently pointing to.
* If the desired place is available in that coverage angle of the camera, we show the desired place as an indicative image.

**Finding the angle of the place intended:**

We need a relative line from which we can find the angle to the desired location. For that, we need a relative line from which the angle is to be drawn. We choose the line joining the current place with North Pole as the relative line. Although it is curvy in the real map, we consider it straight as we zoom to represent at most 1-2 km.

Desired place

Your Current Location

North

Here we need to find out the angle. All the details that we have, is the latitude and longitude of our current location and the desired location.

So, to get that angle we just consider the map as plotted in a graph with latitudes and longitude representing the Y and X – axis respectively. However, the desired angle is between the line and Y-axis (pointing towards North pole).

We create a right-angled triangle in the graph with the latitude and longitude data available.

Now, we have a right-angled triangle with points (c1, c2), (d1, d2) and (d1, c2).

(d1, c2)

Desired location (d1, d2)

Current Location (c1,c2)

North

Now, we need to find the desired angle.

So, we first find the distance between the points

* (d1, d2) , (d1,c2) which is absolute value of (d2-c2) since the desired place may be at any quadrant, in the graph resulting in either positive or negative values.
* (c1, c2), (d1, c2) which is the absolute value of (d1-c1) since the desired place may be at any quadrant, in the graph resulting in either positive or negative values.

With this we apply the famous of tan-1, which is:

tan-1(opp/adj)= θ.

But sadly, this value is in the range of [0, π/2]. We need to figure out the quadrants of the place on our own.

**Figuring out the quadrant:**

To spread out the value of range [0,90] to [0,360] correctly, we use the sign of the values we get from the sign of (d2-c2) and (d1-c1).

If

* Both are positive, it means that we need not shift the quadrant
* (d2 - c2) is negative and (d1- c1) is positive, shift to the fourth quadrant
* (d1- c1) is negative and (d2- c2) is positive, shift to the second quadrant
* Both are negative, shift to the third quadrant.

**Overlaying the image on camera:**

As the pre-math done correctly, we need to overlay the pin point over the camera.

At first we find the angle of the device pointing. Since the android devices have sensor to detect the angle with the Magnetic north, we use that sensor to detect the angle at which the device points with respect to magnetic north.

We can measure the coverage angle of the camera. Read android documentation for more details.

To superimpose, we need to calculate the width and height of the device. Since, the demonstration is done on android device, we don’t have a fixed value. We get the width and height of the device dynamically.

If the difference in angle for camera and desired place with respect to the north equals to 0o, we display that place image in the center of the screen. We move the desired image, left or right based on the angle difference.

Alternatively, placing all the places in the same height in the device looks unlikely. So, we get the distance between the place and our current location and use it to determine the location where it should be placed. As, our intuition expects, the distant one is placed at the top of the camera and the closest comes to the bottom.

**Magnetic Declination effect:**

Did you notice that we mentioned that we can get angle of Magnetic North from the device and not North Pole? So, we need to correct the angle between, magnetic north and North Pole. This is called as Magnetic declination. It varies based on time and location. In android, however, we can get the magnetic declination using one of the classes in the android framework. This is used to correct the correctness of the angular difference in terms of locating the places.