Environment Sensing Based In-door Location Monitoring and Topology Mapping

Muhammad Umar Hassan

# Abstract

While most of the location measurement systems like GPS and GSM incorporate some form of RF beacons and triangulation to pinpoint the location of the target, this method describes a way to sense and report the location where the target itself tries to actively perceive the topology of the surrounding by constructing a dynamic three-dimensional map and reports the position with reference to this map. The technique may not be used as a primary means of location but can be coupled with other triangulation-based techniques to keep track of the position even when the RF signals are blocked or there are not enough beacons available for triangulation. Furthermore, this technique is very energy efficient and suitable for locating small pocket-sized targets that are usually handheld, are carried by moving carriers like human beings.

# Introduction and working principle

The self-locating device, which we call “self-positioning dot” in the following, as placed at an unknown location with an unknown altitude in a three dimensional space first tries to map the topology of the surrounding while working like a mini radar. Well, this similarity is in fact only because the process is functionally similar to that of radar but the means to do that are very different. Unlike a conventional radar, it uses light time-of-flight sensors to judge the distance of the first optical obstruction in a particular direction. Also unlike a conventional radar, instead of continuously rotating the sensing device along a particular axis, it has multiple sensor mounted on spherical pitch circle pointing radially outwards. The device also has a casing that in addition to providing the mechanical support, provides blinkers to the sensors so that the field of view of each sensor does not completely interfere with that of its neighbors.

# The Dot

In order to explain the working in a three-dimensional space, let us first describe it in two dimensions. Figure 1 shows a two dimensional sensing object held stationary in a field.

|  |
| --- |
| Chart, icon  Description automatically generated |
| Figure 1: An illustration of a two dimensional “Dot”. N sensors are mounted on the pitch circle of a mechanical base. The base also provides optical blinkers to limit the field of view (FoV) shown yellow color. |

In this particular example, there are ten range sensors facing outwards and spaced equally along the circumfarence of the mechanical base. The sensors are limited by (at least):

1. a maximum field of view due to the blinkers
2. a possibility of false negative detetecion which renders them blind to an object in front of them with an inverse probability of the fraction of cross sectional FoV area the object intersects with, and
3. other uncertainintes arising due to physical limitations such as clock jitter, analog to digital conversion and quantization.

# Methodology

The ToF sensors send a short IR pulse and try to measure the time of flight for this pulse to get reflected from an emissive surface and returning to the sensor. Based on the time of flight, individually, each sensor can measure the one-dimensional distance of the nearest obstruction in front of them. However, this individual reading is not quite sufficient as the task of two-dimensional positioning and topology mapping begs to combine the measurement of all the sensors. Principally, this is very similar to other tomography techniques and one can utilize a technique used in other tomography devices.

Of all the algorithms and techniques used in tomography and topology mapping, one is based on a principal I call “probability mapping”. To understand what probability mapping is, see Figure 2 which shows our self-positioning dot held stationary in a two-dimensional environment, surrounded by an irregular optical obstruction.

|  |
| --- |
| Chart, radar chart  Description automatically generated |
| Figure 2: The Dot is placed in a two dimensional environment. The whole space is divided in equilateral probability bins. The surrounding of this environment is an optical obstruction and the dot can move freely in the free space. |

The space is discretized into small probability bins. With respect to the irregular shape of each section of the environment situated in the FoV of a particular sensor, the sensor will give out a stream of distance readings distributed in the range of minimum and maximum distance of the obstruction. The shape of the dataset distribution itself contains a lot of information but to keep things simple for now, we can assume that depending upon the kind of statistical method used to extract one representative reading out of this set, the sensor will assume that there is only one object in front of it with a known distance and a dynamic, but known, uncertainty. So if the sensor detects a distance D, the probability of finding that object in all the bins lying withing the FoV and at the detected distance D will be the same. So by assuming a fixed starting position for itself, the dot increments the probability counter in all such bins by one unit. Nothing further happens unless the dot changes its position or orientation in which case, there will be a new set of probability bins that get hit by the sensors. The dot, once again, increments the probability in the bins by one unit and waits for further movement. It must be noted that in this new sample, not all the bins that are marked by the sensors belong to the set of bins in the previous sample. Some of the bins would now be getting the hit for the first and some for the second time. After “feeding” this sample to all the bins, the recurring bins now indicate relatively double probability of finding an obstruction.

The dot keeps on moving and rotating randomly in the workspace with its carrier in an unpredictable manner, and new bins keep getting hit by the sensors and start indicating a probable presence of an obstructive surface. While some of the old bins will also be hit which will indicated more probability of actually finding an obstruction at the respective position of those bins. One can normalize the probability value in all the bins, may be, based on the maximum possible probability counter throughout the known space and color code each bin with respect to the probability it represents. Such a color map soon start to develop a discrete map of the surrounding.

To verify this hypothesis and such a progression of a probability map, I’ve written a proof-of concept computer program to simulate this scenario where the position of the dot is controlled by the on-screen cursor and the map is developed as the cursor moves around the screen. The screenshot below shows the progression of the probability map as the dot maneuvers only a small portion of the space without even rotating a bit.

The screenshots also show a black line passing through the boundary bins that constructs an estimate of how exactly the obstruction looks like. It must also be noted here that even if a sensor gets continuously blocked by an obstruction sticking to the Dot itself, only if the sensor can detect the problem, the rest of the sensors will keep on developing the map without the help of the blocked *sensor.*

*So in each sampling cycle, to develop the probability map, the algorithm only has to:*

1. *collect samples of scalar distances of the possible obstructions,*
2. *for each sensor , find all the probability bins that lie within the FoV of each sensor and at a distance ,*
3. *starting from the strongest bin find out the nearest bin that has a reasonable probability value and construct a line between the two bins, and*
4. *keep on constructing this line until a reasonable bins have been connected and the line connects back to the first bin to form a polygon*.

It is quite clear that this algorithm is light is both space and complexity and can be replicated easily in 3D space where:

1. the circular dot will be replaced by a spherical one,
2. the 2D space will be converted into a real-world 3D space,
3. the square bins are replaced by cubes, and
4. the line segments integrated as a representation of the boundary are replaced by triangular facets which integrate into a human perceivable 3D surface,

which means that by locating individual triangular facets, the dot will be constructing a three dimensional surface representing the surrounding environment and can graphically show its whole journey in this space.

There is one important thing that now needs to be discussed. For this continuous probability mapping to work, the dot must know its own exact location and orientation in the space to report the relative position of the bins to store the probability. Although, the orientation can be determined using an SoC orientation sensor like the MPU9250, which is a 9 axis motion sensor packed in a tiny package; about location, one might wonder that wasn’t the dot supposed to report the location on its own in the first place? The answer to this question is in the assumption that initially, to build up even a partial probability map, the dot needs to be aided using other location sensing techniques. Such as the GPS/GSM/Bluetooth triangulation or the MPU9250 itself. MPU9250 isn’t an absolute position sensor, rather it can report the rate of rate of change in this position only (which is called the acceleration), Given the accelerometer readings along with the orientation quaternions computed using the gyro-compass, one can integrate the accelerations in discrete domain to get the change in position from the initial position. However, this position will only be an estimate because all kinds of accelerometers currently available have intrinsic drifts which tend to sum up to large errors with time. For short durations of time, however, this data can be used to develop the probability map and then the map itself can be used to detect and remove the drift in the sensor.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A picture containing text, umbrella, accessory, vector graphics  Description automatically generated | Chart  Description automatically generated | | Chart  Description automatically generated | | Chart  Description automatically generated with medium confidence | | Radar chart  Description automatically generated with medium confidence |
| (a) | (b) | | (c) | | (d) | | (e) |
| Chart, shape, radar chart  Description automatically generated | Chart, radar chart  Description automatically generated | | Chart, radar chart  Description automatically generated | | Chart, radar chart  Description automatically generated | | Chart, radar chart  Description automatically generated |
| (f) | (h) | | (i) | | (j) | | (k) |
|  | | Chart, shape  Description automatically generated | | Shape, arrow  Description automatically generated | |  | |
|  | | (l) | | (m) | |  | |
| Figure 4: Progression of the probability map in the simulator. A dot is surrounded by an irregular obstruction (a) shows the least bins with at least some bins with a probable obstruction data. The dot keeps on moving at known fixed location and the numbers of bins keep on increasing through (b), (c) … (m) where almost all of the topolgy has been discretely mapped. | | | | | | | |

# Implementation

Figure 1 shows a proof-of-concept kind of model for such a positioning device based on the Texas Instruments VL53l0X ToF range finders. Numerous ToF sensors, are mounted on a flexible PCB, cut in a CAD driven layout. This way, the PCB cab be folded to form a sphere with the microntroller and battery inside a 3D printed mechanical base. As a proof of concept, the device will not use any triangulation aid and will try to transmit the perceived topography using thin wire. Alternatively, it can use a TCP/UDP wireless communication channel in case an SoC controller such as ESP32s is used as the primary device.

|  |  |
| --- | --- |
| A picture containing sky  Description automatically generated |  |
| (a) | (b) |
| A picture containing outdoor, yellow, chain  Description automatically generated |  |
| (c) |  |
| 1. Flexible PCB with multiple VL53l0X sensors soldered. (b) The two parts of the spherical chassis on top which the flexible PCB (a) can be assembled. (c) | |

# Proof-of-concept to actual device

The problem is as hard to be implemented as easy it sounds like in the above explanation. There are numerous challenges which will be faced to practically engineer the device. To name some, here is a short list of problems and there possible solutions.

## The ToF Sensor

VL53l0X, works on a patented technology yet still has poor characteristics, the most significant of which are a) low sample throughput, b) little tolerance to optical noise c) high uncertainties and standard deviations. It can surely be used to develop a proof of concept with low spatial resolution and slowly moving targets but probably not in an actual device.

Instead of using a general purpose microcontroller, a custom IC can be developed which uses vernier interpolation or long tapped delay lines to measure the time of flight. The circuit can easily be tested on an FPGA with a low cost FPGA before being deployed to an IC.

## The complexity of the Algorithm

To keep up the track in a bigger environment such as an office space with multiple rooms and long corridors the dot not only needs to implement a way to offload the unused portions of the map sitting on its RAM onto a more efficient form such as an SD card. In presence of moving objects, and some of the sensor continuously blocked by the carrier, there must also be an algorithm that can analyze the constructed maps to filter out moving or blocking objects.

# Possible applications

* Integration in home appliances which tend to get misplaced frequently. The constructed map can be transmitted wirelessly to the user’s cellphone/smart TV pinpointing the last known position of the dot.
* While mounted on a robotic arm or a UGV or a UAV, the dot can be used to created 3D maps of inaccessible locations like caves and tunnels, while requiring extremely small computing power.