Multiple Device Indoors Localization System

Filipe Miguel Marques Tavares, Student, MEIC-T, Instituto Superior Técnico - Taguspark, Universidade de Lisboa, Porto Salvo, Portugal, filipe.tavares@tecnico.ulisboa.pt

Abstract—With the increasing demand for interactive places and with the rise of augmented reality applications, the accurate localization of a device in the environment has become a big issue for the app developers, specially indoors, due to the fact that most localization technologies do not work that well in closed environments (such as GPS), and the solutions for indoors environments are prohibitively expensive, and most of the times require specific equipment in order to work. This project aims to study some of the existing technologies and solutions that try to solve this problem. Our solution tries to implement a cheap and reliable indoors positioning system with a high level of accuracy (<20cm error). This solution uses a Radio based technology (Bluetooth/WiFi), and Image based technology (Marker Tracking) and a Movement based technology (Inertial Sensors), which through sensor fusion using Kalman filter, tries to locate the device in a 2D space. It was chosen to use an Android device for two main reasons: first it already has all the components we need to perform the tracking (camera, Wifi and inertial sensors), and second because it is a fairly common and cheap device, meaning that almost anyone could use one. Some experiments where made to ensure each component works as it should and that each as the desired accuracy. Lastly we made an experiment with the full setup on a room. The first results were unsatisfactory due to sensor noise. On a later experiment we only used the markers and the results were much better, in terms of accuracy, but the tracking in real time was sacrificed.

Keywords—Indoors Localization; Marker Tracking; Smart Places; Mobile Localization; Inertial Navigation Unit; Radio Localization; High Accuracy Localization, Augmented Reality, Internet of Things.

I. INTRODUCTION

Nowadays when someone thinks of localization the Global Positioning System (GPS) is almost immediately mentioned, as it is the most common means of localization for the average person. Almost everyone has some sort of device that has GPS, be it a smartphone, a tracking device or a GPS map. However, the GPS can only take you as far the entrance of your destination. Once inside of a building it stops working, as the signal is too weak to penetrate the materials of the walls of the building. Also there is a certain level of inaccuracy associated with the GPS in the order of several meters which would make it hard to track someone or something inside a building accurately.

A. Motivation

With the increasing popularity and growth of domotics and interactive places with or without Augmented Reality (AR), the search for a cheap and reliable means of localization for an indoor environment has also increased. Giving a device the ability to accurately locate itself in a room would significantly increase its interactivity. For instances we could have restaurants where we could check which tables are free, and put our orders, for each individual place. Other good use would be in a museum, where we could check points of interest, and even have some AR to enhance the experience of the visitor.

With the increasing growth of the mobile industry the devices are getting cheaper and have more sensors that can be used for tracking purposes. For instance, almost every smartphone from the last 3-4 years has the required hardware to be able to form a somewhat high accuracy tracking, as for example the Inertial Navigation Unit (INU) which gives us the three dimensional acceleration of the device.

B. Goal

The goal of this work is to demonstrate that its possible to create and develop a low cost (if not free) indoor localization application with high accuracy (<20cm error). We achieved these results by using already widely spread technologies meaning that most of them are already embedded in most of our daily used devices such as the smartphone, therefore there is no need to purchase any third party devices.

The project uses one Android Smartphones (Samsung Galaxy S4) that serve as the device that we want to track in the environment.

C. Contribution

The main contribution of this work is to prove that it is possible to develop a extremely cheap and accurate positioning system, with hardware already in your pocket. This way any establishment would be able to implement their own system in order to enhance the costumers experience.

D. Document Overview

In Section II we are going to study some of the techniques and technologies used in indoors localizations, starting with Radio Techniques and Technologies (Subsection II-A) which talks about some technologies that use radio, what are their problems and what techniques are used to enhance their precision, then Image Techniques and Technologies (Subsection II-B) and finally some Other techniques (Subsection II-C) that do not fall in any of the previous categories, such as the INU. In Section III we are going to discuss which technologies where chosen and why (Subsection III-A), then we are going to address the inner components of the solution, how they are made and how they interact with each other (Subsection III-B). In Section IV we are going to discuss which metrics we are going to evaluate and why we have chosen them, after we made and how they interact with each other (Subsection III-B).
results (Subsection IV-A). At last, in Section V we will talk about what was studied, what happened, what could have gone better and what were our conclusions (Subsection ??).

II. STATE OF THE ART

There are three two main categories of technologies and techniques for localization: Radio and Image. There is other category which includes other technologies which are mostly movement based. Radio-based technologies rely on radio signals to pinpoint the object we are trying to find. This is done by measuring the strength of the signal, the time of flight of the signal, the angle of arrival, or a combination of them. These solutions require, in most cases, a Line Of Sight (LOS) for additional precision. Also the radio signal suffers from heavy reflections and attenuations from the various materials (walls, windows, etc) that exist indoors.

Image-based technologies, on the other hand, do not suffer from Radio-Based problems, however there are other issues with these technologies. However some may suffer from occlusion of the image or even image deformations which might lead to some bad readings.

Movement-based technologies suffer from none of the problems above mentioned, but they require integration of the data to get a position from the device movement. Also as it has no way of actually knowing where the device is through an absolute value, the error gets bigger though time if its not dealt with.

A. Radio Techniques and Technologies

To better understand how the positioning via radio works, we will present Position Estimation Techniques. These techniques are very similar through all radio technologies and we will discuss the following:

1) Received Signal Strength (RSSI): RSSI estimates a correspondence between the signal power and the distance to the signal source. Hence, most often it requires calibration to set up the system based on several factors, such as the signal source, its position and its reach.

2) Angle of Arrival (AOA): AOA calculates the angle at which the signal was received. This is only possible if we have several antennas (antenna array) measuring the difference in time that signal arrives at each antenna. Then it uses simple geometric relations to calculate the position of the emitter.

3) Time of Arrival (TOA): TOA uses the absolute time which took the signal to travel from the transmitter to the receiver. The estimated time is then translated to distance based on a known signal velocity and the environment features (humidity, temperature, density, etc)[1].

4) Time Difference of Arrival (TDOA): TDOA is much similar to TOA, however TDOA uses the interval of time that the signal takes to arrive at the several stations, meaning that it requires more than one receiver to be used. Although its generally more accurate than TOA it is also more expensive due to the increasing number of stations it requires and the underlying synchronization they need. TDOA suffers from the same ills than TOA such as the speed at which the signal travel on different environments and temperatures.

The problem with TDOA, TOA, AOA and all time based techniques is that they require a LOS to increase the level of accuracy [2], [1].

Often in radio positioning Techniques there is the need to calculate the position of the objects from several beacons. Such techniques are used to combine the information from several beacons and computing objects position by using simple geometric relations. The most common techniques are:

5) Triangulation: Triangulation makes use of trigonometry angles to calculate the position of a point. By knowing the position of two points we are able to calculate a distant point, simply by measuring the angles between the target point and the known points.

6) Trilateration: Trilateration is somewhat similar to triangulation, but instead of using angles as the main measurement, it uses distances by measuring the TDOA and the RSSI of the beacon. Figure 1 demonstrates how beacons are used to calculate the position of a device in space.

Radio technologies for indoors environments have been used and studied for many years with several goals in mind, such as security, tracking, health-care and monitoring [3]. In this project several of those where studied to compare and better understand which would be better to implement in a tracking system at a low cost. The main technologies studied were:

Each of these technologies has its advantages and disadvantages. We are going to discuss some of them.

7) Ultra Wide Band (UWB): UWB uses high frequency radio signals. Large Bandwidths bring some major advantages to positioning applications[4], [5], [1], [2]: Penetration through obstacles due to lower fading effects; Improved multiple access; Lower probability to interference; Increased accuracy on position estimation; Low power transmission; However having a high bandwidth reduce the distances covered by the signal. UWB Solutions frequently use RSSI, TDOA/TOA and AOA as position estimation techniques, but sometimes other techniques are also used to further improve its accuracy[6], [5].

![Fig. 1. Beacon Trilateration](image-url)
8) Radio Frequency Identification (RFID): RFID uses radio waves to identify tags, which have their unique ID. These tags can have short, medium and long ranges. Most solutions that use RFID make heavy use of RSSI, making it extremely prone to signal reflection and attenuations[7], [8].

9) ZigBee: ZigBee is defined as a "standards-based wireless technology designed to address the unique needs of low-cost, low-power wireless sensor and control networks. ZigBee needs little power to operate". Its setup is very similar to a WLAN network. Most of ZigBee solutions make use of RSSI measurement, with some RSSI smoothing algorithms, as the main localization technique, because it is preferred in a No Line Of Sight scenario. There are also solutions that use ZigBee with other technologies in an effort to further increase the accuracy of the system[9].

10) Bluetooth and Bluetooth Low Energy (BTLE): Bluetooth was developed as an alternative to cable technology but with properties such as low cost and high availability, it could also be used for the development of several indoor position position systems [10]. However it would consume too much power for portable devices. BTLE made this power consumption much lower, enabling much smaller and portable devices to be able to advertise themselves to other devices [11]. One of the greatest advantages in using Bluetooth is that it is already a standard for wireless communications. Almost every mobile device from the past 10 years is Bluetooth enabled, allowing them to be used as possible beacons [10].

B. Image Techniques and Technologies

Images has been used for localization for a long time especially in automated vehicles because they can achieve higher accuracy levels[12], [13], [14].

The main techniques used for these type of technologies are:

1) Simultaneous Localization And Mapping (SLAM): SLAM is define as a technique used by robots and autonomous vehicles to build a map while simultaneously localizing themselves in said map. Most solutions involving SLAM either use a LIDAR (Laser Radar) or a infrared camera to create the map. However the LIDAR is extremely expensive making quite inaccessible. The infrared camera suffers from interferences from infrared emitter, such as the sun.

2) Marker Tracking: Marker Tracking is achieved by placing an artificial marker in the environment and then have a camera read it. These markers have specific patterns which translates to their IDs. After the marker has been recognized by the software, it processes the image according to the position, scale and orientation of the marker, as shown in Figure 2.

3) Marker-less Tracking: Marker-less tracking tries to find some points of interest in the image, such as intersections and places where there is high contrasts, and tries to track their movement. This makes the use of markers unnecessary. However it requires more processing power[15].

---

1http://www.zigbee.org/About/AboutTechnology/ZigBeeTechnology.aspx
sent to the marker detection algorithm, that raw image is then converted to gray scale in order to better find boarders in the picture. Then, after the marker has been identified, it is projected in the camera space so that it may be possible to calculate its position and rotation. With all that data finally available we are able to draw anything on the marker position. The algorithm described, as most of the marker tracking algorithms are, is for Augmented Reality, but with some changes to the process we are able to use it as a localization technique for the camera and not the marker, and with great accuracy (within 20cm).

Fig. 3. How Marker Tracking Works (http://www.vrworld.com/2009/10/26/zombies-hit-nvidiae28099s-tegra-thanks-to-augmented-reality/)

2) Radio Technology - Bluetooth / WiFi: The Radio Technology proposed was Bluetooth Low Energy which had been advertised for having a high accuracy within a few centimeters, and most of the newer smartphones already have Bluetooth Low Energy capabilities it seemed the perfect choice. After we acquired some beacons, and were preparing to begin testing them, we came across a problem, the Android version that was current available for our devices was not compatible with the beacons, thus it would not be possible to detect and measure the signal from them. As such we would have to use, for the time being, the WiFi network as a place holder for the Bluetooth. The technique used would remain the same, as it is possible to use it with the WiFi Signal, however this would surely carry a greater error, thus lower accuracy.

3) Aiding Technologies: Since most smartphones already have an Inertial Navigation Unit (INU) it was decided to use it enhance our tracking algorithm. The INU is used to measure the movement of the object, and that is exactly what we are aiming for by using it, to be able to predict where the device is going to be before the next measurement of any of the other technologies in use, however it is not an accurate measurement but a merely prediction in order to try to reduce the errors from those other technologies.

B. Implementation

There are 5 main components in this implementation as shown in Figure 4:

Fig. 4. Implementation of the solution components, including the server, tracker and client.

1) Clients/Devices: As already stated, this project is using Android devices, mainly smartphones, as the devices to track, as such it has a limited amount of energy and reduced processing power, so the objective is to make it last for as long as possible with the least processing as possible. Two options were thought, both relying on server. The first option would be to share the data with the server, thus making the load of the server smaller but having the device do some of the required processing. Second option would be to let the server do the whole processing bit while the device would only be responsible for gathering the data and sending it to the server, this might overload the server, but the device would last longer. The client is responsible for starting the connections with the server, as well as maintaining it as long as the server is alive or willingly terminates it. Two connections are to maintain with the server, one for sending sensory data and another to send the image.

2) Server: The server application is deployed in a Personal Computer, with a i7-3630QM (8 processors) 2.40GHz, 8Gb of RAM, running Windows OS. The server is responsible for managing the creation of incoming connection and for the management of the client objects, it is also responsible for sending the information to any displaying program that is already connected. All the data and objects are stored in volatile memory for two main reasons, faster access, and not being necessary to keep a state for the simple fact that if the server crashes we do not want to keep the previous positions, because in the mean time the server takes to restart, the devices will not be in the same place, so the tracking has to be done all over again anyway.

3) Client Object: The Client Object acts as a representation of the physical device to the server. It receives the data from the device and stores all data of the sensors, and localization. In this way, we have access to both raw and processed data at all times, however it does not keep track of previous values in order not to overload the memory with too much data. When
the Client Objects is initialized it also creates a Tracker, with the purpose of delegating the processing of all the data and encapsulating the localization algorithms in one object. This way if we want a different algorithm to be used, we just have to initialize a different tracker.

4) Tracker: The Tracker has the core algorithms of the project. This component as the ability to process and merge all the data from all the sensors and camera. Every Client Object has its own Tracker and, this enables a greater flexibility in individual configurations.

There are 3 main steps, all of which are independent from each other:

- Maker Tracking - the marker tracking technology is the video technology of this project, and one of the two absolute localization technologies, meaning that the values that we receive from this algorithm is the direct value (excluding errors) of our position in space. To implement this algorithm a open source library was chosen, ArUco\(^2\), which is a C++ open source library for augmented reality application, based on OpenCV. ArUco detection algorithm follows several steps:
  - First it finds the contours of the image by using Adaptive Thresholding, in this step every contour of the image is found, including markers and visual garbage.
  - Then it removes borders with few points.
  - Contours that do not appear to be similar to 4 vertex polygons (rectangles) are discarded.
  - Rectangles that have narrow sides are also removed.
  - At last the inside of the marker needs to be verified, but first the perspective projection needs to be removed so that it may get a frontal view of the inside of the marker, then it checks the code and verifies if the marker is valid and its ID;
  - After the marker is verified and is valid, if the camera parameters are provided, then the extrinsics of the marker is calculated;

At the end of these steps, we get the extrinsic matrix of any marker that is in the image. The extrinsic matrix of the marker is represented as a rotation vector expressed as the Rodrigues Formula \((x, y, z, \theta)\) and a translation vector \((x, y, z)\). This gives us the position of the marker in relation to the camera space, but we need to get the camera in the marker space. To achieve this it is required that we compute the inverse of the extrinsic matrix of the marker. Being \(R_{33}\) the destination 3x3 matrix for the rotation matrix, we obtain the desired matrix:

\[
R_{33} = \begin{bmatrix}
R_{00} & R_{01} & R_{02} \\
R_{10} & R_{11} & R_{12} \\
R_{20} & R_{21} & R_{22}
\end{bmatrix}
\]

and we already have the translation vector, \(T_{vec}\), by transposing that vector we get a 3x1 matrix:

\[
T_{vec}^t = \begin{bmatrix}
T_{00} \\
T_{10} \\
T_{20}
\end{bmatrix}
\]

at last we can compose a 4x4 transform matrix by adding the rotation matrix with a the translation vector:

\[
T_{Marker} = \begin{bmatrix}
R_{00} & R_{01} & R_{02} & T_{00} \\
R_{10} & R_{11} & R_{12} & T_{10} \\
R_{20} & R_{21} & R_{22} & T_{20} \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

As already stated this is the representation of the marker in the camera space, now by computing the inverse of this matrix we get the representation of the camera in the marker space:

\[
T_{Marker}^{-1} = \begin{bmatrix}
R_{00} & R_{01} & R_{02} & T_{00} \\
R_{10} & R_{11} & R_{12} & T_{10} \\
R_{20} & R_{21} & R_{22} & T_{20} \\
0 & 0 & 0 & 1
\end{bmatrix}^{-1} = T_{Camera}
\]

The data is now ready to be fed to the Kalman Filter where it will be fused with the remaining data from other sensors.

- Sensory Data Processing - The sensory data is the information received from the INU. Is composed of data from the gyroscope, which gives the angular velocity of the device, and the data from the accelerometer, which is related to the forces exerted on the device, both in 3D space. This type of data is very susceptible to error, especially due to the environment and the way the device is handled. To try to mitigate these, errors two solutions were proposed, the first is simply apply a low-pass filter to try and smooth the acceleration curve, this will make

\[\text{Low Pass Filter} = \begin{bmatrix}
(1 - k) & k \\
k & (1 - k)
\end{bmatrix}\]

\[\text{Low Pass Filter} = \begin{bmatrix}
0.7 & 0.3 \\
0.3 & 0.7
\end{bmatrix}\]

\[\text{Low Pass Filter} = \begin{bmatrix}
0.5 & 0.5 \\
0.5 & 0.5
\end{bmatrix}\]
Kalman Filter - The Kalman Filter is the reference algorithm for fusing sensor data, because it allows the use of measures over time from many different sensors, as long as the measures are supported by the system model. The filter then tries the calculate the real position of the device by using a weighted average with more weight on variables that have a higher certainty. The filter can be used in real time and only needs the previous known state of the system, and the current measures in order to calculate its current state.

The physical model that represents the device are simply the laws of motion:

\[
\begin{align*}
    x_{k+1} &= x_k + v(x_k)\cos(\theta_k)\Delta t + \frac{1}{2}a(x_k)\cos(\theta_k)\Delta t^2 \\
    y_{k+1} &= y_k + v(y_k)\sin(\theta_k)\Delta t + \frac{1}{2}a(y_k)\sin(\theta_k)\Delta t^2 \\
    \dot{x}_{k+1} &= v(x_k)\cos(\theta_k) + a(v(x_k))\cos(\theta_k)\Delta t \\
    \dot{y}_{k+1} &= v(y_k)\sin(\theta_k) + a(v(y_k))\sin(\theta_k)\Delta t \\
    \theta_{k+1} &= \theta_k + \dot{\theta}(\theta_k)\Delta t
\end{align*}
\]

Before showing the Kalman matrices, it is worth noting that it was decided to break the Kalman in three different Kalman filters, one for each coordinate and angle. Thus we get \(K_{x}, K_{y}, K_{\theta}\), where \(K_{x}\) means Kalman filter of \(x\). The reasoning behind this is simply because it is easier to verify a 3x3 matrix instead of a 8x8 matrix. Keeping that in mind, converting these equations to the Kalman matrices to work with each Kalman we get:

\[
A_x = \begin{bmatrix}
    1 & \cos(\theta_k)\Delta t & \frac{1}{2}\cos(\theta_k)\Delta t^2 \\
    0 & 1 & \cos(\theta_k)\Delta t \\
    0 & 0 & 1
\end{bmatrix}
\]

\[
A_y = \begin{bmatrix}
    0 & \sin(\theta_k)\Delta t & \frac{1}{2}\sin(\theta_k)\Delta t^2 \\
    0 & 1 & \sin(\theta_k)\Delta t \\
    0 & 0 & 1
\end{bmatrix}
\]

\[
A_{\theta} = \begin{bmatrix}
    0 & 0 & 0 \\
    0 & 0 & 0 \\
    1 & 0 & 0
\end{bmatrix}
\]

Then we have the measurement matrix which gathers all the measurements from each sensor:

\[
H_x = \begin{bmatrix}
    \text{Marker}_x & 0 & 0 \\
    0 & \text{WiFi}_x & 0 \\
    0 & 0 & \text{Accel}_x
\end{bmatrix}
\]

\[
H_y = \begin{bmatrix}
    \text{Marker}_y & 0 & 0 \\
    0 & \text{WiFi}_y & 0 \\
    0 & 0 & \text{Accel}_y
\end{bmatrix}
\]

\[
H_{\theta} = \begin{bmatrix}
    \text{Marker}_\theta & 0 & 0 \\
    0 & \text{WiFi}_\theta & 0 \\
    0 & 0 & \text{Gyro}_\theta
\end{bmatrix}
\]

As we can see there are no measurements for the velocity of the device, because the measure is only related to the device absolute position (Markers and WiFi) and the acceleration from the gyroscope. For the angle, we do measure both the absolute angle from the marker, and the angular velocity from the accelerometer. When the Marker is not detected, the position of the marker in the matrix \(H_x\) should be 0, so that the Kalman does not take in to account that row of the matrix. This is required, because even if you just set the measured results of the marker to 0, that would make the kalman think the device is at (0,0), and as the marker has a greater weight, most of the time it would be trying to get the device to (0,0) position.

5) Displays: The only function of the display components is to visualize or use the localization data of the devices. These programs are not contemplated by this project, as this is just a generic idea for a possible output program.

IV. EVALUATION

There is only one main requirement, which is the accuracy of the system. The accuracy goal is to be able to pin point the device in the room with high accuracy (high is defined in this work as <10cm). As such we want to test if our solution is highly accurate in a instantaneous measurement as well as in a continuous measurement.

A. Testing environments

1) Inertial Navigation Unit: The INU measures the forces exerted on the device, in this case the acceleration. However if we only use the raw data, the results will be flooded with error. The INU has a major role in this solution, in order to track the position of the device in the absence of markers or radio measurements. This test was made to demonstrate how much error there is in just tracking the device through sensory gathering, especial when the requirement is high precision. The test uses a simple A4 paper sheet (21x30cm) and the objective is to track the device inside that paper sheet by moving it from...
the center to the edges and back to the center again. The used procedure is as follows (following 7):

- First we position the device in the center of the sheet (Position 0);
- Then we move the device to the edge of the sheet (Position 1) and we measure its position;
- After, we move the device back to the center (Position 0) and measure again. And so forth;
- The sequence should be 0 ⇒ 1 ⇒ 0 ⇒ 2 ⇒ 0 ⇒ 3 ⇒ 0 ⇒ 4 ⇒ 0, taking measures at each point.

Fig. 7. INU Test Path

If the INU was perfect, it would be expected to see many points near the predefined path, but the results do not show that. As it is possible to see in Figure IV-A1, there are very few points near the path and they drift away from it very fast.

Applying Kalman Filtering slightly improves the results, as seen in Figure IV-A1 we do have slightly more points near the path, and closer to it as well, but still there is too much drift.

Adding the movement window, the results are improved drastically, as show in Figure IV-A1. The improvement is such that not a single measure goes beyond an area of 30cm from the center of the sheet. Also this brings almost no overhead to the processing time of the position, only 0.3ms more on avg than the other solutions, thus making this our implementation for the INU.

2) WiFi:

The WiFi test was made especially to assess its accuracy. No algorithm was implemented because it is an absolute value that we read, there is no room to try and figure out the error like with the INU. One common way to get a better accuracy would be by having many WiFi emitters, and use a triangulation algorithm. However there was only one available at the time the tests where made. The test was quite simple:

- Place the WiFi emitter at a known location (if possible with a clear line of sight);
- Place the device at a known distance from the emitter and measure the distance through the known formula;
- Repeat step 2 at different distances to measure the error.

The WiFi emitter used was a TP-Link and the formula for the distance was taken from TP-Link website (http://www.tplink.com/be/support/calculator/#1):

\[
\text{Distance (cm)} = 10 \times (\text{dB} - 27.55 - 20 \log_{10} (\text{frequency})) / 20
\]

here 27.55 is the constant for having the frequency measured in MHz and the distance in cm.

The results obtained show that the error is within 120cm as shown in Figure 11. The black line represents where the device was, while the blue circle represents an average range of the distances that were measured (while the device was resting on the same place). As it can be seen, there is a big error while measuring with the WiFi, almost as big as the GPS, even though there was a clear line of sight from the device to the emitter. This error is even greater if there is an obstacle in the way.

3) Markers:

There where two tests made for this component in order to assert its accuracy: Distance Measurement and World Positioning.

Distance Measurement is intended to measure the relation between marker distance and accuracy. The marker used should be the marker defined as the origin (coordinate 0,0). This way, every place the marker is set, is going to be considered the world origin. The Test procedures are:

- Define the Origin Marker (marker that represents the origin in the World reference);
Fig. 10. INU Kalman and Window Results

Fig. 11. Wifi Result

- Place the device 1 meter away from the Marker at 0 degrees;
- Leave the Device in place and measure the distance obtained, as well as the fluctuations on that distance (The frame are always slightly different due to light variations);
- Add one extra meter to the distance and repeat from Step 2 until the marker is no longer detected.

<table>
<thead>
<tr>
<th>Distance (cm)</th>
<th>1 meter</th>
<th>2 meters</th>
<th>3 meters</th>
</tr>
</thead>
<tbody>
<tr>
<td>X (cm)</td>
<td>0.15 ~ 0.55</td>
<td>-12 ~ 2</td>
<td>-10 ~ 20</td>
</tr>
<tr>
<td>Y (cm)</td>
<td>94.4 ~ 98.6</td>
<td>190 ~ 192</td>
<td>296 ~ 290</td>
</tr>
<tr>
<td>Angle (°)</td>
<td>-1 ~ 0</td>
<td>-4.1 ~ 2</td>
<td>-1 ~ 3</td>
</tr>
</tbody>
</table>

The results presented by Table I show that this technology is quite precise. As we can see the highest error (distance wise) is about 10cm at 3m in this setup. However there are some problems with the positions obtained. In a perfect test these should be somewhere near X = 0cm, Y = Distance, Angle = 0°. At the 1 meter mark, the results obtained are extremely accurate, less then 1cm error and the angle is within 1 degree, this is as good as it gets for tracking an object. But as the distance keeps increasing the errors become bigger. At the 2 meters mark, the X coordinates already have a 14cm error and the angle as a 2 degree error. At 3 meters mark, the error becomes even greater, 30cm in the X coordinates and 4 degree error in the Angle.

Thus the conclusion we can obtain from this test is that the error becomes greater with the distance, as expected. There are two main reasons for this increasing error, the marker size on the image and the image blur.

The World positioning enable us to test how precise the markers are at calculating the position of the device in the world referential. The markers themselves provide the position in 3D coordinates but we only want the position in 2D coordinates, thus we project the “up” coordinate to 0 (which in this case, due to the handling of device, it is the Y coordinate).

As the previous test, and for the same reasons, we use the origin marker, however as the device is no longer right in front of the marker, the position at which the device is might suffer some error due to human error. The same is true for the space at which we want to locate the device. Also its worth noting that this test was made in a very small space, which in this case is a table. It was made like so to show how accurate the technology can be in small spaces, as most of the others technologies mentioned before are not able to do so.

The Tests went as following:
- Define the Origin Marker;
- Place the device at a pre-measured spot.
- Measure the device coordinates and angle;
- Repeat from Step 2 until all measurements have been made.

As we can see in Figure 12, the results are extremely accurate. At the Position 1 (Figure 12a) the value should be \( x = -58 \text{cm} \) and \( y = -55\text{cm} \), and the values measured in Figure 12b are \( x = -52.58 \) and \( y = -54.82 \), this means that we have an error nearing the 5cm, which is extremely good. In all the measurements in this test (< 1m distance), the measurements remained under the 10cm error mark. Also the blur problem was not a major one as long as there are no rapid movements, this is due to the closeness with the marker.

4) Full Setup Test: This test was designed to test all the components working together. Figure 13 depicts the test layout. The Room is 4.25 meters wide and 4 meters long. There are 4 markers (represented by the yellow rectangles), one in each wall. There are 5 pre-measured marks which are represented by the white squares (also they represent the corners of the testing path).

The test consists in following the predetermined path. Starting at position 1, you should go through every position, and then back to position 4, facing the east wall. Since the point of the test is to check if we are able to track in real time the movement of the device with some degree of accuracy, there is not much more to it.

The results obtained, by using all the sensors, where very unsatisfactory. Figures 14 a test made with this setup, and as we can see the results are very unstable, it is barely possible
to see which was the followed path. However we can see that the device is stopped near position 4, which is the end of the test, with an error within 10 to 20cm. These results bring us to the conclusion that the noise must be coming from the INU, since this only works when the camera is not able to track the markers. So, Some other tests were made.

This time the INU was shut down. And as it is possible to see from Figures 15 the results seem to be much better, and in fact, the error measured is almost always within 30cm error except for position 4, where sometimes there is a heavy wobble from the tracking algorithm due to the distance from the markers. However the results still remain partially unsatisfactory. This is because, the tracking is not actually done in real time, in fact the tracking is only done when the marker is captured by the camera, meaning that most of the time the device appears to be stopped. When the camera captures the mark, the Kalman filter is activated and tries to interpolate the last known position (last captured marker) with the new position.

The reason why both tests performed so poorly is mainly the same, image blur. In the case of the Full setup, when there is enough blur for the camera to not be able to track the markers, the INU starts working, but the same reason that causes the image to blur also causes the INU to read noise, which is the shaking of the device, mainly due to the steps and/or the handling of the device, and the longer the markers take to detect, the higher the error from the measurements, due to its cumulative properties. Also the Kalman Filter tries to interpolate the position of the device, by fusing the information of both the sensors, so if the INU drifts 1 meter in the X axis, but not on the Y axis, and the marker gets tracked again 1 meter ahead in the Y axis, the results are a erratic shaking until the markers get tracked consistently. For the Marker Only setup, there is no shaking, but the device is not tracked either, as explained before. As we can see in Figures 16 steps or turning the device, makes the blur serious enough for the algorithm not to be able to track the markers. Even with reduced exposure, the blur remains, and it is still to strong, this gets worse as the distance grows.

V. CONCLUSION

This work set out to find if it was possible to use cheap and commonly available technologies and techniques to locate a device indoors with a high degree of accuracy. As such, some technologies where studied in order to understand their strengths and weaknesses. During this study, it was recognized that using visual techniques, radio techniques and movement based techniques together would be better because they have strengths that complement each others weaknesses. Also it was decided that a smart-phone was going to be the device used to
perform the tracking, because it already has all the necessary components.

With the appearance of IoT, some use cases come to mind that might make good use of this technology. For example in a restaurant, where the application could lead you to your designated seat, and then you could make your order, and it would know in which place on the table you would be.

The solution proposed would use sensor fusing. The video technique would be marker tracking. The inertial sensor, would keep track of the motion of the device, finally the Radio technology would complement these two, by triangulation its position.

The tests showed that the Inertial Sensors (INU) was extremely noisy and any vibration would make the measurements go off by meters. As such a low pass filter was used to try and reduce its noise, however it was not enough. A threshold based algorithm was implemented, that measured only accelerations that would stand out. The threshold algorithm proved to be better.

As for the radio solution, originally was set to use Bluetooth as the chosen technology, but the device OS was not compatible with the latest Bluetooth technology (BTLE), so it shifted to WiFi, as a placeholder. The tests done with WiFi showed that the error was beyond the acceptable level.

The test for the video solution with marker tracking, showed that it was possible to track the device with very high accuracy (<10cm). However this was only true at close range.

The idea behind the full setup would be that when the camera was not tracking markers or in between frames, the inertial sensor would get activated, and try to deduce the position of the device through its movement. However the tests with the full setup on the room, were very disappointing. There was just too much noise from the INU mostly due to steps. This again, because of the image blur, means that the camera cannot track, thus the INU will stay on more time than originally thought.

Another test was made just with Markers. The obtained results where much better in terms of accuracy. However, image blur would again cause problems, making the localization not being tracked in real time. Every time the image was blurred the tracking would stop, and the device would appear as if it was stopped, but when the camera tracked a marker, the Kalman Filter would try to calculate the traveled path.

With these results we conclude that, with this architecture and components, only the markers have been proven to really work even better than expected. It is possible to track the devices in very small spaces (<100cm) with very high accuracy (<10cm error), and in big rooms, as long as there are many markers around, but not actually in real time.

Although it might not be perfect yet, its accessibility and the development is in reach of anyone who owns any type of smartphone, meaning that there are billions of potential developers around the world. This technology would enable us to have devices that could have augmented reality with a high level of precision, amongst other applications. The possibilities can be almost limitless.

**REFERENCES**


