

# Estudio de la calidad de la información de la geolocalización proporcionada por un móvil

David Sánchez Pita

**Resumen**—La motivación de este proyecto es otorgar una mejora en las aplicaciones de geolocalización, principalmente para las personas que tienen visibilidad limitada, ya que estos no pueden detectar si el GPS (*Global Positioning System*) les está otorgando una geolocalización errónea o si las indicaciones que les indica la aplicación son correctas. El GPS fue creado para uso militar, y no para uso civil. Por este motivo, la geolocalización proporcionada por los satélites contiene un error. En este proyecto se estudiará la geolocalización proporcionada por un *smartphone* y se utilizará el Filtro de Kalman para estudiar si se puede mejorar esta geolocalización. Finalmente, se consigue cumplir el objetivo y se logra reducir el *accuracy* en ciertos recorridos.

**Palabras clave**—geolocalización en *smartphones*, Filtro de Kalman en *smartphones*, Filtro de Kalman en Android, mejorar geolocalización *smartphone*

**Abstract**—The motivation of this project is to implement an improved application which provides location. It is mainly intended to be used by people with reduced sight, who can not detect if the GPS (*Global Positioning System*) is providing a bad location or if it is providing wrong suggestions. The GPS was created for military use and not for civil use. Due to this, the location provided by the satellite has an error. In this project we will study the location provided by the satellites in smartphones and we will use the Kalman Filter to study if we can improve that location. Finally, the objective is achieved and the accuracy is reduced in certain travels.

**Index Terms**—smartphones' geolocation, Kalman Filter for smartphones, Android's Kalman Filter

## 1 INTRODUCCIÓN

DESDE la aparición de los *smartphones*, la geolocalización ha crecido exponencialmente debido a los beneficios que aporta a la sociedad, la cual es cada vez más tecnológica. Entre estos beneficios encontramos:

- Conocer en cualquier momento donde estamos.
- Saber llegar a nuestro destino en cuestión de segundos.

Los satélites se encargan de obtener esta geolocalización a través de la trilateración [Imagen 1] [1].

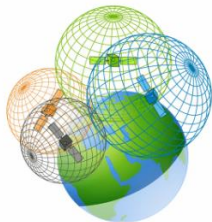


Imagen 1 – Trilateración de cuatro satélites

La trilateración calcula la distancia entre nuestro dispositivo y tres satélites distintos. La diferencia con la triangulación [2] es que este mide los ángulos, mientras que la trilateración mide las distancias. Es frecuente encontrar información que habla sobre triangulación mediante GPS, pero el término exacto es trilateración.

A partir de estos cálculos, obtenemos dos posibles geolocalizaciones, una de las cuales es improbable, por lo tanto, será descartada. En el caso de que obtengamos dos geolocalizaciones que probablemente sean correctas, se puede hacer el cálculo con un cuarto satélite para obtener, finalmente, la geolocalización correcta.

Sin embargo, la geolocalización que nos otorgan los satélites contiene un error, ya que estos fueron creados para uso militar.

Cada satélite emite un mensaje de navegación a 50 bits por segundo en la frecuencia transportadora de microondas. Estos constan de tres señales: señal L1, emitida a 1575,42 MHz; señal L2, emitida a 1227,6 MHz; y señal L5, emitida a 1176,45 MHz. La señal L2 es de uso militar, mientras que la señal L1 y la señal L5 son de uso civil[3].

- E-mail de contacto: david.sanchezpi@e-campus.uab.cat
- Mención realizada: Computación
- Trabajo tutorizado por: Marta Prim (Departamento de Microelectrónica y Sistemas Electrónicos)
- Curso 2018/19

El uso civil apareció posteriormente al uso militar, y solo puedes obtener una geolocalización más precisa pagando por el servicio. Aún así, la geolocalización seguirá siendo menos precisa que la otorgada para el uso militar.

Desde la aparición del GPS se ha utilizado la señal L1 emitida por los satélites para comunicarse con el receptor GPS. Esta señal proporciona dos elementos para obtener la geolocalización: el código de adquisición aproximada y los códigos de cifrado de precisión; y tiene asociado un error mínimo de aproximadamente 5 metros, el cual puede incrementar en trayectorias múltiples. Estas trayectorias aparecen cuando la comunicación entre los satélites y el receptor GPS contiene obstáculos (edificios, montañas, etc) y la señal rebota en ellos.

Actualmente, los procesadores Qualcomm Snapdragon 820 y Exynos 780 son los mejores procesadores para la obtención de la geolocalización con esta tecnología. Esto es debido a su compatibilidad con la GNSS (*Global Navigation Satellite System*) [4], en la que pueden utilizar los sistemas GPS, GLONASS, BeiDou y Galileo; y GPS, GLONASS y BeiDou, respectivamente.

En el 2018, salió al mercado una nueva tecnología llamada “frecuencia dual”. Esta tecnología capta las señales L1 y L5. La señal L5 está dividida en dos partes: el código L5 en fase (códigos de navegación) y el código cuádrfase (adquisición de datos de navegación más fácilmente). La función de la señal L5 es obtener información adicional de los satélites para reducir el error en trayectorias múltiples.

Esta tecnología se implementó por primera vez en los procesadores Qualcomm Snapdragon 845 y Kirin 980, implantados en los dispositivos Xiaomi Mi8, Xiaomi Mi8 Pro y Xiaomi Mi Mix 3; y en todas las versiones de Huawei Mate 20.

La frecuencia dual tiene dos problemas principales: solo ciertos dispositivos contienen esta tecnología, aunque en un futuro se espera que todos los dispositivos móviles la contengan; y que no todos los satélites transmiten la señal L5. A estos problemas, también hay que añadirle que Xiaomi y Huawei tuvieron problemas con la API de Android GnssStatus [5], la cual recibe los datos de los satélites. Ese problema fue solucionado posteriormente y ya se puede disfrutar de esta nueva tecnología.

La motivación del proyecto es intentar reducir el error que contiene la geolocalización a través del Filtro de Kalman, para que aquellas personas que tengan dificultades para observar la interfaz gráfica sepan exactamente donde están y poder seguir las indicaciones auditivas correctamente.

Este documento está dividido en diferentes apartados. Empezaremos explicando los objetivos de este proyecto. En el apartado 3 expone toda la información teórica necesaria

para entender cómo se obtiene una localización. En los apartados 4 y 5, la explicación teórica del Filtro de Kalman y cómo se ha implementado en la aplicación. A partir de aquí, se expondrán la metodología y planificación aplicadas al proyecto. En el apartado 7 el diseño del software y todos sus parámetros de configuración. Por último, se explicarán las variables que se han tenido en cuenta a la hora de hacer las pruebas, los resultados obtenidos con estas variables y las conclusiones respectivas. Finalmente se expondrán las conclusiones del proyecto, así como implementaciones futuras.

## 2 Objetivos del proyecto

El objetivo principal del proyecto es reducir el error que otorgan los satélites a la geolocalización. Para ello, tendremos que implementar una API (*Application Programming Interface*) [6] en Android, en la cual recibiremos la geolocalización adquirida por el receptor GPS del *smartphone*, le aplicaremos el Filtro de Kalman y, finalmente, enviaremos la geolocalización filtrada a las diferentes aplicaciones de nuestro dispositivo móvil.

Android [7] es un sistema operativo para dispositivos móviles con pantallas táctiles, basado en Kernel de Linux[8] y en otros software de código abierto[9].

## 3 TRILATERACIÓN

La trilateración es un método matemático que calcula la distancia entre un satélite y un receptor GPS.

El satélite emite una señal, utilizando la frecuencia L1, y el receptor GPS capta la señal con una hora y distancia determinadas. Con esta señal, se localiza el receptor en una ubicación con un radio posible de localización [Imagen 2].

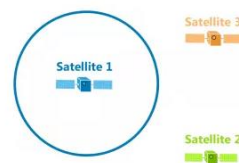


Imagen 2 -Ubicación posible con la señal del satélite 1

Al realizar este proceso tres veces con tres satélites diferentes, obtenemos la intersección entre 3 satélites. Esta intersección determina la geolocalización del receptor GPS.

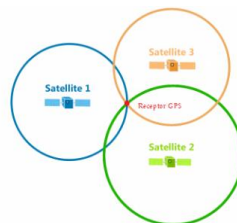


Imagen 3 – Intersección entre 3 satélites

Este proceso puede dar como resultado dos intersecciones, ya que con la distancia entre el satélite y el receptor GPS se genera una esfera [Imagen 1], una de ellas es una geolocalización improbable. Para solucionarlo, se puede utilizar un cuarto satélite y con ello obtendremos una única intersección [Imagen 1].

#### 4 FILTRO DE KALMAN

El Filtro de Kalman [10] es un algoritmo recursivo que se utiliza para obtener el estado oculto de un sistema dinámico lineal que contenga ruido blanco. Este algoritmo escoge la ganancia de Kalman óptima cuando conoce las varianzas del ruido que afecta al sistema. El Filtro de Kalman se puede usar en tiempo real utilizando únicamente las mediciones de entrada actuales, en nuestro caso, la ubicación actual.

El ruido blanco [11] es una señal aleatoria caracterizada porque sus valores no tienen una correlación en dos tiempos diferentes.

#### 5 Algoritmo del Filtro de Kalman

El algoritmo del Filtro de Kalman [Imagen 4] está compuesto por dos tipos de ecuaciones: las que relacionan la variable de estado con las variables observables (ecuaciones principales) y las que determinan la estructura temporal de las variables de estado (ecuaciones de estado).

Las estimaciones de las variables de estado se realizan basándonos en la dinámica de estas variables (dimensión temporal), así como las mediciones de las variables observables que se van obteniendo en cada instante de tiempo (dimensión transversal).

La dinámica se divide en dos pasos:

1. Etapa de predicción: estimar las variables de estado utilizando su propia dinámica.
2. Etapa de corrección: mejorar la estimación de la etapa de predicción utilizando la información de las variables observables.

Las variables del algoritmo se diferencian según su etapa: las variables de la etapa de predicción contienen el signo “-”, mientras que las variables de corrección no contienen este signo. Por otro lado, también se diferencia según su estado. El estado actual es “i” e “i-1” es el estado anterior.

Las variables utilizadas en la etapa de predicción del algoritmo son las siguientes:

- $x_i^-$ : estimación a priori del estado actual.
- $A$ : matriz de transición de estados.
- $x_{i-1}$ : vector del estado anterior.
- $B$ : modelo de control de entrada.
- $u_i$ : vector de control del estado actual.
- $P_i^-$ : matriz de covarianza de error de la estimación.

a priori.

- $Q_i$ : covarianza del ruido del proceso

Las variables utilizadas en la etapa de corrección del algoritmo son las siguientes:

- $K_i$ : ganancia de Kalman.
- $H$ : modelo de observación.
- $R$ : matriz de covarianza del ruido de proceso.
- $x_i^+$ : estimación a posteriori.
- $z_i$ : vector de mediciones.
- $P_i^+$ : matriz de covarianza a posteriori.
- $I$ : matriz identidad.

También se utilizarán variables de la etapa de predicción [Imagen 4].

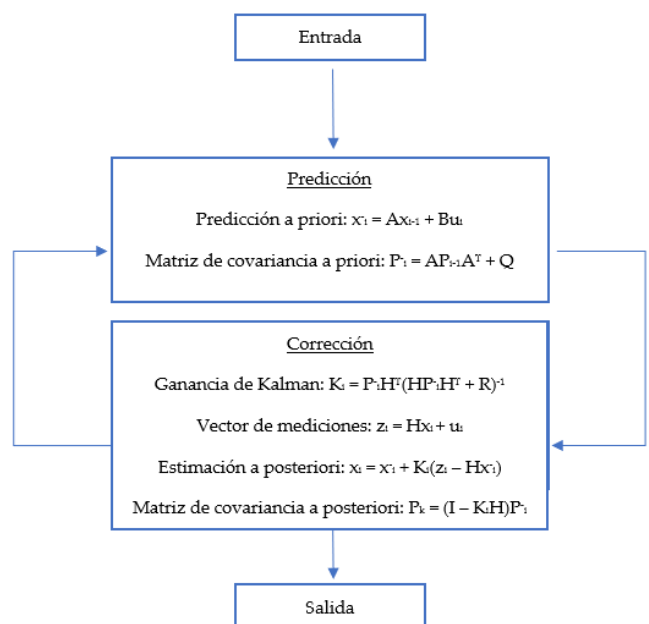


Imagen 4 – Algoritmo del Filtro de Kalman

#### 6 METODOLOGÍA Y PLANIFICACIÓN

Durante el proyecto se utilizó una metodología de ciclo de vida clásico [12] hasta que se llegó a las tareas de implementación y test del algoritmo del Filtro de Kalman, donde se cambió la metodología a iterativa e incremental [13].

Para desarrollar el proyecto, este se ha dividido en tres entregas, las cuales contienen diferentes tareas. Se pueden visualizar las entregas y las tareas en el diagrama de Gantt [Apéndice 1].

Durante la primera entrega, se buscó información sobre los satélites, los chipsets de los *smartphones*, que contienen el receptor GPS, y sobre la comunicación de estos. Con estas tareas obtuvimos la información teórica necesaria para conocer como un *smartphone* podía ser geolocalizado.

Durante la segunda entrega, se buscó información sobre programación en Android y sobre las API [Imagen 5].

Con estas tareas, obtuvimos el conocimiento básico de programación en Android y sobre las librerías utilizadas para la geolocalización.

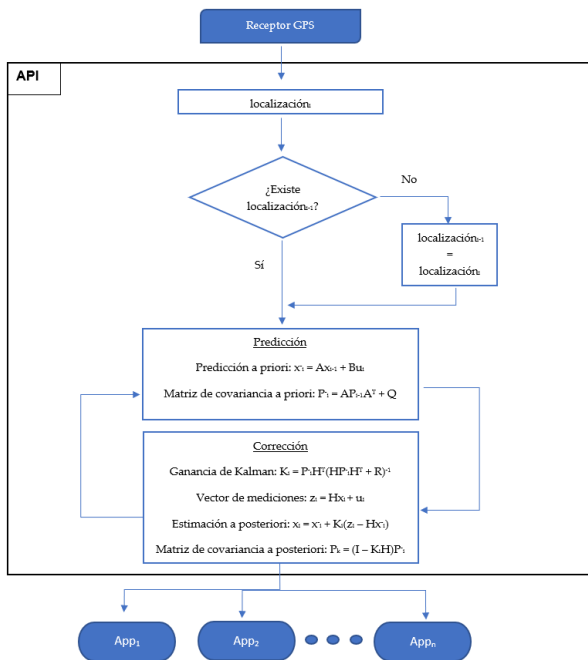


Imagen 5 – API a desarrollar

Con esta información, se creó una aplicación “Fake-GPS”, la cual modificaba la ubicación actual del *smartphone* por una ubicación simulada predeterminada [Imagen 6].

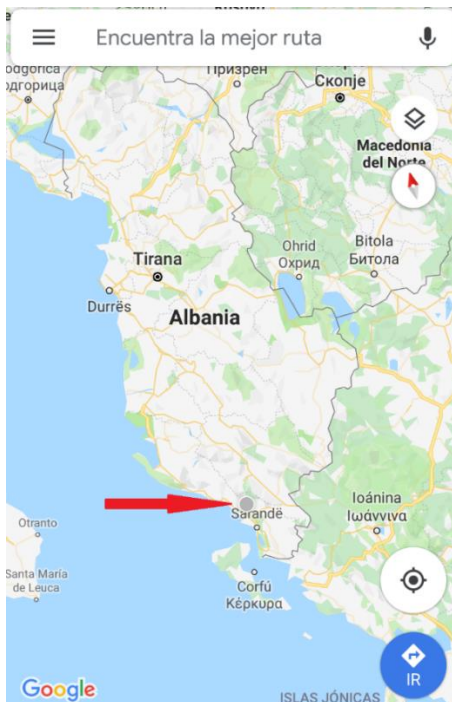


Imagen 6 – Ubicación simulada en Sarandë, Albania en la aplicación “Google Maps”

Esta aplicación ayudó a saber como implementar una API que modificara la ubicación actual del *smartphone* en el resto de las aplicaciones instaladas en este.

En la tercera entrega se procedió a implementar y testear el Filtro de Kalman. En este momento se cambió la metodología ya que los resultados no eran los esperados, y pasamos de una metodología de ciclo de vida básico a una metodología iterativa e incremental, ya que repetimos la implementación y las pruebas del Filtro de Kalman hasta que nos dio unos resultados correctos.

En esta entrega aparecieron tareas que no se habían tenido en cuenta: implementación de inicio y fin del proceso; implementación de un mapa para poder observar los resultados; importación y exportación de resultados; configuración de parámetros por el usuario y configuración de la visualización de los resultados.

Durante la entrega final, se buscó una manera de mejorar los resultados del Filtro de Kalman, ya que este no era capaz de predecir un cambio de dirección, por lo que, a la hora de girar, no hacia correctamente el giro.

## 7 Diseño del software

A pesar de que la aplicación tenga la finalidad de convertirse en una API, se ha creado una interficie gráfica [Imagen 7] con las siguientes funciones:

a) **Información:** nos indica información de interés de los datos que está administrando la aplicación. Estos datos están separados por un guión. A la izquierda de este, se encuentran los datos que nos otorga el receptor GPS; a la derecha, están los datos filtrados con el Filtro de Kalman. En el caso de que suceda algún error, también se nos mostrará en la interfaz.

b) **Proceso:** nos permite iniciar o parar la aplicación.

c) **Configuración de parámetros:** permite configurar los parámetros con los cuales ejecutaremos las diferentes pruebas.

d) **Configuración del mapa:** nos permite abrir el mapa y cambiar los parámetros de visualización de los resultados en este. Podemos activar o desactivar los círculos (zona en la cual puede estar el *smartphone*), las líneas (unión entre la localización actual y la anterior), el *accuracy* máximo permitido para mostrar las localizaciones y si queremos el mapa en modo normal o en modo satélite.

e) **Resultados:** nos permite importar y exportar los resultados. Los resultados se guardan en formato `timestamp_tiempo_distancia.txt`, en el cual los datos estarán separados por “&”. El tiempo y la distancia son las variables que configuran la actualización de la localización. También nos permite cambiar el nombre del archivo que, posteriormente, será leído.

**Kalman Filter**

**INFORMACION**  
 Latitud: 41.4341287 - 41.4341287  
 a Longitud: 2.0918628 - 2.0918628  
 Accuracy: 32.1 - 18.5069  
 Size List: 1 - 1  
 Error: ---

**PROCESO**  
 b **START**  
**STOP**

**CONFIGURACION DE PARAMETROS**  
 Tiempo: 0  
 Cambiar tiempo (ms)  
 c Distancia: 0  
 Cambiar distancia (metros)  
 Error: 15  
 Cambiar distancia (metros)  
**GUARDAR CONFIGURACION**

**CONFIGURACION DEL MAPA**  
 d **MAPA**  
**CIRCULOS: TRUE**  
**LINEAS: TRUE**  
**ERROR: TRUE**  
**TIPO MAPA: NORMAL**

**RESULTADOS**  
 e **GUARDAR RESULTADOS**  
**CARGAR RESULTADOS**  
 Nombre del archivo a cargar:  
 Nuevo nombre del archivo  
**GUARDAR NOMBRE ARCHIVO**

Imagen 7 – Interficie gráfica de la aplicación

## 8 Test

En este apartado se explicará que parámetros y condiciones se han tenido en cuenta para hacer las pruebas.

Se han utilizado dos configuraciones de la aplicación para observar cual de las dos nos otorgaba unos resultados más cercanos a la realidad. Los parámetros configurables por el usuario son los siguientes: el tiempo de refresco de la localización y cada cuántos metros pedimos la localización. Para que la aplicación obtenga una nueva localización, solo se tendrá que cumplir con uno de los dos parámetros anteriores.

También se ha tenido en cuenta los entornos donde se han realizado las pruebas, los cuales los clasificaremos en entornos abiertos (Les Planes, Sant Cugat del Vallès y Universitat Autònoma de Barcelona) y en entornos cerrados (Barcelona). Los entornos cerrados se caracterizan por estar rodeados de obstáculos grandes, por lo que hemos elegido el centro de Barcelona, donde podemos encontrar calles muy estrechas con edificios muy altos. Por otro lado, los entornos abiertos pueden ser cualquier tipo de entorno sin obstáculos.

Por otro lado, se han podido realizar pruebas con diferentes climatologías, durante días soleados y días lluviosos.

Por último, también se ha tenido en cuenta el medio de transporte utilizado. Se han hecho pruebas en recorridos a pie y en coche.

Las configuraciones que se han utilizado para obtener los resultados han sido las siguientes:

- Configuración 1: tiene un tiempo de refresco de 0 segundos y de 0 metros, es decir, cada vez que el receptor GPS reciba una nueva localización, se la enviará a la aplicación.
- Configuración 2: tiene un tiempo de refresco de 3 segundos y de 5 metros (a pie) o 50 metros (en coche). Cada vez que se cumpla una de estas condiciones, el receptor GPS nos enviará la localización.

Finalmente, hay que tener en cuenta con que dispositivo se han obtenido los datos. Para ello, hemos utilizado dos dispositivos móviles [Imagen 8] con el mismo chipset, el Qualcomm Snapdragon 636, el cual sólo recibe la señal L1 de los satélites, es decir, no contiene la tecnología *dual-frequency*. Para poder hacer una comparación entre las dos configuraciones, se han obtenido los datos con los dos dispositivos móviles de forma paralela.



Imagen 8 – Dispositivos utilizados para hacer las pruebas



## 9 RESULTADOS

En este apartado se compararán las localizaciones otorgadas por el receptor GPS con las localizaciones obtenidas después de filtrar las localizaciones con el Filtro de Kalman.

Durante los siguientes resultados, hablaremos del término *accuracy*. En geolocalización, el *accuracy* es la longitud del radio de la posible ubicación, por lo tanto, queremos obtener el menor *accuracy* posible.

También hablaremos de recorridos. Definimos recorrido como la unión en orden de obtención de los centros de las localizaciones.

### 9.1 RESULTADOS EN UN ENTORNO ABIERTO EN COCHE

Utilizaremos las dos configuraciones anteriormente explicadas para observar cual de las dos otorgan mejores localizaciones. Podremos observar en las siguientes imágenes en color azul la localización otorgada por el receptor GPS y en color verde las localizaciones filtradas.

Podemos observar que con la configuración 1 [Imagen 9] se obtiene un recorrido muy limpio, actualizando la localización cada vez que esté disponible y estando en un entorno abierto, no haría falta aplicar el Filtro de Kalman, ya que los resultados del GPS tienen una localización correcta y un *accuracy* menor del que obtenemos con el Filtro de Kalman.

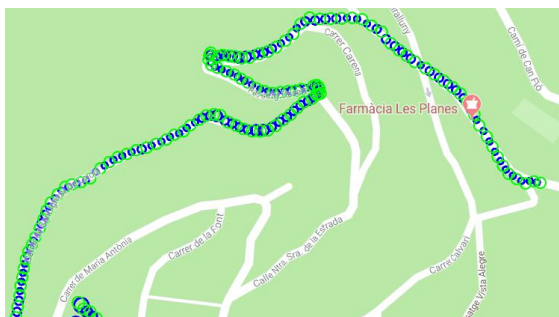


Imagen 9 – Resultados con la configuración 1 en un trayecto en coche en un entorno abierto

Podemos observar que con la configuración 2 [Imagen 10] se obtiene un recorrido discontinuo. Esto es provocado por la configuración, ya que la API pide la localización al receptor GPS cada más tiempo o cada más metros que con la configuración anterior.

También podemos observar que, a pesar de haber realizado las dos pruebas simultáneamente en paralelo, el *accuracy* es distinto [Gráfico 1], lo que provoca el resultado contrario al que buscamos, que es obtener el mínimo *accuracy* posible.

Por último, podemos observar el problema principal de utilizar el Filtro de Kalman en los cambios de dirección. Este filtro no puede detectar un cambio de sentido por si solo, ya

que predice la siguiente localización según la orientación de la localización anterior, lo que implica que se salga de la carretera en las curvas. A partir de la siguiente localización, este empezaría a predecir correctamente el camino.

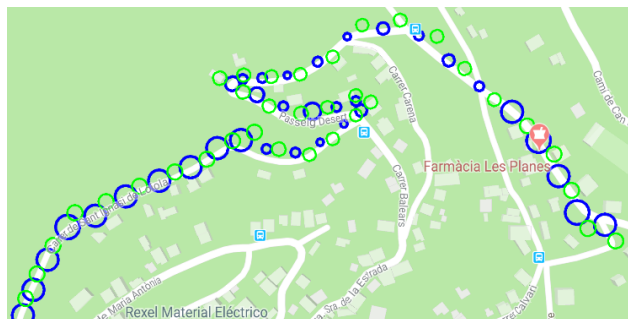


Imagen 10 – Resultados con la configuración 2 en un trayecto en coche en un entorno abierto

Se pueden extraer los siguientes resultados de estas pruebas:

- La configuración 1 nos otorga un *accuracy* menor que la configuración 2, por lo tanto, debemos obtener la siguiente localización lo antes posible para tener un recorrido correcto.
- Podemos observar que en la configuración 2 [Gráfico 1], el GPS + Filtro de Kalman obtiene un menor *accuracy* que el GPS, pero, sin embargo, si nos fijamos en el recorrido [Imagen 10], podemos observar que el recorrido del GPS + Filtro de Kalman es erróneo, ya que esta prueba se hizo en coche, y hay diversas localizaciones que no están dentro de la carretera por la cual se circuló.

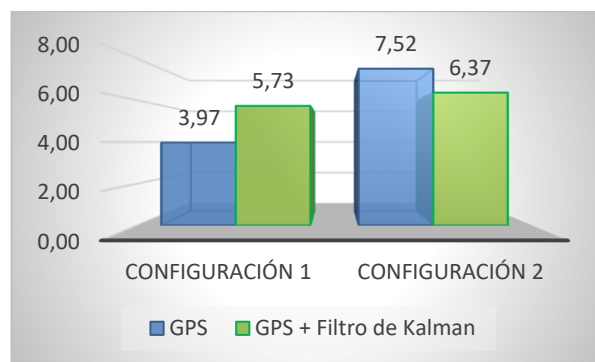


Gráfico 1 – Comparativa de la media del *accuracy* con las distintas configuraciones en un entorno abierto

### 9.2 RESULTADOS EN UN ENTORNO ABIERTO A PIE

Durante las pruebas en un entorno abierto, se ha considerado como entorno abierto un entorno sin obstáculos que dificulten la señal entre los satélites y el receptor GPS, pudimos detectar que a veces el receptor GPS pierde la señal y no obtiene una localización. Cuando esto sucede, la locali-

zación es obtenida de las antenas de telefonía, las cuales tienen un *accuracy* mucho mayor al otorgado por el GPS [Imagen 11].

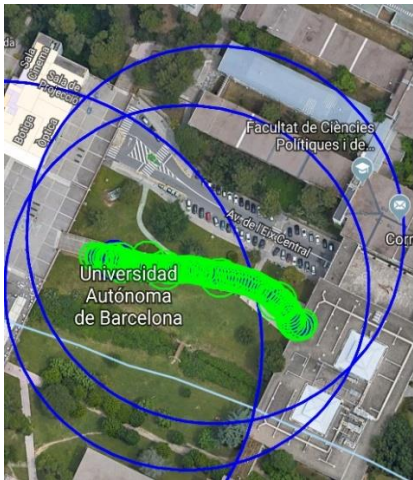


Imagen 11 – Pérdida de señal GPS y ubicaciones obtenidas a través de las antenas de telefonía

Cuando esto sucede, la aplicación continúa funcionando y el Filtro de Kalman obtiene un *accuracy* muy pequeño comparado con el que las antenas de telefonía otorgan [Gráfico 2].

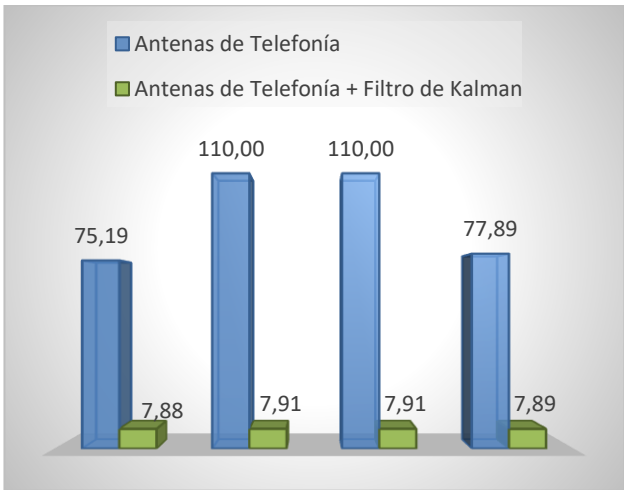


Gráfico 2 – Comparativa del *accuracy* de las localizaciones obtenidas con la señal de las antenas de telefonía con estas mismas filtradas con el Filtro de Kalman

Por otro lado, si no perdemos la señal GPS, no tendremos ningún motivo para usar el Filtro de Kalman ya que el receptor GPS obtiene un *accuracy* menor al que nos otorga el Filtro de Kalman [Imagen 12].

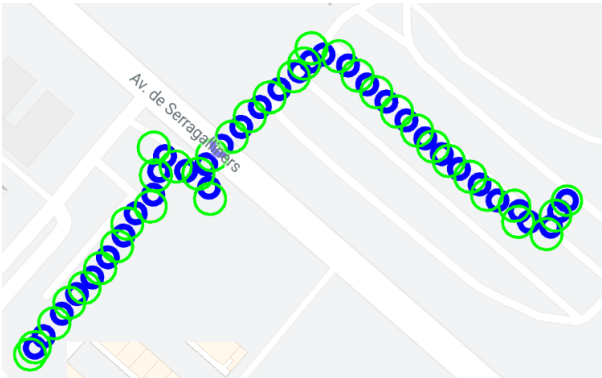


Imagen 12 – Recorrido a pie en un entorno abierto sin que el dispositivo pierda la señal GPS

### 9.3 Resultados en un entorno cerrado

Estas pruebas se han realizado solamente a pie, ya que han sido realizadas en Barcelona, por calles peatonales, donde los edificios podían intervenir en la calidad de la localización.

También se han realizado con climatologías diferentes para poder observar si también afectaba al *accuracy*.

Los resultados de estas pruebas nos confirman que actualizar lo antes posible la localización [Gráfico 3], otorga un mejor *accuracy*, como se ha visto en los resultados de entornos abiertos [Gráfico 1].

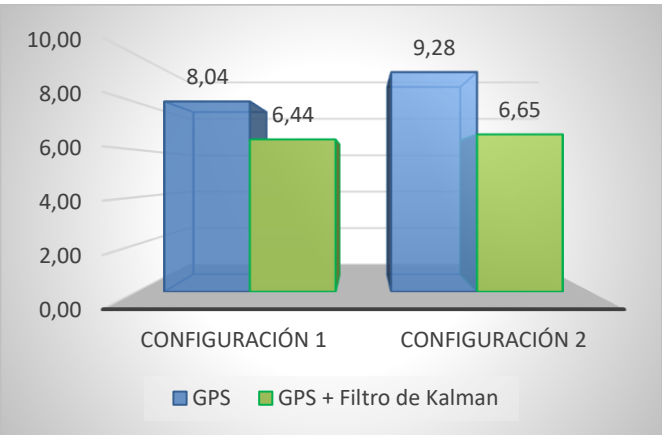


Gráfico 3 - Comparativa de la media del *accuracy* con las distintas configuraciones en un entorno cerrado durante un día soleado

También, como en las pruebas del entorno abierto, la configuración 1 [Imagen 13] nos proporciona un recorrido más similar al recorrido real que la configuración 2 [Imagen 14].



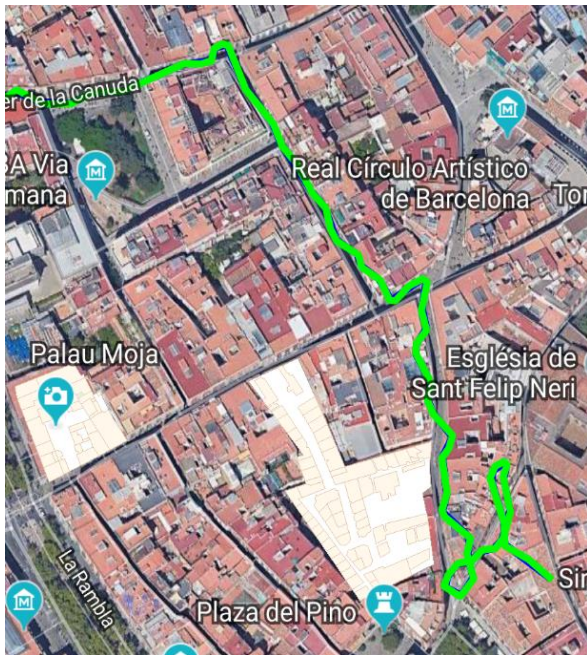


Imagen 13 – Recorrido durante un día soleado en un entorno cerrado con la configuración 1

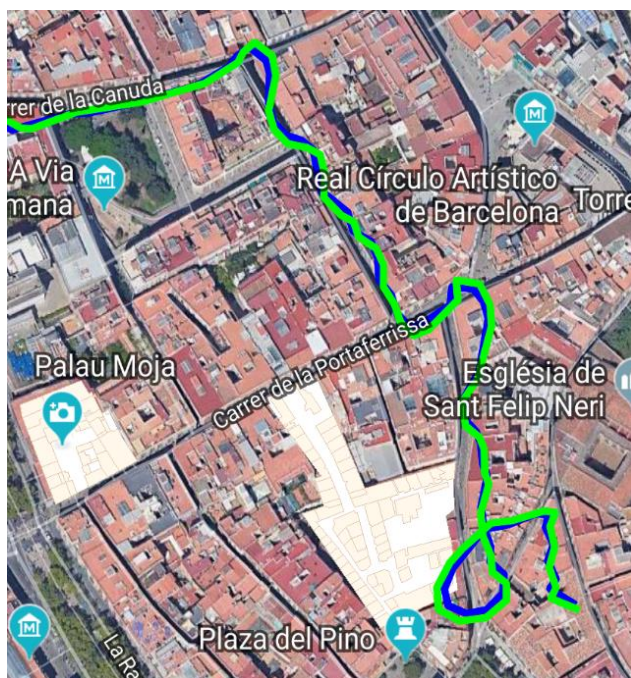


Imagen 14 – Recorrido durante un día soleado en un entorno cerrado con la configuración 2

Por otro lado, podemos confirmar que la climatología también afecta al *accuracy* de la localización, ya que, durante un día soleado, obtenemos la localización con una media de 1 metro más preciso que durante un día lluvioso [Gráfico 4].

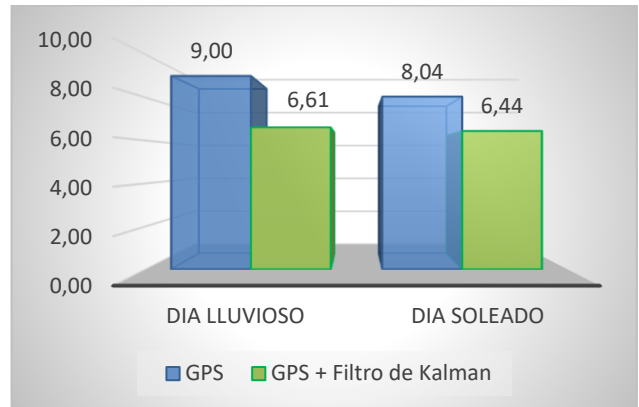


Gráfico 4 – Comparativa de la media del *accuracy* durante un día lluvioso y un día soleado

Sin embargo, aparecen distintos problemas, uno de los cuales ya habíamos observado en las pruebas en entornos abiertos, que es la predicción del giro [Imagen 15].

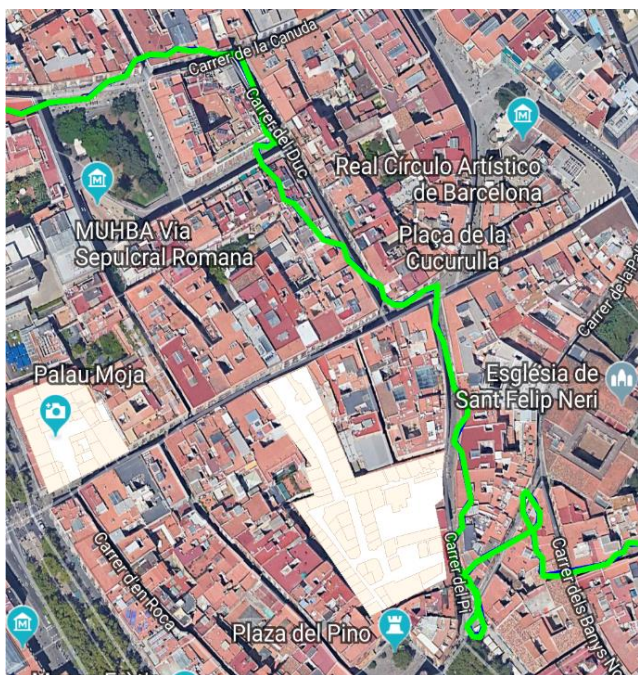


Imagen 15 – Recorrido original (rojo) con localizaciones de GPS (azul) y GPS + Filtro de Kalman (verde)

Por otro lado, durante las pruebas en un día lluvioso, la lluvia influye bastante en los resultados, hasta el punto de obtener la mayoría de los centros de las localizaciones donde están ubicados los edificios [Imagen 16].

En las pruebas durante un día soleado, se observa todo lo contrario, casi todos los centros de las localizaciones están en la calle [Imagen 13].





*Imagen 16 – Recorrido durante un día lluvioso en un entorno cerrado con la configuración 1*

## 10 CONCLUSIÓN

Observando los resultados de las pruebas ejecutadas, podemos extraer las siguientes conclusiones:

- En entornos abiertos, el *accuracy* de la localización es muy bajo, y, por lo tanto, muy preciso, y sólo se ha conseguido mejorar cuando el dispositivo perdió la señal GPS y se obtuvo la localización a partir de antenas de telefonía.
- En entornos cerrados, se consigue disminuir el *accuracy*, y, por lo tanto, obtenemos una localización más precisa.
- En los dos entornos, a pesar de utilizar la configuración óptima (configuración 1), el Filtro de Kalman no puede predecir un giro inesperado, por lo que puede provocar que, en las curvas, la localización no esté dentro del radio de este.

Con estas conclusiones, podemos afirmar que se ha conseguido el objetivo, ya que hemos estudiado la geolocalización y se ha conseguido reducir el *accuracy* de las localizaciones en entornos cerrados, los cuales son los más problemáticos. Aun así, tiene muchas implementaciones futuras por delante.

Durante el proyecto han surgido diversos problemas, sobre todo con la programación en Android, lenguaje que desconocía por completo. Estos problemas se han solucionado consultando foros y las librerías oficiales de Android. Por otro lado, la dificultad matemática del Filtro de Kalman hizo que la planificación se retrasara, pero no afectó mucho

ya que se alcanzó el objetivo y se hizo todo lo preestablecido.

El proyecto me ha permitido aplicar ciertos conocimientos obtenidos durante la carrera y me ha obligado a trabajar continuamente para poder cumplir con las tareas preestablecidas.

## 11 IMPLEMENTACIONES FUTURAS

El problema principal de esta aplicación es que el Filtro de Kalman no logra predecir los giros, habría que crear un algoritmo que lo permitiera. Por otro lado, también se podría intercalar las localizaciones del GPS con las del Filtro de Kalman y utilizar la que tenga un *accuracy* menor. Solucionaríamos los problemas que tiene el Filtro de Kalman con los giros y siempre tendríamos la localización con el menor *accuracy* posible.

Actualmente, la aplicación se tiene que ejecutar en primer plano y sin que se bloquee la pantalla del dispositivo. Habría que implementarla para que se ejecutara siempre en segundo plano.

Al utilizar la aplicación como proveedora de ubicación a otras aplicaciones, aparece el siguiente problema: la aplicación obtiene la última localización conocida, y en el momento de establecer la ubicación filtrada, ésta pasa a ser la última conocida. Habría que estudiar la forma de que la aplicación envíe la localización filtrada y que la última ubicación conocida sea la que otorga el receptor GPS.

La dificultad de este algoritmo es que utiliza la última ubicación conocida y el estado anterior. Con estos datos no se pueden predecir ciertos giros. Esto provoca que los cambios de dirección sean calculados erróneamente. Habría que implementar un algoritmo que fuese capaz de detectar cuando el dispositivo ha cambiado de dirección.

Por último, habría que probar la aplicación en *smartphones* que contengan la tecnología *dual-frequency* para comprobar si realmente la señal L5 ayuda a reducir el error provocado por los obstáculos. En el caso de que sea cierto, obtendríamos unos mejores resultados en entornos cerrados (rodeados con obstáculos). Actualmente, la localización por satélites es errónea y, por lo tanto, el Filtro de Kalman también.

## AGRADECIMIENTOS

Agradecer a Marta Prim su implicación en este proyecto, realizando las funciones de tutora y adaptándose a mi disponibilidad.

También agradecerles a Jordi Roig y a Marc Vallibera su implicación en el proyecto, aportando información y conocimientos técnicos sobre el Filtro de Kalman.

Por otro lado, agradecerle a mi pareja, Meritxell Ar-

gente, por su ayuda durante las pruebas, cediendo su dispositivo para realizarlas de manera paralela y por el apoyo aportado durante el proyecto.

Por último, agradecerles a mis padres, Miguel Sánchez y Teresa Pita, por hacer posible que estudie esta carrera.

## BIBLIOGRAFÍA

[1] Trilateración [Febrero 2019]. Available: <https://es.wikipedia.org/wiki/Trilateraci%C3%B3n>

[2] Triangulación [Febrero 2019]. Available: [https://es.wikipedia.org/wiki/Triangulaci%C3%B3n#Triangulaci%C3%B3n\\_mediante\\_GPS](https://es.wikipedia.org/wiki/Triangulaci%C3%B3n#Triangulaci%C3%B3n_mediante_GPS)

[3] Información sobre los GPS [Febrero 2019] Available: <https://es.wikipedia.org/wiki/GPS>

[4] Sistema Global de Navegación [Febrero 2019]. Available: [https://es.wikipedia.org/wiki/Sistema\\_global\\_de\\_navegaci%C3%B3n\\_por\\_sat%C3%A9lite](https://es.wikipedia.org/wiki/Sistema_global_de_navegaci%C3%B3n_por_sat%C3%A9lite)

[5] Librería GnssStatus [Febrero 2019]. Available: <https://developer.android.com/reference/android/location/GnssStatus>

[6] *Application Programming Interface* [Febrero 2019]. Available: [https://es.wikipedia.org/wiki/Interfaz\\_de\\_programaci%C3%B3n\\_de\\_aplicaciones](https://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones)

[7] Android [Junio 2019] Available: <https://es.wikipedia.org/wiki/Android>

[8] Kernel de Linux [Junio 2019] Available: [https://es.wikipedia.org/wiki/N%C3%A9cleo\\_Linux](https://es.wikipedia.org/wiki/N%C3%A9cleo_Linux)

[9] Software de Código abierto [Junio 2019] Available: [https://es.wikipedia.org/wiki/Software\\_de\\_c%C3%B3digo\\_abierto](https://es.wikipedia.org/wiki/Software_de_c%C3%B3digo_abierto)

[10] Filtro de Kalman [Marzo 2019]. Available: [https://es.wikipedia.org/wiki/Filtro\\_de\\_Kalman](https://es.wikipedia.org/wiki/Filtro_de_Kalman)

[11] Ruido blanco [Marzo 2019]. Available: [https://es.wikipedia.org/wiki/Ruido\\_blanco](https://es.wikipedia.org/wiki/Ruido_blanco)

[12] Metodología de ciclo de vida básico [Marzo 2019]. Available: [https://es.wikipedia.org/wiki/Desarrollo\\_en\\_cascada](https://es.wikipedia.org/wiki/Desarrollo_en_cascada)

[13] Metodología de desarrollo iterativo e incremental [Abril 2019]. Available: [https://es.wikipedia.org/wiki/Desarrollo\\_iterativo\\_y\\_creciente](https://es.wikipedia.org/wiki/Desarrollo_iterativo_y_creciente)

APÉNDICE

APÉNDICE 1. Diagrama de Gantt

Durante el proyecto se ha seguido el siguiente diagrama de Gantt:

