



Institución Universitaria

# Metodología de posicionamiento robusto de UAVs en interiores utilizando técnicas de inteligencia computacional para supervisión y vigilancia

**Julián Andrés Román Franco**

Instituto Tecnológico Metropolitano  
Facultad de Ingeniería, Maestría en Automatización y Control  
Medellín, Colombia  
2018

# Metodología de posicionamiento robusto de UAVs en interiores utilizando técnicas de inteligencia computacional para supervisión y vigilancia

**Julián Andrés Román Franco**

Tesis o trabajo de grado presentada(o) como requisito parcial para optar al título de:  
**Magister en Automatización y Control**

Directores:

Doctor, Ricardo Andrés Velázquez Vélez  
Magister, David Andrés Marquez Viloría

Línea de Investigación:

Sistemas de control y robótica

Grupo de Investigación:

Automática, Electrónica y Ciencias Computacionales

Instituto Tecnológico Metropolitano  
Facultad Ingeniería, Maestría en automatización y control  
Medellín, Colombia

2018

## Resumen

Cuando un espacio cerrado como un edificio, una fabrica, una bodega, un escenario deportivo o algún otro tipo de lugar de índole similar, se encuentra en emergencia ante alguna catástrofe humana o natural como avalanchas, inundaciones, incendios o derrumbes, el personal de rescate pone en riesgo su vida, mientras buscan mitigar y evitar al máximo posible las perdidas humanas. La utilización de vehículos aéreos no tripulados UAVs puede propiciar la disminución del tipo de riesgos anteriormente mencionados, pero para que la intervención de estos sea efectiva se deben solucionar antes diversos problemas que el entorno presenta para su correcto funcionamiento. Uno de estos problemas es el posicionamiento en interiores, el cual ha sido abordado a través de diversas técnicas como Pseudolites, Ultra-sonido, Visión, Magnéticas, entre otras. Sin embargo, las anteriores alternativas cuentan con la dificultad de requerir una infraestructura o equipos especiales propios en el sitio de la aplicación, lo que puede conllevar a que si en medio de una catástrofe alguno de los equipos es afectado, el posicionamiento será afectado también. Para dar solución a este problema hemos propuesto a través de este trabajo tres metodologías de posicionamiento en interiores utilizando la información de estaciones radiales AM y FM, las cuales están disponibles en cualquier ciudad del planeta y no requieren de ningún tipo de intervención o adecuación para su uso. Las metodologías propuestas utilizan algoritmos de aprendizaje profundo y su desempeño fue comparado con el del algoritmo de posicionamiento KNN Regresivo, el cual es comunemente utilizado para posicionamiento en interiores. Por último las técnicas fueron llevadas a un sistema embebido en donde se puso a prueba su respuesta en tiempo real. Obteniendo resultados para posicionamiento en interiores con un error inferior al que presentan los dispositivos GPS en exteriores.

**Palabras clave:** Posicionamiento en interiores, localización en interiores, FM en interiores, AM en interiores, Radio Frecuencia en interiores.

## Abstract

When a closed space such as a factory, a warehouse, a sports arena or some other similar place, is in emergency due to some kind of natural catastrophes such as avalanches, floods, fires or landslides, the rescue personnel puts their lives in risk, while they seek to mitigate and avoid human losses as much as possible. The use of unmanned aerial vehicles UAVs, may decrease the risk mentioned above, but for the intervention of these to be effective, it is necessary to solve before the various problems that the environment presents for its correct functioning. One of these problems is indoor positioning, which has been addressed through various techniques such as Pseudolitos, Ultra-sound, Vision, Magnetic, among others. However, these techniques have the difficulty of requiring an infrastructure or special

equipment at the application site, this means that if one of the equipment is affected during the catastrophe, positioning is also affected. To obtain a solution to this problem, we have proposed through this work three indoor positioning methods using the information of AM and FM radio stations, which are available anywhere in the world and do not require any type of intervention or adaptation to be used. The proposed methodologies use deep learning algorithms, the performance of these was compared with that of a positioning algorithm that uses a regressive KNN, which is commonly used for indoor positioning. Finally, the techniques were taken to an embedded system where their response was tested in real time, reaching results for indoor positioning with an error lower than that presented by GPS devices outdoors.

# Contenido

<b>Resumen</b>	<b>iii</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Hipótesis . . . . .	2
1.2. Objetivos . . . . .	3
1.2.1. Objetivo General . . . . .	3
1.2.2. Objetivos Específicos . . . . .	3
1.3. Organización del documento . . . . .	3
<b>2. Marco Teórico y Estado del Arte</b>	<b>4</b>
2.1. Transmisiones radiales AM y FM . . . . .	4
2.2. Regulaciones radiales locales y nacionales y sus características . . . . .	4
2.3. Posicionamiento a través de Fingerprint . . . . .	5
2.3.1. K vecinos mas cercanos (KNN) . . . . .	8
2.3.2. Aprendizaje Profundo (Deep Learning) . . . . .	8
2.4. Técnicas de posicionamiento en interiores . . . . .	10
2.4.1. Posicionamiento con Pseudolites . . . . .	10
2.4.2. Posicionamiento con Campos Magnéticos . . . . .	11
2.4.3. Posicionamiento con Luz Visible . . . . .	11
2.4.4. Posicionamiento con Unidades de Medida Inercial . . . . .	11
2.4.5. Posicionamiento con Técnicas de Visión Artificial . . . . .	12
2.4.6. Posicionamiento con Ultrasonido . . . . .	12
2.4.7. Posicionamiento con Radio Frecuencia . . . . .	13
2.4.8. Sistemas de Posicionamiento Híbridos . . . . .	13
<b>3. Metodología</b>	<b>15</b>
3.1. Hardware . . . . .	15
3.1.1. Comando rtl_power . . . . .	17
3.2. Montaje Experimental . . . . .	18
3.2.1. Área . . . . .	19
3.2.2. Radio Estaciones Utilizadas . . . . .	19
3.2.3. Construcción de base de datos con el índice de potencia RSS . . . . .	20
3.3. Procesamiento de datos . . . . .	21
3.3.1. FM . . . . .	21

3.3.2.	AM . . . . .	23
3.3.3.	AM - FM . . . . .	24
3.4.	Entrenamiento . . . . .	24
3.4.1.	Métricas . . . . .	25
3.5.	Sistema de Posicionamiento Embebido . . . . .	26
3.5.1.	Compatibilidad . . . . .	27
3.5.2.	Capacidad de procesamiento y costos de implementación . . . . .	27
3.5.3.	Implementaciones futuras . . . . .	27
3.5.4.	Manejo remoto para pruebas . . . . .	28
3.6.	Instalación de Drivers y librerías RTL2832u en Raspberry pi . . . . .	28
<b>4.</b>	<b>Resultados Experimentales</b>	<b>30</b>
4.1.	Posicionamiento con radio-estaciones FM . . . . .	30
4.1.1.	Exploración de la configuración de la red . . . . .	31
4.1.2.	Evaluación de desempeño . . . . .	32
4.1.3.	Tiempo de Ejecución . . . . .	36
4.2.	Posicionamiento con radio-estaciones AM . . . . .	40
4.2.1.	Exploración de la configuración de la red . . . . .	40
4.2.2.	Evaluación de desempeño . . . . .	41
4.2.3.	Tiempo de ejecución . . . . .	44
4.3.	Posicionamiento híbrido radio-estaciones FM - AM . . . . .	47
4.3.1.	Exploración de la configuración de la red . . . . .	47
4.3.2.	Evaluación de desempeño . . . . .	48
4.3.3.	Tiempo de Ejecución . . . . .	51
4.4.	Implementación de redes neuronales en la Raspberry pi . . . . .	54
4.5.	Prueba de redes neuronales en Raspberry Pi . . . . .	54
4.5.1.	Lectura de los datos y aplicación de las redes . . . . .	54
4.5.2.	Tiempo de estimación de la posición . . . . .	56
4.5.3.	Consumo de potencia . . . . .	57
4.5.4.	Prueba de técnicas en movimiento . . . . .	59
4.6.	Comparación de resultados . . . . .	63
<b>5.</b>	<b>Conclusiones y recomendaciones</b>	<b>66</b>
5.1.	Conclusiones . . . . .	66
5.2.	Recomendaciones . . . . .	66
<b>A.</b>	<b>Instalación y Configuración RTL2832u</b>	<b>68</b>
	<b>Bibliografía</b>	<b>70</b>

# 1. Introducción

Los sistemas de posicionamiento global (GPS) son una herramienta fundamental, no solo en aplicaciones militares, sino también, en un sinnúmero de aplicaciones civiles: navegación terrestre, marítima y aérea, sistemas de información geográfica, etc [1]

Un problema esencial de los sistemas de posicionamiento global actuales es su incapacidad para proporcionar una posición precisa al interior de edificaciones. Esta limitación de los sistemas GPS dificulta el despliegue de aplicaciones que requieren posicionamiento preciso dentro de edificaciones.

Tradicionalmente, los sistemas de posicionamiento en interiores (IPS por sus siglas en inglés) han sido utilizados para localizar personas u objetos en grandes edificaciones y lugares cerrados. Por ejemplo, localizar pacientes en un hospital, encontrar gente atrapada en un edificio colapsado o en llamas, encontrar trabajadores en una planta, etc. Otras posibles aplicaciones de los IPS son navegación personal en aeropuertos, estaciones ferroviarias, y grandes superficies comerciales; y navegación de sistemas robóticos dentro de edificios para labores de aseo, vigilancia, correspondencia, además de un sin número de aplicaciones que estos pueden ofrecer [2,3].

La utilidad y posibilidades de un IPS son indiscutibles, y la comunidad académica ha propuesto diversas tecnologías para la implementación de estos sistemas: magnéticos, ultrasonido, pseudolites, ópticos, RF, entre otros. De Angelis et al. en [4] y Blankenbach et al. en [5], proponen un IPS utilizando principios magnéticos que puede desplegarse en áreas pequeñas y que además utiliza sensores disponibles en teléfonos celulares, los cuales a su vez actúan como estación móvil. La tecnología de ondas mecánicas de ultrasonido también ha sido trabajada por diversos autores [6–10]. En particular, Yazici et al. en [9] propone un sistema de posicionamiento por ultrasonido llamado SESKON que utiliza la técnica de diferencia en tiempo de llegada (TDOA por sus sigla en inglés), la cual no requiere sincronización entre los transmisores y la estación móvil, alcanzando un error máximo de 16mm a velocidades de 200 mm/s. En un intento de reutilizar los receptores GPS para posicionamiento en interiores, algunos autores han propuesto el uso de satélites falsos (*pseudolites*) que emiten señales análogas a las del GPS [11–14]. Los pseudolites emiten señales que permiten que un receptor GPS comercial estime la posición con errores en el orden de centímetros. IPSs que utilizan como principio la propagación de señales de radiofrecuencia (RF) han sido propuestos con casi todos los tipos de sistemas de comunicación inalámbrica: WiFi, GSM, ZigBee, Ultra Wide Band, FM, y otras tipo de sistemas de radiofrecuencia menos comunes. En esta categoría existen diferentes técnicas para estimar la posición: potencia recibida (RSSI) de las fuentes

de RF, el tiempo de vuelo (TOF), TDOA, fingerprinting, etc. En el capítulo 2 se profundizará un poco más en las diferentes técnicas utilizadas en posicionamiento en interiores.

Cada una de las tecnologías mencionadas tiene ventajas y desventajas que deben ser evaluadas a la hora de implementar un IPS, por ejemplo, la precisión de algunas de estas solo se alcanza bajo ciertas condiciones controladas, y otras exigen el montaje de infraestructura adicional lo cual las hace costosas. En este sentido, los IPSs basados en RF permiten ahorrarse los costos de infraestructura, ya que en general utilizan infraestructura de comunicaciones previamente instalada. Adicionalmente, el avance de la tecnología de semiconductores permite no solo integrar transceptores RF en casi cualquier dispositivo electrónico, sino también realizar los cálculos necesarios para estimar la posición en pequeños sistemas embebidos de bajo costo. Por otro lado, las técnicas de posicionamiento que utilizan RF y fingerprinting permiten solventar algunos de los problemas comunes de línea de vista, efectos multirayectoria y pérdidas de señal asociadas a elementos estructurales de edificaciones. El fingerprinting consiste en caracterizar puntos de interés en un área de navegación. De esta forma se construye una base de datos con mediciones por ejemplo de RSSI en dichos puntos. Esta base de datos permite luego estimar la posición comparando respecto a mediciones hechas por la estación móvil [15].

Este trabajo presenta tres técnicas de posicionamiento en interiores utilizando señales emitidas por radio estaciones comerciales: posicionamiento con radioestaciones FM, posicionamiento con radioestaciones AM, y posicionamiento con radioestaciones AM y FM. Las técnicas propuestas, utilizan fingerprinting y algoritmos de aprendizaje profundo (Deep Learning) para estimar la posición de la estación móvil. El desempeño de las técnicas se evalúa respecto a técnicas previamente utilizadas en la literatura [16, 17] que estiman la posición con algoritmos KNN Regresivo (k-vecinos mas cercanos) y fingerprinting. A diferencia de otros trabajos donde los experimentos se realizan en lugares vacíos, nuestros experimentos se realizaron en un espacio residencial amoblado y habitado. Por ultimo, las técnicas propuestas son implementadas y evaluadas en un sistema embebido.

En este contexto, nos proponemos responder con este trabajo la siguiente pregunta de investigación:

¿ Cómo posicionar de forma robusta UAVs en interiores utilizando las señales de Radio Frecuencia de estaciones radiales AM y FM sin necesidad de modificar o alterar las señales transmitidas ?

## 1.1. Hipótesis

Utilizando información espectral y de potencia de las señales electromagnéticas producidas por estaciones radiales de AM y FM y sin alteración de las señales emitidas, es posible posicionar UAVs con niveles de precisión similares a los de un receptor GPS en espacios abiertos.

## **1.2. Objetivos**

### **1.2.1. Objetivo General**

Proponer una metodología de posicionamiento en interiores utilizando la información conjunta de estaciones radiales AM y FM sin modificación de las señales transmitidas y que pueda ser aplicada a UAVs.

### **1.2.2. Objetivos Específicos**

- Caracterizar las técnicas de posicionamiento en interiores y exteriores que utilicen señales de radio AM y FM.
- Proponer una técnica híbrida entre AM y FM para mejorar la precisión en la estimación de la posición en interiores.
- Validar la metodología de posicionamiento en interiores mediante la integración en una plataforma de hardware embebido para control automático de UAVs

## **1.3. Organización del documento**

Este trabajo esta organizado de la siguiente manera: En el capítulo 2 tenemos el Estado del Arte, en donde encontrará una revisión de las técnicas de posicionamiento en interiores y el marco conceptual. En el capítulo 3 se describe toda la metodología necesaria para proponer las técnicas de posicionamiento. En el capítulo 4 se presentan los resultados experimentales de las técnicas de posicionamiento propuestas, comparadas con la técnica de KNN regresivo e implementadas en un sistema embebido. Por último, en el capítulo 5 se presentan las conclusiones y recomendaciones.

## **2. Marco Teórico y Estado del Arte**

### **2.1. Transmisiones radiales AM y FM**

La necesidad de transmitir información como la voz humana (300-3300 Hz) o cualquier sonido dentro del rango audible (20 - 20000 Hz), ha permitido el desarrollo de diversas técnicas y tecnologías que hacen uso del espectro radio-eléctrico. Para el mundo de hoy, la aparición de los radios comerciales y su evolución, hacen que las tecnologías de transmisión radial en amplitud modulada AM y en frecuencia modulada FM, a través de las estaciones radiales, se conviertan en acompañantes del día a día de un sinnúmero de personas.

Las estaciones radiales AM transmiten en un rango de frecuencia entre los 500 KHz y los 1600 KHz. Cuando se habla de Amplitud Modulada, se hace referencia a la modulación en la amplitud, la cual consiste en transmitir las señales de audio y voz, haciendo uso de una frecuencia portadora que está en el rango de frecuencia ya mencionado y cuya amplitud cambia (es modulada) con la señal que se desea transmitir.

Las estaciones radiales FM, transmiten en un rango de frecuencia entre los 88 MHz y los 108 MHz, haciendo uso de modulación en Frecuencia Modulada (FM). De manera similar al AM, para la modulación, utiliza una frecuencia portadora en el rango de frecuencias dado, pero la diferencia con AM, es que FM no modula la amplitud sino la frecuencia. Tanto AM como FM realizan la radiodifusión utilizando una antena transmisora acorde con la frecuencia de transmisión y con características como longitud, ubicación y potencia. Mientras que las emisoras radiales se ubican en diferentes puntos de una ciudad, las antenas transmisoras se instalan comúnmente en los cerros o puntos más altos, alrededor de las ciudades. Todo esto, para tener una línea de vista mayor que permita propagar la señal alrededor de la ciudad, evitando el máximo de interferencias electromagnéticas. Para comunicar las radio estaciones distribuidas en toda la ciudad con sus antenas transmisoras, se utilizan radio enlaces directivos en FM, tanto para radio estaciones AM como FM [18].

### **2.2. Regulaciones radiales locales y nacionales y sus características**

Para el uso y distribución adecuada del espectro, existen diversos organismos a nivel mundial encargados de asesorar y dar recomendaciones a los diferentes países sobre el uso del espectro radio eléctrico, una de estas es la Unión Internacional de Telecomunicaciones (UIT). La ley

1341 de 2009 establece de acuerdo a los lineamientos dados por UIT la reglamentación para el espectro radio eléctrico. De acuerdo a esta Ley y teniendo en cuenta la distribución del espectro para estaciones de radio comerciales AM y FM dadas en la Resolución 3041 de 27 de Diciembre de 2017, el Ministerio Colombiano de las Tecnologías de la Información y las Comunicaciones MinTIC publicó el “PLAN TÉCNICO NACIONAL DE RADIODIFUSIÓN SONORA AMPLITUD MODULADA(A.M.)” [19] y el “PLAN TÉCNICO NACIONAL DE RADIODIFUSIÓN SONORA FRECUENCIA MODULADA(F.M.)” [20], esto con el fin de establecer un marco técnico para la adjudicación de canales radioeléctricos en los diferentes municipios del país. Además buscando facilitar la asignación de canales y racionalización de este recurso.

En [19] y [20] se brindan los lineamientos y regulaciones principales de operación para las radio estaciones tanto AM como FM, además de las especificaciones de frecuencias de operaciones, frecuencias de enlace, altura de la antena y la clase de estación de acuerdo con la cantidad de potencia radiada.

La información de frecuencias de operación y potencia de las radio estaciones es de gran utilidad durante este trabajo, ya que es el principal insumo para el Fingerprint.

La búsqueda de la información de las radio estaciones se centró en la ciudad de Medellín y los municipios aledaños que conforman el Valle del Aburra (Envigado, Itagú, Bello, Sabaneta, La Estrella).

Tomando como base la información consignada en [19], [20] y la información recopilada y actualizada de manera periódica por radio aficionados [21], en las cuales se presenta la información de frecuencia y potencia de las radio estaciones AM y FM de la ciudad de Medellín a la fecha Abril de 2018, se construyen las tablas **2-1** y **2-2** respectivamente. En estas tablas se presenta la información de interés sobre las radio estaciones AM y FM ubicadas en el Valle de Aburra. La mayoría de las antenas de radio difusión se encuentran ubicadas en los cerros noroccidentales de la ciudad.

## 2.3. Posicionamiento a través de Fingerprint

La metodología de posicionamiento a través de Fingerprint consiste en fijar una serie de puntos de interés en el área de experimentos. Luego, se recolectan las características propias cada punto, construyendo de esta forma una base de datos que sirve para entrenar y probar diferentes algoritmos de inteligencia computacional como clasificadores y estimadores [15]. La principal característica usada en esta metodología para señales de radio frecuencia es el índice de potencia de las señales de radio frecuencia o RSS. A continuación hablaremos un poco de las dos técnicas que se utilizarán durante este trabajo y que se basan en posicionamiento a través de fingerprint.

---

**Tabla 2-1.: Frecuencias Radio Estaciones Comerciales**


---

<b>EMISORAS AM CIUDAD DE MEDELLÍN</b>		
<b>Frecuencia Portadora</b>	<b>Potencia(KW)</b>	<b>Emisora</b>
590	50	W RADIO
670	50	ANTENA 2
710	10	RADIO SUPER
750	50	CARACOL RADIO
790	15	MUNERA EASTMAN RADIO
830	25	RADIO RELOJ
870	5	VIDA AM
910	10	LA VOZ DEL RIO GRANDE
990	100	RCN MEDELLIN
1020	10	EMISORA CLARIDAD
1050	10	RADIO UNICA
1080	10	LA 1.080
1110	10	RADIO BOLIVARIANA
1140	10	RADIO PAISA
1170	10	RADIO NUTIBARA
1230	10	MINUTO DE DIOS
1260	5	RADIO AUTENTICA
1320	5	RADIO MARIA
1350	5	ONDAS DE LA MONTAÑA
1380	2,3	RADIO INTERNACIONAL
1410	5	EMISORA CULTURAL U. DE A.
1440	5	COLMUNDO
1470	5	RADIO POPULAR
1530	5	LA VOZ DE JESUCRISTO
1590	5	BBN

---

**Tabla 2-2.: Frecuencias Radio Estaciones Comerciales**

<b>EMISORAS FM CIUDAD DE MEDELLÍN</b>		
<b>Frecuencia Portadora</b>	<b>Potencia(KW)</b>	<b>Emisora</b>
88,9	5	GUASCA FM
89,9	15	MIX
90,3	5	CARACOL RADIO
90,9	15	W RADIO
91,9	15	LA Z
92,4	5	RADIO BOLIVARIANA
92,9	15	LA MEGA
93,9	15	RADIO UNO
94,4	5	RADIO FANTASTICA
94,9	45	LA VOZ DE COLOMBIA
95,9	15	CÁMARA FM
96,4	15	POLICÍA NACIONAL
96,9	15	RADIO CRISTAL
97,9	5	BLU RADIO
98,9	5	TROPICANA
99,4	15	LOS 40 PRINCIPALES
99,9	100	RADIONICA
100,4	15	U.N. RADIO
100,9	1	LATINA ESTEREO
101,4	20	LA ESQUINA RADIO
101,9	15	EMISORA Cultural U DE A
102,9	15	OXIGENO
103,9	15	LA X
104,3	1	ESTRELLA ESTEREO
104,9	15	OLIMPICA ESTEREO
105,9	15	RADIO TIEMPO
106,3	5	CANDELA
106,9	5	LA FM
107,9	5	EL SOL

### 2.3.1. K vecinos mas cercanos (KNN)

Es una técnica de aprendizaje supervisado que permite realizar clasificación de un conjunto de muestras tomando como base un conjunto de datos de entrenamiento.

El algoritmo del KNN consta básicamente de dos fases. En la primera fase se almacenan los datos de entrenamiento, cada uno con una etiqueta correspondiente a la clase a la que pertenece. La segunda fase es la clasificación, cuando llega un nuevo dato o un nuevo conjunto de datos, se calcula la distancia entre este y cada uno de los puntos de entrenamiento almacenados. Luego se elige los K puntos más cercanos (vecinos) y se le asigna al nuevo punto la clase que mayor frecuencia tiene entre los puntos elegidos.

Para posicionamiento en interiores con KNN, el Fingerprint del índice de potencia para cada punto coordinado es utilizado para la fase de entrenamiento y los datos o mediciones posteriores son utilizados en la segunda fase [22–27]. Otra forma muy común de utilizar el algoritmo como se ve en los trabajos ya citados anteriormente, es utilizar un porcentaje de los puntos del Fingerprint para entrenamiento y otro porcentaje menor para las pruebas del KNN.

La técnica KNN también puede ser utilizada con carácter regresivo promediando los K vecinos mas cercanos y obteniendo así un valor intermedio a las clases. De esta manera el algoritmo ya no funcionaría como un clasificador sino como un estimador [28, 29].

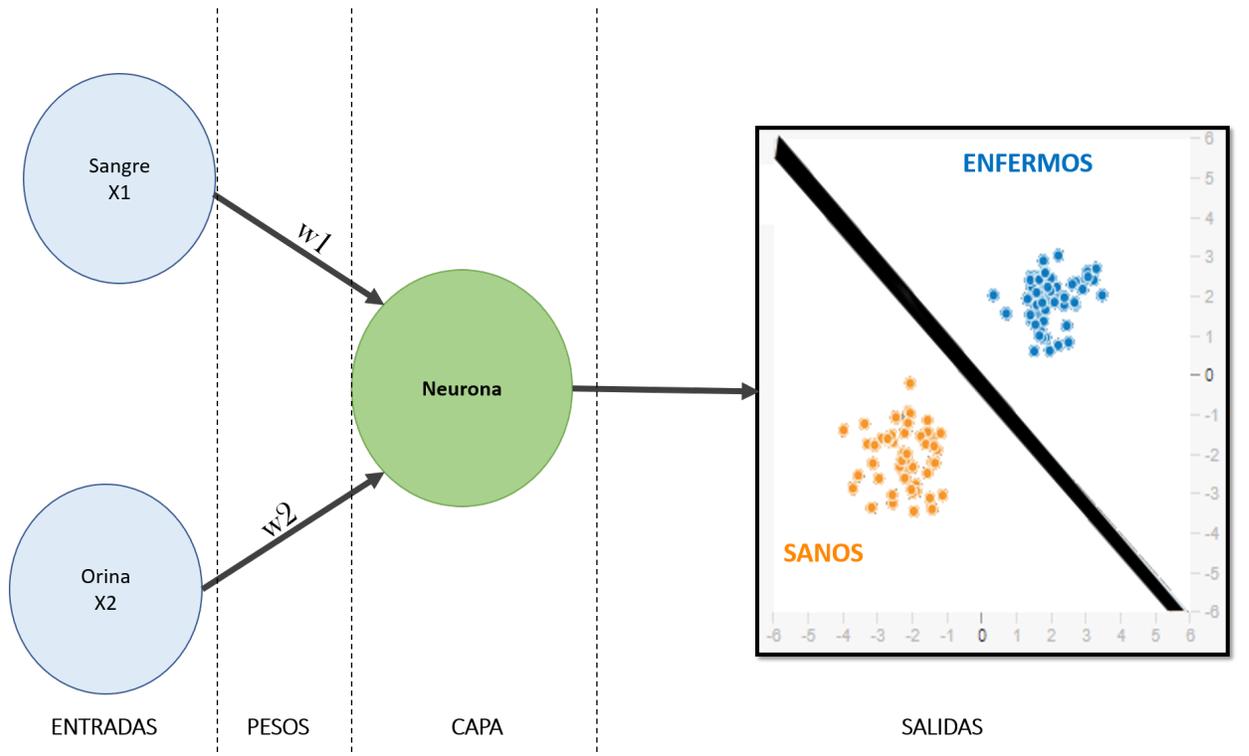
Durante este trabajo se utilizó el enfoque de KNN regresivo para comparar el desempeño de las metodologías propuestas.

### 2.3.2. Aprendizaje Profundo (Deep Learning)

El Deep Learning es una técnica de aprendizaje automático que recibe un conjunto de datos de entrada para que un computador aprenda a identificar o clasificar los datos mediante ejemplos, de una manera similar a como aprendemos los humanos. De esta manera se puede entrenar un sistema para reconocer y clasificar tipos de vehículos, clases de objetos, tonos de voz, y un sinnúmero de posibilidades, con un porcentaje de acierto que en ocasiones supera al humano. Aunque las teorías del Deep Learning existen desde los años 80, solo hasta hace algunos años se han empezado a implementar, ya que estas requieren de un alto costo computacional para su entrenamiento [30, 31].

El Deep Learning trabaja principalmente haciendo uso de redes neuronales, el término Deep (profundo), es empleado precisamente para hacer alusión al número de capas ocultas de la red.

Existen varios tipos de arquitecturas de redes neuronales utilizadas para el Deep Learning como Feed Forward Network, Back Forward Network, Belief Network, Convolutional Network, Residual Network [32, 33], entre otras. Teniendo en cuenta que para esta investigación se cuenta a nivel computacional con procesadores multi-nucleo gama media, se optó por utilizar Deep Learning con la arquitectura de red neuronal Feed Forward Network y función de activación sigmoide. Esta es una arquitectura que se caracteriza dentro de las anteriormente



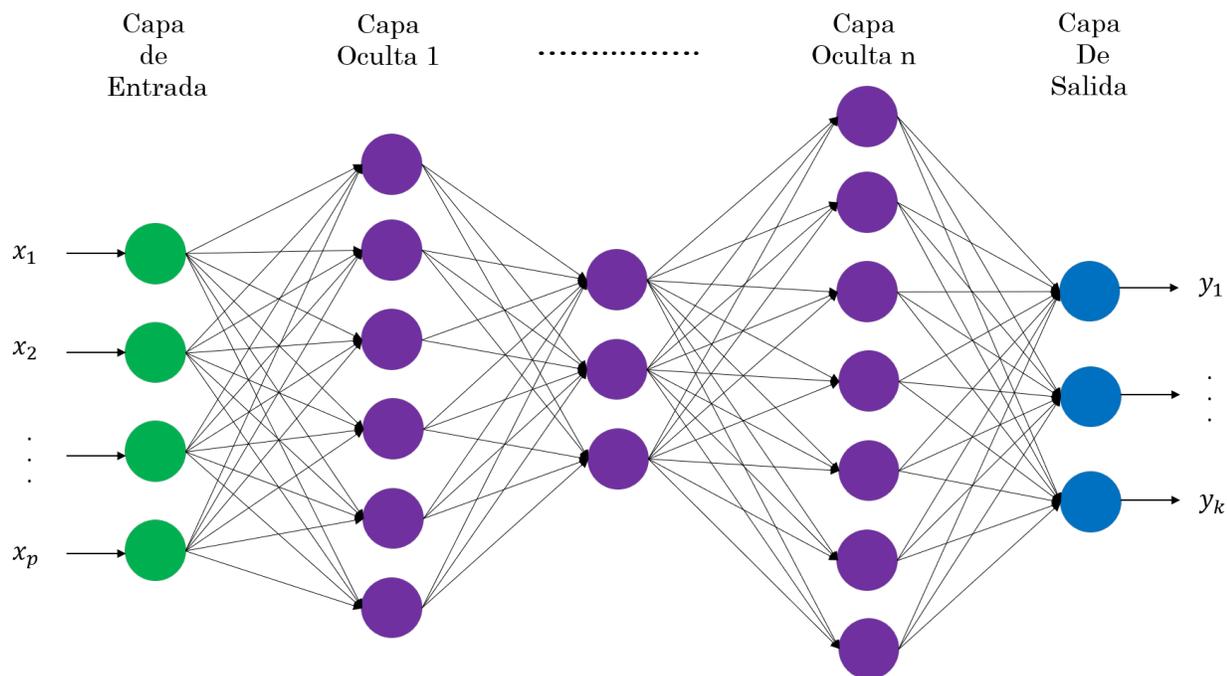
**Figura 2-1.:** Perceptrón simple con dos entradas y una salida

mencionadas por su bajo consumo de recursos a nivel computacional, tanto en la etapa de entrenamiento como en la etapa de prueba.

Las redes neuronales se comportan como un modelo computacional, con un conjunto de entradas que pueden viajar a través de una configuración de neuronas y capas hasta llegar a una salida. Como entradas de la red consideramos comúnmente la información conocida como datos, a la salida de la red esperamos encontrar una clasificación o estimación de acuerdo a la información de la entrada. Teniendo en cuenta el error en la salida, la red es reconfigurada hasta obtener una salida cercana a la esperada, a este proceso se le conoce como entrenamiento de la red y es el que mayor costo computacional requiere.

Imaginemos por ejemplo que tenemos una base de datos de 2000 pacientes con los resultados de un análisis de la sangre y concentración de PH en la orina y queremos de acuerdo a esto determinar cuales pacientes están enfermos y cuales sanos. Utilizando la red neuronal mas simple, el Perceptrón, podemos realizar una clasificación de los pacientes tomando como entrada la información de la sangre y la orina. En la Figura 2-1 podemos observar la configuración de la red con dos entradas, una neurona y una salida. La conexión entre las entradas y la neurona esta sujeta a los pesos  $w_1$  y  $w_2$ , estos valores son reajustados hasta obtener a la salida una respuesta como la observada en la figura.

La configuración Feed Forward es una arquitectura unidireccional compuesta por un conjunto de neuronas como la vista anteriormente en el Perceptrón, pero que extiende su configuración



**Figura 2-2.:** Red Neuronal Feedforward

a varias capas y neuronas por capa como se observa en la Figura 2-2. Como vemos en la figura, a partir de las capas ocultas cada neurona esta interconectada con todas las neuronas de la capa anterior con un peso asociado hasta llegar a la capa de salida.

## 2.4. Técnicas de posicionamiento en interiores

Como ya fue mencionado al inicio de este trabajo, muchas han sido las formas de abordar el problema de posicionamiento en interiores utilizando diferentes tecnologías. En este capítulo hacemos un revisión del Estado del Arte para las tecnologías utilizadas en IPS.

### 2.4.1. Posicionamiento con Pseudolites

Debido a que los receptores GPS presentan problemas al intentar conectarse en espacios cerrados, algunos trabajos han optado por sistemas de pseudolites o también llamados pseudo-satélites [12,34,35] los cuales son una configuración de antenas en el exterior de un edificio que simulan el comportamiento del GPS en el interior de este. Sin embargo, la implementación de éste tipo de técnica requiere de una infraestructura de antenas existente para su correcto funcionamiento o adecuaciones estructurales complejas alrededor de las edificaciones.

### 2.4.2. Posicionamiento con Campos Magnéticos

Como alternativa al posicionamiento con Pseudolites, en [4,36] se presentan técnicas basadas en la generación de campo magnético artificial utilizando una corriente alterna (AC por sus siglas en inglés) y una bobina receptora móvil en la cual es inducida una corriente debido al campo magnético generado por las bobinas transmisoras. Esta corriente es utilizada para realizar una estimación de la posición en un espacio cerrado de pruebas de 3m x 3m, logrando una precisión de alrededor de 10cm. Otros autores como [5], han propuesto un enfoque similar pero utilizando una corriente directa (DC por sus siglas en inglés) y como receptor móvil un sensor de campo magnético. El área de pruebas de este es de 15m x 15m y el error promedio en la posición es de 0.5m. Sin embargo para lograr una estimación de la posición en lugares de mayor tamaño a los presentados en las pruebas, se requiere un aumento considerable de la corriente necesaria para la generación de campos magnéticos adecuados para cubrir el área de interés, lo cual implica costos, tanto energéticos como de implementación.

### 2.4.3. Posicionamiento con Luz Visible

Algunos autores han propuesto técnicas utilizando sensores o receptores de luz y diferentes configuraciones de bombillos o paneles LED para realizar una buena estimación de la posición. En [37,38] se valen de la potencia de la señal lumínica recibida, utilizando modulación o configuración de las señales emitidas por los LED. Para brindar una mayor precisión a este tipo de estimación, autores como [39] proponen técnicas alternativas que no requieran de la potencia de la señal incidente como la técnica TDOA explicada con anterioridad en las técnicas de RF, en este caso para señales de luz. Para brindar una mayor robustez y garantizar una mayor precisión en la estimación de la posición, en [40] se propone un sistema similar a los mencionados inicialmente pero con la adición de un acelerómetro como sensor para estimar la posición en tres dimensiones. Aunque estas técnicas representan un error en la estimación de la posición del orden de centímetros, siguen teniendo la limitante de una infraestructura preconfigurada y la necesidad de la energía eléctrica, lo que podría ser una gran dificultad en situaciones de emergencia.

### 2.4.4. Posicionamiento con Unidades de Medida Inercial

Las unidades de medida inercial (IMU por sus siglas en inglés) son sensores que combinan un acelerómetro y un giróscopo permitiendo con esta combinación realizar una estimación de la posición. En [41] muestran como haciendo uso de una IMU y con un mapa previamente almacenado en una base de datos, es posible lograr una estimación de la posición con poca dependencia de una infraestructura previa, lo cual puede ser de gran utilidad en situaciones de emergencia como incendios, donde otras técnicas como las mencionadas anteriormente podrían correr un alto riesgo debido a daños o afectaciones en la infraestructura. Sin embargo, debido al carácter acumulativo del cálculo de la posición por parte de la IMU, hay una

alta probabilidad de que la posición calculada se desvíe de la posición real debido a la acumulación sucesiva de errores. Aunque se han propuesto diseños de IMU con un tamaño más pequeño y comunicación inalámbrica integrada para evitar costos de hardware adicional [42], el comportamiento de este sigue siendo similar en cuanto a acumulación del error. Otras alternativas como crear un sistema con múltiples IMUs han sido presentados en la literatura como [42], logrando una mejoría en la precisión de la posición, pero no dando una solución definitiva, ni robusta a la prevención del error acumulativo que estos presentan.

### 2.4.5. Posicionamiento con Técnicas de Visión Artificial

Diferentes técnicas de procesamiento de imágenes realizan estimación de la posición valiéndose de cámaras, condiciones especiales de luz y apoyándose de otras metodologías de estimación de la posición existentes. Muchas de estas se valen de marcas o pistas agregadas previamente en el ambiente, con las cuales se puede desarrollar un sistema de referencia como lo proponen los autores [43]. Pero adquirir la posición solo con marcas puede ser complejo debido al número de marcas a utilizar y la línea de visión de la cámara, por lo cual esta técnica es usualmente combinada con otras como la técnica de Odometría [44]. La Odometría calcula la posición relativa teniendo como referencia un punto inicial de movimiento, esta suele ser muy propensa a la acumulación de errores los cuales se tratan de corregir con las etiquetas o referencias procesadas por imágenes. También valiéndose de la Odometría pero esta vez realizando una captura de imágenes y un mapeo en tiempo real para futuros recorridos han sido propuestas técnicas de posicionamiento como [45]. Otros sensores han sido combinados con el procesamiento de imágenes como los sensores de unidad inercial; utilizando una sola cámara y algoritmos de corrección de errores se busca lograr una estimación de la posición aceptable como es presentado en [46, 47]. Existen otras técnicas que hacen uso de dos o más cámaras llamadas técnicas de Visión Estereo, las cuales facilitan sin el uso de sensores adicionales el cálculo de la posición como [48]. Sin embargo esta técnica al igual que las otras presentan limitantes como la necesidad de una infraestructura especial para su funcionamiento y la alta capacidad de procesamiento que es necesaria para el trabajo con imágenes.

### 2.4.6. Posicionamiento con Ultrasonido

El posicionamiento por ultrasonido se ha estudiado ampliamente debido al bajo costo de este tipo de sensores algunos autores proponen el uso de sensores de ultrasonido y la técnica TOA (Time Of Arrival) [10]. Para calcular el tiempo de viaje de la señal entre el transmisor y el receptor. Esta técnica bajo condiciones con obstáculos se ve altamente afectada en su línea de vista, además debido a la frecuencia de funcionamiento de estos sensores, los ruidos o perturbaciones sonoras en el ambiente pueden afectar considerablemente el sistema.

### 2.4.7. Posicionamiento con Radio Frecuencia

Diversas técnicas y metodologías en esta corriente han sido diseñadas y probadas en los últimos años. Técnicas como DoA (Direction of Arrival) utilizan la dirección entre transmisor y receptor para hacer una estimación de la posición [49]. El problema con este tipo de técnicas es que los receptores requieren línea de vista. Técnicas como TDOA (Time Differential Of Arrival) la cual consiste en calcular el tiempo de viaje de la señal entre un transmisor, varios receptores y hacer uso de este tiempo de manera conjunta con la velocidad de la onda para hacer una estimación de la posición, han sido propuestas [42]. La mayor dificultad de este tipo de técnicas radica en el alto grado de sincronía de tiempo que debe haber entre los receptores, lo cual aumenta las probabilidades de que cualquier pérdida en la sincronía genere grandes errores en el posicionamiento.

Las técnicas más estudiadas se valen de la señal WiFi adquirida de puntos de acceso (APs por sus siglas en inglés) instalados en la edificación y realizan un posicionamiento con la potencia de la señal o Received Signal Strength (RSS), haciendo además un previo mapeo de la zona o toma de huella del lugar conocido como Fingerprinting [27, 50–60]. Esto debido en gran parte a que la tecnología WiFi se encuentra comúnmente en muchos lugares cerrados y edificios. Sin embargo, hay zonas que no cuentan con este tipo de señal, además, debido a su longitud de onda, las señales emitidas por los transmisores son atenuadas fácilmente ante obstáculos como las paredes y otros tipos de objetos de diferentes materiales comúnmente encontrados en interiores [61]. Para solventar esta dificultad algunos autores han propuesto utilizar señales de una menor frecuencia como GSM [24–26, 62] y FM [16, 17, 23, 63–65]. Los trabajos mencionados anteriormente que son basados en FM requieren de la instalación de algún tipo de infraestructura como antenas para transmitir sus propias señales FM y facilitar la estimación de la posición. En [63] se hace uso de las radio estaciones comerciales FM lo que evita la instalación de una infraestructura propia para la generación de las señales.

La gran mayoría de técnicas o algoritmos de posicionamiento utilizados con Fingerprint en los trabajos anteriormente citados se enfocan en técnicas de aprendizaje de máquina como KNN, wKNN, SVM y GP para clasificar y determinar la posición estimada. Esto implica un alto costo computacional al momento de calcular la posición.

Para solucionar esto, algunos autores han propuesto utilizar Deep learning en el contexto de redes WiFi [66–68]. El cual, si bien requiere un alto costo computacional para su entrenamiento, simplifica de manera considerable la ejecución de los algoritmos de prueba. Sin embargo, este enfoque no ha sido explorado hasta el momento utilizando señales FM de radio estaciones comerciales existentes y mucho menos de radio estaciones AM en donde no se encontraron técnicas en la literatura aún.

### 2.4.8. Sistemas de Posicionamiento Híbridos

Para aumentar la exactitud en la medición de la posición de un objeto en espacios cerrados hay varias propuestas en la literatura en las cuales se mezclan algunas de las técnicas men-

cionadas con anterioridad. Como en [69] donde los autores proponen una combinación de una señal GPS, una IMU y de procesamiento de imágenes para realizar una estimación de la posición en un espacio abierto. Esta técnica puede ser adaptada para interiores utilizando pseudolites, pero el uso de imágenes propuesto se puede ver afectado por situaciones de riesgo como incendios o de poca visibilidad. Otra de las técnicas propuestas es la combinación de posicionamiento RFID apoyándose con el procesamiento de imágenes, en [70] los autores proponen un sistema utilizando ambas técnicas para un espacio cerrado, allí valiéndose de la captura de movimiento y posición de un Kinect y de Tags RFID se estima la posición de una persona. Aunque este enfoque podría ser de utilidad en espacios cerrados de área pequeña tiene algunas limitantes como el hecho de la existencia de una infraestructura previa, la distancia de alcance efectiva y precisión del Kinect y la limitada cantidad de tags RFID utilizados. En otra técnica encontrada en la literatura [71] los autores muestran como a través de la adquisición de imágenes, la utilización de un IMU y realizando escaneo de área con láser, un grupo de UAVs logran mejorar la precisión en su posicionamiento. Técnicas más robustas haciendo uso de múltiples sensores y sistemas de comunicación han sido propuestas [72] pero con la gran limitante de altos costos de instalación e infraestructura.

# 3. Metodología

En este capítulo se describe la metodología implementada en este trabajo. En la Figura 3-1 tenemos un diagrama de bloques que resume las principales etapas de la metodología. Iniciando por un proceso de adquisición de datos con el índice de potencia de radio estaciones comerciales AM y FM. Construyendo con esta información una base de datos con el índice de potencia RSS para cada uno de los 23 puntos de un plano de experimentos. Esta información es procesada de manera posterior para ser utilizada en el entrenamiento de los algoritmos de Deep Learning y KNN Regresivo que permitirán estimar la posición de acuerdo al índice de potencia de la señal y finalmente implementar las técnicas resultantes en un sistema embebido.

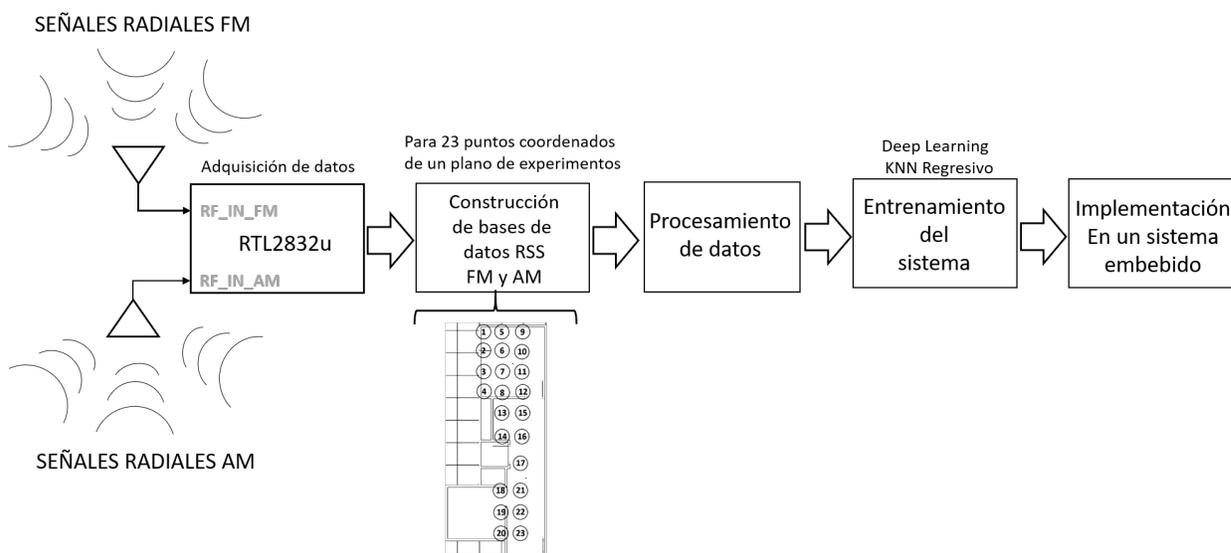
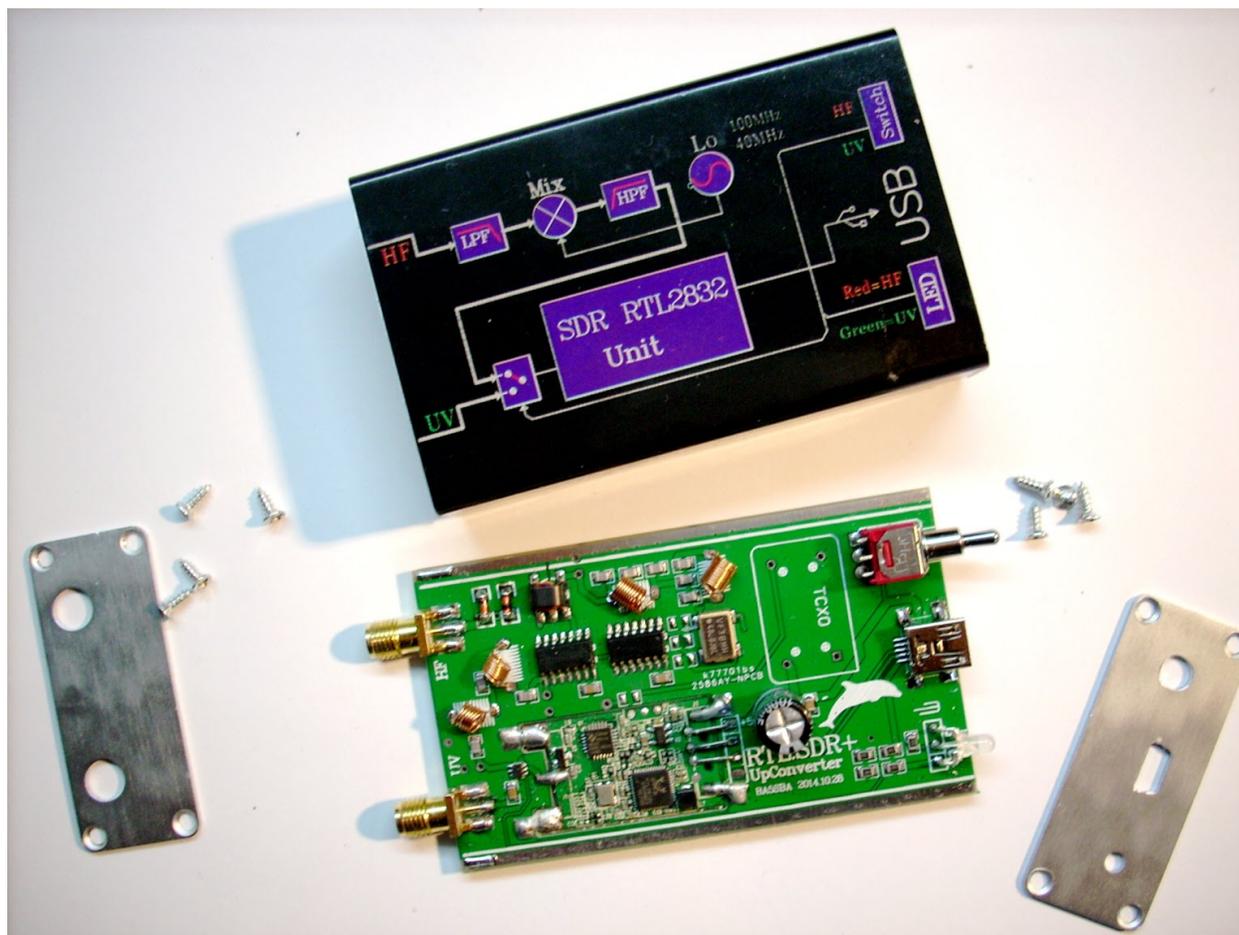


Figura 3-1.: Diagrama de bloques Metodología

## 3.1. Hardware

Para la adquisición de datos se utilizó un sistema de Radio Definido por Software (SDR) RTL-SDR basado en el chip RTL2832u presentado en la figura 3-2. Este dispositivo está compuesto por 2 chips: un sintonizador de señales de radio Raphael Micro R820T que de manera conjunta con una antena permite la adquisición de señales de radio en un rango de frecuencia entre 100 Khz a 1700 Mhz y un chip conversor ADC de 8-bit Realtek RTL2832u.



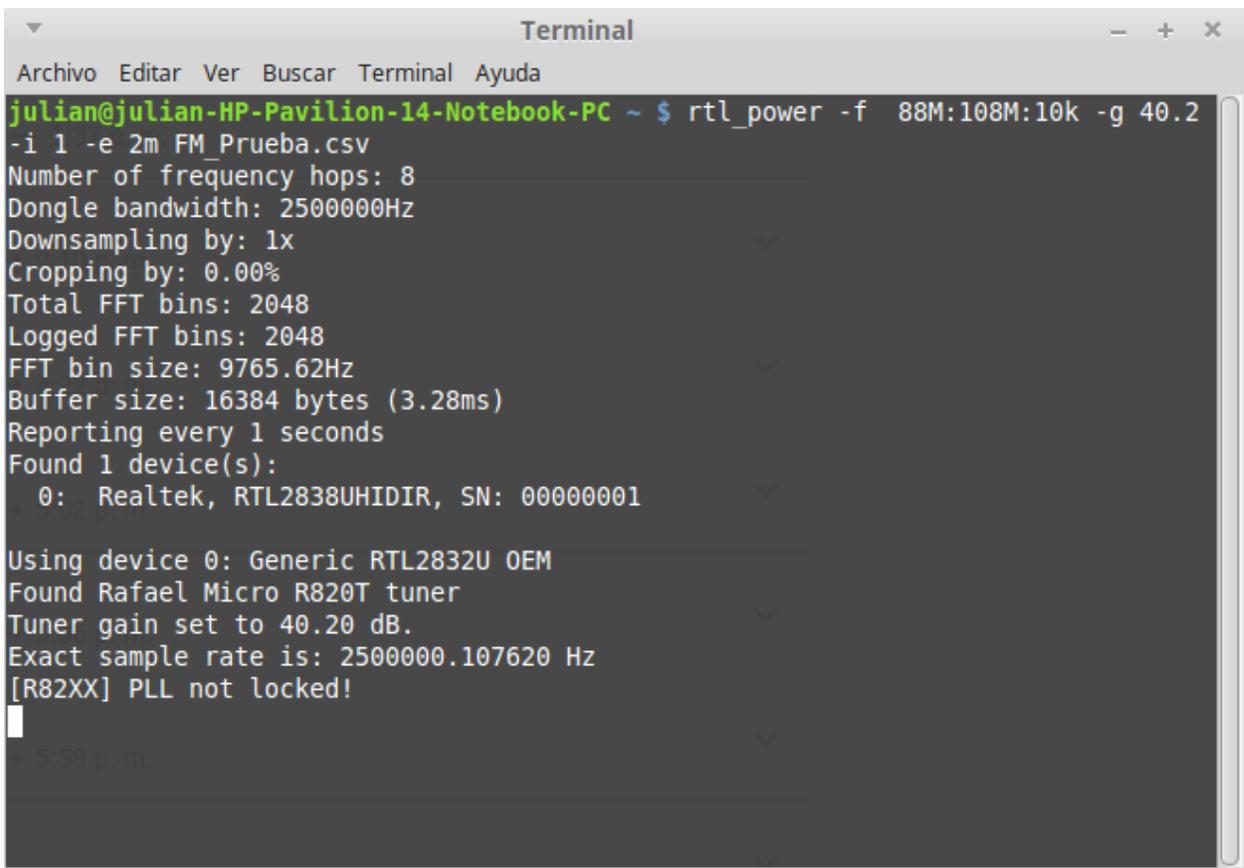
**Figura 3-2.:** Sistema de Radio Definido por Software RTL2832u

El RTL2832u se encarga de transformar la señal analógica recibida por el chip R820T de una antena telescópica común, sacada de un radio receptor de bajo costo con una longitud de 52 cm de largo para adquisición de las señales FM y de una antena pasiva sintonizable de onda media Tecsun AN-200 para AM, en una señal digital compuesta por dos componentes uno Real/Inphase(I) y uno Imaginario/quadrature(Q). Esta señal es por último acondicionada y llevada a una interfaz USB permitiendo así leer, almacenar y procesar esta información en un computador o sistema embebido.

Para leer la información del RTL-SDR se puede usar el software SDR#, Matlab y GNU Radio para Windows OS y GQRX para Mac OS y Linux. Una alternativa al uso de software comercial es la posibilidad de aplicar comandos propios de los drivers desde una consola como CMD de Windows o Terminal de Linux

En este trabajo se optó por esta última posibilidad, ya que a través de consola podemos adquirir los datos brutos utilizando comandos como *rtl\_sdr*, *rtl\_power*, *rtl\_fm* y *rtl\_test*. El comando utilizado durante este trabajo fue *rtl\_power*, el cual entrega información del espectro de frecuencia como un índice de potencia de la señal o RSS en un rango de frecuencias

deseado. Esta información es usada para capturar las características de un punto de interés. El sistema operativo Linux fue elegido en este trabajo con el objetivo de que los experimentos realizados en una computadora personal sean migrados posteriormente a un sistema embebido como el Raspberry Pi que permita ampliar el espectro de aplicación del sistema de posicionamiento propuesto como la navegación de drones o robots móviles en interiores. Para realizar la lectura de datos a través de un terminal de Linux, es necesario instalar inicialmente los controladores USB requeridos para que el dispositivo sea leído correctamente y posteriormente descargar, modificar, instalar y compilar las librerías necesarias para la captura de información AM y FM. Este procedimiento es descrito en el Anexo A



```
Terminal
Archivo Editar Ver Buscar Terminal Ayuda
julian@julian-HP-Pavilion-14-Notebook-PC ~ $ rtl_power -f 88M:108M:10k -g 40.2
-i 1 -e 2m FM_Prueba.csv
Number of frequency hops: 8
Dongle bandwidth: 2500000Hz
Downsampling by: 1x
Cropping by: 0.00%
Total FFT bins: 2048
Logged FFT bins: 2048
FFT bin size: 9765.62Hz
Buffer size: 16384 bytes (3.28ms)
Reporting every 1 seconds
Found 1 device(s):
 0: Realtek, RTL2838UHIDIR, SN: 00000001

Using device 0: Generic RTL2832U OEM
Found Rafael Micro R820T tuner
Tuner gain set to 40.20 dB.
Exact sample rate is: 2500000.107620 Hz
[R82XX] PLL not locked!
```

Figura 3-3.: Aplicación del comando `rtl_power` para FM en terminal de Linux

### 3.1.1. Comando `rtl_power`

El comando `rtl_power` permite realizar un muestreo (recolección de datos) en un rango de frecuencia deseado, utilizando una resolución o tamaño de paso específico durante un intervalo de tiempo asignado. Por ejemplo en la Figura 3-3, podemos ver la configuración de un muestreo de datos en FM entre 88 Mhz y 108 Mhz con una resolución de 10 khz durante 2 minutos, tomando muestras cada segundo. Esto quiere decir que el dispositivo es configurado

para que cada segundo a partir del segundo 1, hasta cumplidos 2 minutos tome muestras en las frecuencias 88.8 Mhz, 88.81 Mhz, 88.82 Mhz, . . . , 108.79 Mhz, hasta 108.8 Mhz. Las cuales serán almacenadas en un archivo con formato de datos CSV. Sin embargo es importante entender como se realiza este proceso, ya que los valores solicitados por comandos, están sujetos a las condiciones y limitaciones reales de funcionamiento del dispositivo. Al digitar el comando `rtl_power` con la frecuencia, tiempo y resolución deseada, este llama al script `rtl_power.c` , el cual realiza la siguiente secuencia con los valores de entrada.

**Organizar la solicitud de muestreo:** Teniendo en cuenta que el ancho de banda máximo del dispositivo RLT2832u es de 2.8 Mhz, la información a muestrear debe ser organizada en secciones (hops) que no excedan esta condición. Para esto, se hace inicialmente una resta entre la frecuencia superior y la frecuencia inferior. Luego, el valor resultante es dividido por un valor entre 1 y 1500 (hops) hasta que el resultado arroje un valor inferior a los 2.8 Mhz. En el ejemplo el anterior la diferencia entre la frecuencia superior e inferior es de 20 Mhz, la cantidad de hops que permiten realizar el muestreo es de 8 hops, arrojando un ancho de banda de 2.5 Mhz.

**Determinar la resolución:** Aunque dentro del comando se asigna una resolución, este valor sirve solo como referencia para que el dispositivo calcule la resolución real posible. Debido a que el dispositivo trabaja con potencias de 2, la resolución real dependerá de estas potencias. El algoritmo `rtl_power.c` determina esta resolución dividiendo de manera iterativa por potencias de 2 la diferencia de la frecuencia superior y la inferior hasta encontrar el valor inferior mas próximo a la resolución deseada. En el ejemplo anterior divide 20 Mhz por potencias de 2 iterativamente hasta llegar a un valor de 2048, con el cual se obtiene una resolución aproximada a los 10 khz de 9.76562 khz. Esto quiere decir que las muestras serán tomadas realmente con un espaciamento equivalente en frecuencia de esta última resolución.

## 3.2. Montaje Experimental

Para el montaje experimental se definió la utilización de un área de experimentos organizada como un plano de dos dimensiones y con 23 puntos coordenados utilizados para la recolección de muestras y construcción del Fingerprint de entrenamiento. En estos puntos se recolectaron en un primer momento muestras en el espectro de FM, luego en un segundo momento muestras en el espectro de AM y en un último momento muestras conjuntas en ambos espectros. Las muestras recogidas en cada momento sirvieron como insumo para el entrenamiento y prueba de los algoritmos propuestos con Deep Learning y del algoritmo comparativo KNN Regresivo. A continuación detallaremos un poco más el montaje experimental.

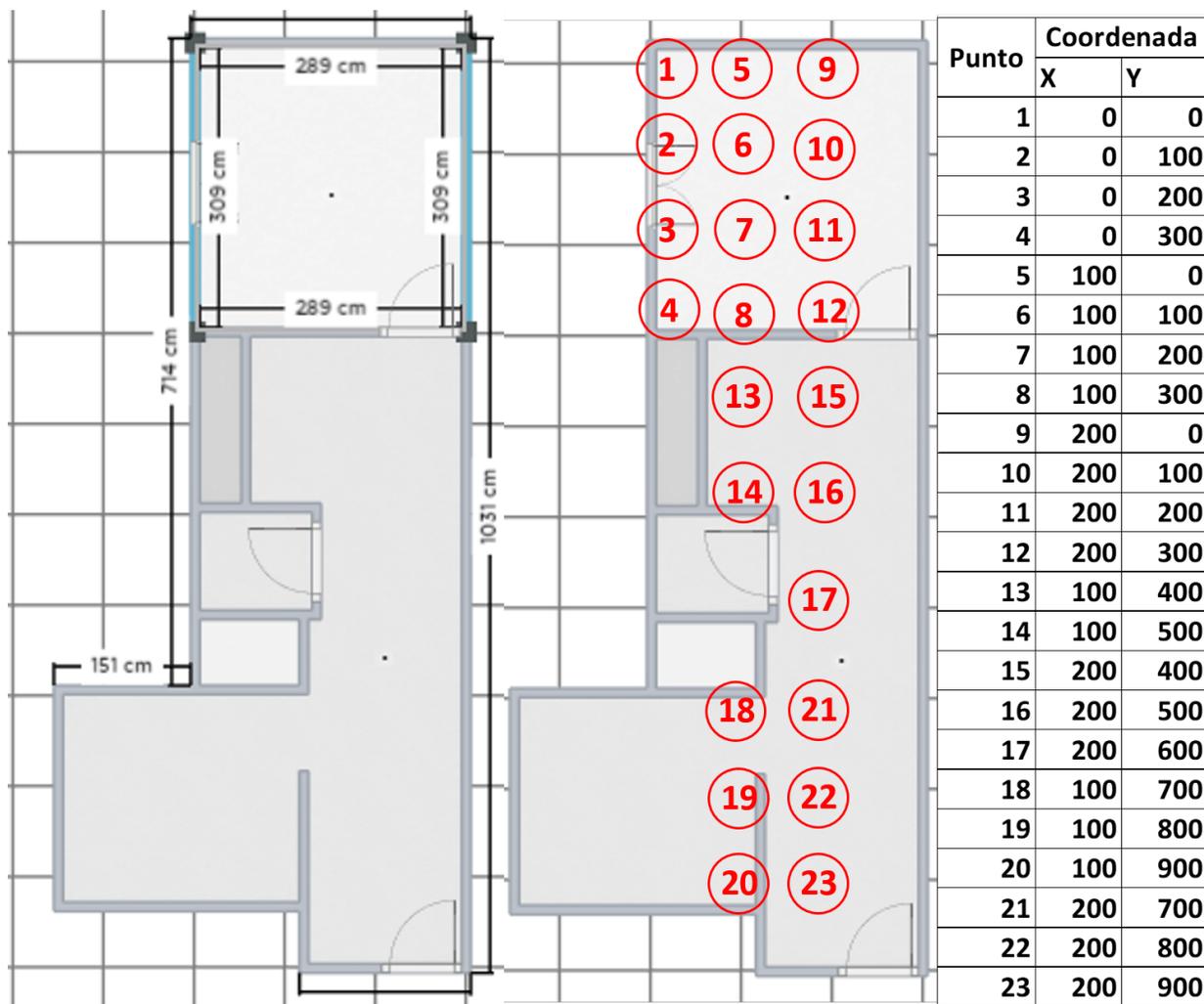


Figura 3-4.: Plano del área de experimentos

### 3.2.1. Área

Para el montaje experimental se utilizó un espacio cerrado ubicado en un tercer piso residencial y habitado el cual cuenta con muros de concreto con espesor de 10cm, ventanas y diversos obstáculos como muebles y electrodomésticos. Como se observa en la Figura 3-4, este está dividido básicamente en 4 espacios: dos habitaciones, una cocina y un pasillo, abarcando estos un área de 289 cm X 1011 cm. En este espacio se distribuyeron 23 puntos cada uno correspondiente a una coordenada (x,y) y distanciado de sus puntos vecinos 100 cm.

### 3.2.2. Radio Estaciones Utilizadas

Para el experimento, se tuvo en cuenta todas las radioestaciones FM y AM comerciales presentes en la ciudad de Medellín, Colombia.

## FM

Las radioestaciones FM utilizadas están distribuidas en el espectro radioeléctrico desde los 88.9 Mhz hasta los 107.9 Mhz, para un total de 29 radioestaciones que aportan información del índice de potencia RSS en los diferentes puntos utilizados.

## AM

Las radioestaciones AM utilizadas están distribuidas en el espectro radioeléctrico desde los 590 Khz hasta los 1590 Khz, para un total de 25 radioestaciones que aportan información del índice de potencia RSS en los diferentes puntos utilizados.

### 3.2.3. Construcción de base de datos con el índice de potencia RSS

La base de datos con el índice de potencia RSS de las radioestaciones fue construida ubicando el dispositivo RTL-SDR y la antena receptora, en cada uno de los de puntos o coordenadas mostradas en la Figura 3-4. Para cada coordenada se realizó un total de 6 sesiones de toma de datos. Estas sesiones fueron realizadas en días y horas diferentes con el fin de incluir las variaciones causadas por el entorno y el clima. El tiempo de adquisición fue de 5 minutos por sesión de toma de datos a una frecuencia de 1 muestra/segundo, para un total de 300 muestras por sesión. Teniendo así para cada coordenada un total de 1800 muestras. Las muestras obtenidas para FM contienen información de potencia de las 29 radioestaciones FM. Por su parte para AM contienen información de potencia de las 25 radioestaciones AM. Cada muestra contiene los datos del índice de potencia de las frecuencias encontradas, para el caso de FM cada 10khz entre 88.4 Mhz y 107.9 Mhz. Para el caso de AM cada 10Khz entre 560 Khz y 1840 Khz. Todas las muestras de cada sesión de toma de datos fueron obtenidas automáticamente con el comando *rtl\_power* de Linux y almacenadas en un archivo CSV. Para adquisición de datos AM se utilizó el comando de la siguiente manera:

```
rtl_power -f 560k:1840k:10k -g 40.2 -i 1 -e 5m -D 04_01AM.csv
```

En donde la letra -f indica el rango de frecuencia a medir y el tamaño de paso, el cual empieza en 560 Khz y termina 1840 Khz tomando muestras cada 10 Khz. La letra -g indica la ganancia en decibelios que le dará a la señal recibida el RTL2832u, en este caso elegimos una ganancia de 40.2 dB tanto para AM como FM. La letra -i indica cada cuanto se van a tomar los datos y -e durante cuanto tiempo, en este caso estamos tomando datos cada 1 segundo durante 5 minutos. El tiempo mínimo de toma de datos que permite el dispositivo es 1 segundo. La letra -D es necesario utilizarla para habilitar el canal de muestreo directo que es el que permite realizar mediciones entre 100 Khz y 30 Mhz sino se escribe esta letra el dispositivo tomará mediciones únicamente entre 25 Mhz y 1700 Mhz. Por último copiamos

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	
1	2018-05-25	00:45:05	560000	1840000	10000.00	4928	-41.36	-41.46	-40.39	-22.69	-40.12	-41.11	-41.20	-41.15	-41.07	-40.43	-38.34	-16.36	-38
2	2018-05-25	00:45:06	560000	1840000	10000.00	9408	-41.47	-41.19	-39.35	-22.54	-38.92	-40.95	-41.12	-40.98	-40.82	-40.03	-37.62	-16.20	-37
3	2018-05-25	00:45:07	560000	1840000	10000.00	9728	-41.45	-41.54	-41.00	-22.77	-40.86	-41.30	-41.19	-41.15	-40.94	-40.42	-38.35	-16.27	-38
4	2018-05-25	00:45:08	560000	1840000	10000.00	9728	-41.33	-41.32	-39.89	-22.50	-39.59	-40.93	-41.20	-41.07	-40.97	-40.38	-38.14	-16.28	-38
5	2018-05-25	00:45:09	560000	1840000	10000.00	9600	-41.44	-41.34	-40.30	-22.70	-40.06	-41.14	-41.20	-41.20	-41.11	-40.46	-38.46	-16.25	-38
6	2018-05-25	00:45:10	560000	1840000	10000.00	9600	-41.41	-41.55	-41.03	-22.77	-40.95	-41.21	-41.27	-41.08	-40.94	-40.13	-37.65	-15.99	-37
7	2018-05-25	00:45:11	560000	1840000	10000.00	9728	-41.33	-41.34	-39.40	-22.49	-39.01	-41.00	-41.14	-41.21	-41.04	-40.50	-38.62	-16.37	-38
8	2018-05-25	00:45:12	560000	1840000	10000.00	9728	-41.39	-41.34	-39.92	-22.59	-39.60	-41.06	-41.18	-41.18	-41.03	-40.55	-38.85	-16.34	-38
9	2018-05-25	00:45:13	560000	1840000	10000.00	9536	-41.32	-41.27	-39.26	-22.47	-38.85	-40.97	-41.14	-41.06	-41.05	-40.52	-38.79	-16.39	-38
10	2018-05-25	00:45:14	560000	1840000	10000.00	9600	-41.44	-41.21	-39.27	-22.55	-38.99	-40.98	-41.22	-41.14	-41.02	-40.58	-38.88	-16.34	-38
11	2018-05-25	00:45:15	560000	1840000	10000.00	9408	-41.40	-41.37	-39.98	-22.58	-39.70	-41.05	-41.12	-41.09	-40.99	-40.31	-37.86	-16.19	-38
12	2018-05-25	00:45:16	560000	1840000	10000.00	9472	-41.33	-41.48	-39.98	-22.67	-39.78	-41.10	-41.26	-41.21	-41.17	-40.77	-39.24	-16.44	-39
13	2018-05-25	00:45:17	560000	1840000	10000.00	9728	-41.25	-41.34	-39.72	-22.49	-39.41	-41.02	-41.15	-41.15	-41.02	-40.50	-38.69	-16.29	-38
14	2018-05-25	00:45:18	560000	1840000	10000.00	9728	-41.41	-41.50	-40.47	-22.66	-40.30	-41.26	-41.24	-41.21	-40.98	-40.37	-38.41	-16.13	-38
15	2018-05-25	00:45:19	560000	1840000	10000.00	9216	-41.48	-41.65	-41.10	-22.78	-40.89	-41.35	-41.30	-41.13	-41.12	-40.54	-38.49	-16.25	-38
16	2018-05-25	00:45:20	560000	1840000	10000.00	9408	-41.35	-41.19	-39.29	-22.56	-39.04	-40.99	-41.14	-41.02	-40.91	-40.38	-38.14	-16.26	-38
17	2018-05-25	00:45:21	560000	1840000	10000.00	9536	-41.26	-41.32	-39.32	-22.52	-39.07	-40.92	-41.10	-41.08	-40.92	-40.21	-37.79	-16.21	-37
18	2018-05-25	00:45:22	560000	1840000	10000.00	9408	-41.39	-41.29	-39.78	-22.51	-39.60	-41.07	-41.17	-41.16	-40.97	-40.35	-38.22	-16.37	-38
19	2018-05-25	00:45:23	560000	1840000	10000.00	9728	-41.43	-41.45	-40.47	-22.77	-40.35	-41.25	-41.12	-41.14	-40.94	-40.31	-37.53	-16.20	-37
20	2018-05-25	00:45:24	560000	1840000	10000.00	9792	-41.35	-41.34	-39.48	-22.55	-39.14	-40.99	-41.19	-41.20	-41.08	-40.53	-38.66	-16.17	-38
21	2018-05-25	00:45:25	560000	1840000	10000.00	9728	-41.32	-41.33	-39.69	-22.52	-39.33	-40.92	-41.22	-40.99	-40.91	-40.27	-37.66	-16.02	-37
22	2018-05-25	00:45:26	560000	1840000	10000.00	9536	-41.44	-41.67	-40.96	-22.76	-40.97	-41.17	-41.20	-41.08	-41.00	-40.21	-37.39	-16.22	-37

**Figura 3-5.:** Formato de archivo CSV con RSS del espectro radio-eléctrico

el nombre del archivo .csv que es el tipo de archivo a utilizar.

El comando anterior creará un archivo como el que se muestra en la Figura 3-5, el cual contiene información general del muestreo entre las columnas A y F, como la fecha, la hora, el rango de frecuencias, entre otros. De la columna G en adelante el archivo contiene la información del RSS de potencia del espectro radio-eléctrico para las frecuencias muestreadas. Por ejemplo en la columna J podemos observar información del RSS de la radio estación 590 AM “W Radio”, en la columna R información de la radio estación 670 AM “Antena 2”, de manera similar las columnas siguientes corresponden a las otras radio estaciones.

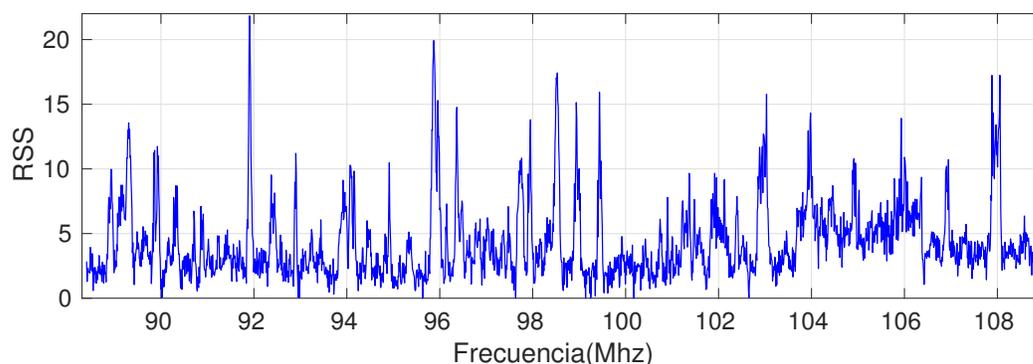
Para la adquisición de datos FM se utiliza el comando:

```
rtl_power -f 88.4M:108.8M:10k -g 40.2 -i 1 -e 5m 17_03FM.csv
```

## 3.3. Procesamiento de datos

### 3.3.1. FM

Para el procesamiento de datos en FM se utilizó un script realizado en el software Matlab que se encarga de unir los diferentes archivos CSV en una matriz que consolida la información de todos los muestreos realizados. Con la matriz consolidada se normalizaron los datos sobre un offset de cero para facilitar el tratamiento matemático.



**Figura 3-6.:** Espectro de magnitud FM entre 88.4 Mhz y 108.8 Mhz

**Tabla 3-1.:** Frecuencias Radio Estaciones Comerciales FM

Frecuencias FM (Mhz)					
89,21	89,9	90,3	90,97	91,91	92,4
92,9	93,97	94,42	94,9	95,89	96,42
96,93	97,92	98,94	99,42	99,94	100,4
100,9	101,42	101,92	102,92	103,93	104,32
104,98	106	106,37	107	108	

La gráfica de la información de la matriz consolidada nos presenta el espectro de magnitud en el rango de frecuencia de las radio estaciones como se observa en la Figura 3-6. Podemos observar que los picos de mayor potencia en el espectro están formados por los puntos de la gráfica que se encuentran cercanos a las frecuencias de las portadoras otorgadas a cada radio estación comercial, es decir, el ancho de banda ocupado por cada una de ellas. En la Tabla 3-1 se muestra la frecuencia de portadora de cada una de las radioestaciones FM utilizadas. El ancho de banda de cada radioestación es 200 khz, y el tamaño de paso del espectro capturado es de 10 khz. Como resultado tenemos un total de 20 puntos relevantes para cada radioestación. Teniendo en cuenta lo anterior hay información del muestreo que no presenta una relevancia significativa en el espectro por lo cual podemos simplificar la cantidad de puntos del Fingerprint solo con aquellos que se encuentran dentro del ancho de banda alrededor de las frecuencias fundamentales y así obtener únicamente 29 columnas de datos cada una correspondiente a una radioestación FM.

Aunque inicialmente se pensó en utilizar 19 de los 20 puntos relevantes alrededor de la frecuencia central para cada radio estación, luego de realizar algunas pruebas de entrenamiento del sistema, no se logró una convergencia satisfactoria, por lo cual se propuso reducir estos 19 valores realizando alguno de los siguientes 3 tratamientos de los datos:

### Sumatoria de datos

$$x_i = \sum_{j=-9}^9 f_{i,j} \quad \text{Donde } i \in \{1, \dots, 29\}$$

Se suman los 19 datos alrededor de cada frecuencia portadora de cada una de las radioestaciones, esta información se convierte la nueva matriz de Fingerprint.

### Promedio de datos

$$x_i = \frac{\sum_{j=-9}^9 f_{i,j}}{19} \quad \text{Donde } i \in \{1, \dots, 29\}$$

Se promedian los 19 datos alrededor de cada frecuencia portadora de cada una de las radioestaciones, esta información se convierte la nueva matriz de Fingerprint.

### Valor Máximo en rango

$$x_i = \max_{-9 \leq j \leq 9} (f_{i,j}) \quad \text{Donde } i \in \{1, \dots, 29\}$$

Se calcula el valor máximo entre los 19 datos alrededor de cada frecuencia portadora de cada una de las radioestaciones, esta información se convierte la nueva matriz de Fingerprint.

### 3.3.2. AM

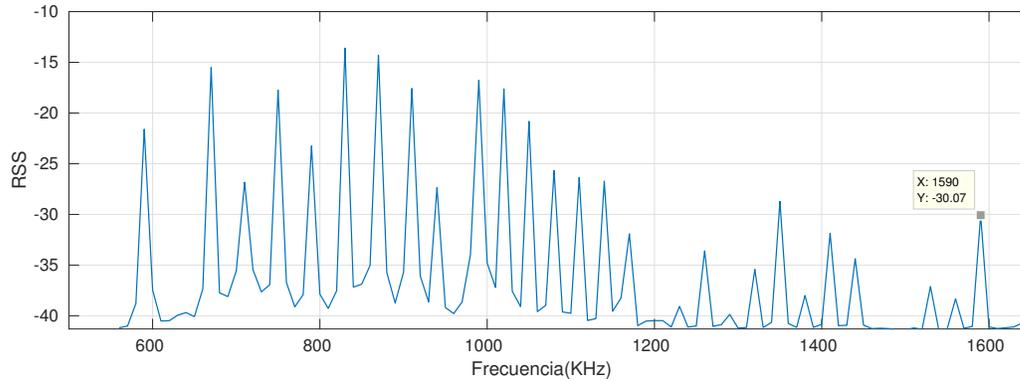
De manera similar a FM, para el procesamiento de datos en AM se utilizó un script realizado en el software Matlab que se encarga de unir los diferentes archivos CSV en una matriz que consolida la información de todos los muestreos realizados.

La gráfica de la información de la matriz consolidada nos presenta el espectro de magnitud en el rango de frecuencia de las radio estaciones AM como se observa en la Figura 3-7.

Podemos observar que los picos de mayor energía en el espectro están formados por los puntos de la gráfica ubicados en las frecuencias de las portadoras otorgadas a cada radio estación comercial AM.

En la Tabla 3-2 se muestra la frecuencia de portadora de cada una de las radioestaciones AM utilizadas.

Teniendo en cuenta en la Figura 3-7 que los mayores picos de la gráfica están ubicados exactamente en las frecuencias portadoras de la Tabla 3-2, podemos omitir los demás puntos registrados durante el Fingerprint, ya que no presentan información de relevancia. Para



**Figura 3-7.:** Espectro de magnitud AM tomando muestras cada 10 KHz entre 560 KHz y 1840 KHz

**Tabla 3-2.:** Frecuencias Radio Estaciones Comerciales AM

Frecuencias AM (KHz)									
590	670	710	750	790	830	870	910	940	990
1020	1050	1080	1110	1140	1170	1230	1260	1320	1350
1380	1410	1440	1530	1590					

seleccionar los datos a entrenar en el algoritmo basta con elegir aquellos que coincidan con las posiciones de un vector con la información de la Tabla **3-2** así:

$$x_i = (f_i) \quad \text{Donde } i \in \{1, \dots, 28\}$$

### 3.3.3. AM - FM

Para el procesamiento de datos con información de AM y FM se mezclaron las técnicas propuestas anteriormente para elegir la información relevante inicialmente de FM y posteriormente de AM. Luego se creó una matriz con la información conjunta FM - AM preprocesada, teniendo un total de 54 columnas con información de entrada para el entrenamiento. Las primeras 29 con la información de las radioestaciones FM y las 25 restantes con la información de las radio estaciones AM.

## 3.4. Entrenamiento

Para el entrenamiento del sistema se utilizó el criterio de validación cruzada 80-20 para los datos del Fingerprint. Tanto para las técnicas propuestas con Deep Learning como para la técnica de KNN Regresivo que servirá de comparación. Como se muestra en la Figura **3-8**

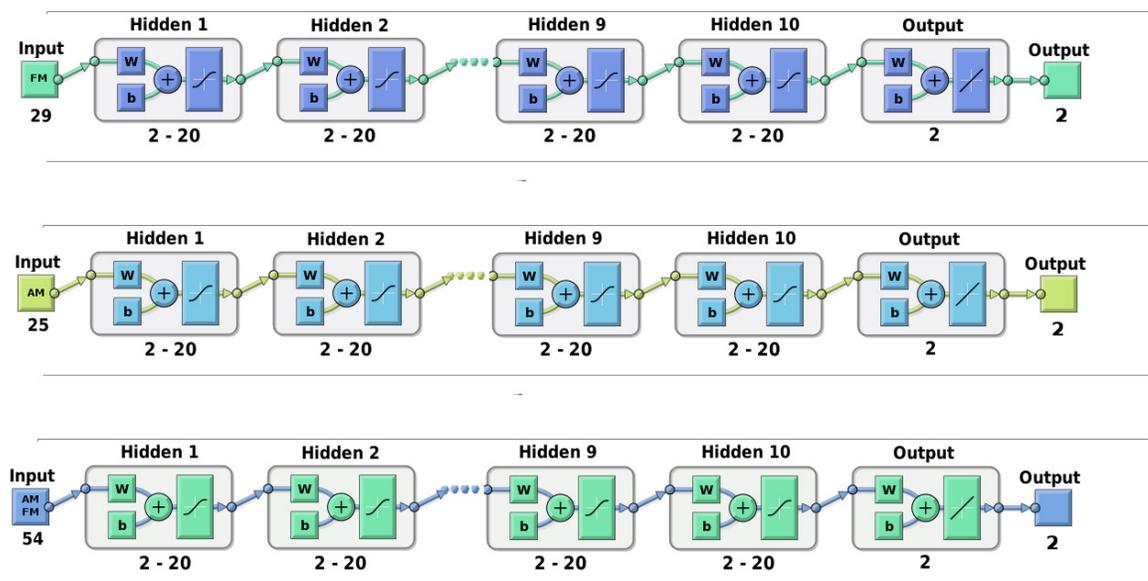


Figura 3-8.: Redes Neuronal Feedforward para FM, AM y AM - FM

para el experimento se realizó una exploración de la configuración de la red, variando las capas ocultas de la arquitectura desde 1 hasta 10 y el número de neuronas por capa entre 2 y 20 neuronas para un total de 180 posibles arquitecturas. Tomando como entradas la información procesada en FM, AM y AM - FM y obteniendo como salidas las coordenada (x,y) pertenecientes al plano de experimento presentado en la Figura 3-4. Previamente a esto se realizaron pruebas con diferentes configuraciones de redes (las cuales no se incluirán en el documento), observando que el error asociado al utilizar más de 10 capas y 20 neuronas por capa, no disminuía de manera considerable, mientras que el tiempo de procesamiento sí aumentaba. Durante estas pruebas también se observó que utilizar igual número de neuronas en cada capa mejoraba la respuesta al error de la red. Esto nos llevó a delimitar el espacio de exploración de las redes, a las presentadas en la Figura 3-8.

Para cada una de las arquitecturas probadas se realizó el cálculo del promedio del error absoluto y se eligieron las mejores arquitecturas de menor a mayor error.

### 3.4.1. Métricas

#### Error absoluto

El Error Absoluto es el valor absoluto resultante de la diferencia entre el valor encontrado y el valor esperado. Esta métrica puede servir para establecer un punto de comparación entre las diferentes arquitecturas de red.

$$AE_i = \|Y_i - X_i\|$$

Donde  $Y_i$  es el vector de coordenadas (x,y) resultantes a la salida de la red neuronal y  $X_i$  es la vector de coordenadas (x,y) esperado.

### Promedio del error absoluto

El promedio del error absoluto (MAE) es la sumatoria del valor absoluto resultante de la diferencia entre el valor encontrado y el valor esperado sobre la cantidad de muestras. Esta métrica al igual que el Error Absoluto puede servir para establecer un punto de comparación entre las diferentes arquitecturas de red.

$$MAE = \frac{\sum_{i=1}^n \|Y_i - X_i\|}{n}$$

Donde  $Y_i$  es el vector de coordenadas (x,y) resultantes a la salida de la red neuronal,  $X_i$  es la vector de coordenadas (x,y) esperado y  $n$  es la longitud del vector.

### Desviación Estándar

Esta métrica nos permitirá identificar que tan dispersos están los puntos de medición unos de los otros para cada una de las técnicas.

$$\sqrt{\frac{\sum |Y_i - \bar{X}_i|^2}{n}}$$

Donde  $Y_i$  es el vector de coordenadas (x,y) resultantes a la salida de la red neuronal y  $\bar{X}_i$  es la media del vector de coordenadas (x,y) esperado.

## 3.5. Sistema de Posicionamiento Embebido

Para dar un mayor alcance a las técnicas de posicionamiento propuestas, es importante ampliar el espectro de estas a sistemas que permitan integrar de manera ágil, económica y fácil los algoritmos presentados anteriormente. Adicionalmente que puedan trabajar de manera conjunta con aplicaciones o proyectos que requieran de posicionamiento en interiores. Es por eso que en esta sección se analizan algunos parámetros necesarios para la elección de un sistema embebido que permita la migración de las técnicas propuestas. Para esto se analizarán factores como: compatibilidad, capacidad de procesamiento y costo de implementación, implementaciones futuras y manejo remoto para pruebas.

### 3.5.1. Compatibilidad

Como vimos en la sección 3.1, para la configuración del RTL2832u y adquisición de datos de este, se utilizó un computador con sistema operativo Linux y distribución Debian. Acorde con esto entonces si queremos buscar una alta compatibilidad en la configuración del dispositivo y en la adquisición de datos, es importante elegir un sistema embebido que pueda trabajar bajo características similares de sistema operativo y distribución.

### 3.5.2. Capacidad de procesamiento y costos de implementación

Teniendo en cuenta que sobre el dispositivo se utilizarán arquitecturas de las redes neuronales y no técnicas de alto costo computacional como el KNN regresivo, podemos utilizar algún sistema embebido de última generación como los presentados en la Tabla **3-3**

En esta Tabla se presentan las características de un Raspberry Pi Modelo B, un ODROID XU4 y un UDOO C86 Advance Plus. Como podemos observar al comparar los sistemas, la capacidad de procesamiento y memoria RAM es directamente proporcional al precio del dispositivo. Si miramos entre los 3 sistemas, el que mejores características de procesamiento y memoria presenta es el UDOO con un procesador Quadcore de 2.24 GHz, una memoria RAM de 4 GB, un precio de \$165 y posibilidad de integrarle WiFi por aparte. En el lado opuesto el que menor capacidad de procesamiento y memoria RAM presenta es el Raspberry Pi modelo B con un procesador quad core de 1.2 GHz y 1 GB de RAM, pero con una gran ventaja en cuanto a su bajo precio de alrededor de los \$35, casi 5 veces menos que el precio del UDOO. Contando además con las características como WIFI integrado y de consumir menos potencia, lo que puede alargar la vida de la batería para aplicaciones que hagan uso de esta.

### 3.5.3. Implementaciones futuras

Las técnicas propuestas se diseñaron pensando en el posicionamiento de UAVs en interiores. Sin embargo debido a que en las regulaciones de la Aeronáutica Civil Colombiana consignadas en la CIRCULAR REGLAMENTARIA 002 de los REQUISITOS GENERALES DE AERONAVEGABILIDAD Y OPERACIONES PARA RPAS, se prohíbe el pilotaje de UAVs en interiores y lugares donde haya población civil cerca [73]. Se realizará la implementación de las técnicas de posicionamiento en el sistema embebido de una manera general, abriendo el espectro de aplicación no solo a UAVs sino a cualquier otra aplicación que requiera de posicionamiento en interiores. No obstante a la hora de elegir el sistema embebido es importante tener en cuenta una implementación futura que integre tanto las técnicas de posicionamiento como la capacidad de construir un UAV a futuro. De las plataformas embebidas de la Tabla **3-3**, se encontraron aplicaciones de UAVs principalmente con el sistema Raspberry Pi. Como en [74] donde un UAV es utilizado en la agricultura para detectar cambios significativos en el suelo como humedad, sequía o inundaciones. En [75] se puede observar como un UAV con

una arquitectura similar al anterior es utilizado para detectar vida salvaje. Otros ejemplos de aplicaciones que hacen uso del Raspberry como sistema de control de un UAVs pueden ser encontrados en la literatura. Como referente adicional se encontró un trabajo donde describen de manera detallada como construir un UAV utilizando la plataforma Raspberry Pi [76], el cual puede servir de guía para una implementación futura del sistema.

### 3.5.4. Manejo remoto para pruebas

Para evitar al máximo cualquier tipo de interferencia en la recepción de las señales con la cercanía del cuerpo, es importante que el sistema embebido a utilizar tenga la opción de manejarse de manera remota a través de un software como Team Viewer. De las plataformas embebidas vistas anteriormente se encontró que las tres cuentan con la posibilidad de controlarse remotamente, pero el Raspberry pi 3 modelo B cuenta con la ventaja adicional de tener conexión a WIFI integrado, lo que evita costos y configuraciones adicionales de tarjetas externas. Por las condiciones analizadas anteriormente y la facilidad de conseguir en el mercado local, el sistema embebido que empleamos para las pruebas de las redes neuronales fue el Raspberry Pi 3 Modelo B.

## 3.6. Instalación de Drivers y librerías RTL2832u en Raspberry pi

El Raspberry Pi desde sus primeras versiones se ha caracterizado por su sistema operativo sugerido, Linux bajo la distribución Raspbian. Esta es una distribución basada en Debian pero adaptada para sistemas embebidos Raspberry. Al ser una distribución basada en Debian, no existen muchas diferencias en los procedimientos de instalación de drivers y librerías del RTL2832u respecto a lo visto en la sección 3.1. Por lo cual se puede seguir el mismo procedimiento descrito. Sin embargo debido a que el sistema operativo Raspbian no es de 64 bits sino de 32 bits necesitamos hacer un cambio adicional en el archivo *rtl\_power.c* cambiando todas las variables definidas con tipos de datos long a *int64\_t*. Este cambio es necesario para que haya compatibilidad con el tipo de memoria y no se obtengan datos con valores “nan” durante la adquisición de datos.

**Tabla 3-3.:** Tabla comparativa de Sistemas Embebidos

	<b>Raspberry Pi 3 Modelo B</b>	<b>ODROID-XU4</b>	<b>UDOO C86 Advanced Plus</b>
<b>CPU</b>	ARM Cortex-A53 (64-bit) 1.2GHz quad core	ARM Cortex-A15 / ARM Cortex-A7 (32-bit) 2GHz octa core	Braswell (1.60GHz with 2.24GHz burst) (64-bit) 2.24GHz quad core
<b>GPU</b>	VideoCore IV	Mali-T628 MP6	640MHz Intel HD Graphics Gen 8-LP
<b>RAM size</b>	1GB	2GB	4GB
<b>Wi-Fi</b>	802.11b/g/n	No	Optional Wi-Fi 802.11ac
<b>GPIO pines</b>	40	30	14
<b>I2C</b>	1	1	2
<b>Voltaje</b>	4.8V - 5.2V	4.8V - 5.1V	3.3V
<b>Consumo Potencia</b>	600mA - 1.6A	1A - 4A	1A - 3A
<b>Linux</b>	Rasbian (Basado en Debian)	armbian (basado en Debian)	Debian
<b>Precio</b>	\$35	\$74	\$165

## 4. Resultados Experimentales

Como se muestra en la Figura 4-1, en este capítulo se presentan los resultados de la estimación de la posición con las técnicas propuestas utilizando Deep Learning y se compara con la estimación de la posición obtenida con un algoritmo de KNN Regresivo para FM, AM y AM combinado con FM. Posteriormente se presentan los resultados del proceso de implementación de las técnicas propuestas en un sistema embebido Raspberry pi y su comportamiento en tiempo real.

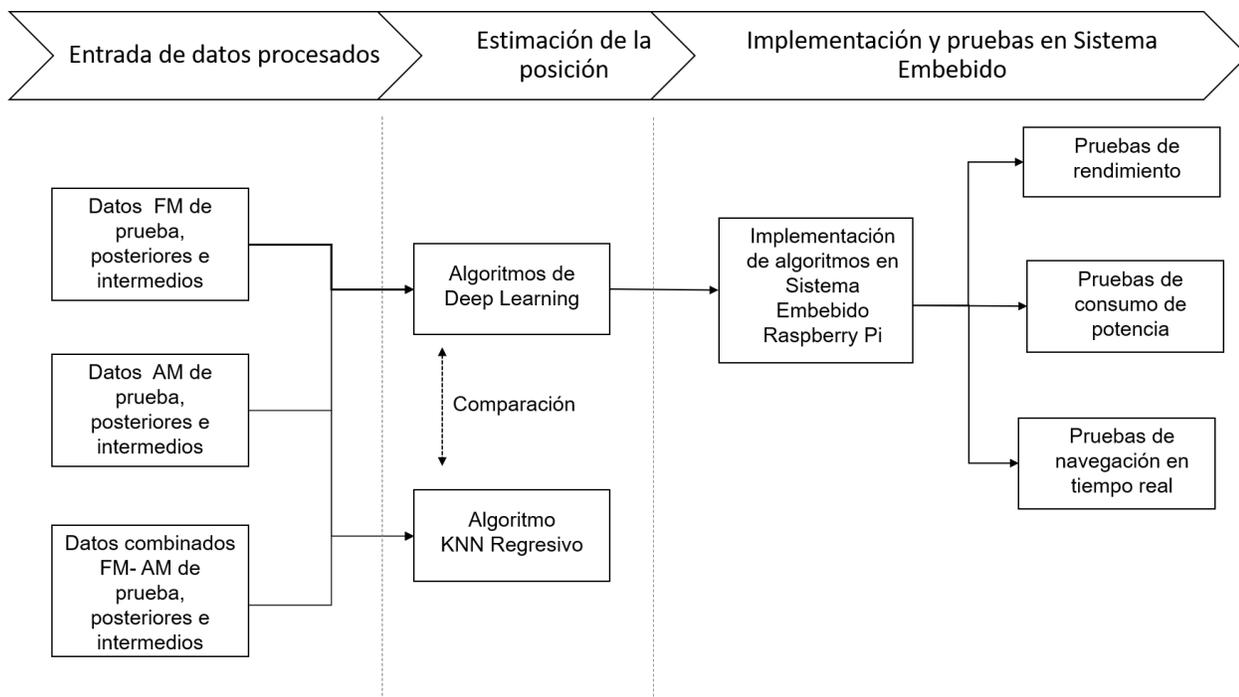


Figura 4-1.: Diagrama de bloques Resultados Experimentales

### 4.1. Posicionamiento con radio-estaciones FM

En esta sección se estableció como fue la elección de la configuración de la red neuronal a utilizar para FM, tomando como insumo la base de datos con el RSS de FM procesado con información de los 23 puntos coordinados descrita en el capítulo anterior. Una vez la configuración de la red neuronal es establecida, se realiza la validación de la misma (estimación



promedio de error absoluto de 0.05 cm en la coordenada X y de 0.11 cm en la coordenada Y y el segundo con un promedio del error absoluto de alrededor de 0.07 cm en la coordenada X y 0.15 cm en la coordenada Y.

Teniendo en cuenta los resultados de estas dos configuraciones utilizaremos las redes neuronales correspondientes a estas para la estimación de la posición con FM, junto con el algoritmo comparativo de KNN Regresivo.

### 4.1.2. Evaluación de desempeño

En esta sub-sección se mostró entre las dos configuraciones de red anteriormente mencionadas y un algoritmo de KNN regresivo, cuál presenta mejores resultados al realizar una estimación de la posición para cada una de las 23 coordenadas elegidas con la información de las radio estaciones FM.

Para ello los resultados fueron presentados de dos maneras, una gráfica y una numérica utilizando una tabla de resultados con el promedio del error absoluto MAE para cada una de las 23 coordenadas. La gráfica hace uso del MAE como la distancia en X y Y de una elipse cuyo centro es la coordenada real esperada. Se utiliza la elipse por la facilidad visual que esta figura ofrece en la comparación de resultados. A menor tamaño de la elipse y mayor cercanía de su centro a la coordenada original, mayor será la precisión de la estimación para ese punto.

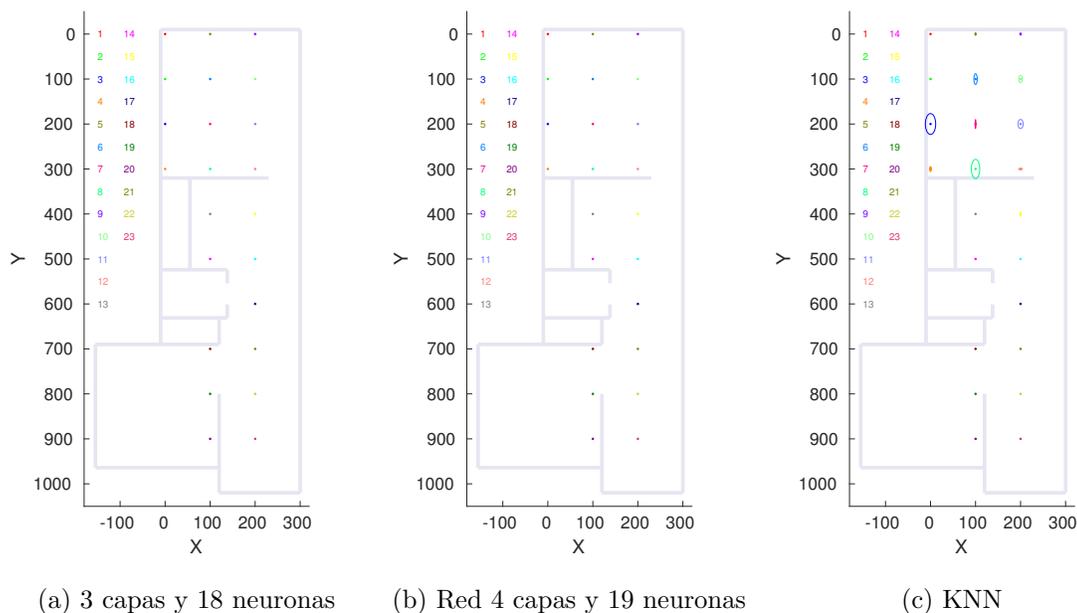
#### Datos de prueba

En la Figura 4-3 y en la tabla 4-1 es posible observar los resultados para las dos configuraciones de red elegidas y para el KNN regresivo en términos del promedio del error absoluto para los datos de prueba obtenidos durante la construcción de la base de datos de entrenamiento del sistema. De manera general, observamos que las redes neuronales presentan una mejor estimación de la posición que el KNN regresivo con promedios de error absoluto inferiores a 1 cm para todos los puntos, mientras que el KNN presenta un promedio de error absoluto para puntos como el 3 y el 8 superior a los 20 cm para la coordenada Y. Que si bien no es un error grande comparado con otras técnicas, nos permite verificar un comportamiento con una mejoría del 95 % de las redes para los puntos de prueba respecto al KNN.

#### Datos posteriores

Días posteriores a la construcción del Fingerprint, se recolectaron datos adicionales en cada una de las 23 coordenadas coincidentes con la grilla. Estas mediciones fueron probadas de manera similar a la sub-sesión anterior, con los algoritmos de redes neuronales y el KNN regresivo obteniendo los resultados presentados en la Figura 4-4 y en la tabla 4-2.

En la tabla 4-2 y en la Figura 4-4 observamos que el promedio del error absoluto para algunos puntos aumenta un poco respecto al visto para los puntos de prueba de la Figura 4-1. Esto



**Figura 4-3.:** Promedio del error absoluto FM para puntos de prueba

se debe a varios factores como a que desde el momento en que se recolectaron los datos para entrenamiento y prueba hasta los datos posteriores e intermedios, hubo cambios físicos en el espacio de experimentos en la distribución de muebles y electrodomésticos. Como vemos en [61, 77, 78] los efectos multi - trayectoria y perdidas de señal pueden cambiar de acuerdo a la distribución del espacio, afectando así el RSS. Otro factor a tener en cuenta según [79] es que los metales, los techos y demás superficies reflectivas en un espacio cerrado pueden causar falsos positivos en el RSS de algunas ubicaciones del espacio. Un último factor que puede afectar las mediciones posteriores son los cambios de diales que se dan mas comunmente en radioestaciones FM a principio de año, lo que puede afectar el índice del RSS por la potencia de emisión de cada radioestación.

En la tabla 4-2 y en la Figura 4-4a vemos que para la red neuronal con configuración de 3 capas y 18 neuronas el promedio del error absoluto tiende a ser menor a 20 cm en las coordenadas X y Y para la mayoría de los puntos, excepto para puntos como el 5, el 11, el 22, el 1 y el 17, alcanzando para estos dos últimos un promedio del error absoluto en Y de alrededor de los 80 cm. Por otro lado la red neuronal de 4 capas y 18 neuronas de la Figura 4-4b presenta un comportamiento similar a la red anterior para algunas de las coordenadas, llegando a un máximo promedio del error absoluto de 50 cm en X y 75 cm en Y en el punto 7. Por su parte el KNN Regresivo de la Figura 4-4c presenta para todas sus coordenadas un promedio de error absoluto inferior a los 15 cm en la coordenada X, pero errores superiores

**Tabla 4-1.:** Promedio del error absoluto FM para puntos de prueba

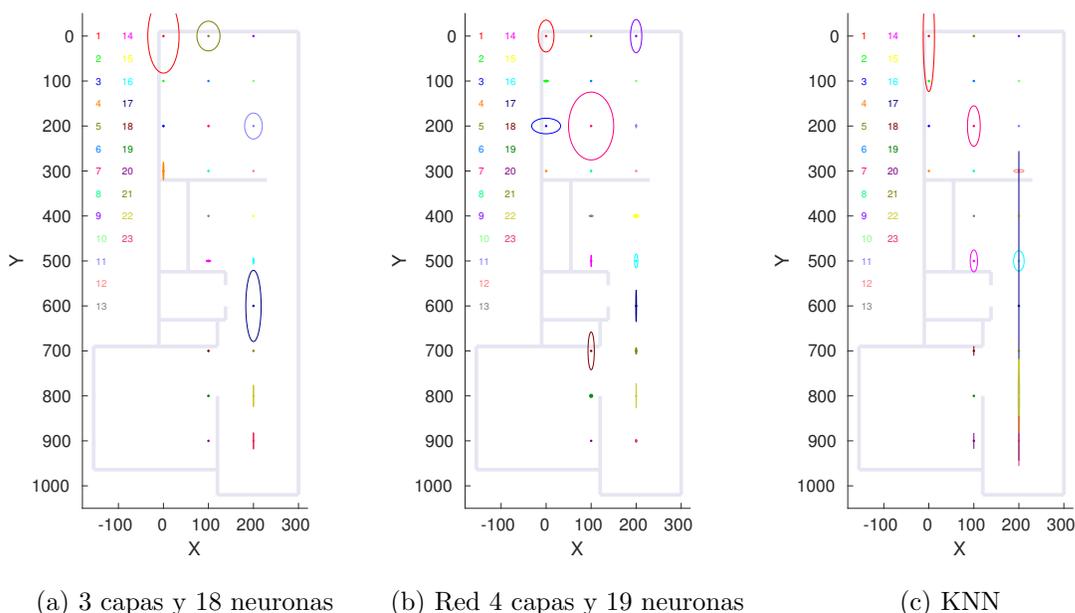
Punto	Red 3 - 18		Red 4 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)	x (cm)	y (cm)
1	0,012	0,024	0,071	0,073	0,238	0,000
2	0,036	0,020	0,191	0,171	0,000	0,000
3	0,090	0,097	0,034	0,073	11,463	23,171
4	0,025	0,050	0,041	0,049	1,786	5,357
5	0,010	0,012	0,041	0,045	0,698	2,868
6	0,280	0,493	0,268	0,395	3,704	11,852
7	0,019	0,040	0,057	0,206	0,889	9,889
8	0,040	0,116	0,059	0,072	9,322	21,412
9	0,029	0,245	0,148	0,416	0,000	3,577
10	0,034	0,034	0,047	0,063	3,977	7,953
11	0,045	0,481	0,051	0,074	5,789	10,175
12	0,320	0,405	0,069	0,092	4,138	1,552
13	0,012	0,035	0,093	0,051	0,000	0,818
14	0,032	0,055	0,107	0,215	0,000	0,000
15	0,026	0,035	0,063	0,246	0,690	4,138
16	0,008	0,054	0,029	0,041	0,000	1,538
17	0,048	0,071	0,040	0,057	0,058	0,175
18	0,025	0,033	0,056	0,110	0,000	0,000
19	0,015	0,037	0,046	0,055	0,000	0,000
20	0,037	0,023	0,029	0,054	0,000	0,000
21	0,018	0,036	0,036	0,127	0,000	0,000
22	0,007	0,027	0,037	0,055	0,000	0,000
23	0,011	0,036	0,050	0,633	0,000	0,000

a los 80 cm para puntos como el 1 y el 22 para la coordenada Y, llegando a un promedio del error absoluto superior a los 320 cm para el punto 17. Obteniendo así aproximadamente un 25 % menos de error en la estimación de la posición para puntos posteriores con las redes neuronales.

### Coordenadas intermedias

Teniendo en cuenta que los algoritmos utilizados tienen un carácter regresivo, estos fueron utilizados para realizar la estimación de la posición en algunos puntos coordinados intermedios diferentes a los utilizados en el entrenamiento.

En la tabla 4-4 y en la Figura 4-5 podemos observar los resultados obtenidos en la estimación



**Figura 4-4.:** Promedio del error absoluto FM puntos posteriores

de la posición para las coordenadas de la Tabla 4-3. Tanto en la tabla como en la figura, es evidente que el promedio del error absoluto aumenta para este tipo de estimación. En el caso de la configuración de red 3-18 de la Figura 4-5a y la tabla 4-4 vemos que el promedio del error absoluto no excede los 45 cm en X y los 67 cm en Y, a excepción de los puntos 8 y 9 donde el error está alrededor de los 120 cm para la coordenada Y. Por su parte para la red 4-19 de la Figura 4-5b y para el KNN de la Figura 4-5c, se alcanzan a estimar un promedio del error absoluto superior a los 600 cm para estos mismos puntos. Aunque los resultados anteriores presentan un promedio del error absoluto mayor, vemos que con redes neuronales como la 3-18 podemos llegar a tener un comportamiento regresivo aceptable para la estimación de posiciones intermedias. Como vemos en [80–84] el promedio de los errores mínimos para la mayoría técnicas de radio frecuencia para posicionamiento en interiores ronda alrededor 1m y las técnicas que logran superar este umbral requieren de altos costos de implementación de infraestructura o las pruebas son realizadas en áreas pequeñas, lo que minimiza el error. Al igual que lo que mencionamos en la sección anterior, es posible también que las alteraciones del espacio y cambios en los diales de las emisoras, puedan influir también en el grado de imprecisión para algunas coordenadas.

**Tabla 4-2.:** Promedio del error absoluto FM para puntos posteriores

Punto	Red 3 - 18		Red 4 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)	x (cm)	y (cm)
1	34,739	82,436	17,446	35,419	12,500	123,333
2	0,036	0,021	5,325	1,768	0,000	0,000
3	1,050	1,508	32,132	17,256	0,000	0,000
4	1,154	19,774	0,730	0,365	0,000	0,000
5	25,680	33,060	0,317	0,932	0,000	0,000
6	0,061	0,026	0,398	0,429	1,458	0,000
7	0,364	1,357	50,134	75,375	14,167	45,000
8	0,185	0,434	0,219	1,730	0,000	0,000
9	0,013	0,076	12,550	36,915	0,000	0,000
10	0,018	0,020	0,823	1,046	1,667	1,667
11	19,618	28,834	0,627	4,054	0,000	0,000
12	0,196	0,558	0,630	0,363	11,042	3,125
13	0,030	0,036	4,525	1,293	0,000	0,000
14	5,140	1,732	0,810	12,719	8,125	24,375
15	0,373	0,336	6,099	3,424	2,083	1,458
16	0,274	7,322	3,607	15,472	11,667	22,708
17	17,030	79,162	0,798	34,883	0,000	344,167
18	0,668	1,127	6,990	41,932	0,000	10,625
19	1,274	1,264	3,376	3,577	0,000	0,000
20	0,003	0,020	0,054	0,209	0,000	17,708
21	0,161	0,212	1,408	7,128	0,000	0,000
22	0,440	24,108	0,249	26,846	1,667	82,083
23	0,631	18,448	1,440	3,212	0,000	55,625

### 4.1.3. Tiempo de Ejecución

En la sub-sesión anterior se analizó el comportamiento de las técnicas utilizadas en cuanto a la estimación de la posición. En esta sesión analizaremos como se comportan estas técnicas en cuanto al Tiempo de Ejecución para la estimación tanto para los puntos del Fingerprint como para los puntos posteriores.

En la Tabla 4-5 observamos los tiempos de ejecución de cada una de las técnicas, este tiempo se tuvo en cuenta como el tiempo promedio que tarda cada técnica en realizar la estimación de la posición en 20 iteraciones diferentes. Estos tiempos son relativos al desempeño y velocidad del procesador y capacidad de RAM del computador, por lo que varían de una máquina a otra. En nuestro caso se utilizó un computador con un procesador Intel Core i7-4510U y con

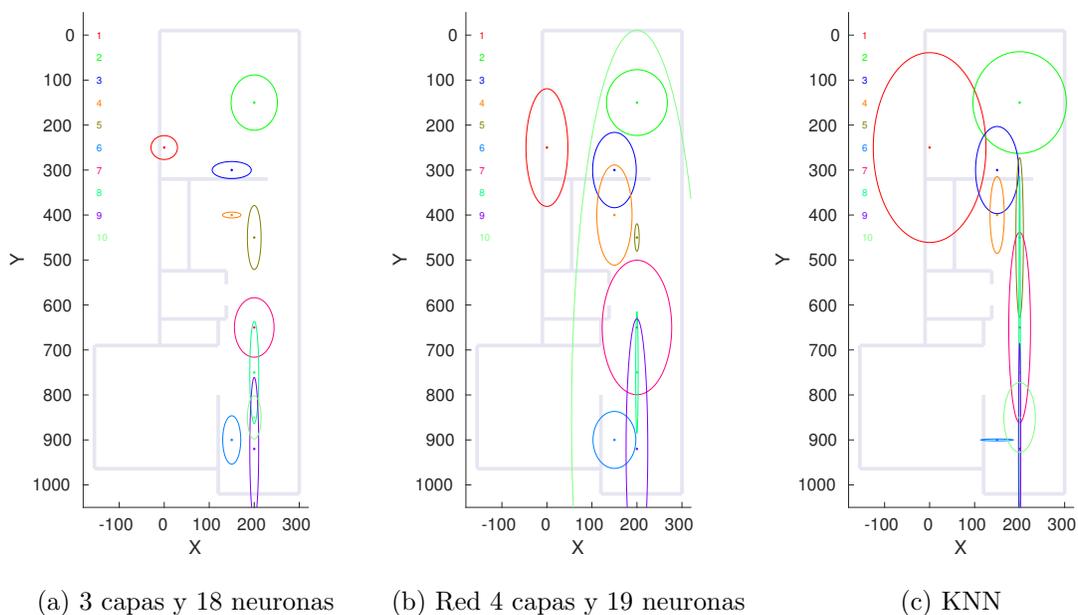
**Tabla 4-3.:** Coordenadas Intermedias

<b>Punto</b>	<b>X (cm)</b>	<b>Y (cm)</b>
1	0	250
2	200	150
3	150	300
4	150	400
5	200	450
6	150	900
7	200	650
8	200	750
9	150	700
10	200	850

una capacidad de memoria RAM de 8 GB.

Con los resultados presentados en la Tabla 4-5, confirmamos que el comportamiento de las redes se destacan más que el del KNN regresivo tanto para los puntos del Fingerprint como para los puntos posteriores. Esto se debe a que el KNN requiere siempre entrenar el sistema completo antes de hacer la estimación de una coordenada.

También podemos observar que para los puntos del Fingerprint la configuración de 3 capas 18 neuronas se comporta un poco mas rápido que la configuración de 4 capas 19 neuronas. Para los puntos posteriores esta diferencia no es tan perceptible y ambas configuraciones tienen un tiempo de ejecución equiparable.



**Figura 4-5.:** Promedio del error absoluto FM puntos intermedios

**Tabla 4-4.:** Promedio del error absoluto FM para puntos intermedios

Punto	Red 3 - 18		Red 4 -19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)	x (cm)	y (cm)
1	29,479	26,560	46,563	130,841	125,208	211,042
2	51,447	61,474	67,573	73,121	103,750	113,125
3	43,394	18,959	48,464	83,739	47,917	96,875
4	20,454	6,400	39,150	111,469	15,833	85,417
5	15,235	71,150	5,035	30,486	8,333	177,500
6	20,027	53,676	47,588	63,315	36,250	2,083
7	44,017	66,169	77,251	149,503	24,167	211,667
8	9,904	113,669	2,718	135,115	1,667	435,000
9	9,865	159,037	24,107	289,177	2,500	233,750
10	15,145	47,882	145,353	861,136	35,000	77,500

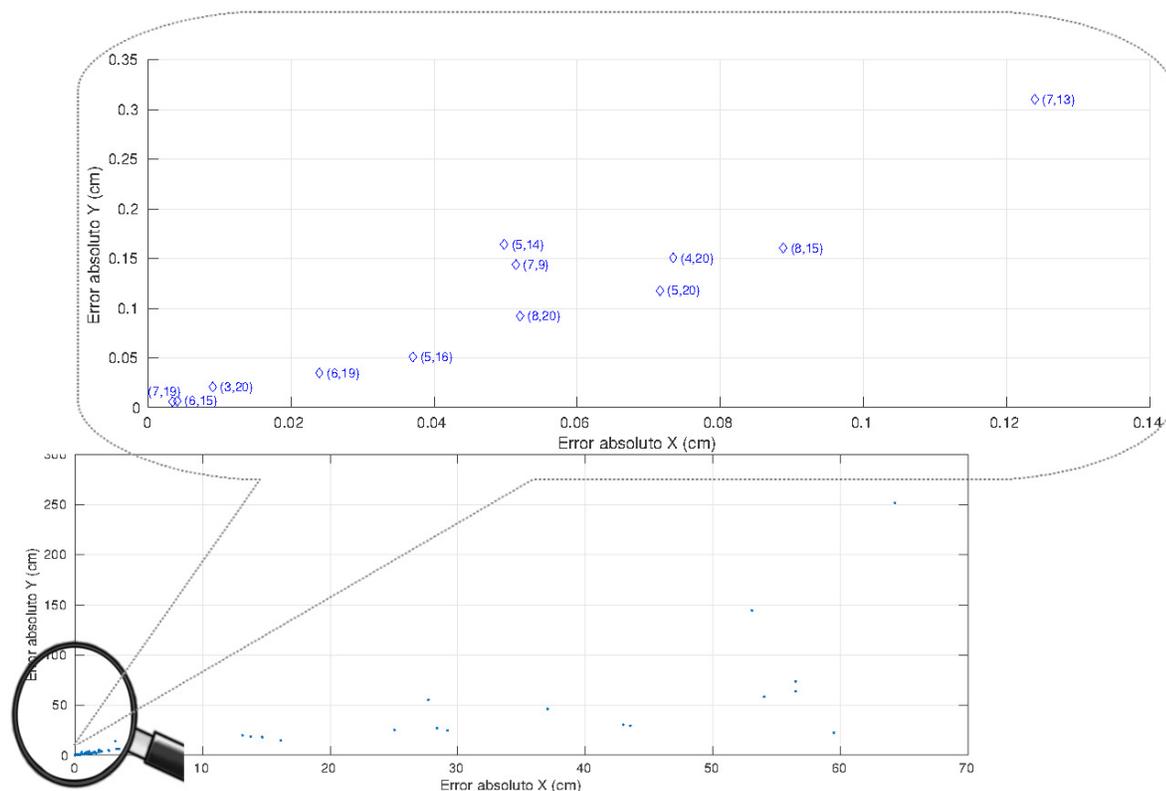
**Tabla 4-5.:** Tiempo de ejecución algoritmo estimador de la posición en FM

<b>Tiempo de ejecución (s)</b>		
Algoritmo	Puntos Fingerprint	Puntos Posteriores
Red 3 – 18	0,0497	0,041
Red 4 – 19	0,0637	0,0425
KNNR	8,2418	2,6169

## 4.2. Posicionamiento con radio-estaciones AM

En esta sección se estableció como fue la elección de la configuración de la red neuronal a utilizar para AM, tomando como insumo la base de datos con el RSS de AM procesado con información de los 23 puntos coordenados descrita en el capítulo anterior. Una vez la configuración de la red neuronal es establecida, se realiza la validación de la misma (estimación de la posición) con 3 muestras de datos diferentes. En una primera instancia los datos de prueba, luego en una segunda instancia con un conjunto de datos capturados posteriormente y que coinciden con la grilla de medición y por último datos intermedios a la grilla.

### 4.2.1. Exploración de la configuración de la red



**Figura 4-6.:** Errores absolutos AM según configuración de la red

De manera similar que para FM, pero en este caso eligiendo como datos de entrada la información procesada para AM, se realizó la exploración de las 180 configuraciones posibles de la red para determinar las mejores configuraciones de capas y neuronas por capa. El criterio para determinar las mejores configuraciones fue el promedio del error absoluto de los datos de salida de la red. Los resultados obtenidos de las 180 configuraciones se pueden observar en la Figura 4-6 donde se gráfica el promedio del error absoluto relacionado con el

número de capas y neuronas por capa. Al hacer un acercamiento a los menores errores de la gráfica se observa que el menor error corresponde a la configuración de 7 capas - 19 neuronas con un promedio de error absoluto de 0.0034 cm en la coordenada X y de 0.006 cm en la coordenada Y.

Acorde con los resultados anteriores utilizaremos la red neuronal de 7 capas y 19 neuronas para la estimación la posición con AM, junto con el algoritmo comparativo de KNN Regresivo.

### 4.2.2. Evaluación de desempeño

Similar a como se hizo para FM, en esta sub-sección se analizó entre la configuración de la red neuronal de 7 capas y 19 neuronas y un algoritmo de KNN regresivo, cuál presenta mejores resultados al realizar una estimación de la posición para cada una de las 23 coordenadas elegidas con la información de las radio estaciones AM.

Para ello los resultados se presentan también a través de una gráfica y una tabla numérica siguiendo la misma dinámica que en la sección anterior.

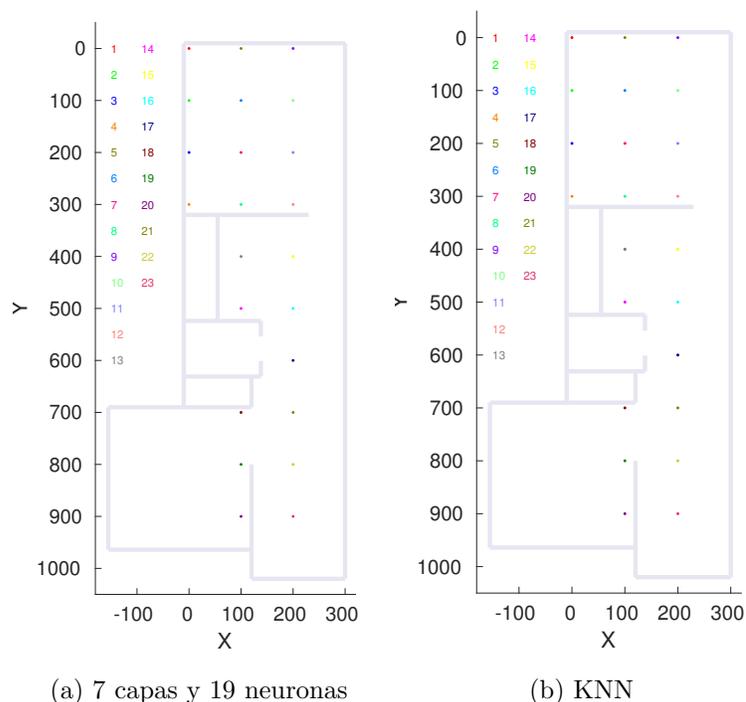


Figura 4-7.: Promedio del error absoluto AM puntos de prueba

### Datos de prueba

En la tabla 4-6 y la Figura 4-7 es posible observar los resultados para la configuración de red elegida y para el KNN regresivo en términos del promedio del error absoluto utilizando

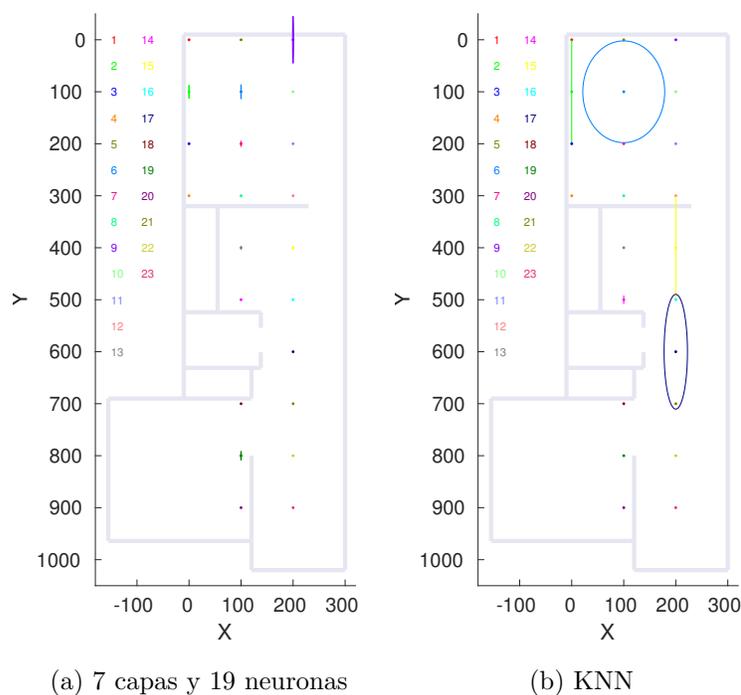
los datos de prueba. De manera general, se observa que tanto la red neuronal como KNN regresivo mejoran los resultados obtenidos con los datos de prueba de FM en la estimación de la posición vistos en la sección anterior. Llegando un máximo promedio del error de alrededor de 0.008 cm la red neuronal y 1.75 cm el KNN. Por lo que ambos algoritmos pueden ser utilizados de manera indistinta en cuanto precisión para los datos de prueba de este tipo de señales.

**Tabla 4-6.:** Promedio del error absoluto AM para puntos de prueba

Punto	Red 7 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)
1	0,00551	0,00784	0,00000	0,00000
2	0,00433	0,00770	0,00000	0,94094
3	0,00323	0,00995	0,03653	0,69406
4	0,00261	0,00426	0,00000	0,00000
5	0,00904	0,00594	0,00000	0,00000
6	0,00256	0,00792	0,31703	1,26812
7	0,00378	0,01347	0,00000	0,00000
8	0,00125	0,00624	0,00000	0,00000
9	0,00261	0,00429	0,00000	0,00000
10	0,00306	0,00403	0,00000	0,00000
11	0,00254	0,00490	0,00000	0,00000
12	0,00167	0,00281	0,00000	0,00000
13	0,00339	0,00639	1,75111	1,71556
14	0,00571	0,00302	1,35870	1,35870
15	0,00240	0,00718	0,93948	1,35501
16	0,00444	0,00486	0,00000	0,31339
17	0,00286	0,00706	0,44818	0,86835
18	0,00278	0,00484	0,00000	0,29326
19	0,00240	0,00357	0,00000	0,27701
20	0,00405	0,00299	0,27778	0,55556
21	0,00354	0,00588	0,00000	0,29070
22	0,00336	0,00783	0,00000	0,00000
23	0,00168	0,00503	0,00000	0,00000

### Datos posteriores

De manera análoga a como se hizo con FM, para AM también se recolectaron datos adicionales diferentes a los del fingerprint para cada una de las 23 coordenadas. Estos fueron



**Figura 4-8.:** Promedio del error absoluto AM puntos posteriores

probados nuevamente con ambos algoritmos, obteniendo los promedios del error absoluto presentados en la tabla 4-7 en la Figura 4-8.

Como vemos en al comparar las Figuras 4-8a y 4-8b, la red neuronal presenta mejores resultados en la estimación de la posición para la mayoría de las coordenadas, con errores inferiores a los 2 cm en la coordenada X y a los 14 cm en la coordenada Y a excepción del punto 9 donde el error en Y se aproxima a los 45 cm. Por su parte el KNN regresivo supera los 90 cm de error en la coordenada Y para puntos como el 17, el 15, el 2 y el 6, superando además los 70 cm en X para este último. Aunque en puntos diferentes a los mencionados el KNN regresivo no presenta errores tan significativos, vemos que el comportamiento de la red neuronal para puntos posteriores con señales AM es mejor que el del KNN.

### Coordenadas intermedias

Al igual que para FM para AM también se prueba el carácter regresivo de los algoritmos utilizados. Para establecer un punto de comparación futuro utilizaremos las coordenadas intermedias de prueba presentadas en la Tabla 4-3 del capítulo anterior.

Al analizar la tabla 4-8 y la Figura 4-9 de manera conjunta con la Tabla 4-3, vemos que los resultados de la red neuronal presentados en la Figura 4-9a son muy superiores a los del KNN regresivo de la Figura 4-9b. Teniendo un promedio del error absoluto de máximo para la red neuronal de alrededor de los 73 cm en la coordenada X y de 72 cm en la coordenada

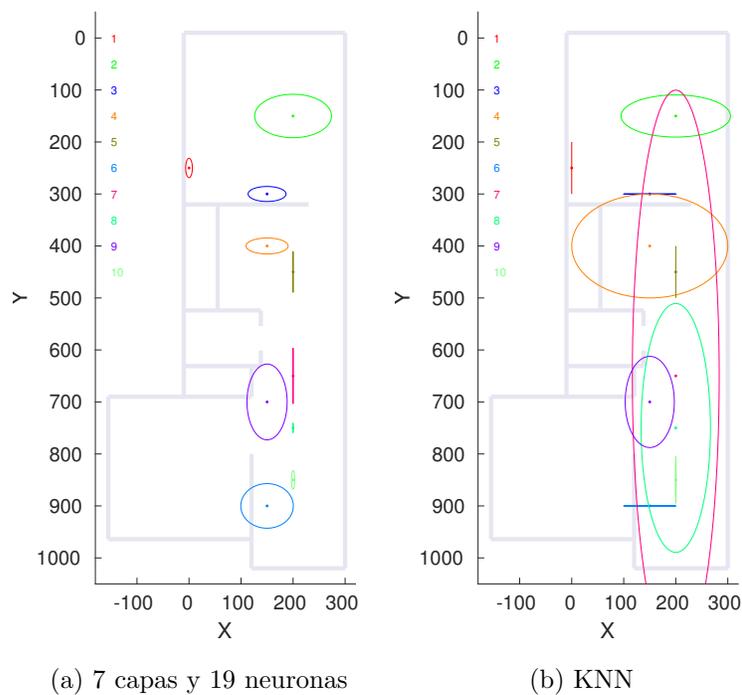
**Tabla 4-7.:** Promedio del error absoluto AM para puntos posteriores

Punto	Red 7 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)
1	0,06835	0,44068	0,00000	0,83333
2	0,15110	12,54403	0,00000	100,00000
3	0,00994	0,05717	0,00000	0,00000
4	0,00349	0,02115	0,00000	0,00000
5	1,33959	0,59194	0,00000	0,00000
6	0,26395	13,56661	78,72222	97,83333
7	0,26013	5,09778	0,16667	0,16667
8	0,00102	0,03842	0,00000	0,00000
9	1,07122	45,11546	0,00000	0,00000
10	0,00741	0,10396	0,00000	0,00000
11	0,00583	0,04526	0,00000	0,00000
12	0,05325	1,45964	0,00000	0,00000
13	0,12911	2,99754	0,00000	0,00000
14	0,01351	0,01080	0,00000	8,16667
15	0,51309	2,44446	0,00000	100,00000
16	0,00803	0,12628	0,00000	0,00000
17	0,00824	0,09864	22,44444	110,38889
18	0,00487	0,06574	0,00000	0,00000
19	0,52720	8,42376	0,00000	0,00000
20	0,01157	0,09813	0,00000	0,00000
21	0,00611	0,31916	0,00000	0,00000
22	0,00609	0,02732	0,00000	0,00000
23	0,00123	0,01062	0,00000	1,27778

Y, mientras por su parte el KNN llega a presentar errores hasta de 100 cm en la coordenada X y 550 cm en la coordenada Y.

### 4.2.3. Tiempo de ejecución

Utilizando el mismo procedimiento que para FM llegamos a resultados similares para AM. Como se observa en la tabla 4-9, la red neuronal utilizada al no requerir de el conjunto de datos completo de entrenamiento tarda mucho menos que el KNN regresivo en estimar la posición. En el caso de los puntos del Fingerprint la relación de tiempos es 1/210, mientras que para puntos posteriores es 1/78.



**Figura 4-9.:** Promedio del error absoluto AM puntos Intermedios

**Tabla 4-8.:** Promedio del error absoluto AM para puntos intermedios

Punto	Red 7 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)
1	6,422896	18,65737	0	50
2	73,82271	41,795	105,2222	40,55556
3	36,43986	14,44041	50	0
4	40,43973	15,33084	150	100
5	0,05334	39,16324	0	50
6	50,49277	42,88914	50	0
7	0,365949	53,32838	83,5	549,8333
8	0,12666	9,31225	66,66667	239,3889
9	38,61924	72,87955	47,22222	87,72222
10	3,096684	18,14661	0,555556	44,83333

**Tabla 4-9.:** Tiempo de ejecución algoritmo estimador de la posición en FM - AM

<b>Tiempo de ejecución (s)</b>		
Algoritmo	Puntos Fingerprint	Puntos Posteriores
Red 7 – 19	0,035	0,029
KNNR	7,378	2,28

### 4.3. Posicionamiento híbrido radio-estaciones FM - AM

En esta sección determinamos como fue la elección de la configuración de la red neuronal a utilizar haciendo uso de la señal híbrida FM - AM, tomando como insumo la base de datos con el RSS de FM y AM procesados con información de los 23 puntos coordenados descrita en el capítulo anterior. Una vez la configuración de la red neuronal es establecida, se realiza la validación de la misma (estimación de la posición) con 3 muestras de datos diferentes. En una primera instancia los datos de prueba, luego en una segunda instancia con un conjunto de datos capturados posteriormente y que coinciden con la grilla de medición y por último datos intermedios a la grilla.

#### 4.3.1. Exploración de la configuración de la red

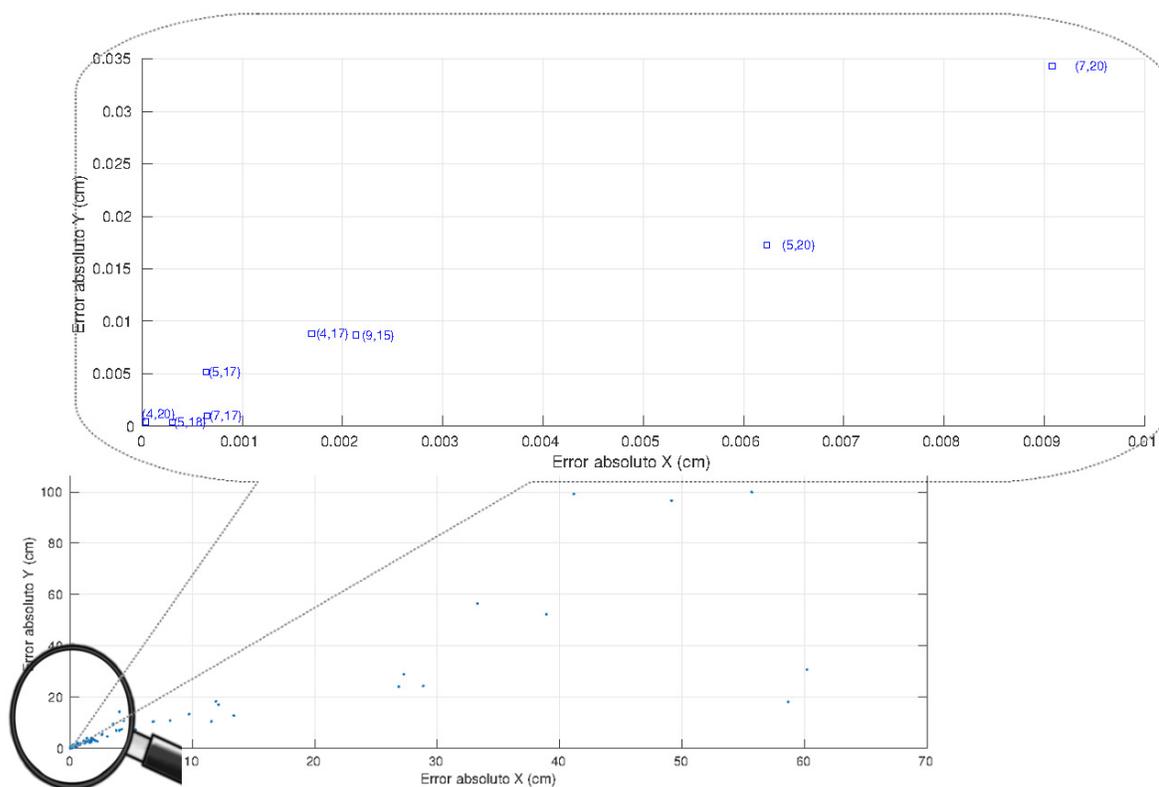


Figura 4-10.: Errores absolutos FM-AM según arquitectura de la red

Al igual que para las dos técnicas anteriores, en la Figura 4-10 se observa el promedio del error absoluto para 180 arquitecturas de prueba FM-AM. Es evidente como se puede observar que al mezclar la información de FM con la de AM el promedio del error tiende a disminuir para la mayoría de las arquitecturas. Distribuyéndose la mayoría en un rango inferior a los 10 cm en el eje X y a los 20 cm en el eje Y. Al hacer un zoom sobre los puntos mas próximos

a cero, vemos que la arquitectura de red que mejores resultados presenta es la red de 4 capas y 20 neuronas con un error de 0.000038 cm en el eje X y 0.00041 cm en el eje Y.

### 4.3.2. Evaluación de desempeño

En esta sección se muestra el resultado de combinar el RSS de la señales de radio FM y AM para la estimación de la posición utilizando la arquitectura de red neuronal de 4 capas y 20 neuronas y KNN regresivo.

Similar a como se hizo en las sesiones anteriores, los resultados se presentan en términos del promedio del error absoluto MAE para cada una de las 23 coordenadas con las respectivas tablas y gráficas elípticas vistas en las secciones anteriores.

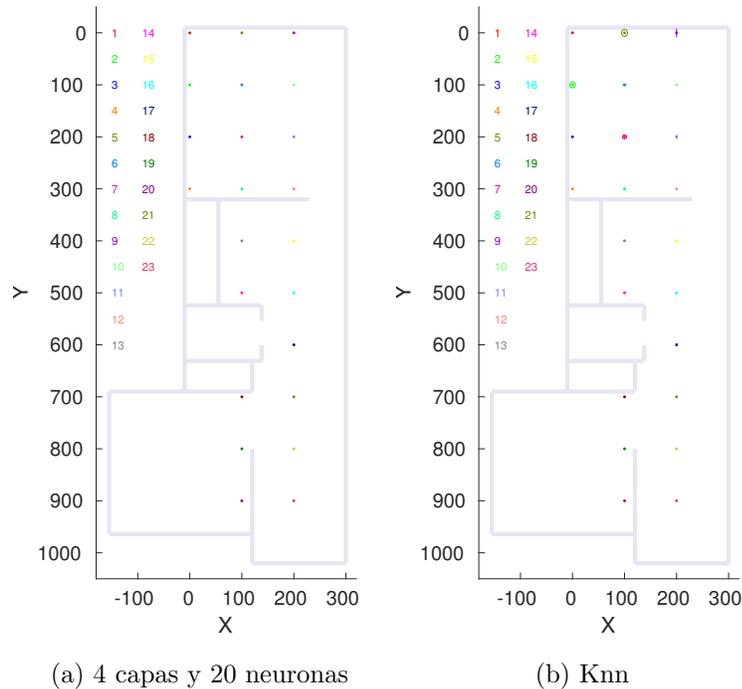


Figura 4-11.: Promedio del error absoluto FM-AM puntos de prueba

### Datos de prueba

En la tabla 4-10 y en la Figura 4-11 se observan los resultados para la configuración de red elegidas y para el KNN regresivo en términos del promedio del error absoluto. Al igual que para las técnicas anteriores, vemos que con los datos de prueba la tendencia del promedio del error absoluto es baja y es posible establecer la posición tanto con ambas técnicas con errores muy bajos. Sin embargo el KNN regresivo excede los 5 cm en el promedio de error absoluto para puntos como el 2 y el 5, mientras que el error promedio del error absoluto de la

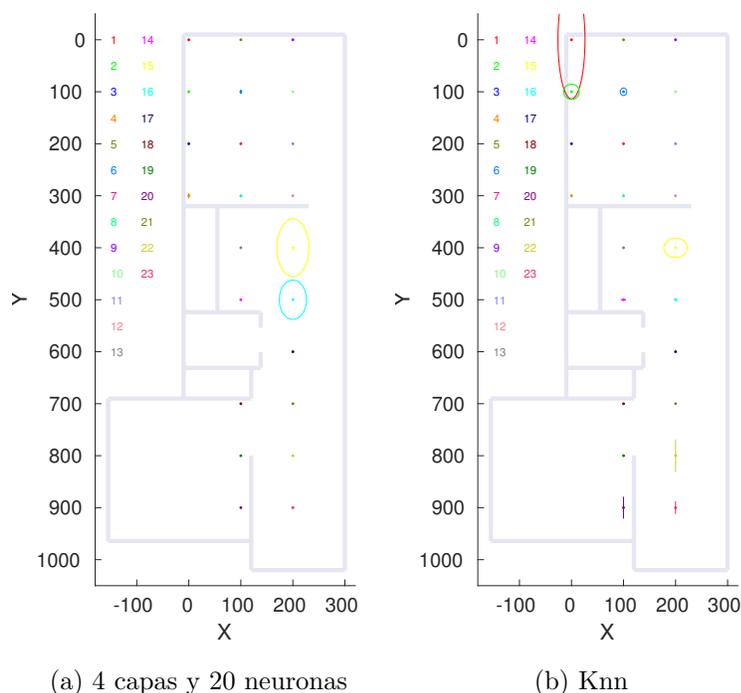
**Tabla 4-10.:** Promedio del error absoluto FM - AM para puntos de prueba

Punto	Red 7 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)
1	3,01E-05	0,000131	0,606884	1,34058
2	3,93E-05	0,000188	5,487179	5,487179
3	5,04E-05	0,000226	0	0,070175
4	2,65E-05	0,000164	0	0
5	6,80E-05	0,000124	5,668403	6,796875
6	1,81E-05	0,00018	2,843137	0,261438
7	2,60E-05	0,000113	3,909574	3,909574
8	3,52E-05	3,64E-05	0	0
9	5,61E-05	0,000141	0	8,231293
10	1,18E-05	0,000145	0	0
11	2,60E-05	0,00014	0	3,777778
12	6,57E-05	9,62E-05	0	0,018519
13	7,47E-05	0,00274	0,558882	0,199601
14	8,79E-05	0,000865	0	0,581395
15	3,33E-05	0,000206	0,869565	1,032609
16	1,42E-05	0,000159	0	0,610465
17	2,67E-05	0,000113	0,15873	1,081349
18	4,93E-05	9,08E-05	0	0
19	1,86E-05	9,95E-05	0	0,009208
20	4,23E-05	0,00011	0	0
21	2,88E-06	7,84E-05	0	0
22	4,83E-05	0,003246	0	0
23	3,32E-05	7,05E-05	0	0,017825

red neuronal esta por debajo de 1 cm tanto para la coordenada X como para la coordenada Y.

### Datos posteriores

Para la recolección de datos posteriores se intercaló la recolección de datos en cada coordenada, tomando primero la información de FM y de manera consecutivamente los de AM. Esto debido a que el dispositivo RTL2832u solo permite seleccionar un rango de frecuencias para la toma de datos a la vez como ya lo habíamos en el capítulo anterior. La información adquirida de FM y AM se mezcló y fue probada con los dos algoritmos, obteniendo el promedio del error absoluto para los datos de salida.



**Figura 4-12.:** Promedio del error absoluto FM-AM puntos posteriores

En la tabla 4-11 y en la Figura 4-12a se observa que la red presenta un promedio del error absoluto que supera los resultados de las secciones anteriores para la mayoría de puntos, exceptuando en los puntos 15 y 16 en donde presenta un promedio del error absoluto de alrededor de los 30 cm en la coordenada X y de los 50 cm en la coordenada Y. Por su parte el KNN Regresivo en la Figura 4-12b presenta un promedio del error absoluto bajo en la mayoría de sus puntos, exceptuando los puntos 2, 6, 15, 20, 22, 23 y 1. Este último presentando un promedio del error absoluto de alrededor de los 25 cm en la coordenada X y de 110 cm en la coordenada Y.

### Coordenadas intermedias

Siguiendo el mismo procedimiento que en los capítulos de FM y AM realizamos la prueba del comportamiento regresivo de la red neuronal y del KNN Regresivo. Esto utilizando las coordenadas de la Tabla 4-3.

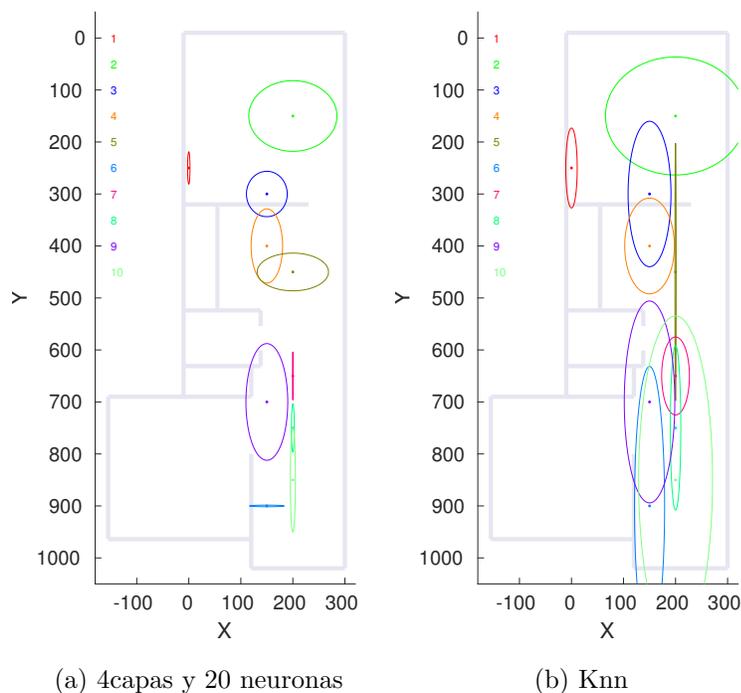
Al analizar la tabla 4-12 y la Figura 4-13 de manera conjunta con la Tabla 4-3, se observa que al igual que para las técnicas de FM y de AM, el promedio del error absoluto para las redes neuronales es inferior que para el KNN regresivo. Teniendo un máximo promedio del error absoluto para las redes de alrededor de 84 cm en la coordenada X y de 112 cm en la coordenada Y. Por su parte el KNN regresivo alcanza errores alrededor de los 135 cm en la coordenada X y de 270 cm en la coordenada Y.

**Tabla 4-11.:** Promedio del error absoluto FM - AM para puntos posteriores

Punto	Red 7 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)
1	0,000183636	0,095975	26	113,7444
2	0,281303775	0,464263	14,96667	14,74444
3	0,004407325	0,940363	0,2	0,5
4	0,001329135	4,015965	1,111111	0,744444
5	9,11E-05	0,000283	0	0
6	0,004341721	2,919923	6,188889	7,177778
7	9,80E-05	0,000446	0,344444	0,666667
8	2,96E-05	2,39E-05	0	0
9	0,000230795	0,015371	0	0,455556
10	8,49E-06	0,000779	0	0
11	3,40E-05	0,000584	0	0,177778
12	0,001859244	0,370925	0,055556	0,311111
13	0,000128907	0,018571	0,8	1,311111
14	0,000170383	0,003655	3,611111	0,011111
15	30,86371936	55,49671	22,44444	18,57778
16	26,0618464	37,72689	2,444444	0,544444
17	2,59E-05	0,000187	0	0,077778
18	0,000168228	0,000538	0	0
19	5,29E-05	0,009208	0	0
20	0,00153665	0,089062	0	21,06667
21	7,07E-06	0,000312	0	0,166667
22	0,000931645	0,417415	0	31,34444
23	7,49E-05	0,000418	0	12,16667

### 4.3.3. Tiempo de Ejecución

Al igual que para las sesiones de FM y AM, calculamos para la técnica híbrida utilizando un procedimiento similar al de estas sesiones. En la Tabla **4-13** se observa que como era de esperar al utilizar una mayor cantidad de datos (Los de FM y los de AM), el KNN regresivo requiere mucho mas tiempo que la red neuronal respecto a las técnicas anteriores. Obteniendo una relación de tiempos para los puntos del Fingerprint de 1/253 y de 1/197 para puntos posteriores. Lo que nos hace concluir que si bien la diferencia del promedio del error absoluto en cuanto a la estimación de la posición para puntos posteriores no esta tan grande como con otras técnicas, el tiempo de ejecución si lo es.



**Figura 4-13.:** Promedio del error absoluto FM-AM puntos Intermedios

**Tabla 4-12.:** Promedio del error absoluto FM - AM para puntos intermedios

Punto	Red 7 - 19		KNN Regresivo	
	x (cm)	y (cm)	x (cm)	y (cm)
1	2,285236826	31,15008	11	76,88889
2	84,76815449	67,99175	134,7222	113,4444
3	39,20955382	43,65112	41,22222	139,8889
4	30,28854449	71,41091	47,72222	92,11111
5	68,51856853	36,27379	1,166667	247,4444
6	33,36084409	1,514269	28,94444	268,3333
7	0,211683475	46,1334	26,66667	75
8	2,692248619	46,67726	10,05556	158,0556
9	40,21206741	112,2443	48,55556	194,2222
10	4,857909639	100,4838	71,05556	315,5

**Tabla 4-13.:** Tiempo de ejecución algoritmo estimador de la posición con FM-AM

<b>Tiempo de ejecución (s)</b>		
<b>Algoritmo</b>	<b>Puntos Fingerprint</b>	<b>Puntos Posteriores</b>
Red 4 – 20	0,049	0,038
KnnR	12,3926	7,489

## 4.4. Implementación de redes neuronales en la Raspberry pi

Si bien para la construcción de las redes neuronales y la validación de su funcionamiento se utilizó el software Matlab, para la implementación de las redes en tiempo real en un sistema embebido es necesario buscar una alternativa de software de pruebas que consuma menos recursos de procesamiento que Matlab. Por su similitud con Matlab y facilidad de uso en la Raspberry Pi, elegimos el lenguaje de programación interpretado Python. Este viene instalado por defecto con la distribución Raspbian, cuenta con una variada documentación en la web y es utilizado ampliamente en el ámbito investigativo a nivel mundial.

Para convertir las redes neuronales encontradas en Matlab a un formato que permita hacer la transcripción a Python, Matlab cuenta con la función `genFunction`. Esta recibe como parámetro de entrada una red neuronal junto con los datos del RSS y retorna la posición estimada como parámetro de salida. En la Figura 4-14 se muestra el contenido de la función única para la red neuronal de 7 capas y 19 neuronas obtenida para AM en Matlab.

```

1 function [Y,Xf,Af] = neural_functionAM7_19jun4(X,-,-)
2
16
17
18 %ok<->RPMT0>
19 %==== NEURAL_NETWORK CONSTANTS ====
20
21 % Input 1
22 x1_step1.xoffset = [-24.36;-19.75;-34.98;-20.04;-37.15;-16.84;-19.24;-23.39;-33.44;-23.91;-40.9;-23.87;-37.8;-34.2
23 x1_step1.gain = [0.687285223367698;0.333333333333333;0.169491525423729;0.591715976331361;0.105374077976818;0.32679
24 x1_step1.ymin = -1;
25
26 % Layer 1
27 b1 = [1.0706601353638363072;-1.3452128908292027617;-1.08461492240493865872;-0.088059932915318134272;0.8040979504961
28 IW1_1 = [-1.1233093824942752548 -0.10562191072807669556 -0.44086242059792390968 -0.48900442935135046696 1.22785813
29
30 % Layer 2
31 b2 = [1.484147437879357847;-1.0838343719371279494;-1.2274839413251601439;1.1318258717838840521;0.8734070896355242
32 LW2_1 = [-0.092866023951427109639 0.29620063922869999359 -0.076039516200655056249 1.5970080396838080006 -0.7841962
33
34 % Layer 3
35 b3 = [1.430215940533154928;-1.6318089760549669975;1.602905714450133523;-1.4046542002279944761;-0.01649587969749498
36 LW3_2 = [0.4505596750841753817 0.33074326349736143937 -0.011421650869126193095 -0.19152944105479691173 0.130459264

```

```

119 ~!$setfa, f = cell2mat(t); end
120 end
121
122 %==== MODULE FUNCTIONS =====
123
124 % Map Minimum and Maximum Input Processing Function
125 function y = mapminmax_apply(x,settings)
126 y = bsxfun(@minus,x,settings.xoffset);
127 y = bsxfun(@times,y,settings.gain);
128 y = bsxfun(@plus,y,settings.ymin);
129 end
130
131 % Sigmoid Symmetric Transfer Function
132 function a = tangsig_apply(n,-)
133 a = 2 ./ (1 + exp(-2*n)) - 1;
134 end
135
136 % Map Minimum and Maximum Output Reverse-Processing Function
137 function x = mapminmax_reverse(y,settings)
138 x = bsxfun(@minus,y,settings.ymin);
139 x = bsxfun(@rdivide,x,settings.gain);
140 x = bsxfun(@plus,x,settings.xoffset);
141 end

```

Figura 4-14.: Función de Red Neuronal Generada en Matlab

Como en la Figura 4-14 la función utiliza comandos e instrucciones propias en Matlab, por ello fue necesario encontrar su equivalente para Python. En la Figura 4-15 se muestra la función implementada en Python, allí podemos observar que el grado de complejidad para pasar de un lenguaje al otro no es muy alto.

## 4.5. Prueba de redes neuronales en Raspberry Pi

### 4.5.1. Lectura de los datos y aplicación de las redes

Para realizar la lectura de datos RSS del espectro en tiempo real utilizamos los mismos comandos descritos en la sección 3.2, tanto para FM, AM, FM-AM. En la Figura 4-16 se puede observar como se hace la lectura de datos desde un script de Python para las tres técnicas propuestas. En las líneas finales de cada script, podemos observar como una vez



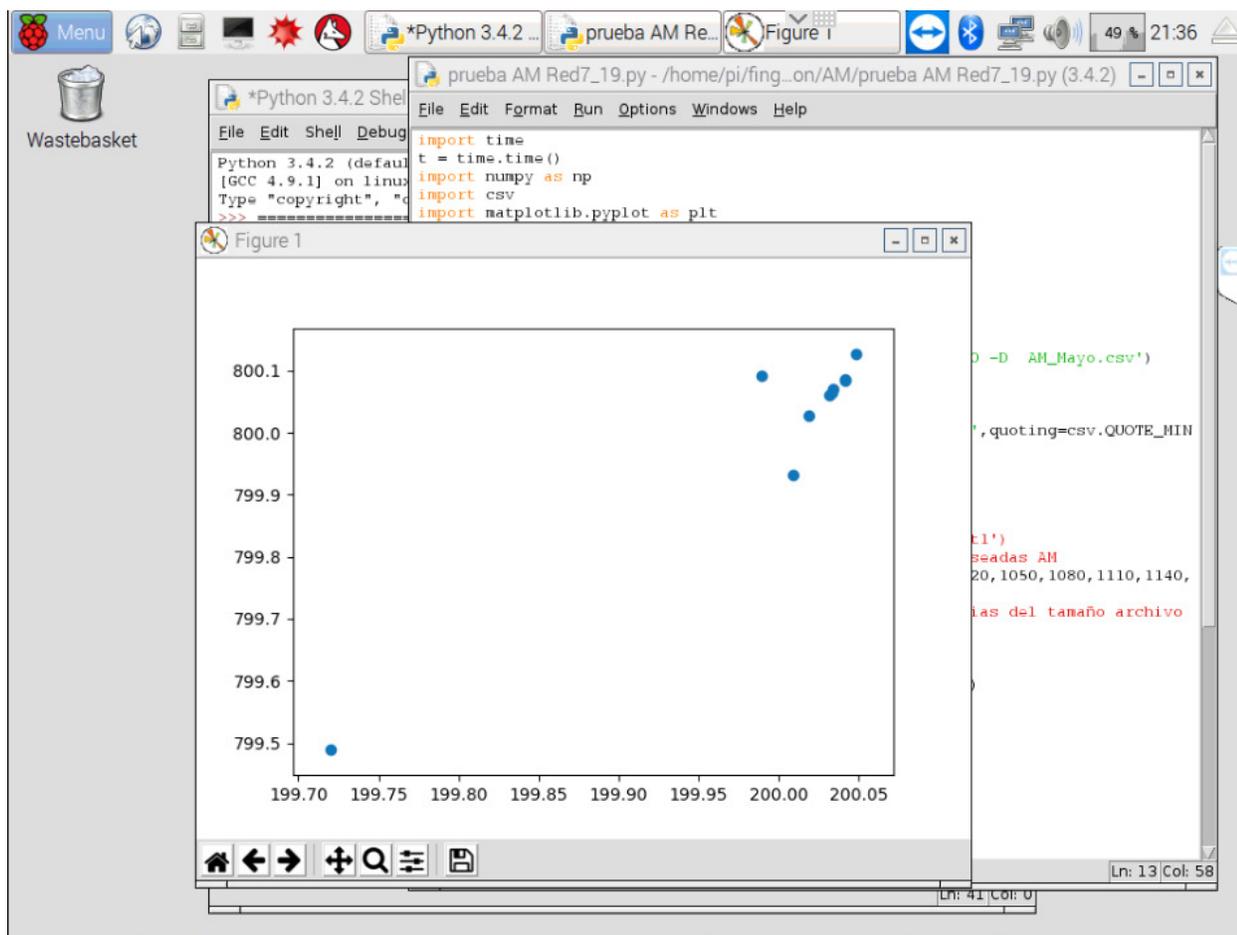


Figura 4-17.: Gráficos de prueba de las Redes Neuronales

#### 4.5.2. Tiempo de estimación de la posición

Tabla 4-14.: Tiempo de ejecución en segundos para cada una de las técnicas en la Raspberry Pi 3 Modelo B

Técnicas	AM (seg)	FM (seg)	AM_FM (seg)
Cargando librerías y Red	3,06389	3,157303	3,02635
Adquisición mínima de datos	1,23	2,93	3,41
Procesamiento de los datos	0,2267	0,5581	0,6247
Tiempo total	4,52059	6,645403	7,06105

Para calcular el tiempo de ejecución para las técnicas de posicionamiento FM, AM y FM-AM implementadas en la Raspberry Pi 3, se configuró el RTL2832u para 1 segundo de

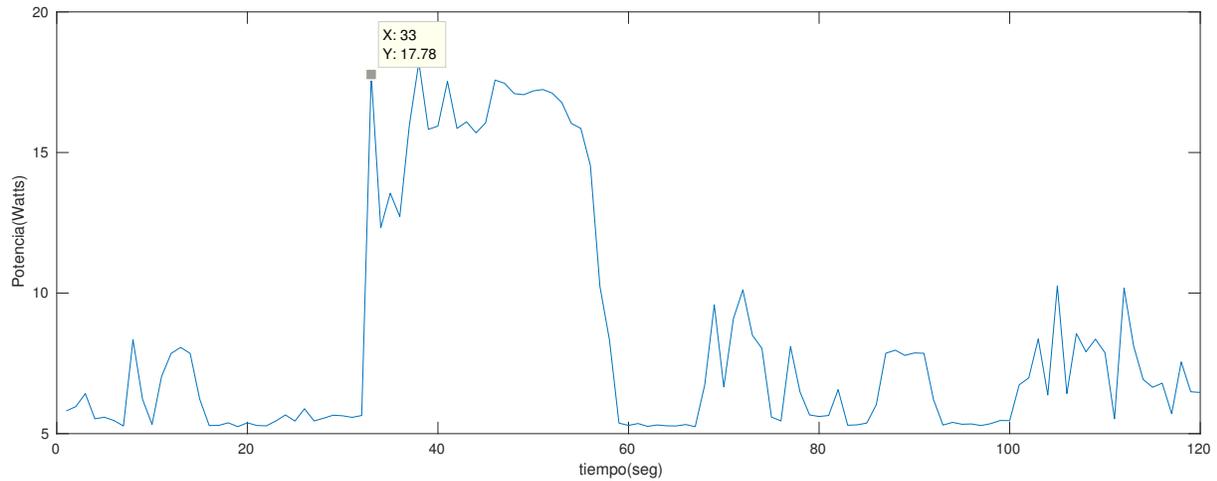
adquisición del RSS. Este es el tiempo mínimo de adquisición de datos manejado por el dispositivo. Para calcular el tiempo de estimación de la posición no es suficiente con conocer esta información, sino que también debemos incluir el tiempo que tarda el Script de python cargando las librerías y la red neuronal y el tiempo de procesamiento de los datos. En la Tabla 4-14 se observa el tiempo de ejecución necesario para la estimación de la posición para cada una de las técnicas. La que menor tiempo tarda en establecer una posición es la técnica de posicionamiento AM con 4.52 segundos, seguida por la técnica de FM con 6.635 segundos. Por último esta la técnica híbrida FM - AM con un tiempo total de 7.061 segundos.

### 4.5.3. Consumo de potencia

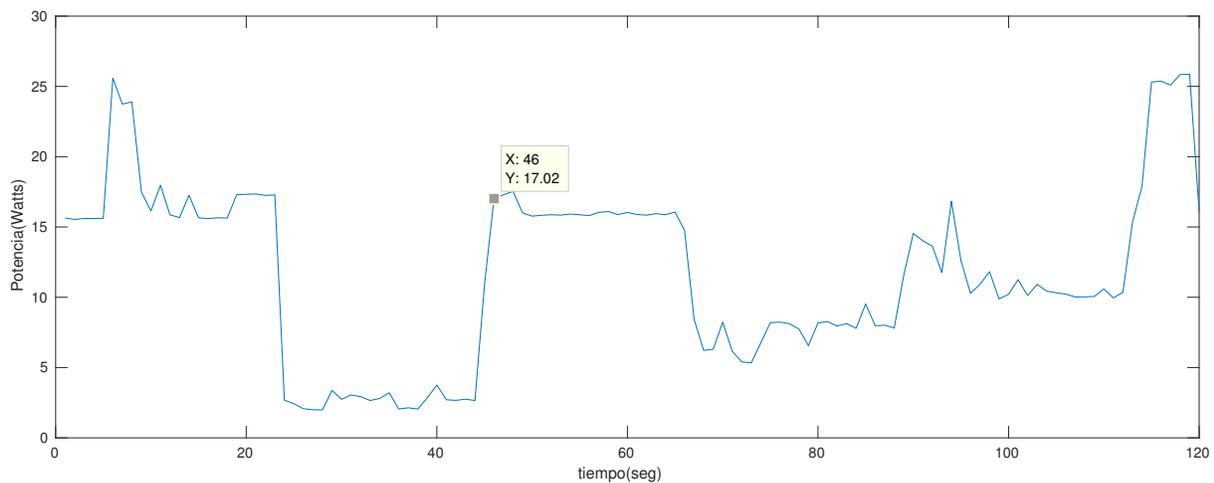
Para establecer un parámetro adicional de comparación entre técnicas se realizó una medición del consumo de potencia promedio por segundo en FM y en AM. Para ello se utilizó un dispositivo usb ZY1276 que sirve de puente entre una batería ADATA P20000D y el Raspberry Pi 3 B. El ZY1276 entrega información del voltaje y la corriente de consumo de la Raspberry Pi en tiempo real y permite almacenarla en un archivo CSV, para luego ser analizada.

**FM:** Para probar el consumo promedio de potencia del posicionamiento FM, se realizó una petición al dispositivo RTL2832u para adquirir datos durante 20 segundos en el rango de señales FM, aplicando posteriormente la respectiva técnica de posicionamiento en FM. En la Figura 4-18 se puede observar la petición realizada en dos momentos diferentes. Para un primer momento, como se observa en la Figura 4-18a, al ejecutar el algoritmo FM en el segundo 33, aumenta de manera considerable el consumo de potencia hasta el segundo 57. Aunque la petición únicamente se realiza para 20 segundos, recordemos como se vió en la Tabla 4-14 que los 3 primeros segundos de ejecución del algoritmo de posicionamiento se utilizan para cargar las librerías, los 20 segundos restantes se usarían para la adquisición de datos y el segundo restante para el procesamiento de estos. El consumo promedio de potencia por segundo para el primer momento es de 16.2148 Vatios(Watts) por segundo(s). Haciendo un análisis similar para el segundo momento de la Figura 4-18b, obtenemos que el consumo promedio de potencia para este es de 16.1203 W/s. Por lo que la adquisición de datos con el algoritmo de FM genera un consumo de potencia de alrededor de 16 W/s.

**AM:** De manera similar que para AM para medir el consumo promedio de potencia del posicionamiento AM, se realizó una petición al dispositivo RTL2832u para adquirir datos durante 20 segundos, pero esta vez en AM. En la Figura 4-19 se puede observar la petición realizada en dos momentos diferentes. Para un primer momento, como se observa en la Figura 4-19a, al igual que para FM, cuando ejecutamos el algoritmo en el segundo 37, aumenta de manera considerable el consumo de potencia hasta el segundo hasta el segundo 61. Tomando 4 segundos mas ya analizados con anterioridad en la Tabla 4-14. El consumo promedio



(a) Medición de potencia 1 FM

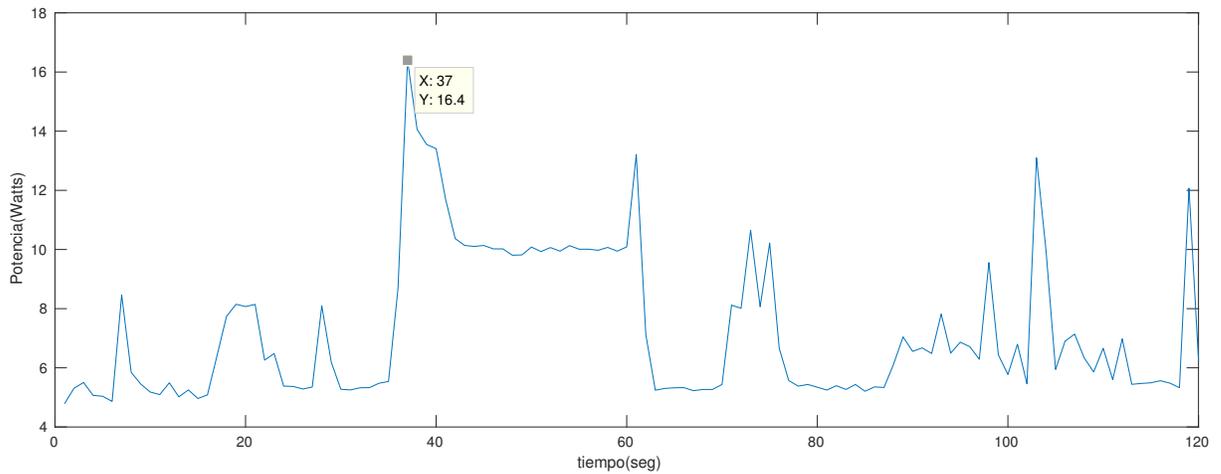


(b) Medición de potencia 2 FM

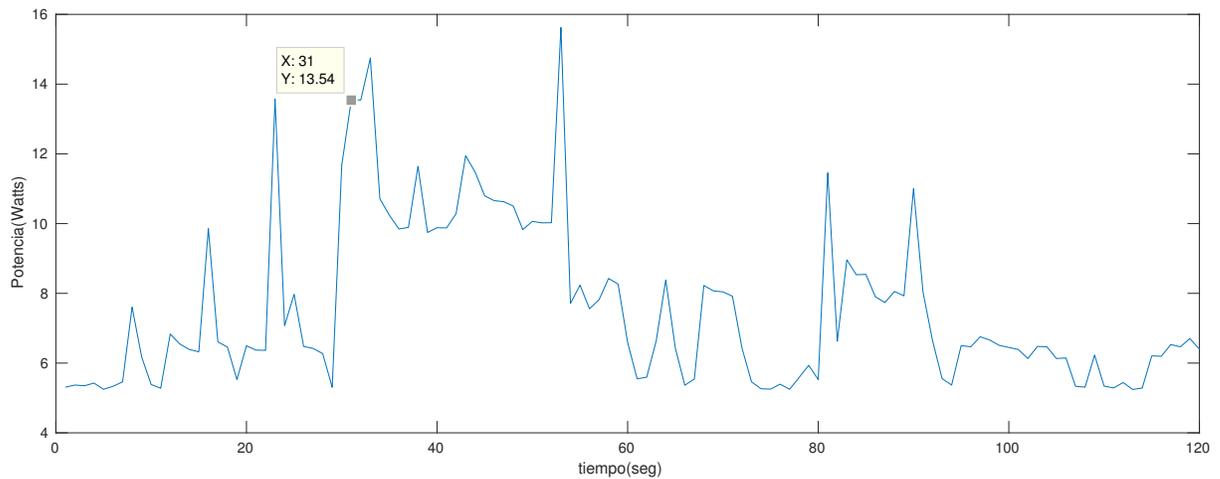
**Figura 4-18.:** Consumo de potencia FM para 20 segundos

de potencia por segundo para el primer momento es de 10.9192 W/s. Haciendo un análisis similar para el segundo momento de la Figura 4-19b, obtenemos que el consumo promedio de potencia para este es de 11.1098 W/s. Por lo que la adquisición de datos con el algoritmo de AM genera un consumo de potencia de alrededor de 11 W/s. 5W menos que para FM. Esto posiblemente se debe a que la cantidad de datos que muestrea para FM por cada segundo, es mucho mayor que la muestreada para AM.

**AM - FM:** Como ya conocemos cual es el consumo independiente para posicionamiento AM y FM, además de la dinámica para la adquisición de datos con el algoritmo híbrido, que realiza primero adquisición de un dato en FM y después de un dato en AM, para luego combinarlos. Podemos asumir que la potencia promedio va estar entre los 11W de AM y los 16W FM, con una potencia promedio por segundo 13.5W.



(a) Medición de potencia 1



(b) Medición de potencia 2

**Figura 4-19.:** Consumo de potencia AM para 20 segundos

#### 4.5.4. Prueba de técnicas en movimiento

Aunque anteriormente se realizaron pruebas del carácter regresivo de las redes neuronales en coordenadas intermedias, estas se realizaron con el dispositivo estático. Aprovechando la versatilidad que nos brinda el Raspberry Pi para movilizar los equipos en el plano de experimentos, nuevamente probaremos el carácter regresivo de los algoritmos, pero esta vez en movimiento para dos rutas establecidas. Las pruebas se realizarán con las técnicas de AM y FM - AM, ya que estas fueron las que presentaron mejores resultados para mediciones intermedias.

### Ruta de navegación 1

La ruta de navegación 1 es una ruta lineal que tiene como origen el punto  $x=200$  cm,  $y = 900$  cm. Finalizando en el punto  $x=200$  cm,  $y = 400$  cm. Pasando por los puntos intermedios que comprenden estas dos coordenadas.

**AM:** Para el trazado de la ruta con adquisición AM, se utilizó la red neuronal de 7 capas y 19 neuronas para la estimación de la posición entre la coordenada inicial y la coordenada final. Este experimento se realizó en tres ocasiones diferentes obteniendo los resultados presentados en la Figura 4-20. Los puntos azules denotan el inicio de la ruta, este color se va tornando verde, hasta llegar a un amarillo al final de la ruta. En las Figuras 4-20b y 4-20c, vemos que el comportamiento general en la navegación aceptable, teniendo en algunas coordenadas intermedias del Fingerprint un error que no supera los 100 cm y que no es necesario instalar una infraestructura de emisión de señales en el espacio de experimentos respecto a la gran mayoría de técnicas por Radio Frecuencia. En la Figura 4-20a vemos que aunque la navegación no se sale de la ruta, hay algunos puntos que denotan imprecisiones, sin embargo el error al igual que en el caso anterior, no excede los 100 cm.

En la Tabla 4-15 vemos que el valor máximo del promedio del error absoluto para AM es de 13.3964 cm en la coordenada X y 0.9087 cm en la coordenada Y para la prueba 3.

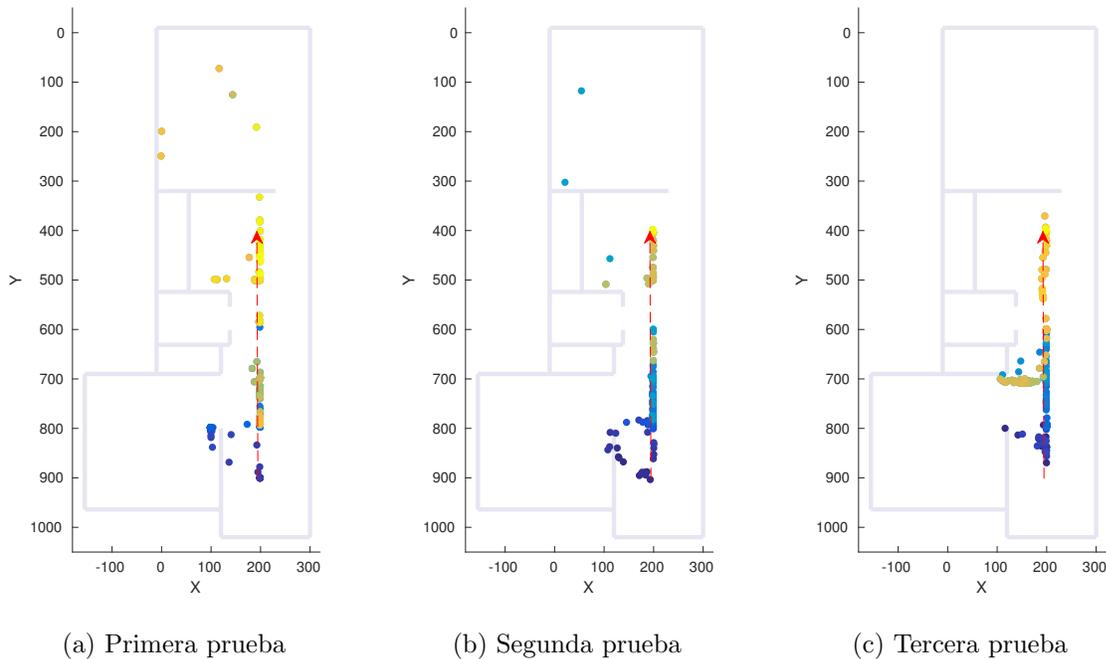
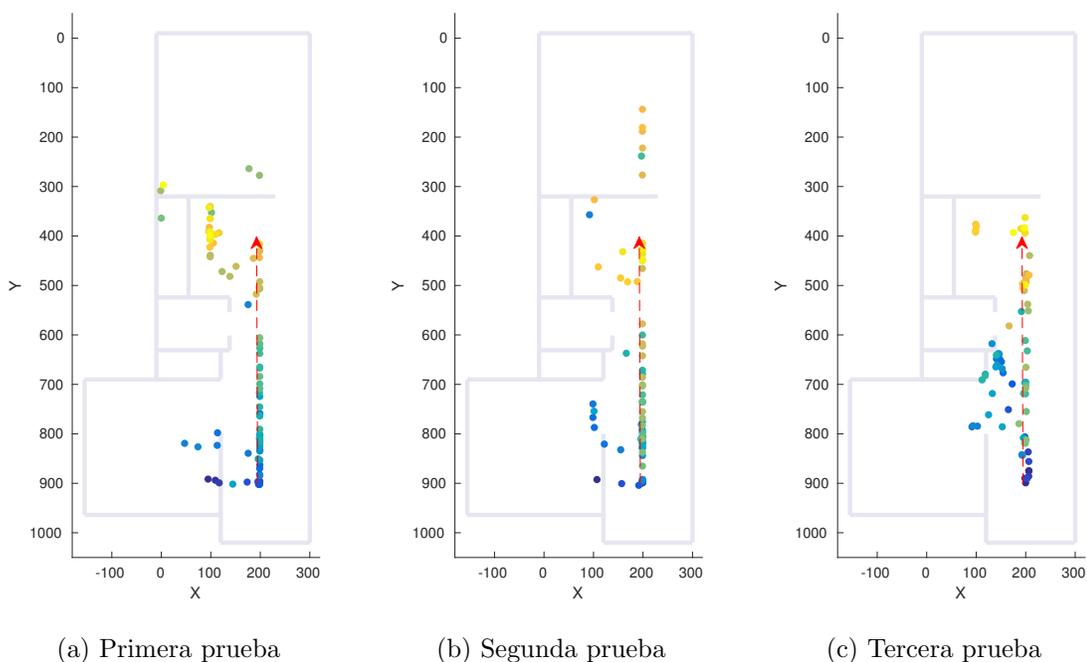


Figura 4-20.: Ruta de Navegación 1 en AM

**AM - FM:** Para el trazado de la ruta con adquisición FM - AM, se utilizó la red neuronal de 4 capas y 20 neuronas para la estimación de la posición entre la coordenada inicial y la coordenada final. De manera similar que para AM, se realizó un experimento en tres ocasiones diferentes obteniendo los resultados presentados en la Figura 4-21. En las Figura 4-21 vemos que al igual que para AM la posición tiende a estar sobre la ruta esperada. Sin embargo es importante recordar que las señales FM son mas propensas a interferencias y atenuaciones en interiores mucho mayores que para AM, por lo que la estimación de la posición puede verse en algún momento afectada por esto.

En la Tabla 4-15 vemos que el valor máximo del promedio del error absoluto para FM - AM es de 40.73 cm en la coordenada X para la prueba 1, muy superior al encontrado en AM.



**Figura 4-21.:** Ruta de Navegación 1 en FM - AM

**Tabla 4-15.:** Máximo Error Absoluto Ruta 1

Técnica	No. Prueba	Máximo Error Absoluto Ruta 1	
		X(cm)	Y(cm)
AM	1	12.6361	7.8198
	2	4.8238	1.7176
	3	13.3964	0.9087
FM AM	1	40,7319	9,1299
	2	14,7428	17,1754
	3	27,2588	2,4113

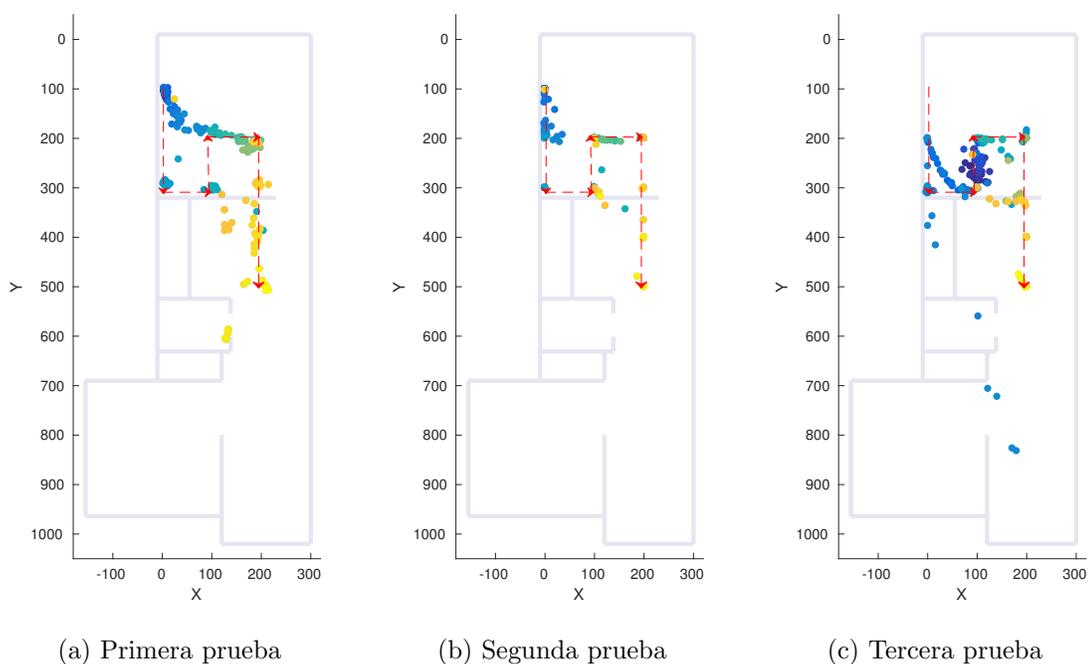
## Ruta de navegación 2

La ruta de navegación 2 se eligió buscando utilizar un espacio diferente a los de la ruta 1 pero terminando cerca de donde finaliza esta. Esta ruta utiliza dos espacios separados por un muro, la habitación principal y la cocina. La coordenada de origen es el punto  $x=0$  cm,  $y = 100$  cm. La coordenada final es el punto  $x=200$  cm,  $y = 500$  cm. La trayectoria realizada a diferencia de la ruta de navegación 1, es una trayectoria no lineal. En las figuras **4-22** y **4-23** se observa en color rojo la trayectoria de navegación de la ruta 2.

**AM** Al igual que para el trazado de la ruta de navegación 1, para la ruta de navegación 2 se utilizó la red neuronal de 7 capas y 19 neuronas para la estimación de la posición entre la coordenada inicial y la coordenada final describiendo el movimiento denotado con rojo en la Figura **4-22**. En las pruebas realizadas presentadas en las Figuras **4-22a**, **4-22b** y **4-22c**, observamos que la tendencia de la estimación es a converger en puntos cercanos a la trayectoria deseada, presentando inestabilidad los puntos cuya coordenada en X es igual a 0 cm. Dichos puntos se encuentran cercanos a una ventana, lo cual nos hace suponer que esta es posiblemente la fuente de oscilación.

En la Tabla **4-16** vemos que el valor máximo del promedio del error absoluto para AM es de 17.0896 cm en la coordenada X para la prueba 1.

**AM - FM** Para el trazado de la ruta de navegación 2 con adquisición FM - AM, se realizó un experimento similar al de AM pero con la red neuronal de 4 capas y 20 neuronas para la estimación de la posición entre la coordenada inicial y la coordenada final de la Figura **4-23**. En las Figura **4-23** se puede observar que con esta técnica la oscilación en  $X = 0$  cm, tiende a disminuir de manera considerable, pero tienen aumentar para otros puntos, esto se debe posiblemente a las afectaciones y atenuaciones que presenta el ambiente para la señal FM.



**Figura 4-22.:** Ruta de Navegación 2 en AM

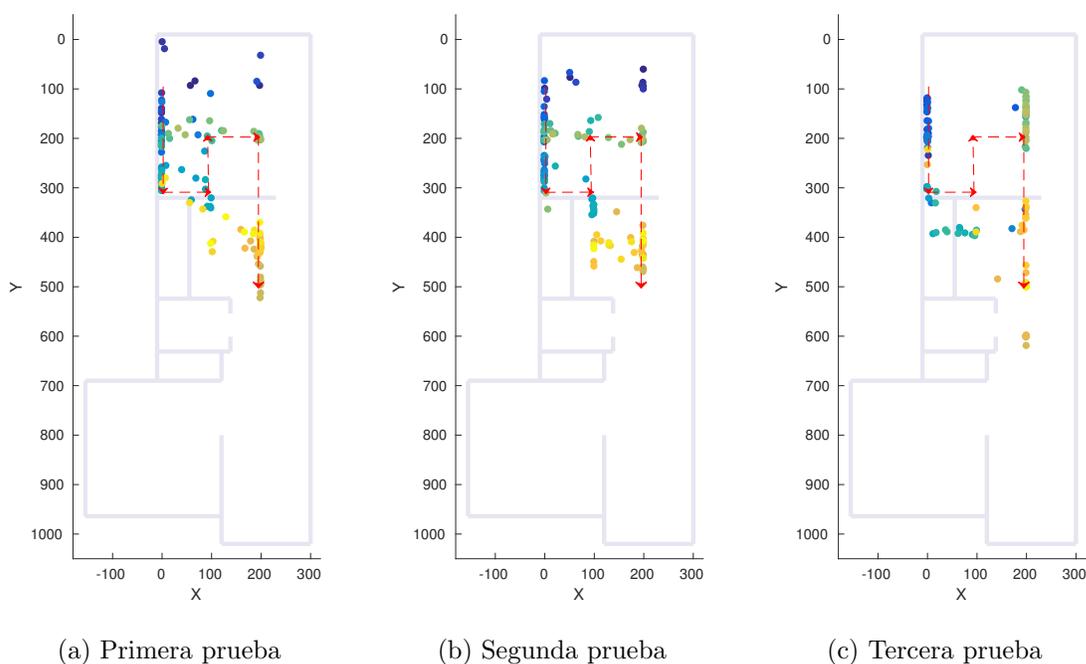
En la Tabla 4-16 se puede observar que el valor máximo del promedio del error absoluto para la ruta de navegación 2 con FM-AM es de 26.7565 cm en la coordenada Y para la prueba 2.

**Tabla 4-16.:** Promedio Error Absoluto Ruta 2

Técnica	No. Prueba	Promedio Error Absoluto Ruta 2	
		X(cm)	Y(cm)
AM	1	17.0896	6.7152
	2	1.4527	1.0798
	3	4.8770	6.0132
FM AM	1	7.4669	10.0955
	2	1.3402	26.7565
	3	12.5168	11.2618

## 4.6. Comparación de resultados

En las secciones anteriores de este capítulo se presentaron resultados inicialmente en términos del promedio del error absoluto para puntos fijos de la grilla y puntos intermedios de la grilla,



**Figura 4-23.:** Ruta de Navegación 2 en FM-AM

resultados del promedio del error absoluto del receptor en movimiento implementado en un sistema embebido, tiempos de estimación de la posición y consumo promedio de potencia. En esta sesión se realizó el gráfico de barras comparativo presentado en la Figura 4-24 recopilando la información anteriormente mencionada para las técnicas propuestas. Para el gráfico de barras correspondiente a los puntos estáticos se promediaron los resultados de las Tablas 4-2,4-4, 4-7, 4-8,4-11 y 4-12, para obtener el promedio del error absoluto general distribuido en el espacio de experimentos.

En la Figura 4-24 se observa de manera general que la técnica de AM presenta el menor tiempo de ejecución con 4.5 segundos, un consumo de potencia de 11W y un menor promedio del error absoluto. Llegando a un valor de alrededor de 10.2 cm en un ambiente de navegación. Este valor supera muchas de las técnicas presentadas para posicionamiento en interiores en el Estado del Arte y en diferentes técnicas analizadas en trabajos como los presentados por [61, 77, 78], además de superar el error asociado a los GPS en exteriores que ronda entre los 2.5 m y los 3 m.

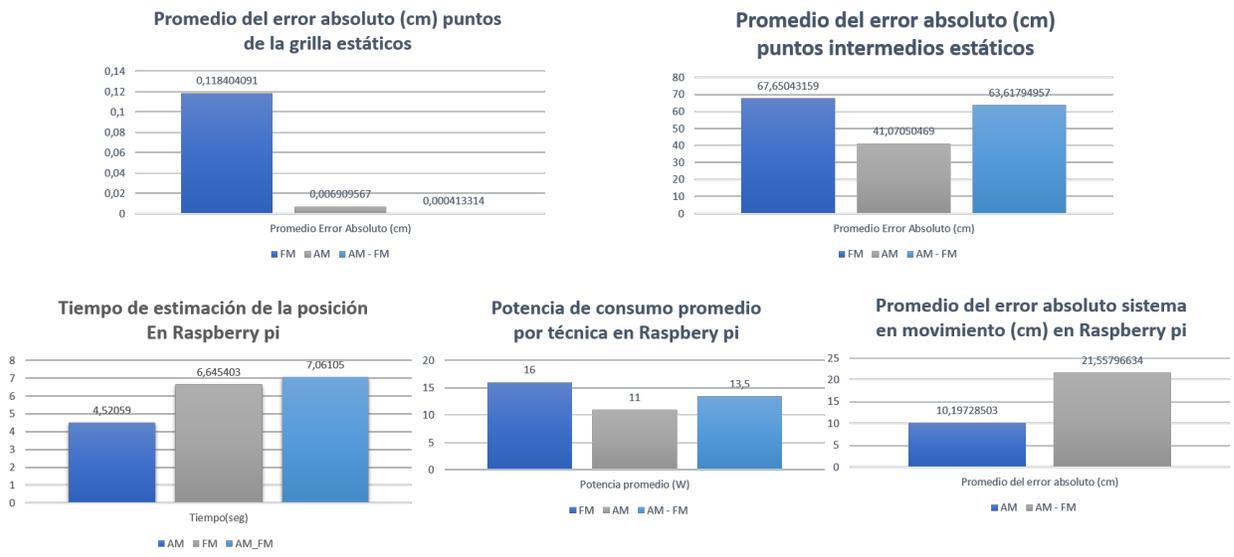


Figura 4-24.: Compilado de resultados Deep Learning

# 5. Conclusiones y recomendaciones

## 5.1. Conclusiones

El posicionamiento en interiores ha sido explorado desde muchas técnicas y tecnologías asociadas, entre ellas las tecnologías de radio frecuencia. Contando con gran cantidad de trabajos que proponen el uso de diversas estrategias para el posicionamiento, en su gran mayoría con la necesidad de adecuar una infraestructura o de utilizar equipos de gran costo y desempeño para su funcionamiento.

Con este trabajo demostramos que no se requiere de costosos equipos, ni adecuaciones espaciales de infraestructura para lograr un posicionamiento en interiores. Utilizando la infraestructura existente de las radio estaciones FM y AM de la ciudad de Medellín, fueron propuestas tres metodologías para posicionamiento en interiores, una con señales de radio FM, otra con AM y una tercera que combina ambas señales.

Algunas técnicas con señales FM que no usan Deep Learning ya habían sido exploradas en el pasado. Sin embargo, con señales AM o la combinación de ambas no se encontró ninguna referencia en la literatura consultada, lo cual aporta nuevas posibilidades con este tipo de señales. Además, los resultados obtenidos para la señales AM y FM-AM, obtuvieron un mejor desempeño que los obtenidos con señales FM.

Las metodologías propuestas se realizaron utilizando Deep Learning, estas fueron comparadas con el algoritmo KNN Regresivo, que es el algoritmo comúnmente utilizado para posicionamiento en interiores con Fingerprint. Obteniendo mejores resultados con Deep Learning en cuanto a precisión y tiempo de ejecución.

Al comparar los resultados obtenidos con muchas de las técnicas propuestas para posicionamiento en interiores vemos como sin necesidad de adecuar o tener una infraestructura implementada en el área de experimentos o cercana a esta, es posible obtener un promedio del error absoluto inferior a los 22 cm para un sistema o punto en movimiento en el interior de un edificio.

## 5.2. Recomendaciones

- Aunque la Agencia Nacional para la Regulación del Espectro ANE, reglamenta el rango de potencia de transmisión de las radio-estaciones, durante la construcción del Fingerprint se identificó que algunas radio-estaciones utilizan mas o menos potencia acorde

con la hora del día. Por lo cual se sugiere al lector que quiera hacer uso del espectro radio eléctrico, tener este factor en cuenta.

- En el transcurso del año en Colombia se realiza asignación de frecuencias a nuevas radioestaciones tanto en AM como en FM o cambios de diales por parte de algunas radioestaciones existentes. Por esto es recomendable realizar un Fingerprint y ajuste de las técnicas por lo menos 3 veces al año o como alternativa adicional utilizar para el entrenamiento de las redes, radioestaciones que lleven varios años sin realizar cambio de dial.

# A. Instalación y Configuración

## RTL2832u

Para la instalación y configuración de los controladores USB es necesario estar conectados a internet y a través de una consola o terminal de Linux introducir los siguientes comandos:

```
sudo apt-get install git
sudo apt-get install cmake
sudo apt-get install libusb-1.0-0.dev
sudo apt-get install build-essential
```

Posteriormente se descargan las librerías rtl-sdr utilizando el comando:

```
git clone git://git.osmocom.org/rtl-sdr.git
```

El comando anterior creará una carpeta en el directorio raíz con el nombre rtl-sdr, en la que se ubican las librerías del dispositivo. Para la adquisición correcta del índice de potencia RSS en el rango de frecuencias entre los 100 Khz y los 30 Mhz, en el cual se encuentran las señales de AM, es necesario modificar la variable *direct\_sampling* y la función *verbose\_direct\_sampling* de la librería *rtl\_power.c* asignándole el valor de 2 como se muestra en la Figura A-1.

Una vez realizado este cambio y guardado el archivo, volvemos a la terminal de linux y digitamos los comandos para compilar e instalar las librerías rtl-sdr en el computador:

```
case 'P':
    peak_hold = 1;
    break;
case 'D':
    direct_sampling = 2;
    break;
case 'O':
    offset_tuning = 1;
    break;

sigaction(SIGPIPE, &sigact, NULL);
#else
SetConsoleCtrlHandler( (HANDLER_ROUTINE) sighandler, TRUE );
#endif

if (direct_sampling) {
    verbose_direct_sampling(dev, 2);
}

if (offset_tuning) {
    verbose_offset_tuning(dev);
}
```

Figura A-1.: Modificación a la librería rtl\_power.c

```
cd rtl-sdr/  
mkdir build  
cd build  
cmake ../  
make  
sudo make install  
sudo ldconfig
```

# Bibliografía

- [1] H. Kaplan, *Understanding GPS Principles and Applications*, 2nd ed. ARTECH HOUSE, INC, 2006, vol. 1.
- [2] R. Mautz, “Indoor positioning technologies,” Master’s thesis, Publisher ETH Zurich, 2012.
- [3] Y. Gu, A. Lo, and I. Niemegeers, “A survey of indoor positioning systems for wireless personal networks,” *IEEE Communications Surveys Tutorials*, vol. 11, no. 1, pp. 13–32, First 2009.
- [4] G. De Angelis, V. Pasku, A. De Angelis, M. Dionigi, M. Mongiardo, A. Moschitta, and P. Carbone, “An Indoor AC Magnetic Positioning System,” *IEEE Transactions on Instrumentation and Measurement*, pp. 1–1, 2015. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7000532>
- [5] J. Blankenbach, A. Norrdine, and H. Hellmers, “A robust and precise 3D indoor positioning system for harsh environments,” in *2012 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2012 - Conference Proceedings*, 2012.
- [6] A. Suzuki, T. Iyota, Y. Choi, Y. Kubota, K. Watanabe, and A. Yamane, “Measurement accuracy on indoor positioning system using spread spectrum ultrasonic waves,” in *Autonomous Robots and Agents, 2009. ICARA 2009. 4th International Conference on*. IEEE, 2009, pp. 294–297.
- [7] D. Ruiz, E. Garcia, J. Urena, J. M. Villadangos, J. J. Garcia, and C. De Marziani, “Performance comparison of correlation-based receive filters in an ultrasonic indoor positioning system,” in *Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, 2014 IEEE International*. IEEE, 2014, pp. 1548–1551.
- [8] S. Murata, C. Yara, K. Kaneta, S. Ioroi, and H. Tanaka, “Accurate indoor positioning system using near-ultrasonic sound from a smartphone,” in *Next Generation Mobile Apps, Services and Technologies (NGMAST), 2014 Eighth International Conference on*. IEEE, 2014, pp. 13–18.
- [9] A. Yazici, U. Yayan, and H. Yücel, “An ultrasonic based indoor positioning system,” in *Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on*. IEEE, 2011, pp. 585–589.

- [10] S. Latif, R. Tariq, W. Haq, and U. Hashmi, “INDOOR POSITIONING SYSTEM using ultrasonics,” *Proceedings of 2012 9th International Bhurban Conference on Applied Sciences & Technology (IBCAST)*, pp. 440–444, 2012. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6177596>
- [11] H. Niwa, K. Kodaka, Y. Sakamoto, M. Otake, S. Kawaguchi, K. Fujii, Y. Kanemori, and S. Sugano, “Gps-based indoor positioning system with multi-channel pseudolite,” in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 905–910.
- [12] A. Vervisch-Picois, I. Selmi, Y. Gottesman, and N. Samama, “Current status of the repealite based approach: A sub-meter indoor positioning system,” in *Programme and Abstract Book 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC 2010*, 2010.
- [13] A. Vervisch-Picois and N. Samama, “First experimental performances of the repealite based indoor positioning system,” in *Wireless Communication Systems (ISWCS), 2012 International Symposium on*. IEEE, 2012, pp. 636–640.
- [14] I. Selmi, A. Vervisch-Picois, Y. Gottesman, and N. Samama, “Optical and radio calibration of the repealite based indoor positioning system,” in *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*. IEEE, 2012, pp. 1–8.
- [15] R. Campos and L. Lovisolo, *RF Positioning: Fundamentals, Applications, and Tools*., ser. GNSS technology and applications series. Artech House, 2015, ch. Fundamentals Of RF Fingerprint, pp. 79–109. [Online]. Available: <https://books.google.com.co/books?id=oLTQCgAAQBAJ>
- [16] V. Moghtadaiee, A. G. Dempster, and S. Lim, “Indoor localization using FM radio signals: A fingerprinting approach,” *2011 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2011*, pp. 3–9, 2011.
- [17] Y. Chen, D. Lymberopoulos, J. Liu, and B. Priyantha, “Indoor localization using FM signals,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 8, pp. 1502–1517, 2013.
- [18] J. Q. Rodriguez, “Emisión y Recepción en Radio,” *Revista digital para profesionales de la enseñanza*, vol. 7, no. ISSN: 1989-4023, pp. 1–13, 2010.
- [19] MinTIC, “PLAN TÉCNICO NACIONAL DE RADIODIFUSIÓN SONORA AMPLI-TUD MODULADA (A.M.),” <http://www.mintic.gov.co/portal/604/articles-62446\plan\tecnico\nacional\radiodifusion\sonora\am\actualizado\diciembre\2017.pdf>, 2017.

- [20] —, “PLAN TÉCNICO NACIONAL DE RADIODIFUSIÓN SONORA FRECUENCIA MODULADA (F.M.),” <http://www.mintic.gov.co/portal/604/articles-62446\plan\tecnico\nacional\radiodifusion\sonora\fm\actualizado\diciembre\2017.pdf>, 2017.
- [21] I. Fundacion Wikimedia, “Anexo:Emisoras de Medellín,” <https://es.wikipedia.org/wiki/Anexo:Emisoras\de\Medellin>, 2018.
- [22] A. Popleteev, V. Osmani, and O. Mayora, “Investigation of indoor localization with ambient FM radio stations,” in *2012 IEEE International Conference on Pervasive Computing and Communications, PerCom 2012*, no. March, 2012, pp. 171–179.
- [23] A. Matic, A. Popleteev, V. Osmani, and O. Mayora-Ibarra, “FM radio for indoor localization with spontaneous recalibration,” in *Pervasive and Mobile Computing*, vol. 6, no. 6. Elsevier B.V., 2010, pp. 642–656. [Online]. Available: <http://dx.doi.org/10.1016/j.pmcj.2010.08.005>
- [24] A. Anastasijevi and A. Neškovi, “A practical realisation of kNN indoor positioning model for GSM,” in *Telecommunications Forum (TELFOR), 2012 20th*, no. m, 2012, pp. 1–4.
- [25] J. Machaj and P. Brida, “Using GSM Signals for Fingerprint-based Indoor Positioning System,” in *ELEKTRO, 2014*, 2014, pp. 64–67.
- [26] A. Varshavsky, E. D. Lara, J. Hightower, A. Lamarca, and V. Otsason, “GSM indoor localization,” *Pervasive and Mobile Computing*, vol. 3, pp. 698–720, 2007.
- [27] J. Niu, B. Wang, L. Cheng, and J. J. P. C. Rodrigues, “WicLoc : An Indoor Localization System based on WiFi Fingerprints and Crowdsourcing,” in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 3008–3013.
- [28] U. de Washington, “Machine Learning: Regression,” <https://es.coursera.org/learn/ml-regression/lecture/8pPF9/k-nearest-neighbors-regression>.
- [29] S. Sayad, “K Nearest Neighbors - Regression,” <http://www.saedsayad.com/k\{-\}nearest\{-\}neighbors\{-\}reg.htm>.
- [30] LeCun Y, Bengio Y, Hinton G, “Deep learning,” *Deep learning Nature*, vol. 521, pp. 436–444, 2015.
- [31] R. Lopez. (2017, enero) Introducción al deep learning. <https://iaarhub.github.io/capacitacion/2017/06/13/introduccion-al-deep-learning/>.
- [32] R. Memisevic, “Deep Learning: Architectures, algorithms, applications,” <https://pdfs.semanticscholar.org/cf9c/1d20c5eb1749b99c5327b37e29f79337ee83.pdf>, aug 2015.

- [33] G. et-al 2016, *Deep Learning*. MIT Press, 2016, ch. 6,9,10, <http://www.deeplearningbook.org>.
- [34] R. Xu, W. Chen, Y. Xu, and S. Ji, "A New Indoor Positioning System Architecture Using GPS Signals," pp. 10 074–10 087, 2015.
- [35] K. Ozsoy, A. Bozkurt, and I. Tekin, "2D Indoor positioning system using GPS signals," *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, no. September, pp. 15–17, 2010.
- [36] G. De Angelis, A. De Angelis, M. Dionigi, M. Mongiardo, A. Moschitta, and P. Carbone, "An accurate Indoor Position-measurement system using mutually coupled resonating circuits," in *2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, 2014, pp. 844–849. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6860862>
- [37] P. Lou, H. Zhang, X. Zhang, M. Yao, and Z. Xu, "Fundamental analysis for indoor visible light positioning system," in *2012 1st IEEE International Conference on Communications in China Workshops, ICCCW 2012*, 2012, pp. 59–63.
- [38] P. Luo, Z. Ghassemlooy, Hoa Le Minh, A. Khalighi, X. Zhang, M. Zhang, and Changyuan Yu, "Experimental demonstration of an indoor visible light communication positioning system using dual-tone multi-frequency technique," in *2014 3rd International Workshop in Optical Wireless Communications (IWOW)*, 2014, pp. 55–59. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6950776>
- [39] T.-h. Do, J. Hwang, and M. Yoo, "TDoA Based Indoor Visible Light Positioning Systems," pp. 456–458, 2013.
- [40] M. Yasir, S. Member, S.-w. Ho, and B. N. Vellambi, "Indoor Positioning System Using Visible Light," vol. 32, no. 19, pp. 3306–3316, 2014.
- [41] A. Perttula, H. Leppakoski, M. Kirkko-Jaakkola, P. Davidson, J. Collin, and J. Takala, "Distributed Indoor Positioning System With Inertial Measurements and Map Matching," pp. 2682–2695, 2014.
- [42] H. K. Parikh and W. R. Michalson, "Error mechanisms in an RF-based indoor positioning system," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 0, 2008, pp. 5320–5323.
- [43] L. Jayatilleke and N. Zhang, "Landmark-based localization for Unmanned Aerial Vehicles," in *SysCon 2013 - 7th Annual IEEE International Systems Conference, Proceedings*, 2013, pp. 448–451.

- [44] B. Bischoff, D. Nguyen-Tuong, F. Streichert, M. Ewert, and A. Knoll, "Fusing vision and odometry for accurate indoor robot localization," in *2012 12th International Conference on Control, Automation, Robotics and Vision, ICARCV 2012*, vol. 2012, no. December, 2012, pp. 347–352.
- [45] P. De Cristoforis, M. Nitsche, T. Krajník, T. Pire, and M. Mejail, "Hybrid vision-based navigation for mobile robots in mixed indoor/outdoor environments," *Pattern Recognition Letters*, vol. 53, pp. 118–128, 2015. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0167865514003274>
- [46] R. Mebarki and V. Lippiello, "Image moments-based velocity estimation of UAVs in GPS denied environments," in *2014 IEEE International Symposium on Safety, Security, and Rescue Robotics (2014)*, 2014, pp. 1–6. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7017659>
- [47] C. Wang, T. Wang, J. Liang, Y. Chen, Y. Zhang, and C. Wang, "Monocular visual SLAM for small UAVs in GPS-denied environments," in *2012 IEEE International Conference on Robotics and Biomimetics, ROBIO 2012 - Conference Digest*, 2012, pp. 896–901.
- [48] Y. M. Mustafah, A. W. Azman, and F. Akbar, "Indoor UAV Positioning Using Stereo Vision Sensor," *Procedia Engineering*, vol. 41, no. Iris, pp. 575–579, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.proeng.2012.07.214>
- [49] F. Belloni, V. Ranki, A. Kainulainen, and A. Richter, "Angle-based indoor positioning system for open indoor environments," *Proceedings - 6th Workshop on Positioning, Navigation and Communication, WPNC 2009*, vol. 2009, pp. 261–265, 2009.
- [50] Y. He, W. Meng, L. Ma, and Z. Deng, "Rapid deployment of APs in WLAN indoor positioning system," in *Proceedings of the 2011 6th International ICST Conference on Communications and Networking in China, CHINACOM 2011*, 2011, pp. 268–273.
- [51] G. B. Moon, M. Beom Hur, and G.-I. Jee, "An indoor positioning system for a first responder in an emergency environment," *International Conference on Control, Automation and Systems*, pp. 1368–1372, 2012.
- [52] G. B. Moon, S. Chun, M. B. Hur, and G. I. Jee, "A robust indoor positioning system using two-stage EKF SLAM for first responders in an emergency environment," in *International Conference on Control, Automation and Systems*, no. Iccas, 2013, pp. 707–711.
- [53] Y. Zhuang, Z. Syed, and J. Georgy, "Autonomous smartphone based WiFi positioning system by using access points localization and crowdsourcing," *Pervasive and Mobile Computing*, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.pmcj.2015.02.001>

- [54] M. Al, A. Amali, M. H. Khir, N. M. Saad, and S. C. Dass, "WiFi Fingerprinting Indoor Positioning with Multiple Access Points in a Single Base Station using Probabilistic Method," *International Journal of Applied Engineering Research*, vol. 12, no. 6, pp. 1102–1113, 2017.
- [55] N. Swangmuang and P. Krishnamurthy, "An effective location fingerprint model for wireless indoor localization," *Pervasive and Mobile Computing*, vol. 4, no. 6, pp. 836–850, 2008. [Online]. Available: <http://dx.doi.org/10.1016/j.pmcj.2008.04.005>
- [56] W. Le, Z. Wang, J. Wang, G. Zhao, and H. Miao, "A Novel WIFI Indoor Positioning Method Based on Genetic Algorithm and Twin Support Vector Regression," in *Control and Decision Conference (2014 CCDC), The 26th Chinese*, no. 61174059, 2014, pp. 4859–4862.
- [57] H. Zou, Y. Luo, X. Lu, H. Jiang, and L. Xie, "A Mutual Information Based Online Access Point Selection Strategy for WiFi Indoor Localization," in *Automation Science and Engineering (CASE), 2015 IEEE International Conference on*, 2015, pp. 24–28.
- [58] J. Zhao and J. Wang, "WiFi Indoor Positioning Algorithm Based on Machine Learning," in *Electronics Information and Emergency Communication (ICEIEC), 2017 7th IEEE International Conference on*, 2017.
- [59] P. Torteeka and X. Chundi, "Indoor positioning based on Wi-Fi Fingerprint Technique using Fuzzy K-Nearest Neighbor," *Applied Sciences and Technology (IBCAST), 2014 11th International Bhurban Conference on*, pp. 461–465, 2014.
- [60] P. Le Dortz, N. and Gain, F. and Zetterberg, "Wifi Fingerprint Indoor Positioning System Using Probability Distribution Comparison," *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 2301–2304, 2012.
- [61] P. Ali-Rantala, L. Ukkonen, L. Sydanheimo, M. Keskilammi, and M. Kivikoski, "Different kinds of walls and their effect on the attenuation of radiowaves indoors," *IEEE Antennas and Propagation Society International Symposium. Digest. Held in conjunction with: USNC/CNC/URSI North American Radio Sci. Meeting (Cat. No.03CH37450)*, vol. 3, pp. 1020–1023, 2003.
- [62] Y. Tian, B. Denby, and U. Pierre, "Hybrid Indoor Localization using GSM Fingerprints, Embedded Sensors and a Particle Filter," in *Wireless Communications Systems (ISWCS), 2014 11th International Symposium on*, 2014, pp. 542–547.
- [63] A. Popleteev, V. Osmani, and O. Mayora, "Investigation of indoor localization with ambient FM radio stations," in *2012 IEEE International Conference on Pervasive Computing and Communications*, 2012, pp. 19–23.

- [64] M. Interaction, “Indoor Positioning Using FM Radio,” *International Journal of Hand-held Computing Research (IJHCR)*, vol. 1, no. 3, pp. 19–31, 2010.
- [65] V. Moghtadaiee and A. G. Dempster, “FM Radio Signals,” *IEEE TRANSACTIONS ON BROADCASTING*, vol. 60, no. 2, pp. 336–346, 2014.
- [66] X. Wang, L. Gao, S. Mao, and S. Pandey, “DeepFi : Deep Learning for Indoor Fingerprinting Using Channel State Information,” in *Wireless Communications and Networking Conference (WCNC), 2015 IEEE*, 2015, pp. 9–12.
- [67] J. Wietrzykowski, “Low-Effort Place Recognition with WiFi Fingerprints Using Deep Learning,” in *Automation 2017. ICA 2017.*, 2017.
- [68] M. Siller, “A Fingerprinting Indoor Localization Algorithm Based Deep Learning,” in *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, 2016, pp. 1006–1011.
- [69] F. Samadzadegan and G. Abdi, “Autonomous Navigation of Unmanned Aerial Vehicles Based on Multi-Sensor Data Fusion,” *20th Iranian Conference on Electrical Engineering*, no. ICEE20 12, 2012.
- [70] C.-s. Wang and C.-l. Chen, “RFID-based and Kinect-based indoor positioning system,” in *2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE)*, 2014, pp. 1–4. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6934458>
- [71] F. WANG, J.-Q. CUI, B.-M. CHEN, and T. H. LEE, “A Comprehensive UAV Indoor Navigation System Based on Vision Optical Flow and Laser FastSLAM,” *Acta Automatica Sinica*, vol. 39, no. 11, pp. 1889–1899, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1874102913600804>
- [72] T. Pany, J. Winkel, B. Riedl, H. Niedermeier, B. Eissfeller, T. Wörz, R. Schweikert, S. Lagrasta, R. Nicolé, G. López-Risueño, and D. Jiménez-Baños, “Experimental results from an ultra-tightly coupled GPS/Galileo/WiFi/ZigBee/ MEMS-IMU indoor navigation test system featuring coherent integration times of several seconds,” in *Programme and Abstract Book - 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC 2010*, no. 1, 2010.
- [73] A. C. de Colombia. (2015, jul) Requisitos generales de aeronavegabilidad y operaciones para rpas (numeral 4.25.8.2). [Online]. Available: <http://www.aerocivil.gov.co/autoridad-de-la-aviacion-civil/certificacion-y-licenciamiento/Documents/PROYECTO%20BORRADOR%20CIRCULAR%20RPAS.pdf>

- [74] J. Polo, G. Hornero, C. Duijneveld, A. García, and O. Casas, “Design of a low-cost wireless sensor network with uav mobile node for agricultural applications,” *Computers and Electronics in Agriculture*, vol. 119, pp. 19 – 32, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169915002999>
- [75] S. Ward, J. Hensler, B. Alsalam, and L. F. Gonzalez, “Autonomous uavs wildlife detection using thermal imaging, predictive navigation and computer vision,” in *2016 IEEE Aerospace Conference*, March 2016, pp. 1–8.
- [76] F. J. CASAS MARÍN. (2016, jun) Desarrollo de un software de control basado en raspberry pi ii para la implementación de algoritmos de control en una plataforma 3d (con 3 grados de libertad). [Online]. Available: <https://riunet.upv.es/handle/10251/65984?>
- [77] S. Adler, S. Schmitt, and M. Kyas, “Path loss and multipath effects in a real world indoor localization scenario,” *2014 11th Workshop on Positioning, Navigation and Communication, WPNC 2014*, 2014.
- [78] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang, “Indoor Fingerprint Positioning Based on Wi-Fi : An Overview,” 2017.
- [79] T. Kuffik, J. Lanir, E. Dim, A. Wecker, M. Corra, and M. Zancanaro, “Indoor Positioning in Cultural Heritage : Challenges and a Solution,” pp. 1–5, 2012.
- [80] A. Khalajmehrabadi, S. Member, N. Gatsis, D. Akopian, and S. Member, “Modern WLAN Fingerprinting Indoor Positioning Methods and Deployment Challenges,” no. c, pp. 1–30, 2017.
- [81] K. A. Nuaimi, A. Ain, and A. Ain, “A Survey of Indoor Positioning Systems and Algorithms,” pp. 185–190, 2011.
- [82] L. Mainetti, L. Patrono, and I. Sergi, “A Survey on Indoor Positioning Systems,” *2014 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2014.
- [83] H. Liu, S. Member, H. Darabi, P. Banerjee, and J. Liu, “Survey of Wireless Indoor Positioning Techniques and Systems,” vol. 37, no. 6, pp. 1067–1080, 2007.
- [84] L. Batistić and M. Tomic, “Overview of Indoor Positioning System Technologies,” *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 473–478, 2018.