

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

---

---

SCUOLA DI SCIENZE

Corso di Laurea in Informatica per il management

**STUDIO E VALUTAZIONE  
SPERIMENTALE  
DI TECNICHE DI RADIO  
FINGERPRINTING PER  
NAVIGAZIONE INDOOR CON  
DISPOSITIVI MOBILI ANDROID**

**Relatore:**  
**Chiar.mo Prof.**  
**Marco Di Felice**

**Presentata da:**  
**Federico Stivani**

**Sessione II**  
**Anno Accademico 2013-2014**

Nessun profumo vale l'odore di quel fuoco

B.-P.

Dedicato a tutti i miei compagni di strada, che mi hanno permesso di essere quello che  
sono.

# Indice

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Tecniche e tecnologie location-based</b>                     | <b>1</b>  |
| 1.1      | Il mondo indoor è ben diverso da fuori . . . . .                | 1         |
| 1.2      | Valutazione delle performance . . . . .                         | 3         |
| 1.3      | Tecniche di localizzazione senza fili . . . . .                 | 4         |
| 1.3.1    | Tecniche basate su Infrarossi (IR) . . . . .                    | 4         |
| 1.3.2    | Tecniche basate su Bluetooth . . . . .                          | 5         |
| 1.3.3    | Tecniche basate su ultrasuoni . . . . .                         | 5         |
| 1.3.4    | Tecniche basate su RFID . . . . .                               | 7         |
| 1.3.5    | Localizzazione mediante tracciamento ottico . . . . .           | 10        |
| 1.4      | Tecniche di localizzazione basate sul raggio d'azione . . . . . | 11        |
| 1.4.1    | Received Signal Strenght Indication (RSSI) . . . . .            | 11        |
| 1.4.2    | Time Of Arrival ( <i>TOA</i> ) . . . . .                        | 11        |
| 1.4.3    | Time Difference Of Arrival ( <i>TDOA</i> ) . . . . .            | 12        |
| 1.4.4    | Angle Of Arrival (AOA) . . . . .                                | 13        |
| <b>2</b> | <b>Il Wi-Fi fingerprinting</b>                                  | <b>15</b> |
| 2.1      | Introduzione . . . . .  | 15        |
| 2.2      | L'infrastruttura . . . . .                                      | 15        |
| 2.3      | La tecnica . . . . .  | 16        |
| 2.4      | Fase offline . . . . .  | 17        |
| 2.4.1    | I Reference Point e la scelta dei cluster . . . . .             | 17        |
| 2.4.2    | La costruzione della matrice delle rilevazioni . . . . .        | 19        |

|          |  |           |
|----------|--|-----------|
| 2.4.3    | Il salvataggio su database . . . . .   | 20        |
| 2.5      | Fase online . . . . .  | 20        |
| 2.5.1    | Il confronto e la risposta di localizzazione . . . . .   | 21        |
| 2.6      | Algoritmi e implementazioni del Wi-Fi fingerprinting . . . . .   | 21        |
| 2.6.1    | Algoritmi di basati su Sub-regioni e Information Gain Theory . . . . .                                     | 21        |
| 2.6.2    | L'uso del Compressive Sensing per ridurre la dimensione della collezione di dati del fingerprint . . . . . | 25        |
| 2.7      | I rischi dell'effetto multipath . . . . .  | 26        |
| 2.7.1    | Fingerprinting . . . . .   | 26        |
| 2.7.2    | Position Finding . . . . .   | 27        |
| 2.8      | Punti di forza e limiti . . . . .  | 28        |
| <b>3</b> | <b>Indoor Localizer</b> . . . . .  | <b>31</b> |
| 3.1      | Introduzione . . . . .   | 31        |
| 3.2      | Progettazione e scelte implementative . . . . .  | 32        |
| 3.2.1    | Il database . . . . .  | 32        |
| 3.2.2    | Funzionalità . . . . .   | 34        |
| <b>4</b> | <b>Analisi dei risultati e sviluppi futuri</b> . . . . .   | <b>43</b> |
| 4.1      | Risultati sperimentali . . . . .   | 43        |
| 4.2      | Deployment del sistema . . . . .   | 44        |
| 4.3      | Valutazione delle performance . . . . .  | 47        |
| 4.3.1    | I parametri fondamentali . . . . .   | 47        |
| 4.4      | Critiche al sistema e possibili sviluppi . . . . .   | 48        |
| <b>5</b> | <b>Conclusioni</b> . . . . .   | <b>51</b> |
| 5.1      | Conclusioni . . . . .  | 51        |

# Elenco delle figure

|     |   |    |
|-----|---|----|
| 1.1 | Confronto fra le tecniche di localizzazione e la loro accuratezza . . . . . | 2  |
| 1.2 | Come funziona un sistema BAT . . . . .                                      | 6  |
| 1.3 | Un tag RFID passivo . . . . .   | 9  |
| 1.4 | Un circuito RFID attivo . . . . .   | 10 |
| 1.5 | Localizzazione mediante TOA . . . . .                                       | 12 |
| 1.6 | Localizzazione mediante TDOA . . . . .                                      | 13 |
| 1.7 | Localizzazione mediante AOA . . . . .                                       | 13 |
| 2.1 | Principio di funzionamento della localizzazione tramite fingerprint . . .   | 16 |
| 2.2 | Esempio di divisione di un edificio in RP . . . . .                         | 18 |
| 2.3 | Il modello ER del database . . . . .  | 20 |
| 2.4 | Approccio tradizionale con il metodo di RSS fingerprint . . . . .           | 22 |
| 2.5 | Il processo costruito mediante IGT e KNN . . . . .                          | 25 |
| 2.6 | I risultati della ricostruzione della mappa mediante l'uso del CS . . . . . | 26 |
| 2.7 | Colori delle intensità di segnale . . . . .                                 | 27 |
| 3.1 | Risultato della scansione . . . . .   | 34 |
| 3.2 | Elenco delle mappe salvate . . . . .  | 37 |
| 3.3 | Risposta dell'applicazione alla localizzazione . . . . .                    | 40 |
| 4.1 | La mappa dell'abitazione usata per il test . . . . .                        | 44 |
| 4.2 | Segnali degli AP rilevati in un RP . . . . .                                | 45 |
| 4.3 | Test di localizzazione . . . . .  | 46 |

# Elenco delle tabelle

|     |   |    |
|-----|---|----|
| 1.1 | Comparazione fra tecnologia RFID attiva e passiva . . . . . | 8  |
| 1.2 | Tecnologie e tecniche a confronto . . . . .                 | 14 |
| 4.1 | Risultati degli algoritmi di clustering e ranking . . . . . | 45 |

# Sommario

Lo studio effettuato raccoglie informazioni al fine di svolgere un'analisi globale delle tecniche attualmente in uso, o in fase di studio, nel campo della localizzazione di dispositivi all'interno di un ambiente chiuso, ovvero laddove non è possibile sfruttare la copertura del sistema GPS. La panoramica è frutto dell'analisi e dello studio di paper tecnici pubblicati dai collaboratori dell'IEEE, fruibili all'interno del portale IEEE Xplore. A corredo di questo studio è stata sviluppata una applicazione per dispositivi Android basata sulla tecnica del Wi-Fi fingerprint; l'applicazione, che rappresenta un primo approccio alle tecniche di localizzazione, è a tutti gli effetti un sistema stand-alone per la localizzazione, consente cioè di costruire sia la mappa per la localizzazione, partendo da un ambiente sconosciuto, sia di ottenere la posizione dell'utente all'interno di una mappa conosciuta. La tesi si conclude con una analisi dei risultati e delle performance ottenute dall'applicazione in un uso comune, al fine di poter valutare l'efficacia della tecnica presa in considerazione. I possibili sviluppi futuri sono analizzati in un capitolo a parte e trovano spazio in ambienti nei quali si vogliono offrire servizi "context-based", ovvero basati sulla posizione esatta dell'utente.



# Introduzione

L'esigenza di conoscere la propria posizione, o la posizione altrui, al fine di potersi meglio muovere all'interno dell'ambiente nel quale viviamo è una necessità che si sta prepotentemente affermando fra gli utenti digitali. Molte delle applicazioni che quotidianamente usiamo su smartphone, volontariamente o meno, sfruttano la nostra posizione per fornire servizi personalizzati volti a migliorare l'esperienza d'uso del dispositivo.

All'interno della stragrande maggioranza dei dispositivi è presente un ricevitore GPS che consente di localizzare facilmente, mediante segnale satellitare, un terminale in uno spazio aperto; ma è proprio quando questa possibilità viene meno anche buona parte dei servizi cui siamo abituati diviene inutilizzabile.

La seguente tesi illustra come si sta cercando di ovviare a questa problematica al fine di fornire una corretta localizzazione anche all'interno di spazi chiusi come edifici pubblici o ambienti domestici.

**Il primo capitolo** tratta lo stato dell'arte, offre dunque una panoramica delle tecnologie utilizzate, divise per importanza e per tipologia;

**Il secondo capitolo** tratta nello specifico la tecnica del Wi-Fi fingerprinting analizzandone gli sviluppi dal punto di vista degli algoritmi che

ne hanno portato all'affermazione come tecnica più utilizzata per la navigazione indoor;

**Il terzo capitolo** presenta la soluzione sviluppata, per terminali Android, come esempio d'uso concreto delle tecniche di localizzazione mediante Wi-Fi fingerprint.

**Il quarto capitolo** tratta la valutazione sperimentale e l'analisi dei risultati rilevati dall'uso dell'applicazione in un contesto d'uso quotidiano.

# Capitolo 1

## Tecniche e tecnologie location-based

### 1.1 Il mondo indoor è ben diverso da fuori

Il Global positioning system (GPS) è un sistema di navigazione globale basato su satelliti; tale tecnologia consente di ottenere un'informazione precisa circa la posizione e l'orario al quale un determinato ricevitore effettua la richiesta di localizzazione. Questo è possibile grazie a una rete satellitare in funzione a ogni ora del giorno e con ogni condizione climatica; la localizzazione pertanto è attendibile a patto che il ricevitore riesca a captare il segnale da almeno quattro satelliti distinti.

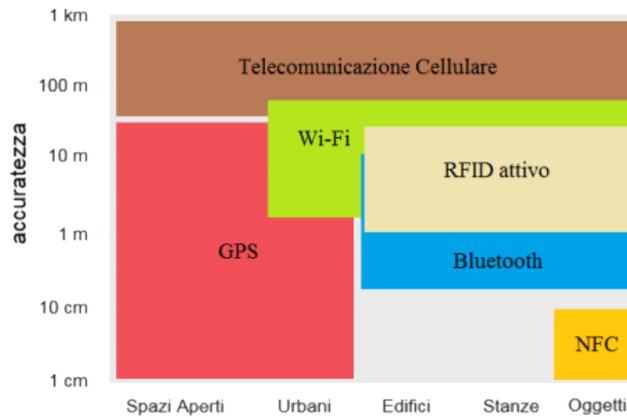
In un ambiente complesso, costituito da muri, aree chiuse, come può essere un palazzo o un ospedale, il segnale GPS risente di forti attenuazioni. Per questo motivo la localizzazione mediante segnale satellitare risulta inefficace.

Questo non vuole però dire che è impossibile ottenere la posizione di un dispositivo all'interno di un'area indoor. Il mondo della navigazione indoor si basa sul concetto di sfruttare l'ambiente e le tecnologie presenti, o comunque di facile reperibilità, per offrire all'utente la possibilità di orientarsi anche in ambienti interni. In molti casi pratici l'ecosistema utile alla localizzazione viene creato sfruttando combinazioni delle tecnologie che a breve verranno illustrate, al fine di ottenere il massimo risultato

dai benefici derivanti dall'uso di una tecnica, minimizzandone i limiti; questo approccio è favorito dalla larga diffusione di device che dispongono nativamente di buona parte di queste tecnologie.

È importante ricordare che il risultato ottenuto da queste tecniche non fornisce un dato di posizione "assoluto", come normalmente restituito da un'interrogazione GPS, bensì viene fornita un'informazione di posizione "relativa" che deve poi essere interpretata correttamente per far comprendere all'utente dove è localizzato.

Le tecniche attualmente più utilizzate sfruttano tecnologie come: infrarossi (IR), Bluetooth, identificazione a radio frequenza (RFID), ultrasuoni, tecniche di riconoscimento mediante tracciamento ottico e tecniche basate sui segnali wireless (RSS techniques).



**Figura 1.1:** Confronto fra le tecniche di localizzazione e la loro accuratezza

## 1.2 Valutazione delle performance

L'accuratezza ottenuta dalla localizzazione non è l'unico parametro da monitorare quando si vogliono valutare le tecniche di localizzazione indoor, è importante tener conto anche di un'insieme di parametri, **bibid** come:

- **Accuratezza** (accuracy o location error): naturalmente è uno dei parametri di maggior interesse quando si parla di sistemi di posizionamento. Solitamente l'accuratezza viene valutata come la distanza Euclidea tra la posizione stimata dal sistema e quella reale nella quale viene effettuata la richiesta. Spesso però l'accuratezza del sistema è un'arma a doppio taglio, difatti per ottenere un buon risultato vengono influenzate negativamente le altre caratteristiche.
- **Precisione** (precision): mentre l'accuratezza valuta solo l'errore sulla differenza fra posizione reale e risultato ottenuto, la precisione considera quanto il sistema lavora bene nel tempo e nel contesto nel quale è inserito.
- **Complessità** (complexity): la complessità è un fattore che dipende da hardware, software e operazioni eseguite su essi. Per quanto riguarda la complessità del software l'analisi si deve concentrare sul costo computazionale, sui tempi di risposta dovuti al calcolo, sull'infrastruttura di server e client. Per quanto riguarda l'hardware è importante prendere in considerazione caratteristiche come la durata, la qualità dei componenti ecc...
- **Robustezza** (robustness): si intende la tolleranza del sistema ai guasti, ovvero la capacità del sistema di operare correttamente anche in assenza di uno dei segnali abitualmente usati per funzionare.
- **Scalabilità** (scalability): la possibilità di adattare il sistema in un nuovo ambiente, solitamente più grande, senza intaccare le performance.
- **Costo** (cost): i costi non sono solo quelli derivanti dall'acquisto e dalla messa in opera del sistema; quando si parla di costi è necessario

prendere in considerazione anche lo spazio, il tempo, la manutenzione, il peso del sistema ecc...

### 1.3 Tecniche di localizzazione senza fili

Di seguito saranno analizzate le tecnologie più diffuse al fine di comprenderne il funzionamento basilare ed i limiti connessi, così da tracciare una valutazione dello stato dell'arte.

#### 1.3.1 Tecniche basate su Infrarossi (IR)

Il principio legato alla tecnica di posizionamento ad infrarossi è sintetizzabile con l'immagine del labirinto, utile a comprendere sia il funzionamento che i limiti di questo sistema. Il dispositivo "ricevitore" deve intercettare il raggio emesso da un apparecchio che funge da "emettitore" cui è associato un ID univoco e la cui posizione è fissa all'interno dell'ambiente. A seconda del segnale ricevuto il dispositivo riuscirà a risalire all'ID dell'emettitore che lo ha mandato e saprà dunque in quale area si trova.

**Limiti della tecnologia** Il **corto raggio** di trasmissione e la necessità di mantenere un **contatto visivo** fra emettitore e ricevitore al fine di ottenere informazioni sulla posizione costituiscono le maggiori critiche fatte a questa tecnica; tali limiti sono dettati dalla tecnologia stessa che non permette la trasmissione attraverso pareti, o laddove il collegamento emettitore/ricevitore può essere ostacolato, da qui l'immagine del labirinto che evidenzia proprio come in un ambiente complesso ci sia bisogno di un grande numero di emettitori strategicamente posizionati per permettere ai ricevitori di localizzarsi.

### 1.3.2 Tecniche basate su Bluetooth

La tecnologia Bluetooth è usata soprattutto in contesti di posizionamento su piccola scala, come una stanza o una piccola abitazione. Il maggior vantaggio derivante dall'uso di questa tecnologia è la reperibilità dell'infrastruttura, l'hardware Bluetooth è comunemente integrato in dispositivi come PDA, PC e smartphone, il che consente di popolare facilmente l'ambiente con terminali che la sfruttano.

Il sistema è composto da un access point Bluetooth, ovvero, un dispositivo che viene identificato come nodo centrale del *piconet*, che funge da riferimento per la localizzazione degli altri terminali presenti in una determinata area. A seconda dell'intensità del segnale ricevuto da questo dispositivo è possibile effettuare una stima della posizione, con una distanza relativa dal nodo centrale.

**Limiti della tecnologia** Il limite di questa tecnologia, se non si prende in considerazione il costo dell'infrastruttura, è dato principalmente dalle interferenze che si generano fra i ricevitori, le quali non consentono di raggiungere un'elevata concentrazione di dispositivi in una piccola area siccome i dispositivi che sfruttano la tecnologia Bluetooth operano anche da emettitori.

### 1.3.3 Tecniche basate su ultrasuoni

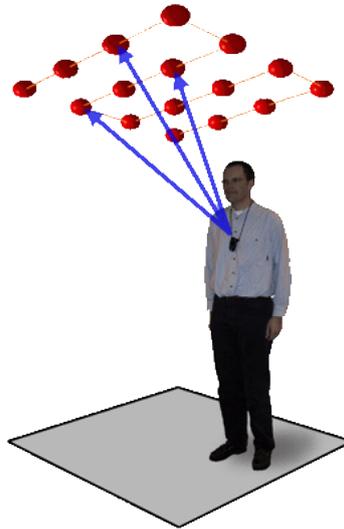
Il sistema a segnali ultrasonici è usato dai pipistrelli per spostarsi di notte e ha ispirato, negli ultimi anni, molti sistemi di localizzazione. Il sistema Active Bat utilizza segnali ad ultrasuoni e tramite la triangolazione di un trasmettitore trasportato dall'obiettivo determina la posizione della persona. Il dispositivo periodicamente emana un impulso ad ultrasuoni. Questo impulso è ricevuto da una matrice di celle montate in posizioni note. Le tecniche utilizzate per ricavare la posizione sono quelle di TOA e triangolazione delle onde ricevute dai vari sensori della matrice. In questo modo è possibile, in base alla complessità della matrice, determinare

anche la posizione 3D della persona. Il dispositivo richiesto è economico e trasportabile, ma specifico per l'utilizzo.

Inoltre il consumo di batteria è limitato e la manutenzione scarsa. Un sistema basato su Active Bat è composto da più controllori centrali, i quali possono monitorare

fino a 3 trasmettitori contemporaneamente con un'accuratezza di 3 cm.

Le performance di tale sistema sono però influenzate dalla presenza di ostacoli che riflettono le onde sonore. Inoltre è importante la posizione della matrice di sensori, che risulta essere complessa e costosa da installare.



**Figura 1.2:** Come funziona un sistema BAT

Altro sistema basato su ultrasuoni è Cricket. Questo sistema è incentrato sull'offrire privacy e sicurezza all'utente. Questo modello è studiato mediante il sistema inverso a quello visto con gli Active Bat. I trasmettitori di segnale ultrasonico sono infatti posizionati in locazioni prestabilite.

I ricevitori invece vengono apposti sull'obiettivo da localizzare. I trasmettitori emettono, oltre alle altre informazioni, messaggi di sincronizzazione in modo da poter operare tramite TOA. La privacy del sistema è garantita dai calcoli per la triangolazione che vengono effettuati all'interno del ricevitore in dotazione all'utente. Non vengono quindi trasmessi, a

meno di specifiche condizioni, i dati relativi alla posizione dal ricevitore al sistema centrale.

Cricket è inoltre un sistema misto poiché il collegamento RF che garantisce il dialogo tra il ricevitore e la centrale può essere utilizzato per ricavare la posizione in caso di mancanza di segnale audio. La precisione del sistema è attorno ai 10cm. Lo svantaggio più grande è dovuto all'impiego massiccio di energia per alimentare il sistema.

#### 1.3.4 Tecniche basate su RFID

Nonostante nascano per l'identificazione di oggetti statici, i sistemi ad identificazione tramite frequenze radio possono essere usati per determinare la posizione di un oggetto in movimento.

Per la localizzazione tramite RFID possono essere distinti due diversi approcci:

- **Posizionamento basato su una singola cella:** i tag RFID sono installati come punti di riferimento attivi, ovvero vengono posizionati in luoghi specifici la cui posizione risulta nota. In questo caso la localizzazione viene effettuata prendendo come riferimento il singolo tag più vicino;
- **Posizionamento mediante multilaterazione:** il dispositivo che riceve il segnale deve essere raggiunto da almeno tre celle distinte, dopodiché l'analisi viene fatta sulla potenza del segnale ricevuto (RSS) mediante un algoritmo di triangolazione.

**Limiti della tecnologia** Perché restituisca un risultato attendibile, l'RFID non necessita del contatto visivo fra emettitore e ricevitore, superando così i limiti imposti da muri e pareti.

Restano presenti problematiche dovute ai disturbi del segnale che potrebbero dare origine ad effetti di **multipath propagation**. A seconda dell'approccio scelto per la localizzazione, la tecnologia porta vantaggi e svantaggi differenti. In termini di precisione, con l'approccio basato su di

|                      | <b>RFID Attivo</b>  | <b>RFID Passivo</b>   |
|----------------------|---|---|
| <b>Alimentazione</b> | Interna al tag  | Il circuito è attivato dal dispositivo utilizzato per leggere il tag  |
| <b>Dimensioni</b>    | Abbastanza grande da contenere una batteria e sostenere un'antenna  | Piccolo come un microchip e sottile come un foglio di carta   |
| <b>Vita media</b>    | Da 5 ai 10 anni, dipende dalla batteria   | Potenzialmente infinita, dipende dall'ambiente  |
| <b>Portata</b>       | Fino a 100 metri  | Fra i 3 e i 5 metri, solitamente molto meno, sfrutta il contatto  |
| <b>Costi</b>         | Attorno a 18€ per tag   | Mediamente 20 centesimi per tag   |
| <b>Vantaggi</b>      | Può essere letto da alte distanze<br>Ha la possibilità di trasmettere un maggior numero di dati<br>Può iniziare la comunicazione<br>Il tag deve essere sostituito al termine della batteria | Non ha limiti di tempo<br>Costo ridotto<br>Piccole dimensioni<br>Resistente agli urti e alle sollecitazioni esterne                   |
| <b>Svantaggi</b>     | Dispositivi abbastanza costosi;<br>Non funzionano senza alimentazione;<br>Il dispositivo è abbastanza grande non risultando ottimale per piccole implementazioni                            | La comunicazione risente molto della qualità del tag;<br>Raggio di copertura ridotto;<br>Difficoltà ad essere letto qualora oscurato. |

**Tabella 1.1:** Comparazione fra tecnologia RFID attiva e passiva

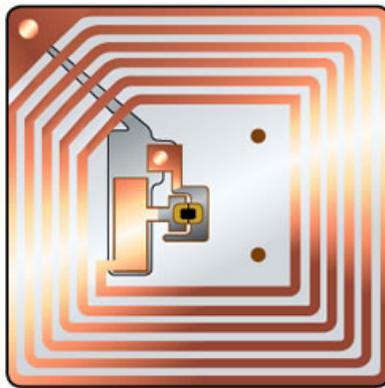
una singola cella è impossibile ottenere una posizione esatta del ricevitore, si potrà solo stimare la presenza o meno all'interno dell'area dell'antenna e di conseguenza, in base alla potenza del segnale, ottenere una stima poco significativa della posizione stessa; se si sfrutta la trilaterazione è possibile ottenere risultati più o meno precisi, in base alla dimensione della cella coperta e dell'affidabilità dell'algoritmo utilizzato.

### **RFID in azione**

La tecnologia RFID, o di **Radio Frequency IDentification** nasce in ambito militare con lo scopo di localizzare oggetti, autoveicoli ed altre attrezzature all'interno di spazi ridotti; col tempo la tecnologia ha trovato spazio anche in altri campi di riconoscimento ravvicinato,

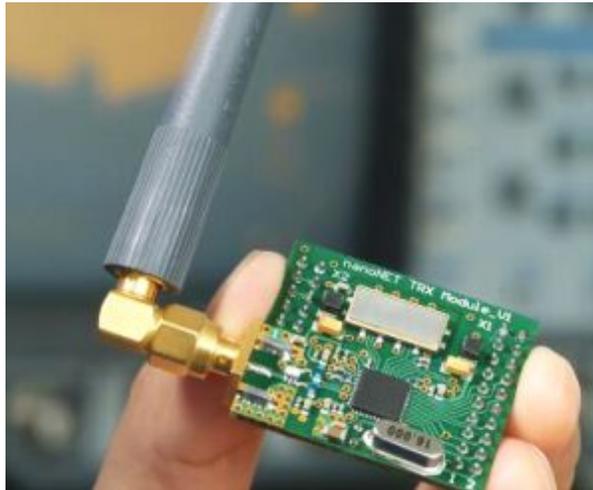
concretizzandosi nella *Near Field Communication (NFC)* per pagamenti ed altre funzionalità di identificazione a corto raggio. Il rilevamento mediante RFID può seguire due diversi approcci: **attivo o passivo**.

**NFC Tag: un caso specifico d'utilizzo** La **comunicazione di prossimità (NFC)**, sfrutta dei dispositivi chiamati tag. Nel caso di rilevamento mediante dispositivi passivi l'ecosistema è composto da tag, ovvero dei micro-circuiti ai quali è assegnato un identificativo e da un dispositivo, uno smartphone ad esempio, in grado di interagire con il tag e di associare ad esso delle informazioni. Verosimilmente il funzionamento è analogo a quello del barcode: quando il "lettore" rileverà la presenza di un determinato tag potrà far partire le operazioni ad esso associato interagendo con l'ambiente circostante. Il limite della tecnologia passiva risiede nel corto raggio d'azione del tag, che non godendo di alimentazione, può essere letto dal dispositivo solo ad una ravvicinata distanza.



**Figura 1.3:** Un esempio di tag RFID passivo

Molto più precisi e potenti sono invece i tag attivi, provvisti di una fonte di alimentazione esterna, come una piccola pila, questi, consentono di memorizzare molte più informazioni trasmissibili tramite una antenna integrata, eliminando di fatto il limite della comunicazione a corto raggio.



**Figura 1.4:** Un circuito RFID attivo

### 1.3.5 Localizzazione mediante tracciamento ottico

La tecnica, principalmente utilizzata nei sistemi di sorveglianza, necessita del contatto visivo fra il soggetto da localizzare e il sistema di localizzazione. Per l'implementazione sono installati sistemi di videocamere per effettuare rilevazioni e localizzazioni in real-time. La rilevazione sfrutta i concetti precedentemente analizzati per la localizzazione mediante telecamere ad infrarossi combinate a sistemi di diodi led o dispositivi foto riflettitori montati sull'oggetto da localizzare. La precisione derivante dall'uso di questo sistema è molto alta, seppure a discapito dei costi dell'infrastruttura che rappresentano il vero limite derivante dalla scelta di questa tecnologia. La possibilità di utilizzare un sistema di questo tipo trova la sua realizzazione in contesti statici dove la localizzazione real-time costituisce un fattore chiave, si pensi ad esempio ad un sistema di sicurezza per musei, dove ogni opera è posizionata in una determinata posizione dalla quale non deve essere spostata; il sistema in questione, tramite la rilettura della posizione dell'oggetto contraddistinto dal "tag" potrà rapidamente identificarne la posizione consentendo un rapida messa in funzione delle operazioni di sicurezza.

## 1.4 Tecniche di localizzazione basate sul raggio d'azione

### 1.4.1 Received Signal Strength Indication (RSSI)

L'algoritmo è basato sulla stima della potenza del segnale trasmesso, calcolata partendo dall'intensità percepita dall'antenna che riceve il segnale, al netto della perdita di intensità dovuta alla distanza dell'emettitore (rilevata mediante una stima che sfrutta l'attenuazione causata dagli effetti di propagazione). Il più grande svantaggio derivante dall'uso di questa tecnica è l'elevata interferenza causata dall'ambiente, basti pensare alle piccole modifiche che possono essere fatte alla posizione di oggetti voluminosi, porte o altri elementi all'interno della stanza che possono interferire sulla qualità del segnale ricevuto. La rilevazione risente pertanto in maniera consistente di effetti di shadowing e multipath facendo sì che, a meno di particolari ottimizzazioni, la precisione di base ottenuta dal sistema sia molto scarsa.

### 1.4.2 Time Of Arrival (TOA)

In base alla velocità e al tempo di propagazione del segnale è possibile determinare la posizione del ricevitore. La localizzazione è caratterizzata da due approcci, il più semplice di questi è il **posizionamento mediante la stima del Tempo di arrivo (TOA)** e il **Time Difference of Arrival (TDOA)**, ovvero localizzazione mediante la differenza fra i tempi di arrivo. Il TOA sfrutta semplicemente il tempo assoluto di trasmissione di un segnale da un singolo emettitore ad un ricevitore remoto ad una velocità nota per misurare la distanza del dispositivo dall'emettitore. In questo caso, una volta ottenuta la stima della distanza, si ha un dato posizionabile all'interno di un cerchio che ha per centro l'emettitore; naturalmente, per far sì che la localizzazione sia precisa occorre effettuare il calcolo della distanza da almeno due trasmettitori differenti, così da poter trovare la posizione corretta in corrispondenza dell'intersezione dei due segmenti che rappresentano la distanza.

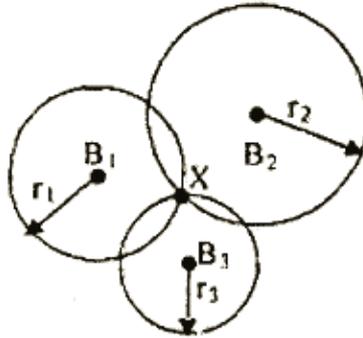


Figura 1.5: Localizzazione mediante TOA

### 1.4.3 Time Difference Of Arrival (*TDOA*)

Il TDOA è alla base delle tecniche di localizzazione che prendono il nome di multilaterazione (*Multilateration*). La tecnica è basata sulla differenza del tempo di arrivo di un segnale emesso da un'insieme di emettitori ad un target; come per il TOA la distanza da un determinato emettitore è calcolata partendo dalla velocità a cui viaggia il segnale in aria e dal tempo impiegato perché esso arrivi al ricevitore.

Definisco come riferimento il primo sensore che riceve il segnale dal target e chiamo A il sensore che riceve il medesimo. Le coordinate della sorgente vengono calcolate tramite l'intersezione di iperboli che sono descritte dalla differenza della distanza tra A e il target, rispetto alla distanza tra il riferimento e il target. Un'iperbole è infatti definita come il luogo geometrico dei punti del piano aventi come costante la differenza delle distanze con i fuochi che, nel nostro caso, sono il sensore di riferimento e il sensore A. Vi è quindi la presenza di  $n-1$  iperboli, con  $n$  il numero di sensori presenti nel sistema, ognuno posizionato nel fuoco dell'iperbole.

Per ottenere una misura univoca in 2D è necessario posizionare in modo strategico almeno tre sensori, con l'aggiunta di un dispositivo di sincronizzazione che allinea i sensori. In questo caso la tecnica si può definire **trilaterazione**. Per ottenere una misura in 3D sono invece necessari almeno 4 sensori più un dispositivo di sincronizzazione. Aumentando il numero di sensori si incrementa la precisione della localizzazione.

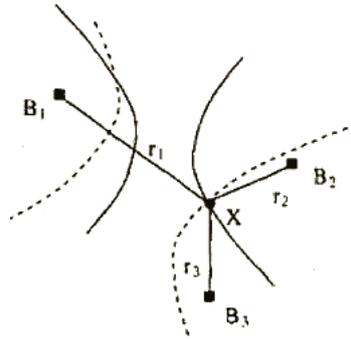


Figura 1.6: Localizzazione mediante TDOA

#### 1.4.4 Angle Of Arrival (AOA)

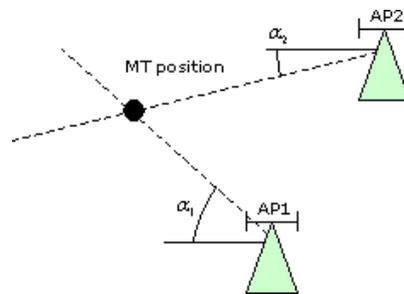


Figura 1.7: Localizzazione mediante AOA

Con l'Angle of Arrival la posizione del dispositivo che effettua la richiesta di localizzazione può essere trovata dall'intersezione di alcune coppie di angoli formati fra i trasmettitori presenti nel raggio della circonferenza che ha per centro il ricevitore.

Il vantaggio dell'AOA risiede nella possibilità di ottenere un risultato attendibile con due sole coppie di angoli, per una risposta in due dimensioni e tre per una in 3D, senza la necessità di informazioni riguardanti i tempi di trasmissione, sfruttando cioè le tecniche di multilaterazione, senza dover considerare i ritardi del segnale.

A fronte del risparmio dal punto di vista computazionale, la tecnica presenta alcuni svantaggi pratici, dovuti al costo dell'hardware, che al fine di restituire informazioni precise, deve essere di qualità; rischiando altrimenti di incorrere in fenomeni di shadowing o multipath che comprometterebbero la misurazione.

| <b>Tecnica di posizionamento wireless</b> | <b>Algoritmo di posizionamento</b> | <b>Precisione ottenibile</b>               |
|---|------------------------------------|--|
| RFID                                      | RSSI                               | Da pochi centimetri ad una decina di metri |
| Bluetooth                                 | RSSI                               | Da pochi centimetri ad una decina di metri |
| Infrarossi                                | AOA                                | Qualche metro                              |
| Tracciamento Ottico                       | Image processing                   | Fino a qualche millimetro                  |

**Tabella 1.2:** Tecnologie e tecniche a confronto

## Capitolo 2

# Il Wi-Fi fingerprinting

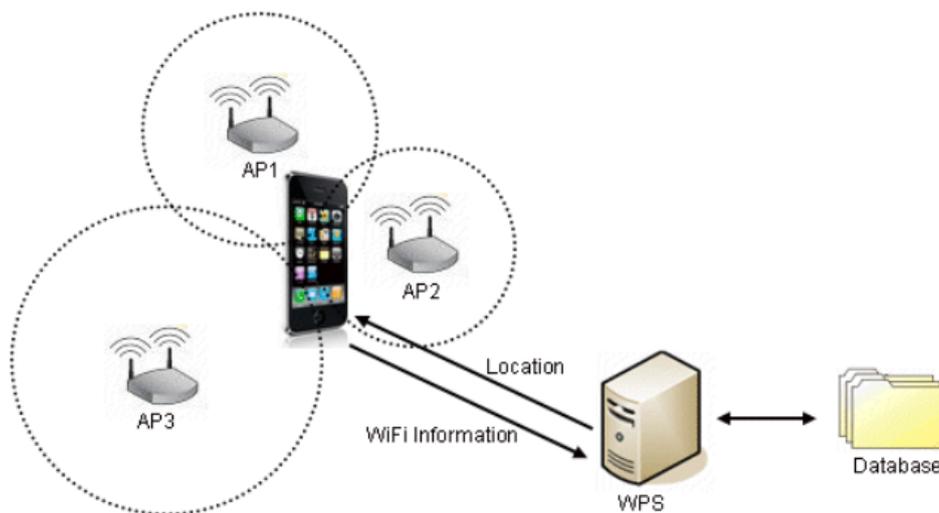
### 2.1 Introduzione

Una delle tecniche che ha maggiormente concentrato gli studi per la navigazione indoor è il **Wi-Fi fingerprinting**, appartenente alle tecniche di localizzazione basate su *Received Signal Strength Indication (RSSI)*; la tecnica sfrutta gli access point presenti all'interno dell'ambiente al fine di restituire la posizione di un dispositivo dotato di antenna. A seconda degli algoritmi scelti si possono ottenere diversi gradi di precisione.

Di seguito sono illustrati gli aspetti maggiormente interessanti derivanti dall'uso del Wi-Fi fingerprint, al fine di comprendere come è possibile implementare la tecnica all'interno di un ambiente indoor, valutando i possibili accorgimenti da adottare per ottenere un risultato preciso e attendibile.

### 2.2 L'infrastruttura

Lo sviluppo e l'attenzione dati all'uso di questa tecnica risiedono nel basso costo di implementazione, siccome alla base è possibile sfruttare il fingerprinting senza un sistema hardware specializzato, facilmente reperibile anche in ambiente domestico. La localizzazione è elaborata unicamente dal software che si occupa di analizzare ed elaborare i dati provenienti dai diversi Access Point (AP). Utilizzando questa tecnica, all'utente non è



**Figura 2.1:** Principio di funzionamento della localizzazione tramite fingerprint

richiesta la connessione ad alcuna rete, a patto che il database contenente le informazioni per la localizzazione sia già stato scaricato in precedenza sul dispositivo che effettua la richiesta. L'unica infrastruttura di supporto diventa pertanto la rete WLAN. L'usabilità, e di conseguenza la precisione del sistema, dipendono dalla densità degli access point e dall'architettura dell'edificio da mappare.

### 2.3 La tecnica

Il nome della tecnica riassume esaurientemente il principio sulla quale si basa cioè il rilevamento dei segnali Wi-Fi presenti in una determinata area al fine di ottenere "un'impronta digitale" univoca dell'ambiente; una volta raccolta, questa impronta, può essere resa disponibile ad uno o più client per consentire loro la localizzazione; la posizione verrà calcolata mediante un algoritmo di cluster-matching, ovvero trovando fra tutte le rilevazioni salvate in precedenza quale meglio si approssima alla rilevazione effettuata a run-time dal terminale dell'utente.

Fin da subito occorre specificare che con questa tecnica non è possibile ottenere informazioni topografiche, ovvero delle coordinate [°LAT,°LNG], ma semplicemente il fingerprint della stanza che più si avvicina a quello

rilevato dal client dotato di antenna Wi-Fi, ovvero una stima relativa e non assoluta.

La localizzazione è costituita da due fasi: **offline**, o di apprendimento, nella quale l'ambiente viene suddiviso in **Reference Points (RP)** per i quali vengono effettuate una o più scansioni degli **access point (AP)** circostanti volte a costruire la mappa dell'ambiente, tale scansione viene poi elaborata e salvata in database per garantirne la persistenza; la seconda fase è detta **online** e riguarda il client che effettua la richiesta di localizzazione, viene effettuata una scansione degli AP che costituisce una sorta di "*Reference Point dinamico*" da confrontare con tutti quelli presenti in database. Al termine della fase online l'utente ottiene un'indicazione della sua posizione corrispondente al RP più simile a quello da lui rilevato.

## 2.4 Fase offline

### 2.4.1 I Reference Point e la scelta dei cluster

Un'area di grandi dimensioni, come può essere un'abitazione, è a sua volta suddivisa in sotto-aree funzionali, come una cucina o un salotto, questa prima divisione consente ad una persona di distinguere esattamente dove si trova all'interno dell'edificio. Al fine di consentire la localizzazione, i sistemi che sfruttano il Wi-Fi fingerprint suddividono a loro volta le stanze in Reference Point, ai quali naturalmente può essere attribuito un riferimento alla stanza di appartenenza. Ad ogni RP è connessa l'informazione relativa agli Access Point rilevati, identificati dal MAC address e dalla loro intensità.

Al fine di ottenere una buona mappatura, l'intero edificio dovrebbe essere coperto da RP correttamente dimensionati, ovvero di aree nelle quali non sono presenti variazioni di intensità di segnale rilevanti; per assurdo, sarebbe inutile cercare di mappare un'intera stanza di qualche decina di metri quadri con un solo punto poiché il segnale ricevuto dai vari router in posizioni differenti, seppur interne alla stanza, sarebbe troppo soggetto ad oscillazioni impedendo di ottenere una buona approssimazione. Per lo

stesso motivo, nello scegliere la suddivisione occorre prestare attenzione alla morfologia della stanza stessa.

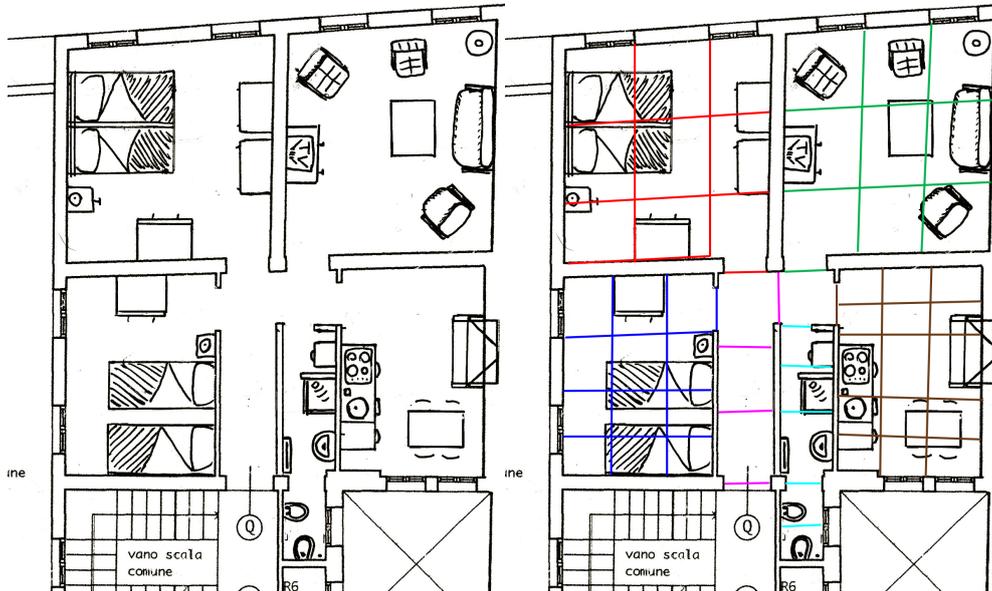


Figura 2.2: Esempio di divisione di un edificio in RP

All'atto pratico la rilevazione viene effettuata ponendo il ricevitore all'interno della stanza, nel Reference Point di competenza; lì vengono effettuate un numero fissato di scansioni ad intervalli regolari. È importante misurare l'RSS proveniente dai vari router lasciando trascorrere un lasso di tempo fra una scansione e l'altra al fine di comprendere come le variazioni usuali del segnale operano in quella determinata posizione; siccome la rilevazione online avverrà in un momento successivo alla raccolta dei dati per la costruzione della mappa il segnale del trasmettitore potrebbe non trasmettere sempre un segnale ad intensità costante, o comunque soggetto ad attenuazioni provenienti da interferenze ambientali che forniscono un risultato differente da quello rilevato nella fase offline.

Al termine di queste scansioni i dati vengono accorpati al fine di ottenere un singolo valore riassuntivo per ogni AP rilevato.

### 2.4.2 La costruzione della matrice delle rilevazioni

L'approccio più semplice per tenere traccia dei dati rilevati dalle scansioni è quello di costruire una cosiddetta **matrice delle rilevazioni** che consiste nel raccogliere i dati rilevati dall'antenna e memorizzarli temporaneamente in una matrice che tenga traccia dell'Access Point cui appartiene il segnale e del numero dell'iterazione al quale viene effettuata la rilevazione.

Siccome questa matrice potrebbe raggiungere grandi dimensioni, soprattutto se si pensa che ne viene generata una distinta per ogni Reference Point, è possibile scegliere di comprimerla in una matrice unaria, ovvero un vettore, che contenga le informazioni aggregate per un determinato Reference Point.

La compressione deve essere pertanto fatta cercando di non perdere le informazioni caratteristiche di tutti i segnali rilevati. L'approccio più semplice consiste nel calcolare la media pesata degli RSS dei vari Access Point.

$$\begin{bmatrix} AP_{0,1} & \cdots & AP_{0,j} \\ \vdots & \ddots & \vdots \\ AP_{i,j} & \cdots & AP_{i,j} \end{bmatrix} \implies \begin{bmatrix} AP_0 \\ \vdots \\ AP_i \end{bmatrix} \quad (2.1)$$

**Interpolazione e previsione dei dati, un'ottimizzazione necessaria** al fine di non prolungare eccessivamente la fase di raccolta dei dati.

Qualora l'ambiente da mappare fosse di grandi dimensioni e volutamente più stabile e meno soggetto a cambiamenti (ne è un esempio un ambiente costituito ad hoc per il Wi-Fi fingerprinting) è possibile adottare un approccio probabilistico di stima dei dati, ne è un esempio la tecnica del **kriging**.

### 2.4.3 Il salvataggio su database

La fase offline termina la sua esecuzione salvando i dati dei vari Reference Point all'interno del database che dovrà essere condiviso fra chi effettua le rilevazioni e i vari client che richiedono la localizzazione. Come già detto, occorre mantenere un database con il minor numero possibile di record, al fine di facilitare la localizzazione, motivo per cui, anche in fase di salvataggio possono essere apportate diverse ottimizzazioni. Una di queste prende il nome di **compressive sensing** e verrà analizzata in seguito.

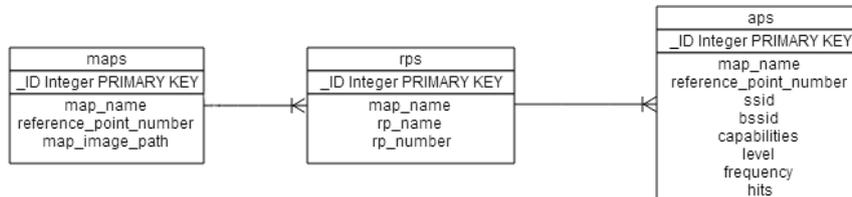


Figura 2.3: Il modello ER del database

## 2.5 Fase online

La fase online, a sua volta, suddivide la localizzazione in due step, per cercare di ridurre la quantità di dati da confrontare: **coarse localization** e **fine localization**. Nella prima fase vengono scremati tutti i punti considerati sufficientemente lontani per tipologia di AP rilevati, dopodiché l'algoritmo procede, in fase di fine localization, con il confronto della rilevazione con i soli RP "simili". A meno di un fattore di tolleranza viene così restituita all'utente l'indicazione di posizione al RP che meglio approssima l'impronta rilevata in quel momento dall'antenna del dispositivo. Naturalmente, se non dovesse essere trovato alcun RP compatibile con la rilevazione fatta occorrerà migliorare la copertura della mappa nella fase offline o aumentare il valore di tolleranza utilizzato per il confronto.

### 2.5.1 Il confronto e la risposta di localizzazione

Una volta ricevuta la richiesta di localizzazione da parte dell'utente, il sistema elabora un fingerprint "live" che deve essere confrontato con tutti quelli presenti in database. L'approccio più semplice consiste nella valutazione della distanza Euclidea fra il livello rilevato e quello memorizzato:

$$|RSS_{live} - RSS_{db}| = \sqrt{(RSS_{live})^2 - (RSS_{db})^2} \quad (2.2)$$

Dopo aver valutato le distanze euclidee fra la rilevazione gli RP, la risposta alla localizzazione si riduce alla risoluzione di un problema di minimo, fra tutte le distanze calcolate.

## 2.6 Algoritmi e implementazioni del Wi-Fi fingerprinting

### 2.6.1 Algoritmi di basati su Sub-regioni e Information Gain Theory

Uno studio interessante, per quanto riguarda gli algoritmi da utilizzare per la localizzazione, è stato condotto dal Centro di Ricerca per la comunicazione dell'Harbin Institute of Technology; la conclusione di questo lavoro, oltre all'analisi degli algoritmi maggiormente utilizzati per la localizzazione, ha dimostrato come l'averne una altissima concentrazione di AP non determini per forza un miglioramento, in termini di accuratezza, della localizzazione. Di seguito verrà analizzato il modello di WLAN indoor positioning utilizzando la famiglia di algoritmi di **pattern matching k-nearest neighbor (KNN)**. Con  $(RP_1, RP_2, RP_3, \dots, RP_n)$  vengono indicati i Reference Points, con  $(TP_1, TP_2, TP_3, \dots, TP_m)$  i test points, ovvero i valori letti dall'antenna in fase di rilevazione.  $RSS_{i,j}$  è l'RSS del  $RP_i$  ricevuta dal  $j$ -esimo AP, mentre  $(\hat{x}, \hat{y})$  è la posizione scelta dall'algoritmo come risultato più attendibile.

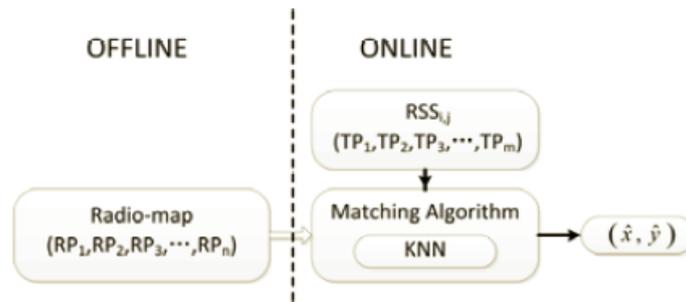


Figura 2.4: Approccio tradizionale con il metodo di RSS fingerprint

### K means clustering

Ogni AP non ha la stessa importanza all'interno di un sistema, motivo per cui, per evitare di immettere nel sistema informazioni non necessarie, anche la scelta di quali segnali considerare deve essere fatta seguendo alcuni criteri di selezione. L'analisi dei vecchi modelli utilizzati ha portato alla luce alcuni evidenti limiti:

- **Il criterio di Fisher**, che permette di selezionare quali AP hanno le migliori discriminanti per identificare una regione. Tuttavia questo metodo si basa su un presupposto irragionevole: che la varianza intra-classe della AP è per natura bassa
- La tecnica del **Joint Clustering** che sceglie come significativi gli AP con RSS più forte. Sfortunatamente questo criterio, considera solo la media della potenza del segnale trasmesso dagli AP, trascurando altri aspetti importanti.
- **Information Gain Theory (IGT)**, che seleziona al meglio gli AP significativi, trascurando del tutto la correlazione che questi hanno fra di loro o nei confronti dell'ambiente nel quale sono collocati, per esempio potrebbe essere scartato un AP che funziona bene unicamente in una sotto area poiché questo non trasmette bene nell'intera regione presa in esame.

Il modello preso in esame ha la necessità di considerare l'influenza che gli AP hanno sull'ecosistema, per cui l'idea è quella di combinare Information

Gain Theory e algoritmi KNN. La fase offline pertanto deve includere non la sola rilevazione, ma anche la valutazione degli AP da memorizzare:

- **Partizione dell'ambiente in cluster:** l'algoritmo utilizzato partiziona l'area in  $K$  sotto-regioni, considerando come discriminante l'RSS percepito da un AP, se l'insieme dei valori di tutti gli AP mantiene nel tempo un valore stabile, allora questo gruppo appartiene alla stessa area:
  - Vengono scelti  $K$  RP come centro delle  $K$  sotto-regioni;
  - Ogni restante RP viene assegnato alla sotto-regione più vicina, confrontando la distanza euclidea del suo **feature vector** con quello del centro della  $K$ -regione;
  - Viene ricalcolato il Reference Point posto a centro della regione come media di tutti i feature vector della sotto-regione appena popolata;
  - Il processo viene ripetuto fino a quando i valori ottenuti come feature vector medio non si discostano, a meno di un valore di tolleranza, dal valore centrale.
- **AP ranking:** l'Information Gain Theory viene applicata a tutte le regioni (cluster) costruite per valutare tutti gli AP, ordinandoli in ordine decrescente all'interno della propria area; il ranking viene effettuato al fine di abbattere il costo computazionale in fase di confronto diminuendo il numero di AP da considerare, escludendo quelli meno significativi o che degradano la precisione del sistema con la loro instabilità.

Nel sistema l'intensità di segnale del RP rilevata per tutti gli access points ( $AP_i, 1 \leq i \leq m$ ), funziona come fingerprint per le sotto-regioni.

La classificazione degli AP è fatta seguendo l'algoritmo:

$$InfoGain(AP_i) = H(C) - H(C/AP_i) \quad (2.3)$$

$$H(C) = \sum_{j=1}^n P(C_j) \log_2 P(C_j) \quad (2.4)$$

$$H(C/AP_i) = \sum_v \sum_{j=1}^n P(C_j, AP = v) \log P(C_j/AP_i = v) \quad (2.5)$$

dove  $H(C)$  è l'entropia del segnale, ovvero la quantità di informazione presente, quando la potenza del segnale è sconosciuta;  $H(C/AP_i)$  misura l'entropia del segnale proveniente dall' $i$ -esimo AP, mentre  $v$  è l'intensità in dBm.

- **Selezione:** i primi 'F' AP di ogni regione sono scelti per identificare una determinata area. Il valore di F è scelto empiricamente a seguito di numerosi esperimenti.

La fase online a sua volta è divisa in due step:

- **Rough Positioning:** l'RSS di tutti gli AP accessibili è utilizzata per restituire una posizione di massima per confinare i Test Point in una determinato gruppo di sotto-regioni possibili;
- **Precise Positioning:** viene applicato un algoritmo di KNN nelle regioni selezionate, per restituire il risultato che meglio identifica i Test Point. Per ogni RP vengono presi gli 'F' AP caratteristici, ai quali viene applicato il seguente algoritmo di KNN:

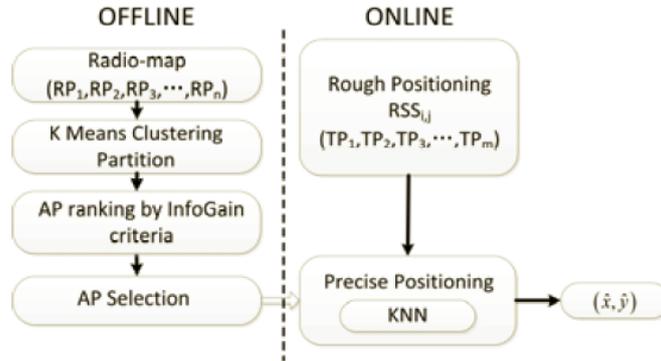
$$d_{jq} = \left( \sum_{i=1}^F |RSS_{ij} - RSS_i|^q \right)^{1/q} \quad j = 1, 2, \dots, n \quad (2.6)$$

$$(\hat{x}, \hat{y}) = \frac{1}{k} \sum_{i=1}^k (x_i, y_i) \quad (2.7)$$

Dove  $d_{jq}$  è la distanza vettoriale fra il TP e il  $j$ -esimo RP, e  $n$  rappresenta il numero di RP nella sotto-regione.  $RSS_{ij}$  è l'intensità di segnale ricevuta nel  $RP_j$  dal  $AP_i$  e  $RSS_i$  è l'intensità ricevuta nel test point dal  $AP_i$ . Il vettore delle distanze è calcolato con la

distanza di Manhattan o con la distanza Euclidea, a seconda che  $q$  sia 1 o 2.  $(\hat{x}, \hat{y})$  è il risultato della localizzazione.

L'intero processo è illustrato e riassunto dalla **Figura 2.5**.



**Figura 2.5:** Il processo costruito mediante IGT e KNN

### 2.6.2 L'uso del Compressive Sensing per ridurre la dimensione della collezione di dati del fingerprint

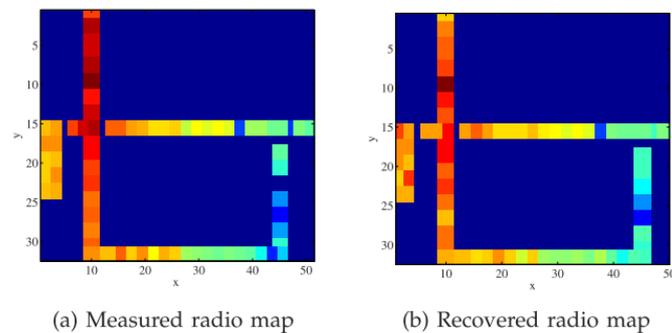
L'obiettivo del *compressive sensing* è quello di ridurre la dimensione dei dati necessari a descrivere il fingerprint di un reference point, riuscendo comunque a ritornare ad un'immagine attendibile senza introdurre errori. Gli esperimenti fatti hanno dimostrato che utilizzando solo il 10% dei dati consente di risalire comunque all'intero sistema con un tasso d'errore inferiore al 16%.

La tecnica si basa sull'interpolazione per ricostruire la *radio map* utile alla localizzazione, un presupposto iniziale è l'indipendenza dell'RSS letto dai vari AP nel tempo.

Il Compressive Sensing aggiunge un nuovo framework per ricostruire segnali sparsi o compressi introducendo un rumore, ovvero un errore, molto basso rispetto agli altri approcci probabilistici, come ad esempio il teorema del campionamento di Nyquist. L'obiettivo è quello di ridurre in maniera significativa il numero di misurazioni; una delle attività più dispendiose in questo genere di sistemi è il mantenimento della radio map, che deve essere aggiornata spesso, per garantire la freschezza dei dati, pena l'impossibilità

di ottenere il giusto posizionamento; con meno dati da dover aggiornare, questa pratica, che di base richiederebbe molto tempo, risulterebbe più snella e meno soggetta ad aggiornamenti. Il limite di questa tecnica risiede però nell'alto costo computazionale richiesto per determinare lo stato del sistema, con un approccio basato su teoria Bayesiana e kernel function, quindi sempre probabilistico; il che rende l'esecuzione lenta e dispendiosa per terminali datati e solitamente proibitiva per dispositivi mobili.

Uno studio interessante condotto da Chen Feng, Wain Sy Anthea Au, Shahrokh Valaee e Zhenhui Tan, gruppo di studiosi dell'IEEE, dimostra come è possibile adattare le tecniche del compressive sensing al fine di creare un sistema di localizzazione "a portata di smartphone".



**Figura 2.6:** I risultati della ricostruzione della mappa mediante l'uso del CS

## 2.7 I rischi dell'effetto multipath

Di seguito sono brevemente illustrati due approcci iniziali al problema della localizzazione tramite fingerprint che evidenziano l'effetto del multipath.

### 2.7.1 Fingerprinting

Come primo passo, vengono scansionati tutti gli access point nei vari Reference Point, di questi vengono memorizzati il MAC address e il *received power level* (in dBm), come media dei livelli di ciascuna rilevazione. Per

|  |                    |
|--|--------------------|
|  | -18 dBm to -22 dBm |
|  | -23 dBm to -28 dBm |
|  | -28 dBm to -32 dBm |
|  | -33 dBm to -37 dBm |
|  | -38 dBm to -42 dBm |
|  | -43 dBm to -47 dBm |
|  | -48 dBm to -52 dBm |
|  | -53 dBm to -57 dBm |
|  | -58 dBm to -62 dBm |
|  | -63 dBm to -67 dBm |
|  | -68 dBm to -72 dBm |
|  | -73 dBm to -80 dBm |
|  | no reception       |

Figura 2.7: Colori delle intensità di segnale

ottenere una buona approssimazione, senza adottare altri accorgimenti, sarebbe bene effettuare circa 50 rilevazioni per ogni cella ( $p_{xyz}$ ).

### 2.7.2 Position Finding

Il primo algoritmo che si può adoperare per la localizzazione è basato sul confronto del segnale ricevuto con quello memorizzato.

Usando l'antenna integrata nel dispositivo. Per ogni MAC address che è anche salvato in database, una variabile contenete il numero di corrispondenze (*hits*) viene incrementata di uno. Allo stesso tempo la differenza fra il livello attuale dell'AP e quello registrato viene calcolato come:

$$D_{AP} = |p_{actual} - p_{xyz}| \quad (2.8)$$

e sommata alla variabile  $dP_{xyz}$ , ottenuta quindi come somma delle differenze fra tutti gli AP in una determinata cella:

$$dP_{xyz} = \text{sum}(D_{AP}) | \text{tutti gli APs max hit} \quad (2.9)$$

Confrontando con il database il MAC address la cella con il maggior numero di corrispondenze è scelta come attuale. Nel caso di due celle con lo stesso numero di corrispondenze viene presa come riferimento quella che più si avvicina a quella salvata in database, ovvero con  $dP_{xyz}$  minore.

In questo caso la maggior parte di falsi risultati è derivante dal fatto che "l'immagine" memorizzata in database risulti essere differente da quella rilevata a runtime. Potrebbe essere dovuto da diverse condizioni di propagazione o dalla propagazione **multipath** che affligge la localizzazione indoor.

Il fenomeno della **multipath propagation** è un'interferenza che si concretizza con l'impossibilità di comprendere l'intensità del segnale in arrivo su di unadeterminata antenna. È un fenomeno che si manifesta nei segnali radio che raggiungono una antenna da due o più percorsi, generalmente è causato da fenomeni di rifrazione o riflessione del segnale nell'aria.

Un algoritmo alternativo attribuisce un peso a ciascuna corrispondenza per ogni AP, sfruttando la differenza di potenza del segnale  $D_{AP}$ ,  $hits = hits + W$ , dove  $W$  è:

$$W = K \text{ if } D_{AP} < 3 \text{ dB} \quad (2.10)$$

$$W = K(D_{AP} - c)/(3dB - c) \text{ if } 3 \text{ db} < D_{AP} < 23 \text{ dB} \quad (2.11)$$

$$W = 0 \text{ if } D_{AP} > 23 \text{ dB} \quad (2.12)$$

dove  $K$  e  $c$  sono fattori di scala, es.  $K=2$  e  $c=23$  dB. In questo modo le corrispondenze vanno da 0 a  $K$ .

## 2.8 Punti di forza e limiti

La tecnica è fra le più utilizzate e studiate nel campo della localizzazione indoor in quanto sfrutta un'infrastruttura già presente nell'edificio che si vuole mappare, consentendo di avere ridotti costi per il deployment dell'infrastruttura e una potenziale fetta di client a costo zero. In più i moderni sistemi operativi per mobile, come Android ed iOS offrono nativamente le interfacce di interazione con le reti senza fili, per cui, disponendo di un terminale adeguato, è possibile utilizzare il solo smartphone per entrambe le fasi della localizzazione, avendo a disposizione un adeguato numero di Access Point.

Per quanto riguarda gli aspetti critici derivanti dall'uso di questa famiglia di tecniche (radio fingerprinting e in generale tutte le tecnologie basate su "Received Signal Strength"), sono facilmente contraddistinguibili quelli che influiscono negativamente sulle rilevazioni, inserendo un'interferenza che l'antenna non riesce a distinguere, falsando così la rilevazione del segnale, all'interno della letteratura presa in considerazione per la realizzazione di questo progetto, il problema che emerge maggiormente è il fenomeno di multipath propagation. Inoltre occorre evidenziare anche un limite riferito al campo di applicazione pratica di queste tecniche: l'ambiente in cui è facile creare una mappa corretta ed affidabile nel tempo, non è quello domestico, poiché troppo soggetto a variazioni improvvise (cambio del numero di AP disponibili, segnali instabili dovuti a possibili cambi dell'infrastruttura, banalmente un semplice spostamento dell'Access Point domestico costituisce una forte variazione per tutto l'ecosistema.

Infine, per far sì che la stima del fingerprint sia accettabile devono essere effettuate un elevato numero di rilevazioni ad intervalli costanti e cadenzati nel tempo, che a seconda dei parametri scelti determinano tempi di apprendimento più o meno elevati.



## Capitolo 3

# Indoor Localizer

### 3.1 Introduzione

Al termine della parte di analisi e studio delle tecniche di localizzazione indoor è stata sviluppata una applicazione per terminali Android, denominata **Indoor Localizer**, volta a studiare la tecnica del Wi-Fi fingerprint. L'applicazione vuole essere un primo approccio alla navigazione indoor e all'ecosistema Android, per verificare quanto analizzato nei capitoli precedenti. L'applicazione è stata sviluppata al fine di valutare l'effettiva possibilità di utilizzare un dispositivo mobile per gestire entrambe le fasi della localizzazione, così da non dover coinvolgere un terminale esterno per l'esecuzione dell'algoritmo di cluster matching. Indoor Localizer deve rilevare i Wi-Fi nel suo raggio d'azione, garantirne la persistenza al fine di poterli richiedere per ottenere la corretta posizione dell'utente. Affinché possa aderire a questo requisito deve disporre di un'infrastruttura adeguatamente progettata. Il terminale sul quale l'applicazione viene eseguita deve disporre di un'antenna in grado di rilevare le reti con standard IEEE 802.11 (b/g/n); l'ambiente attorno alla rete deve essere composto da un numero sufficiente di AP per permettere alla rilevazione di essere consistente e precisa. Per questa particolare implementazione il numero di AP minimo per considerare una scansione affidabile è sei per ogni RP, questo numero è stato scelto empiricamente e non è configurabile direttamente dall'utente.

## 3.2 Progettazione e scelte implementative

L'applicazione, sviluppata per terminali Android, funziona sia da rilevatore per creare la mappa dei fingerprint, che da client per la localizzazione. L'ambiente di sviluppo utilizzato è Android Studio. Per evitare la continua connessione del dispositivo utilizzato per il testing (Samsung Galaxy Note 3) tramite il cavo USB, è stato scelto di utilizzare la connessione al servizio omonimo (Android Debug Bridge) tramite la rete Wi-Fi. Per controllare ed interrogare rapidamente il database, sono state utilizzate le applicazioni SQLite Admin ed SQLite Debugger, disponibili nel Play Store.

### 3.2.1 Il database

Il database è stato implementato sfruttando il DBMS SQL presente in Android, accessibile tramite SQLite, questa scelta permette di conservare l'intera applicazione sullo smartphone, al momento della consegna l'applicazione non permette direttamente l'esportazione del database da condividere con altri terminali.

#### La creazione delle tabelle partendo dal modello dati: DatabaseHelper

Il modello dati utilizzato dall'applicazione (consultabile in fig.2.3) è stato implementato all'interno dell'applicazione sfruttando la classe Android SQLiteOpenHelper, ogni qual volta il database necessita di un refresh sul device occorre unicamente cambiare la versione del DB (parametro DB\_VERSION) perché questo venga aggiornato del tutto.

```
// DatabaseHelper.java
public class DatabaseHelper extends SQLiteOpenHelper {
    public static final String DB_NAME = "access_points.db";
    private static final int DB_VERSION = 1;

    public DatabaseHelper(Context context) {
        super(context, DB_NAME, null, DB_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        Log.d("DATABASE CREATION", "Creating database " + DB_NAME);
        // SQL statement to create APs table
        String CREATE_APS_TABLE = "CREATE TABLE aps ( " +
```

```

_id INTEGER PRIMARY KEY AUTOINCREMENT,"+
"map_name TEXT,"+"reference_point INTEGER,"+
"ssid TEXT,"+"bssid TEXT,"+"capabilities TEXT, " +
"level INTEGER,"+"frequency INTEGER,"+"hits INTEGER");
// create ap table
//...
//Create rps table
//...
}
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int
    newVersion) {
// Drop older aps table if existed
db.execSQL("DROP TABLE IF EXISTS aps");
db.execSQL("DROP TABLE IF EXISTS maps");
db.execSQL("DROP TABLE IF EXISTS rps");
// create fresh aps table
this.onCreate(db);
} }

```

Le interrogazioni al database vengono fatte dall'applicazione mediante la classe `DbManager` che si occupa di gestire gli accessi al database garantendone il corretto funzionamento.

```

//DbManager.java
//...
public long addMap(InfrastructureMap map) {
long ALREADY_IN_DB = -1;
if (!checkMapPresence(map.getMapName())) {
Log.d("Add Map", map.toString());
Log.d("[WRITING MAP TO DB]", "MAP: " + map.getMapName() + " " + map.
    getRpNumber());
ContentValues value = createMapContentValues(map.getMapName(), map.
    getRpNumber(), map.getMapImagePath());
return db.insertOrThrow(DATABASE_MAP_TABLE, null, value);
} else {
updateMapNumberOfRP(map.getMapName());
return ALREADY_IN_DB;
} //...

```

### 3.2.2 Funzionalità

All'avvio, dopo l'immagine iniziale (descritta nell'activity "SplashActivity") l'applicazione chiede all'utente di operare per le funzionalità Localize o Retrieve, presentando due diversi menu con le funzioni riportate di seguito.

#### Visualizzazione dei Wi-Fi attualmente rilevati dall'antenna del dispositivo

L'Activity principale si occupa di visualizzare al suo interno il Fragment contenente le informazioni relative agli AP rilevati in una determinata posizione. La FragmentActivity si occupa di aggiornare una ListView quando viene completata la scansione degli Access Point; tramite l'apposito menù è possibile salvare dunque questa scansione come un nuovo Reference Point per una mappa (nuova od esistente), avviare un refresh della lista che cancella le "vecchie scansioni", recupera i nuovi risultati dal Broadcast Receiver e notifica all'adapter la modifica dei dati da visualizzare nella ListView, tramite il metodo `notifyDataSetChanged()`; Questo aspetto è stato implementato sfruttando una ListActivity il cui adapter è collegato ad un Cursor in ascolto, tramite il meccanismo dei BroadcastReceiver all'evento: `WifiManager.SCAN_RESULTS_AVAILABLE_ACTION`



**Figura 3.1:** Risultato della scansione

```
//ShowWifiFragment.java
public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
    progressBar = (ProgressBar) getActivity().findViewById(R.id.
        wifi_search_progress);
    textView = (TextView) getActivity().findViewById(R.id.
        search_wifi_text_view);
    mAdapter = new SimpleAdapter(getActivity().getBaseContext(), mModel,
        R.layout.wifi_list_item_simple, FROM, TO);
    mAdapter.setViewBinder(new SimpleAdapter.ViewBinder() {
        @Override
```

```

public boolean setViewValue(View view, Object o, String s) {
    final TextView outputTextView = (TextView) view;
    // We have to detect which is the item and show it
    switch (view.getId()) {
        case R.id.ssid:
            String ssid = (String) o;
            outputTextView.setText(getResources().getString(R.string.
                ssid_value_pattern, ssid));
            break;
        case R.id.bssid:
            String bssid = (String) o;
            outputTextView.setText(getResources().getString(R.string.
                bssid_value_pattern, bssid));
            break;
        case R.id.capabilities:
            String capabilities = (String) o;
            outputTextView.setText(getResources().getString(R.string.
                capabilities_value_pattern, capabilities));
            break;
        case R.id.level:
            Integer level = (Integer) o;
            outputTextView.setText(getResources().getString(R.string.
                level_value_pattern, level));
            break;
        case R.id.frequency:
            Integer frequency = (Integer) o;
            outputTextView.setText(getResources().getString(R.string.
                frequency_value_pattern, frequency));
            break;
    }
    return true;
}
});
// Initiate wifi service manager
mainWifi = (WifiManager) getActivity().getSystemService(Context.
    WIFI_SERVICE);

// Check for wifi is disabled
if (!mainWifi.isWifiEnabled()) {
    // If wifi disabled then enable it
    Toast.makeText(getActivity().getBaseContext(), "wifi is disabled..
        making it enabled",
        Toast.LENGTH_LONG).show();
    mainWifi.setWifiEnabled(true);
}

// wifi scanned value broadcast receiver
receiverWifi = new WifiReceiver();

// Register broadcast receiver
// Broadcast receiver will automatically call when number of wifi
    connections changed
getActivity().registerReceiver(receiverWifi, new IntentFilter(
    WifiManager.SCAN_RESULTS_AVAILABLE_ACTION));
mainWifi.startScan();

// Instantiating an adapter to store each items

setListAdapter(mAdapter);

return super.onCreateView(inflater, container, savedInstanceState);
}

```

```

//Broadcastreceiver code
class WifiReceiver extends BroadcastReceiver {

    // This method call when number of wifi connections changed
    public void onReceive(Context c, Intent intent) {

        wifiList = mainWifi.getScanResults();
        for (ScanResult result : wifiList) {
            final HashMap<String, Object> item = new HashMap<String, Object>
                >();
            item.put("ssid", result.SSID);
            item.put("bssid", result.BSSID);
            item.put("capabilities", result.capabilities);
            item.put("level", result.level);
            item.put("frequency", result.frequency);
            mModel.add(item);
        }
        mAdapter.notifyDataSetChanged();
        progressBar = (ProgressBar) getActivity().findViewById(R.id.
            wifi_search_progress);
        textView = (TextView) getActivity().findViewById(R.id.
            search_wifi_text_view);
        progressBar.setVisibility(View.INVISIBLE);
        textView.setVisibility(View.INVISIBLE);
        listView.setAdapter(mAdapter);
    }
}

```

L'uso di un `BroadcastReceiver` permette di immagazzinare le informazioni fornite dal `WifiManager` (classe contenuta nel package `android.net.wifi`) relative alla scansione dei segnali ricevuti dall'antenna, quando viene invocato il metodo `ScanResult`, e di conseguenza l'ottimizzazione apportata nel registrare/deregistrare il Receiver quando l'attività viene messa in pausa/ripresa con i comandi:

```

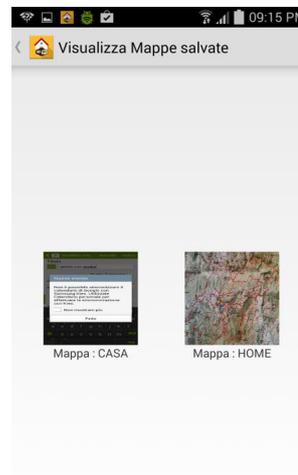
//de-registrazione
unregisterReceiver(riferimentoReceiver)
//registrazione
registerReceiver(riferimentoReceiver, IntentFilter da ascoltare)

```

(nel nostro caso `WifiManager.SCAN_RESULTS_AVAILABLE_ACTION`). Ovviamente ogni volta che il `BroadcastReceiver` riceve una nuova informazione, questa viene salvata nella lista connessa all'adapter collegato alla `ListView` che a sua volta viene notificata per aggiornare il layout corrispondente. Nel caso in cui si voglia salvare la mappa viene visualizzato un Dialog personalizzato ad hoc, che richiede all'utente il nome della mappa per il quale salvare il singolo fingerprint.

### Visualizzazione delle mappe salvate in database e i ReferencePoint connessi a ciascuna di esse

Viene mostrata una lista con le mappe attualmente salvate in database, cliccando su ogni mappa è possibile vedere l'elenco degli AP, ordinati per RP, salvati per quella determinata mappa. Con una pressione prolungata su di una singola miniatura della mappa è possibile accedere a funzionalità aggiuntive, come ad esempio cancellarla dal database. Per sviluppare questa prima vista è stato utilizzato un componente sviluppato da terzi, per gestire il numero di elementi presenti per ogni riga. Per quanto riguarda la descrizione della lista di Access Points è da ritenersi molto simile a quella descritta nel paragrafo precedente, salvo l'utilizzo di un xml differente per descrivere la lista (siccome questi AP hanno anche i campi relativi al Reference Point di riferimento).



**Figura 3.2:** Elenco delle mappe salvate

### Creazione di un nuovo Reference Point

La parte centrale del funzionamento offline è racchiusa in queste due classi. Viene chiesto all'utilizzatore di inserire le informazioni relative al RP da creare, ovvero la mappa ed il nome della stanza in cui ci si trova, dopodiché l'activity "DataRetriever" lancia il servizio che si occuperà di scansionare periodicamente i dati e di salvarli in database. L'Activity si occuperà di richiedere le informazioni, creare la mappa qualora ne sia richiesta una nuova e comunicare il tutto, tramite il meccanismo degli Extra contents al servizio. Una volta avviato il servizio "ScannerService" controllerà periodicamente i Wi-Fi rilevati dall'antenna e li accorperà suddividendoli per Access Point

**L'algoritmo di cluster-matching e di ranking**

L'algoritmo scelto per memorizzare i reference point tenta di seguire l'approccio più semplice studiato, al fine di mantenere basso il costo computazionale a discapito della precisione.

- Tutti gli Access Point vengono letti tramite l'antenna del dispositivo e temporaneamente memorizzati in una struttura apposita che rappresenta il Reference Point;
- La struttura viene scandita, controllando ciascun AP:
  - Se l'AP è già presente allora i valori di RSS vengono sommati e viene incrementato il numero di hits;
  - Altrimenti viene creato un nuovo AP con un solo hit.
- Dopo aver ripetuto la scansione per un numero prefissato di volte i dati vengono aggregati e valutati:
  - Ad ogni AP "complessivo" viene attribuito il valore di RSS pari al valore medio delle rilevazioni  $\sum RSS/hits$ .
- Entra in esecuzione l'algoritmo di ranking, che semplicemente scarta gli AP che compaiono nella lista con un numero di hits inferiore al 75% delle scansioni;
- Si procede al salvataggio su db del RP con i soli AP che hanno superato il ranking.

**Esempio pratico di una rilevazione** per capire come lavora l'algoritmo scelto.

Vengono effettuate 3 rilevazioni in un intervallo 'x' di tempo. Per semplicità riporto nel vettore i soli livelli di intensità del segnale (in dBm):

$$1^a \text{rilevazione} \begin{bmatrix} -67 \\ -78 \\ -90 \\ -88 \end{bmatrix} \quad (3.1)$$

dopo 'x' millisecondi:

$$2^a \text{rilevazione} \begin{bmatrix} -67-69 \\ -78-80 \\ -90-95 \\ -88-85 \end{bmatrix} \quad (3.2)$$

dopo 'x' millisecondi:

$$2^a \text{rilevazione} \begin{bmatrix} -67-69-71 \\ -78-80-76 \\ -90-95-93 \\ -88-85-0 \end{bmatrix} \quad (3.3)$$

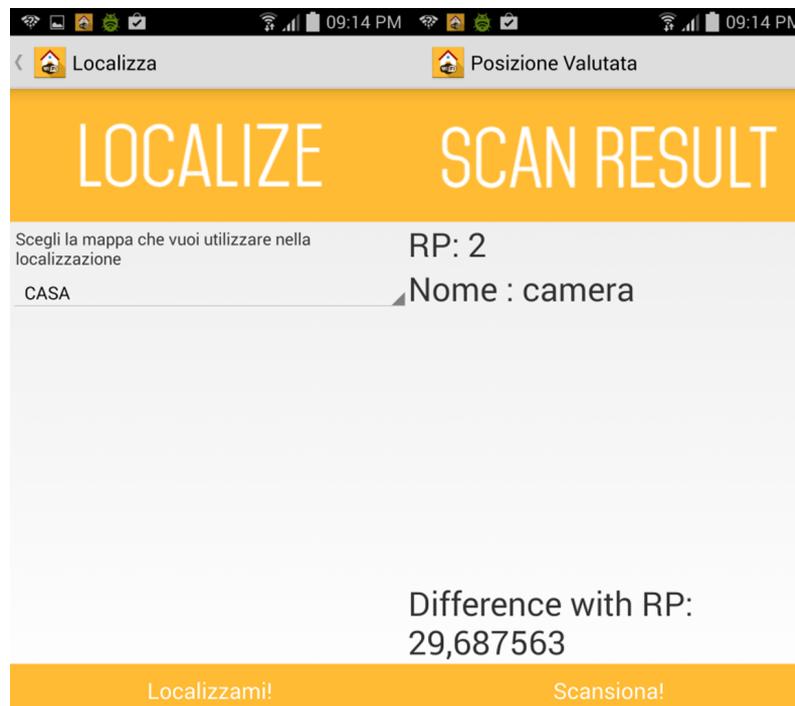
Oltre al singolo dato relativo al RSS di un AP, viene anche salvato il numero di volte in cui un determinato AP viene "colpito", ovvero viene rilevato, questo dato servirà per ottenere un valore medio del segnale più preciso. Il Reference Point risultante sarà, nel nostro esempio, il risultato della media fra le rilevazioni.

$$\begin{bmatrix} -69 \\ -78 \\ -92 \\ -84 \end{bmatrix} \quad (3.4)$$

Dopodiché il RP sarà salvato sul DB del dispositivo. L'utente verrà informato dell'avanzamento mediante Dialog e Notifiche nell'apposita barra.

### Localizzazione

La localizzazione, come già accennato avviene mediante un algoritmo di Cluster Matching. “Localization” si occupa semplicemente di chiedere all’utente su quale mappa vuole effettuare la localizzazione, dopodiché viene avviata l’Activity “PositionEvaluator” che calcola la differenza di segnale fra la rilevazione online e quelle salvate in database e trova quella che, considerando un valore di tolleranza impostato dall’utente, si avvicina maggiormente a quella rilevata, con un algoritmo di ricerca “dell’array di differenze” minimo.



**Figura 3.3:** Risposta dell’applicazione alla localizzazione

**L'algoritmo di confronto per la risposta alla localizzazione**

Anche in questo caso l'approccio scelto è di tipo deterministico, ovvero vengono analizzati tutti i dati disponibili in db senza effettuare previsioni.

- Viene effettuata una scansione degli AP rilevabili nell'area attorno all'antenna, creando così il Test Point (TP);
- Viene confrontato il TP con tutti gli RP presenti in db per quella mappa, e per ognuno di questi viene calcolata la distanza Euclidea;
- L'algoritmo ricerca la minima distanza fra tutte quelle valutate e, se inferiore alla tolleranza definita dall'utente, restituisce all'utente il nome del RP corrispondente al minimo calcolato, con la relativa misura.

```
private String compareRP(){
    SparseArray<ArrayList<AccessPoint>> map = new SparseArray<ArrayList
    <AccessPoint>>();
    SparseArray<ArrayList<Double>> differences = new SparseArray<
    ArrayList<Double>>();
    ArrayList<Integer> ids=new ArrayList<Integer>();
    try{
        dbManager.open();
        int rpNumber=dbManager.getRPNumber(mapName);
        for(int i=1;i<=rpNumber;i++){
            mCursor=dbManager.getAccessPointByMapAndRP(mapName,i);
            ArrayList<AccessPoint> aps=getAParray(mCursor);
            ArrayList<AccessPoint>values = new ArrayList<AccessPoint>();
            int rpId=0;
            for(AccessPoint ap:aps){
                values.add(ap);
                rpId=ap.getRp();
            }
            map.put(rpId,values);
            ids.add(rpId);
        }
        for(int i=0;i<map.size();i++){
            ArrayList<Double> difference = new ArrayList<Double>();
            for(AccessPoint readAP:readAps){
                for(AccessPoint aSavedAP:map.get(ids.get(i))) {
                    if (readAP.getSSID().equals(aSavedAP.getSSID())){
                        Double tmp=EuclideanDifference2(readAP,aSavedAP);
                        difference.add(tmp);
                    }
                }
            }
            differences.put(ids.get(i),difference);
        }
    }catch (SQLException e){
        e.printStackTrace();
    }finally {
        mCursor.close();
    }
}
```

```

        dbManager.close();
    }
    return searchMinimumArray(differences, ids);
}
private String searchMinimumArray(SparseArray<ArrayList<Double>> map,
    ArrayList<Integer>ids) {
    double min=1000000000;
    String rpMin="";
    int rpIdMin=-1;
    SharedPreferences prefs = PreferenceManager.
        getDefaultSharedPreferences(this);
    double tolerance=Double.parseDouble(prefs.getString("tolerance", "
        1.0"));
    for(int i=0;i<map.size();i++){
        double sum=0;
        for(Double value:map.get(ids.get(i))){
            sum+=value;
        }
        if(sum/map.size()<min && sum!=0){
            min=sum/map.size();
            try {
                dbManager.open();
                rpMin=dbManager.getRpName(mapName, ids.get(i));
                rpIdMin=ids.get(i);
                //dbManager.close();
            } catch (SQLException e) {
                e.printStackTrace();
            } finally {
                dbManager.close();
            }
        }
    }
    if(min<tolerance)
        return rpMin+ " nel RP " + rpIdMin;
    else
        return "";
}
}

```

## Capitolo 4

# Analisi dei risultati e sviluppi futuri

### 4.1 Risultati sperimentali

I risultati ottenuti dall'applicazione, in contesto domestico, confermano la correttezza dell'implementazione del WiFi fingerprinting, fornendo delle risposte significative circa la posizione dell'utente, dopo la corretta mappatura dell'edificio. Occorre precisare che la scarsa complessità degli algoritmi utilizzati influisce, talvolta negativamente, sulla risposta di localizzazione in alcune particolari situazioni che verranno analizzate a parte.

Di seguito è riassunto l'esperimento effettuato al fine di valutare le performance dell'applicazione.

La prova è stata realizzata in un reale ambiente domestico, di circa 150 metri quadrati, con l'impossibilità di ricreare una rete di AP ad hoc per la rilevazione, dovendo quindi sfruttare solamente gli access point già presenti nell'edificio, questo, per poter verificare la scalabilità del sistema in tutti i tipi di contesti indoor.

Lo smartphone utilizzato per i test è un Samsung Galaxy Note 3, terminale Android di fascia alta.

L'applicazione, è stata aggiornata al fine di riuscire a tener traccia di tutte le misurazioni effettuate. Tutti i dati raccolti sono stati memorizzati

in appositi file poi rielaborati per valutare le performance del sistema.

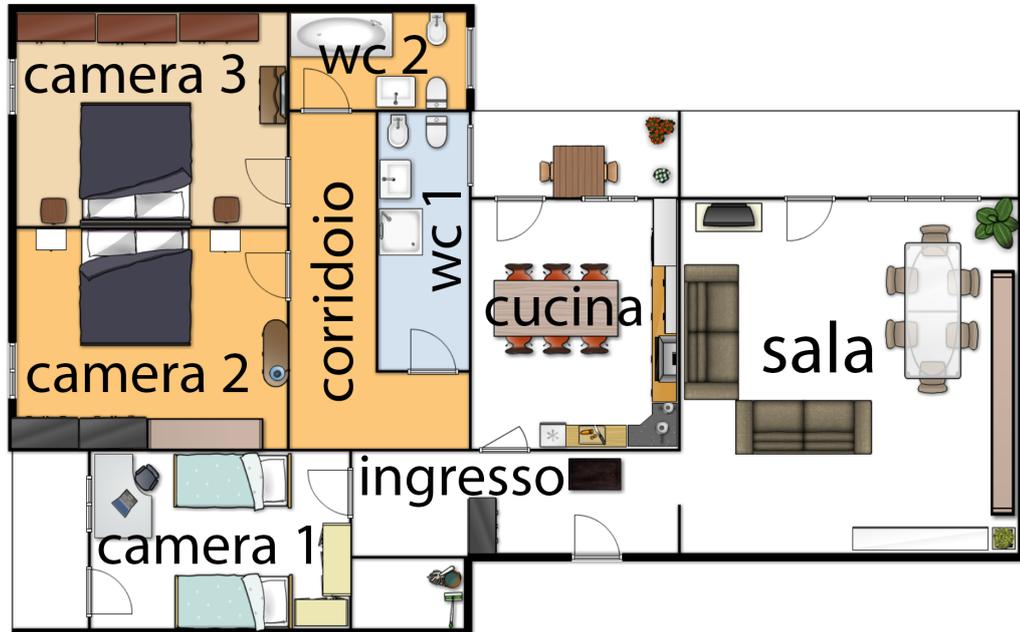


Figura 4.1: La mappa dell'abitazione usata per il test

## 4.2 Deployment del sistema

Prima di poter mettere in funzione il sistema è stato necessario disegnare la mappa dell'edificio, come visibile in Figura 4.1, al fine di suddividere l'appartamento in Reference Points consistenti, ciascuno della grandezza di circa 4 mq.

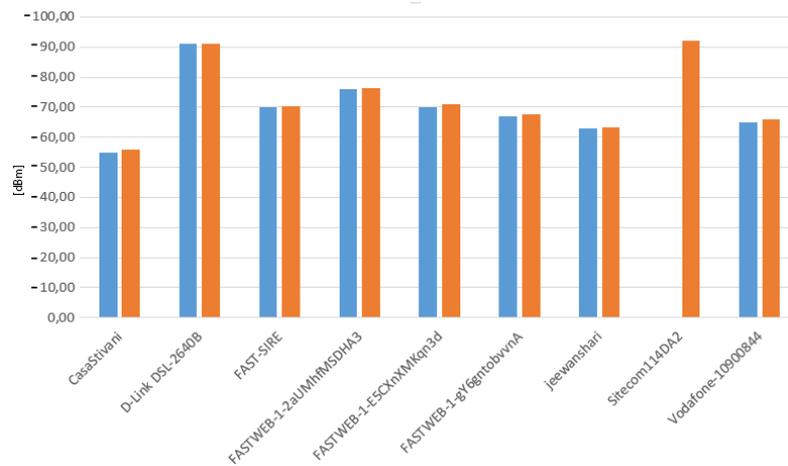
Una volta effettuata la divisione, è stata installata l'app sul device per effettuare la rilevazione all'interno delle stanze. Quanto segue è il risultato derivante dalla scelta di campionare i segnali ogni 20 secondi per un totale di 15 rilevazioni per RP.

L'analisi del risultato di una porzione della mappa è consultabile alla tabella 45, dalla quale si nota la variazione dell'RSS fra le varie stanze, i segnali indicati con 'A' sono assenti in quella stanza, mentre con 'R' sono denotati quelli scartati dopo l'esecuzione dell'algoritmo di ranking, che

|                        | camera fede 1<br>[dBm] | camera fede 2<br>[dBm] | ingresso<br>[dBm] | cucina<br>[dBm] |
|------------------------|------------------------|------------------------|-------------------|-----------------|
| FASTWEB-1-E5CXnXMKqn3d | -79                    | -58                    | -79               | -76             |
| FASTWEB-1-gY6gntobvvnA | -58                    | -67                    | -80               | -79             |
| Vodafone-10900844      | -75                    | -70                    | -84               | -83             |
| FAST-SIRE              | -75                    | -58                    | -57               | -56             |
| jeewanshari            | -69                    | -65                    | -87               | -84             |
| FASTWEB-1-2aUMhfMSDHA3 | -82                    | -78                    | -79               | -67             |
| CasaStivani            | -56                    | -52                    | -50               | -61             |
| FASTWEB-1-uRrK5mkSDLSo | (A)                    | -93(R)                 | -93(R)            | -90             |
| FASTWEB-PIERANTONI     | (A)                    | -94(R)                 | -87               | -93             |
| FASTWEB-1-00219670B6E4 | (A)                    | (A)                    | (A)               | -95(R)          |
| NETGEAR17              | (A)                    | (A)                    | (A)               | -94(R)          |
| Telecom-49495097       | (A)                    | (A)                    | (A)               | -94(R)          |

**Tabella 4.1:** Risultati degli algoritmi di clustering e ranking

filtra gli AP con una presenza inferiore al 75% all'interno della rilevazione.



**Figura 4.2:** Segnali degli AP rilevati in un RP

È interessante vedere come in fase di rilevazione sia stato acquisito anche un segnale che ha ricevuto solo due hit; questo, siccome poco significativo, è stato tagliato dall'algoritmo di ranking prima di essere salvato in database, l'esempio è visibile in fig. 4.2.

Al termine della fase di rilevazione sono stati impostati i parametri di tolleranza per effettuare la localizzazione. Le prime verifiche sono state effettuate con una tolleranza del risultato del 25% ed hanno prodotto risultati validi, riuscendo sempre, ad identificare correttamente il dispositivo all'interno della stanza di competenza, tranne in alcuni test volutamente "al limite", effettuati al confine fra due Reference Point, anche in questo caso il risultato è comunque significativo, i due Reference Point con distanza Euclidea minore sono proprio quelli confinanti (fig. 4.3).

|          | RP               | dist.   | RP               | dist.   |
|----------|------------------|---------|------------------|---------|
| <b>1</b> | RP:[ camera fede | 1 292,2 | RP:[ camera fede | 1 264,2 |
|          | RP:[ camera fede | 2 284,3 | RP:[ camera fede | 2 258,1 |
|          | RP:[ ingresso    | 3 158,8 | RP:[ ingresso    | 3 157,0 |
|          | RP:[ cucina      | 4 151,2 | RP:[ cucina      | 4 85,6  |
| <b>2</b> | RP:[ camera fede | 1 273,9 | RP:[ camera fede | 1 289,2 |
|          | RP:[ camera fede | 2 241,3 | RP:[ camera fede | 2 285,6 |
|          | RP:[ ingresso    | 3 183,5 | RP:[ ingresso    | 3 169,2 |
|          | RP:[ cucina      | 4 181,0 | RP:[ cucina      | 4 217,9 |
| <b>3</b> | RP:[ camera fede | 1 146,9 | RP:[ camera fede | 1 127,6 |
|          | RP:[ camera fede | 2 166,4 | RP:[ camera fede | 2 207,3 |
|          | RP:[ ingresso    | 3 236,8 | RP:[ ingresso    | 3 257,4 |
|          | RP:[ cucina      | 4 253,6 | RP:[ cucina      | 4 248,3 |
| <b>4</b> | RP:[ camera fede | 1 198,0 | RP:[ camera fede | 1 171,3 |
|          | RP:[ camera fede | 2 136,6 | RP:[ camera fede | 2 139,6 |
|          | RP:[ ingresso    | 3 267,0 | RP:[ ingresso    | 3 257,8 |
|          | RP:[ cucina      | 4 286,9 | RP:[ cucina      | 4 290,4 |

Figura 4.3: Test di localizzazione

## 4.3 Valutazione delle performance

A corredo dell'analisi empirica è importante effettuare una stima delle performance del sistema, considerando come indici i parametri fondamentali analizzati nel Capitolo 1. Di seguito il sistema è analizzato alla luce dei risultati ottenuti dai test eseguiti a run-time.

### 4.3.1 I parametri fondamentali

#### **Accuratezza**

L'accuratezza del sistema dipende da tanti fattori. Il risultato ottenuto dimostra che, con gli algoritmi utilizzati, l'accuratezza della risposta di localizzazione è attorno a 1,5 metri, consentendo praticamente sempre di localizzarsi all'interno della stanza giusta. Nei casi limite, ovvero fra una stanza e un'altra, la risposta "ondeggia" fra la scelta di uno dei due RP di confine.

#### **Precisione**

La tolleranza, parametro configurabile dall'utente tramite il meccanismo delle SharedPreferences, è risultata buona per la risposta di localizzazione. Una volta posizionato nel giusto riferimento, la differenza media di segnale memorizzato e il corrispettivo rilevato dall'antenna non ha mai superato il valore impostato, non classificando come aree "non mappate" quelle per le quali era già stata eseguita una scansione.

#### **Complessità**

La complessità del sistema risulta tutto sommato bassa, il costo computazionale è limitato ai soli confronti e a pochi calcoli semplici che rendono adatto il sistema anche a terminali più datati.

In aggiunta al terminale ufficiale utilizzato per i test, l'applicazione è stata provata anche con altri dispositivi con caratteristiche differenti, di fascia inferiore; anche in questo caso i tempi di risposta sono stati nell'ordine del secondo.

**Robustezza**

Il sistema necessita di un aggiornamento programmato, soprattutto in un ambiente soggetto a forti variazioni, come quello domestico. La stabilità del sistema è infatti facilmente intaccabile; è sufficiente spostare un emettitore "fondamentale" (con un alto hit ratio), o addirittura spegnerlo per rendere inefficace la mappa memorizzata e di conseguenza impossibile la localizzazione.

**Costo**

Come previsto, il costo del sistema è relativamente basso, l'infrastruttura è composta unicamente dagli access point e dallo smartphone, non richiede quindi sistemi dedicati. Altro costo è invece quello di messa in funzione e manutenzione del sistema; sicuramente più elevato, dati i tempi necessari alla costruzione della mappa.

**Scalabilità**

Il sistema ha come vantaggio un'alta scalabilità che lo rende adatto sia in ambito domestico, che per l'utilizzo in grandi spazi nel quale risulta ancora più performante.

## 4.4 Critiche al sistema e possibili sviluppi

Il sistema è stato volutamente sviluppato per valutare la tecnica del Wi-Fi fingerprint; nonostante l'approccio didattico l'applicazione è comunque progettata al fine di essere estesa per utilizzi reali, in tal senso è sicuramente possibile utilizzare alcuni accorgimenti volti ad aggiungere servizi specifici per l'utente o algoritmi più performanti.

Tolti alcuni aspetti, volutamente non trattati, l'applicazione non è ancora un sistema totalmente affidabile, necessita, ad esempio, di un sistema in grado di discriminare i punti di frontiera, della navigazione in real-time e della corretta visualizzazione del dispositivo su un supporto grafico più "accattivante".

Per quanto riguarda la complessità degli algoritmi utilizzati, sicuramente si può decidere di utilizzare soluzioni più performanti, partendo dalla localizzazione che potrebbe adottare i concetti di *coarse e fine localization* riducendo drasticamente il confronto fra RP.

#### **Orientamento dell'antenna**

Un aspetto trascurato in fase di sviluppo e in fase di testing è stato l'orientamento dell'antenna rispetto al ricevitore. È stato visto come combinando le informazioni cardinali dell'antenna, sia in fase di mapping che in fase di rilevazione, si possano raggiungere precisioni più elevate, a discapito del costo computazionale.

#### **Integrazione con servizi di mappe indoor**

Indoor Localizer, nella sua prima versione, non supporta la localizzazione su mappa. Siccome in fase di progettazione non è stato identificato un metodo efficace che rispondesse a tale scenario, lo studio non si è soffermato sulla visualizzazione della risposta di localizzazione. È comunque importante un accenno alle tecniche disponibili per visualizzare la posizione su di una mappa.

Il meccanismo più semplice, e meno efficace, consiste nel visualizzare, sul display del device, un'immagine statica sulla quale un *pin* identifica l'utente, quest'approccio però non concede troppa interazione all'utente, al fine di ottenere ulteriori informazioni sulla propria posizione; ben più interessante sotto questo punto di vista è la tecnologia delle Indoor Maps di Google che permettono la navigazione con mappe sofisticate, simili a quelle per l'esterno, seppur per i soli edifici pubblici.



## Capitolo 5

# Conclusioni

### 5.1 Conclusioni

Lo sviluppo delle comunicazioni wireless, dei dispositivi a basso costo ed in generale il facile accesso alle reti, aprono diverse possibilità di applicazione interessanti per il mercato. Basti pensare ad un centro commerciale che vuole fornire servizi personalizzati ai propri clienti, come le offerte fra gli scaffali del supermercato, corsia per corsia; piuttosto che iniziative, più nobili, di navigazione in ambienti complessi per utenti disabili, con un sistema che integra vari aspetti della navigazione, volti a permettere all'utente una "passeggiata sicura" in un luogo dove possono essere presenti barriere architettoniche.

L'interesse per questo settore è confermato dalla discussione viva fra gli esperti in campo di reti e localizzazione, volta allo sviluppo di studi che migliorino le tecniche presentate in questo scritto, al fine di realizzare progetti utili alla società sempre più diretta verso una convivenza con i propri dispositivi mobili.



# Bibliografia

- [1] Jinhong Xiao, Zhi Liu, Yang Yang, Dan Liu, Xu Han, "Comparison and Analysis of Indoor Wireless Positioning Techniques", *Changchun University of Science And Technology*, 2011.
- [2] Messerschmidt Cristoph, Dippold Michael, "Indoor-Localization with WLAN-Fingerprinting and directional antennas", *Munich University of Applied Sciences*, 2013.
- [3] Chen Feng, Wain Sy Anthea Au, Shahrokh Valaee, Zhenhui Tan, "Received-Signal-Strength-Based Indoor Positioning Using Compressive Sensing", *IEEE TRANSACTIONS ON MOBILE COMPUTING*, 2012.
- [4] Igor Bisio, Fabio Lavagetto, Mario Marchese, Andrea Sciarrone, "Performance Comparison of a Probabilistic Fingerprint-based Indoor Positioning System over Different Smartphones", *Dept. of Telecommunication, Electronic, Electric Engineering and Naval Architectures, DITEN, Genoa, Italy*, 2013.
- [5] Lin Ma, Xinru Ma, Xi Liu, Yubin Xu, "WLAN Indoor Positioning Algorithm Based On Sub-Regions Information Gain Theory", *Communication Research Center, Harbin Institute of Technology, Harbin, China*, 2013.
- [6] Hui Liu, Houshang Darabi, Pat Banerjee, Jing Liu, "Survey of Wireless Indoor Positioning Techniques and Systems", *IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS*, 2013.
- [7] B. Li, Y. Wang, H.K. Lee, A. Dempster, C. Rizos, "Method for yielding a database of location fingerprints in WLAN", *2005* 2005.
- [8] Kamol Kæmarungsi, Prashant Krishnamurthy, "Modeling of Indoor Positioning Systems Based on Location Fingerprining", *School of Information Science, University of Pittsburgh*, 2004.

- [9] Sisongkham Phimmasean, Thanapong Chuenurajit, Panarat Chern-tonomwong, "Indoor Localization System based on Fingerprint Technique using RFID Passive Tag", *King Mongkut's Institute of Technology Ladkrabang (KMITL)*, 2013.
- [10] Massimo Carli "Android 4, guida per lo sviluppatore" Ed. Apogeo, 2014.
- [11] <http://en.wikipedia.org/wiki/Multilateration>
- [12] [http://en.wikipedia.org/wiki/Indoor\\_positioning\\_system](http://en.wikipedia.org/wiki/Indoor_positioning_system)
- [13] [http://en.wikipedia.org/wiki/Received\\_signal\\_strength\\_indication](http://en.wikipedia.org/wiki/Received_signal_strength_indication)
- [14] <https://www.google.com/maps/about/partners/indoormaps/>
- [15] <http://www.floorplanner.com/>

*We never fail when we try to do our duty, we always fail when we neglect  
to do it*

— Robert Baden-Powell

## Ringraziamenti

Nemmeno io credo che sia possibile essere arrivato fin qua, partendo dal presupposto che mi sono sempre sentito un emotivo umanista. Penso che la mia carriera universitaria possa essere descritta come l'adolescenza: un miscuglio di sentimenti, di modi, di tempi, di avventure che mi hanno portato a crescere, non tanto come studente, quanto come persona.

I ringraziamenti per questo supporto vanno naturalmente alla mia famiglia: a papà che con la sua presenza mi ha sempre sfidato a fare del mio meglio, non per essere il migliore, ma per ricercare sempre la passione in quello che faccio, con l'alto obiettivo di essere utile agli altri; alla mia dolce mamma che mi ha sempre accompagnato nei momenti di totale abbandono, senza mai pretendere troppo da me, che l'università non è mai stata in cima alle mie priorità; grazie alla mia nonna che mi ha mantenuto in piedi tutto questo tempo e un grande grazie a mia sorella che ogni giorno mi sprona ad esserle d'esempio.

Poi ci sono tutti i miei amici, ciascuno di loro conosce una mia sfaccettatura, che nel complesso mi caratterizza appieno. Li ringrazio per avermi messo in difficoltà ogni volta in cui si è parlato di futuro, di realizzarsi; li ringrazio per avermi spronato a rimettermi in carreggiata, facendomi capire che se non si lavora prima per sé stessi è impossibile donarsi al prossimo.

Infine, ringrazio il mio relatore, per aver portato pazienza con questo studente davvero negligente.

*Bologna, Settembre 2014*

F. S.