Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE
Corso di Laurea in Informatica

Rischi di privacy connessi ai sensori integrati negli smartphone

Relatore:
Dott.
Luca Bedogni

Presentata da:
Alexei Amato

Sessione II
Anno accademico 2017/2018

Indice

Introduzione

Lo smartphone, uno strumento molto utile ormai alla portata di tutti, dai più giovani ai più anziani, da utilizzare per telefonare o anche per navigare in internet. Ma come in tutte le cose, presenta anche alcuni rischi riguardanti l'utilizzo della geolocalizzazione, tra questi quelli connessi alla localizzazione senza necessariamente ricorrere all'uso del GPS. Verrà trattato il tema su come riuscire nel tentativo di recuperare informazioni sulla localizzazione di un possessore di smartphone, senza l'utilizzo dei servizi di rete o localizzazione satellitare, utilizzando quindi dati rilevati da alcuni sensori ormai installati sulla totalità degli smartphone in commercio, il magnetometro e l'accelerometro. Questi sensori consentono di descrivere i movimenti subiti dallo smartphone, in base alla rotazione degli assi tridimensionali e all'accelerazione subita dallo stesso.

Per riuscire ad ottenere i dati dei sensori, basterebbe creare un'applicazione malware che legge i dati rilevati dai sensori, ciò è possibile per il fatto che l'accesso ai sensori non è strettamente legato all'autorizzazione dell'utente. In questo caso la vittima installerebbe un'applicazione che a sua insaputa lancia dei servizi in background e inizia a monitorare i dati dei sensori, nel momento in cui l'utente inizierà a guidare un veicolo, riporrà lo smartphone in una posizione fissa e a quel punto i sensori rileveranno i dati necessari all'analisi del percorso.

Nel corso dell'elaborato si presenterà la sola simulazione del possibile scenario reale, che cercherà di riprodurre più fedelmente possibile una rilevazione dei dati attraverso il movimento di uno smartphone, facendo affidamento a tool e algoritmi implementati ad hoc per la creazione di percorsi reali e la loro ricerca all'interno delle mappe di alcune città prese in esame. Ogni città verrà considerata come un grafo non orientato dove gli incroci corrispondono ai nodi e le strade agli archi, ognuno di questi contenente diverse informazioni utili al raggiungimento dell'obbiettivo.

L'obbiettivo della tesi è proprio quello di trovare l'equilibrio che si trova tra l'invasione della privacy degli utenti e un'utilità per implementazioni future o sperimentali del metodo che, se utilizzate in maniera corretta, potrebbero portare a un progresso su questa tematica.

A questo proposito, si descriveranno alcune metriche e pattern utilizzati per il riconoscimento dei percorsi sulla mappa senza l'utilizzo di algoritmi di machine learning, il che rende la sperimentazione più difficile e interessante. Proprio su queste metriche verrà

fatta un'analisi accurata, attraverso grafici e rappresentazioni grafiche, in diverse città di diverse nazioni per fare svariati confronti e mettere in relazione ciò che le differenzia o le accomuna.

Si discuterà anche delle architetture delle diverse città e come queste influenzano il successo o il fallimento dell'obbiettivo, facendo confronti su piante di città a scacchiera ed altre radiocentriche. Per pianta a scacchiera si intende uno sviluppo delle strade della città disposte a griglia, tipiche delle antiche città romane, formando isolati e forme geometriche definite. La pianta radiocentrica invece, presenta una disposizione delle strade a raggiera convergente verso il centro cittadino, una caratteristica presente per lo più in città di origine medioevale.

A sostegno della sperimentazione verranno descritte alcune sperimentazioni effettuate da altre università riguardo a questo tema, in che modo sono state fatte e i risultati da loro ottenuti, in modo da avere delle linea guida e un confronto continuo sui diversi approcci adottati.

Capitolo 1

Lo stato dell'arte

A supporto di quanto trattato in questo elaborato, esistono alcuni studi che fanno emergere risultati interessanti sulla tematica e in modi differenti. Ciò avviene perchè la diffusione dell'utilizzo dello smartphone è in fase di crescita, quindi sempre oggetto di dubbi e perplessità. L'articolo riguardante uno studio condotto da un team di ricercatori del College of Computer and Information Science Northeastern University di Boston [1] è quello che si accosta maggiormente a questo elaborato, in quanto riporta dati e conclusioni non proprio irrilevanti. Lo studio si basa su dati realmente raccolti tramite un'applicazione, creata dagli stessi autori, che registra i movimenti dello smartphone all'interno di un veicolo.

1.1 Scenario delle motivazioni

Ciò che ha spinto i ricercatori dell'articolo in questione, a condurre studi su questa tematica, sono i possibili attacchi alla privacy degli utenti da parte di hacker o altri tipi di entità. Gli attacchi consistono nell'acquisizione di dati rilevati da accelerometro, giroscopio e magnetometro, sensori che ormai sono parte integrante di ogni smartphone. In questo scenario non vi è uso del GPS, i dati vengono acquisiti tramite un'app creata ad hoc e scaricata dagli utenti negli store ufficiali, ignari delle funzioni maligne della stessa. L'app lavorerà in background, iniziando a registrare i dati non appena l'utente si metterà alla guida che successivamente manderà ai malintenzionati tramite la connessione internet dello smartphone.

Tutto ciò può avvenire perché le app non hanno bisogno di richiedere i permessi per accedere ai dati rilevati dai sensori.

1.2 Le prove

L'entità maligna che deciderà di effettuare un attacco dovrà superare diversi tipi di ostacoli per arrivare a una conclusione esaustiva, come la grandezza dell'area, il rumore sui dati provocato da sensori poco accurati, lo stile di guida del conducente, i percorsi somiglianti tra loro.

Tutte queste caratteristiche vanno a influenzare la fattibilità e/o la riuscita dell'attacco, soprattutto per quanto riguarda il rumore sui dati che, a seconda della sua grandezza, potrebbe vanificare del tutto l'attacco.

1.3 Modello antagonista

Il modello utilizzato dai malintenzionati si basa su poche informazioni riguardanti l'utente e le sue abitudini.

L'applicazione maligna raccoglierà costantemente dati sia attivamente che in background e li invierà ai server predisposti ad intervalli regolari.

La raccolta di dati avviene all'inizio della guida considerando che, nella maggioranza dei casi, lo smartphone rimane nella stessa posizione per tutta la durata del percorso, questo aumenta l'accuratezza dei dati raccolti.

Infine, come informazioni di posizionamento iniziali, non avendo accesso ai servizi di localizzazione, vengono considerate informazioni base di abitazione e spostamenti della vittima.

1.4 L'approccio

L'approccio per un attacco ben riuscito consiste nella raccolta dei dati attraverso l'applicazione maligna e la rielaborazione tramite un server remoto, che classificherà i dati creando una lista ordinata dei percorsi riconoscibili.

Inizialmente, si estrapolano i dati da una mappa e si convertono in una struttura specifica di database, che può essere riutilizzata negli attacchi seguenti. Poi vengono raccolti i dati tramite l'app installata sullo smartphone della vittima e in un secondo momento verranno analizzati dal server remoto che genererà una traccia di spostamento per la vittima.

Infine, verrà eseguito un'algoritmo di ricerca che classificherà i percorsi possibili e li darà in output come lista ordinata.

Nella raccolta dei dati, il sensore che risulta più utile è il giroscopio, perché fornisce dati più accurati rispetto ad altri sensori e rivela gli angoli delle curve negli incroci e le curvature delle strade, elementi particolarmente utili per il riconoscimento. In ogni caso, tutti i dati dei sensori vengono fortemente influenzati da fattori esterni come le condi-

zioni di traffico o della strada, interferenze elettromagnetiche causate da altri dispositivi presenti sul veicolo.

In questo studio, vengono costruiti dei grafi basati su OpenStreetMap [2], una raccolta di mappe open source, sui quali verrà applicato l'algoritmo di ricerca.

La ricerca viene eseguita su un grafo diretto in cui gli incroci della mappa rappresentano i nodi e le strade rappresentano gli archi, per la costruzione del grafo viene considerata l'intera area geografica e agli archi vengono aggiunti dei pesi.

L'algoritmo di ricerca valuta i percorsi visitando il grafo e tiene in memoria quelli riconoscibili, ad ogni step della ricerca vengono esaminati tutti gli archi uscenti da un nodo dato per stabilire quale sarà il prossimo nodo da esaminare. Alla fine della ricerca viene stilata una lista ordinata contenente i percorsi possibili con il relativo punteggio che si basa sulla differenza tra i pesi degli archi e gli angoli di curvatura quando si effettua una svolta.

1.5 Sistema di studio

Per lo studio di questa tematica, sono stati implementati diversi algoritmi di ricerca e classificazione dei dati, l'algoritmo di base assume che chi effettua l'attacco riesca ad accedere solo ai dati del giroscopio, in questo modo l'unica metrica utilizzata per il riconoscimento saranno le curve effettuate dalla vittima.

Successivamente viene implementato un algoritmo più complesso che, oltre agli angoli delle svolte effettuate, tiene conto anche della curvatura della strada e del tempo di percorrenza.

Una grande difficoltà da superare per la riuscita di questo attacco alla privacy sta nell'estrapolazione dei dati dei sensori influenzati dal rumore, diverse sono le cause che comportano questi errori sui dati, dai terreni sconnessi allo stile di guida degli utenti e oltre a questi si aggiungono errori interni come l'errato allineamento dei sensori o la temperatura del dispositivo. Per ovviare al mancato allineamento dei sensori si usa una tecnica di ricalibrazione e rotazione dei dati, per adattarli all'orientamento del dispositivo.

1.6 Valutazione

Per la valutazione, viene utilizzato il confronto tra un'analisi simulata e un'analisi reale. Nell'analisi reale è fondamentale l'accuratezza del giroscopio, in quanto unico sensore a non essere influenzato dall'ambiente esterno.

Per questo studio sono state prese in considerazioni diverse città del mondo con diverse misure di popolazione, estensione geografica e densità della rete stradale. Non sempre queste unità di misura formano un unico tipo di cluster, poiché città come Manhattan,

che risulta molto popolata, ha un grafo più piccolo rispetto ad altre città come Roma, Boston o Londra.

Particolarità della maggior parte delle città americane è la pianta a scacchiera, ciò significa che molti incroci risultano avere angoli di curvatura di 90°, portando a un più difficile riconoscimento dei dati ottenuti.

Per quanto riguarda l'analisi simulata, sono stati generati dei percorsi scegliendo, per una città tra le 11 usate per lo studio, una lunghezza di percorso casuale, successivamente aggiungendo segmenti connessi con angoli di curvatura compresi tra 30° e 150° e tempo di percorrenza superiore a 10 secondi. Ovviamente, per simulare al meglio una situazione reale, sono stati aggiunti vari livelli di rumore ai dati tramite distribuzioni uniformi sia per il magnetometro che per l'accelerometro.

La valutazione, in generale, si basa su diversi scenari, ciò consente di capire al meglio le prestazioni dell'attacco nei diversi ambienti. Lo scenario ideale è quello privo di rumore sui dati, quello peggiore viene considerato con condizioni di traffico estreme e smartphone parecchio datati. Lo scenario tipico rispecchia condizioni di traffico usuali e smartphone attuali, mentre quello futuro tiene conto di un traffico meno intenso e smartphone dotati di sensori molto più accurati.

Nella simulazione sono stati usati i 4 scenari descritti precedentemente per ogni città, in questo modo diventano in totale 44 test e per ognuno di essi sono stati generati 2000 percorsi simulati.

I risultati mostrano come le diverse reti stradali e le diverse tipologie di pianta possano fare la differenza, in termini di risultati ottenuti.

Per confrontare i risultati ottenuti dalla simulazione, sono stati effettuati esperimenti reali di guida nelle città di Boston e Waltham, per ogni città sono stati presi 70 percorsi differenti. Per l'esperimento sono stati chiamati 4 autisti a cui sono state date istruzioni di:

- mettere lo smartphone ovunque all'interno del veicolo, purché fosse in una posizione fissa
- aspettare almeno 10 secondi per iniziare a guidare
- guidare all'interno della città ed effettuare almeno tre svolte nel loro percorso

Queste istruzioni consentono di creare un modello per lo scenario realistico usato nelle analisi dei dati.

Il risultato finale dell'esperimento si colloca tra lo scenario tipico e quello peggiore, con Boston più vicino al peggiore e Waltham più vicino a quello tipico, la situazione viene accentuata dal fatto che non si hanno informazioni preventive sui percorsi dell'utente e dalle condizioni di traffico intense di Boston.

1.7 Fattibilità degli attacchi

La fattibilità degli attacchi va misurata in tempo di esecuzione degli script che processano i dati raccolti e quelli che eseguono la ricerca sul grafo della città in questione, nello specifico i tempi di ricerca dipendono dalla lunghezza dei percorsi e dalla grandezza del grafo stesso.

Chiaramente, più è grande il grafo in oggetto, maggiore sarà il tempo di ricerca dei percorsi, mentre la rielaborazione dei dati raccolti tende a rimanere costante, a prescindere dalle diverse grandezze.

Ciò significa che, con le risorse adeguate, il malintenzionato riesce a recuperare milioni di percorsi possibili in pochissimo tempo.

Altri fattori che possono influenzare la riuscita dell'attacco sono i parametri dell'algoritmo utilizzato e le assunzioni del caso descritti di seguito.

Se i pesi degli archi del grafo vengono azzerati, si nota come il fattore con maggior rilevanza sia la curvatura delle strade, questo si può applicare a quei grafi che hanno molti percorsi unici. Il tempo di percorrenza diventa secondario, poiché può essere influenzato da vari fattori esterni. Per una maggiore probabilità di riuscita, i pesi andranno scelti in base alla grandezza dei grafi in questione.

Filtrare le soglie massime è un altro modo per capirne l'impatto sui risultati, questo consente di eliminare i percorsi peggiori ma rischia di eliminare anche percorsi utili. Si nota che con una direzione maggiormente dritta e un limite sugli angoli di curvatura, le prestazioni diminuiscono leggermente. Se invece si aggiunge un limite abbastanza restrittivo sul tempo di percorrenza, questo farà crollare le prestazioni.

La calibrazione può essere effettuata in un modo differente, tenendo conto del tempo di stazionamento rilevato ed elaborando i movimenti del giroscopio in quel momento. In questo modo, l'accuratezza diminuisce del 10%-15% rispetto al metodo standard.

Avere informazioni preventive sulla cronologia di spostamento della vittima o conoscere il punto di partenza o destinazione, aiuterebbe a migliorare l'affidabilità dei risultati, ma recuperare queste informazioni potrebbe richiedere parecchio tempo.

Una questione delicata da affrontare è quella della posizione dello smartphone all'interno del veicolo, lo scenario tipico considera una posizione fissa durante la guida a causa delle leggi sulla sicurezza dei codici della strada nella maggior parte degli stati. Ciò non toglie che chiunque possa trasgredire le suddette leggi e questo cambierebbe notevolmente lo scenario di partenza, poiché una rotazione del dispositivo porterebbe all'interpretazione di una svolta effettuata. La soluzione a questo problema sarebbe quella di considerare questa condizione quando la variazione dei tre assi del giroscopio è elevata ma per un breve periodo di tempo, in questo caso verrebbero ignorati i dati.

In aggiunto a quanto già utilizzato, un modo per migliorare l'accuratezza dello studio è quello di utilizzare una tecnica simile al conteggio dei passi di Android [3], per stabilire quando l'utente smette di camminare e monta sul veicolo.

La cosa più semplice da effettuare per riconoscere la città in cui si trova la vittima, a causa delle numerose reti Wi-Fi e mobili, è quella di rintracciare gli indirizzi IP attraverso queste reti. Anche i social network, in questo caso, sarebbero d'aiuto al malintenzionato per rintracciare la vittima attraverso luoghi visitati di recente o spesso oppure la condivisione della posizione sulla rete.

Tutte le assunzioni fatte fin ora fanno capo a un'unica assunzione principale, ovvero quella di presumere che gli spostamenti degli utenti siano sempre in avanti, ciò comporta una perdita di informazioni quando avvengono manovre contrarie (i. e. uscita da un parcheggio).

1.8 Contromisure

Il modo migliore per prevenire tutti i problemi descritti in precedenza, è quello di intervenire in termini di sicurezza sui dispositivi quando una determinata applicazione viene installata, ovvero richiedere i permessi per accedere ai dati del sistema per qualunque cosa, dai media ai sensori, agli sms e via discorrendo. Nel caso la prevenzione non dovesse bastare, un modo per intercettare un'attività illecita potrebbe essere quello di attivare le notifiche sull'utilizzo dei vari servizi di sistema, quali GPS, sensori e reti.

Sebbene possano sembrare contromisure efficaci, hanno comunque dei punti deboli, primo il problema dell'implementazione delle sopracitate misure di sicurezza direttamente sul sistema operativo, in aggiunta esistono tool come ddi [4], un applicativo studiato appositamente per strumentalizzare in modo semplice in modo semplice il codice Dalvik, questo si basa su l'inoculazione di librerie e l'intercettazione degli entry points dei metodi. Per rendere tutto più semplice e adattivo, il codice è scritto parzialmente in Java.

Altro tool interessante su questa tematica è Boxify [5], si basa sulla virtualizzazione e la separazione dei processi per incapsulare le applicazioni sconosciute all'interno di un ambiente protetto, tutto ciò eseguito come una normale applicazione senza la necessità di intaccare il firmware e permessi di root.

1.9 Studi correlati

In questa sezione verranno descritte alcune ricerche riguardanti lo stesso tema, in particolare gli studi che utilizzano metodi alternativi per il riconoscimento di percorsi, come sensori, reti Wi-Fi, reti GSM, consumi di energia e controlli di movimento. Questi studi, evidenziando le vulnerabilità della privacy hanno contribuito all'implementazione di vari sistemi di prevenzione per gli utenti finali.

Una ricerca condotta dall'Università del Michigan ha dato vita a LP-Guardian [6], un framework per la protezione della privacy sulla localizzazione di utenti Android, combat-

tendo le minacce di intercettazione e rilevamento della posizione ma sempre mantenendo le funzionalità dell'applicazione, un approccio più risolutivo che preventivo.

Lo studio condotto all'Università del Minnesota [7] descrive la minaccia di localizzazione attraverso le reti mobili GSM, poiché queste reti utilizzano la comunicazione broadcast continua. Per questo motivo è sufficiente restare in ascolto sul broadcast per concludere se un utente si trova in una piccola area oppure fuori da una più grande.

Un articolo pubblicato dal Wichita State University [8] discute dell'efficacia dei meccanismi di protezione dalla tracciabilità degli utenti basati su un insieme di zone differenti, l'esperimento si svolge tracciando 80 volontari per 4 mesi utilizzando gli Access Points. I risultati ottenuti evidenziano che, in uno scenario realistico, pur utilizzando strategie di tracciabilità molto semplici, si ottengono molti successi.

Gli autori Lee e Mase [9] hanno sperimentato se fosse possibile tracciare la posizione di un utente attraverso lo smartphone, cioè capire se l'utente fosse in stato eretto, seduto, sdraiato o se stesse camminando per andare ad un piano superiore o inferiore.

L'approccio utilizzato in questo studio sfrutta, sia il metodo assoluto che quello relativo che, di solito risulta più efficiente in termini di prestazioni e costi. La principale idea di questo approccio è quella di integrare informazioni di movimento incrementali in un determinato periodo, tecnica usata in robotica e conosciuta con il nome di «*dead-reckoning*».

Il loro sistema consiste nell'utilizzo di un palmare equipaggiato di un modulo di sensori, quali accelerometro, magnetometro, un microcontroller e un sensore di velocità angolare. Esso riesce a riconoscere la posizione seduta e la posizione in piedi, più i movimenti attraverso i piani. Per riconoscere i movimenti dell'utente, viene fatto un training al modulo implementato dando in input un set di dati che serve a determinare i parametri dell'unità per il riconoscimento del movimento. Attraverso questi dati, il modulo di riconoscimento determina automaticamente i parametri e il sistema registra la sequenza dei movimenti fatti dall'utente e, usando questa sequenza, costruisce una tabella di transizione.

Il metodo da loro utilizzato risulta più accurato di un semplice smartphone, poiché è stata fatta una modifica a livello hardware e in questo modo sono stati aggiunti parecchi sensori che un normale smartphone non riesce a sfruttare.

I test sono stati effettuati tra i vari ambienti di un ufficio, raccogliendo i dati del cammino di otto soggetti, di cui due femmine e sei maschi con età compresa tra 23 e 51 anni, essi indossavano diversi tipi di scarpe. Per la fase di training, ogni soggetto ha camminato ripetendo 20 cicli di spostamento su più livelli con 3 velocità diverse e spostandosi tra i piani attraverso una scala con 24 gradini.

L'unità di riconoscimento ha estratto i dati automaticamente tenendo conto di media e deviazione standard dei valori dei sensori, per la valutazione dei luoghi proposti sono stati scelti 5 ambienti differenti usati nella quotidianità della vita in ufficio, inoltre sono stati generati 10 vettori di transizione oltre al massimo di 20 transizioni.

Gli esperimenti di transizione tra i vari ambienti sono stati effettuati utilizzando inizialmente percorsi circolari.

I risultati ottenuti mostrano un elevato rateo di successo (91,8%) su 10 transizioni tra i vari ambienti, tuttavia il metodo risulta limitato se utilizzato su percorsi lunghi.

Uno studio della Carnegie Mellon University [14] discute delle potenzialità dei sensori degli smartphone che possono essere sfruttate per il riconoscimento, in particolare dell'accelerometro, poiché è diventato uno dei sensori più diffusi tra diversi tipi di dispositivi, che formano una grande rete di vulnerabilità.

Gli autori mostrano come i dati dei sensori di movimento possono essere usati per effettuare una stima della traiettoria di movimento, anche qui utilizzando un metodo «dead-reckoning» probabilistico chiamato Probabilistic Inertial Navigation che, a differenza di quello classico, non soffre di errori di slittamento che possono causare notevoli differenze sui risultati.

Il metodo probabilistico si differenzia da quello classico per due fattori:

- al posto di usare i valori attuali dei sensori, si considerano le misurazioni come se fossero osservazioni del movimento sottostante che verranno mappate direttamente sugli spostamenti dei veicoli

- il metodo standard calcola lo spostamento deterministicamente in base alla lettura dei movimenti, quello probabilistico proietta le previsioni di spostamento come processo probabilistico, dove una lettura di un sensore può corrispondere a diversi spostamenti con diverse probabilità.

Il modello probabilistico e quello delle traiettorie hanno bisogno di una fase di training prima di poterne stimare la probabilità condizionata, considerando gli spostamenti come informazione nascosta. I dati di training contengono diversi viaggi effettuati in auto mantenendo uno smartphone che raccogliesse dati dai sensori di movimento costantemente, per ogni viaggio è stata registrata la posizione di partenza e quella di arrivo tramite il GPS.

Dopo aver terminato il training, è stato applicato il modello probabilistico alle sequenze di lettura dei dati dei sensori e questi ultimi sono stati decodificati con una traiettoria ottimale.

Una difficile prova da superare è quella di mappare una traiettoria di movimento su un percorso all'interno di una mappa, per farlo è stato sviluppato un algoritmo di matching che da in output un punteggio che indica quanto la traiettoria si avvicina di più a un percorso sulla mappa. L'algoritmo utilizza l'approccio di Greenfield [15] che consiste nel riconoscere la posizione dell'utente più vicino a un incrocio su una mappa.

In questo studio la valutazione si divide in due esperimenti, il primo mostra come il metodo probabilistico viene realizzato tramite i dati raccolti durante la guida, il secondo mostra i percorsi che fanno match sulle mappe e le identificazioni dei punti di partenza.

Il metodo probabilistico è stato configurato usando smartphone di uso comune equipaggiati di accelerometro e magnetometro, successivamente è stato applicato ai dati grezzi dei sensori per tradurli in spostamenti e, infine ,applicando l'algoritmo di matching, è stata ricoperta la traiettoria stimata per identificare e correggere le svolte.

In conclusione si evince che l'accelerometro può essere usato per localizzare il possessore dello smartphone che lo monta, anche se gli altri sistemi di localizzazione sono disabilitati.

Nello studio condotto dall'Università di Cambridge [16], gli autori hanno dimostrato che l'accelerometro e il giroscopio dello smartphone possono essere usati per identificare viaggi significativi indipendentemente dall'orientamento del dispositivo e dalle condizioni di traffico.

La ricerca si basa sul seguente concetto: quando un veicolo è in moto, ogni svolta e inclinazione della strada causa il cambiamento della velocità angolare rilevata dal giroscopio, generando una traccia unica in forma di variazioni di velocità angolare. In questo modo, se l'utente ripete lo stesso percorso effettuato in precedenza, i sensori possono riconoscere il percorso osservando che le variazioni di velocità angolare sono simili a uno dei precedenti.

Per dimostrare l'indipendenza dell'efficacia delle rilevazioni, sono state raccolte tracce della velocità angolare da un telefono all'interno di un veicolo su un tipico tragitto da un punto di partenza A a uno di destinazione B, lungo 12 km della durata di circa 20 minuti in condizioni di traffico ordinarie. Sono stati raccolti dati anche da un altro tragitto simile dal punto A al punto C.

In entrambi i tragitti, lo smartphone è stato posizionato nella console centrale tra guidatore e passeggero, successivamente sono state registrate tracce dei tragitti posizionando il dispositivo in una posizione arbitraria. Confrontando i risultati è stato mostrato che i casi del test erano molto simili tra di loro, dimostrando che la posizione e l'orientamento del dispositivo all'interno del veicolo è ininfluente sul risultato cercato.

Per completezza, sono stati testati diversi dispositivi di diverse case produttrici (Apple, Samsung), confermando ancora una volta il risultato, il quale dimostra che anche il dispositivo diventa ininfluente sui risultati finali.

Una volta che il sistema ha identificato percorsi significativi, si possono dedurre e raccogliere informazioni più dettagliate su quel percorso, ad esempio per un utente tipico si può dedurre che il percorso effettuato più frequentemente sia il viaggio per tornare a casa. Oltre che per recuperare informazioni su un determinato percorso, esistono altri tipi di applicazione del metodo discusso che richiedono il percorso fisico attuale. Ad esempio, un'applicazione che avvisa l'utente che c'è una deviazione sulla sua strada per qualche motivo, servendosi dell'integrazione dei dati del GPS.

Un approccio interessante e sofisticato, che rispecchia l'obbiettivo dell'intero elaborato, è quello di determinare gli angoli di curvatura integrando la velocità angolare e la misurazione della differenza temporale tra una svolta e l'altra.

Nell'articolo di Zhou et al. [17] si discute di una nuova tecnica che analizza le direzioni verbali fornite da un'applicazione di navigazione che utilizza il GPS. Lo studio sostiene che, seguendo la traccia dell'*Address Resolution Protocol* memorizzata in una directory pubblica di Linux, un soggetto malintenzionato può localizzare la vittima con una buona efficacia e anche se gli speaker del dispositivo sono disattivati, si riesce a recuperare informazioni tramite l'API pubblica di Android.

Un approccio chiamato *Application Sandboxing* consente al kernel di Linux di separare l'esecuzione di un'app dalle altre, in questo modo l'app può invocare le API per accedere alle risorse di sistema di Android, così facendo le API che sfruttano risorse sensbili, sono protette da permessi richiesti direttamente all'utente.

Il modello avversario ipotizzato, esegue un'applicazione che non usa richieste di permessi all'utente, in modo da passare inosservato alla vittima e minimizzare l'impatto sulle prestazioni del dispositivo. Oltre a poter raccogliere informazioni dalle directory pubbliche, un'applicazione maligna che non usa permessi necessita di un'insieme di funzionalità per raggiungere il suo obbiettivo, in particolare necessita di un permesso per la connessione alla rete. La soluzione a questo problema sarebbe quella di sfruttare un Intent nella action view dell'applicazione per aprire il browser e caricare i dati che servono.

In questo articolo vengono descritte due tipologie di attacco alla privacy, il primo che profila i luoghi di interesse visitati spesso dalla vittima per capire quando arriva e quando lascia il posto, il secondo che raccoglie un grande numero di percorsi possibili, cercandoli per la sequenza vocale bersaglio.

Per la profilazione dei luoghi di interesse, sono prima stati trovati luoghi di partenza su Google Maps e successivamente è stata eseguita l'applicazione di simulazione, che eseguiva il percorso dalla partenza alla destinazione. Questa procedura ha estratto un insieme di sequenze di lunghezze vocali per i percorsi che portavano ai luoghi di interesse., considerate come firme sui luoghi. Per ogni sequenza ricevuta dall'app sono state estratte delle sottostringhe alla fine delle sequenze, la sottostringa era l'ultimo passo del percorso rilevato dall'applicazione. Se la sottostringa combaciava una qualunque sequenza, si poteva avere la certezza che il dispositivo si trovasse in uno dei luoghi profilati.

Per quanto riguarda la raccolta di percorsi per le sequenze, è stato considerato il fatto che l'abitazione e i luoghi frequentati spesso dall'utente possono essere localizzati quando il dispositivo è abilitato a connettersi automaticamente. E' sufficiente conoscere il punto di partenza o quello di destinazione per recuperare le informazioni necessarie, infatti per andare da una città all'altra di solito si percorrono le medesime strade, così le sequenze vocali possono essere usate per localizzare il punto di entrata nella città di destinazione, considerando la sequenza registrata durante il tragitto.

Utilizzando un *crawler* dell'API Google, è stato possibile scaricare diversi percorsi che collegavano un punto conosciuto agli indirizzi di residenza in una determinata area.

In conclusione, in questo articolo è stato rivelato che le informazioni sensibili degli utenti memorizzate nello smartphone possono essere recuperate ed elaborate analizzando

le applicazioni più popolari.

Lo studio di Michalevsky et al. [18] sostiene che leggendo i dati sul consumo di energia di uno smartphone in un periodo di pochi minuti, un'applicazione può recuperare informazioni sulla localizzazione del dispositivo, anche se il consumo stesso è parecchio influenzato dalla moltitudine di altri componenti e applicazioni che lo attivano contemporaneamente, ma la soluzione a questo problema sarebbe quella di usare algoritmi di machine learning.

Per l'esperimento sono stati usati tre tipi di obbiettivi di tracciamento degli utenti:

- Distinguibilità del percorso: si cerca di capire se un attaccante può indovinare che strada percorrerà l'utente davanti a un insieme di possibili percorsi.
- Tracciamento del movimento real-time: assumendo che l'utente percorrerà una strada conosciuta, si cerca di capire se un attaccante sia in grado di riconoscere la posizione dell'utente lungo il percorso, in tempo reale.
- Deduzione di nuove strade: supponendo che l'utente stia percorrendo un lungo percorso, ci si chiede se un attaccante riesca a riconoscere il percorso attraverso le misurazioni del consumo di energia effettuate in precedenza su molti percorsi più corti nella stessa area.

La localizzazione del dispositivo avviene assumendo che l'attaccante abbia già una conoscenza di base sull'area o sui percorsi che la vittima usa di frequente, questi dati sono stati usati per integrarli alle misurazioni del consumo di energia e nonostante l'elevato rumore sulle misurazioni, sono stati ottenuti risultati ottimi.

Per capire la vulnerabilità del consumo di energia, sono stati effettuati dei test anche sulla potenza del segnale di ricezione, percorrendo due volte lo stesso percorso, il segnale presenta lo stesso andamento nelle stesse posizioni. Successivamente, è stato mostrato che le misurazioni del consumo di energia rivelano un unico pattern stabile per un determinato percorso. Contrariamente alla potenza del segnale, il consumo di energia tende a essere meno regolare perché influenzato dalla reazione del modem ad un cambio di potenza del segnale.

Infine, è stato dimostrato, attraverso test in cui sono stati utilizzati differenti modelli di smartphone, che il consumo di energia rimane stabile a prescindere dal dispositivo usato per la raccolta dei dati.

Con un discreto numero di applicazioni attive durante la raccolta dei dati, i risultati ottenuti sono accurati su 2/3 degli scenari analizzati, mentre attivando anche applicazioni come Facebook [19] o Skype [20], l'accuratezza è scesa ad 1/5.

Capitolo 2

Motivazioni e Modello

In questo capitolo si discuterà delle motivazioni che hanno portato alla concretizzazione dell'elaborato, inoltre verrà illustrato e descritto il modello utilizzato in questo studio, in particolare uno script scritto in Python [10] che ha permesso la riuscita dei vari test ed esperimenti.

Contrariamente a quanto effettuato da altri studi menzionati in precedenza, che utilizzavano una sperimentazione usando test reali, questo elaborato presenta direttamente le fasi di simulazione ed analisi. La simulazione viene effettuata tramite il modello che verrà descritto in seguito, generando dei percorsi possibili su una mappa reale e introducendo vari gradi di rumore sulle misurazioni. In questo modo viene simulata la guida di un veicolo all'interno di una città da un determinato utente, infine l'analisi si effettua sui dati ottenuti attraverso la simulazione.

2.1 La diffidenza degli utenti

In una realtà che ormai gode di una rapida e grande espansione riguardante reti di telecomunicazione e tecnologie utilizzate per sfruttarle al meglio, sono sempre di più gli utenti che utilizzano il proprio smartphone come dispositivo di navigazione, in sostituzione agli ormai «obsoleti» navigatori satellitari. Questo cambio di abitudini è dovuto alla forte diffusione delle mappe sviluppate da Google (Google Maps)[11] e dalla localizzazione GPS, due elementi presenti ormai nella quasi totalità dei dispositivi mobili in commercio.

Ragionevolmente, quando si parla di dati di localizzazione di un dispositivo, si va incontro a numerosi dubbi riguardanti l'effettivo utilizzo di quei dati da parte di «autorità» o «hacker». L'utilizzatore è preoccupato a tal punto da rinunciare all'utilizzo della localizzazione e delle sue funzioni o limitarsi all'uso solo in caso di necessità, ma questa diffidenza non risulta del tutto infondata considerando quanto emerso dagli studi citati in precedenza su questa tematica. Purtroppo, o per fortuna, il GPS non è l'unico siste-

ma di localizzazione, sebbene sia il più preciso e utilizzato, questo studio discute infatti dell'efficacia di metodi alternativi per la localizzazione di un possessore di smartphone.

Un esempio significativo che giustifica la diffidenza degli utenti è quello della cronologia di Google Maps, una funzionalità che consente di tenere traccia degli spostamenti dei dispositivi collegati a un determinato account attraverso l'accesso alla localizzazione continua, condizione che può essere concessa dall'utente anche in maniera inconsapevole. Google utilizza algoritmi di machine learning che classificano gli spostamenti frequenti, in modo da fare previsioni e fornire suggerimenti che un utente inesperto potrebbe interpretare come maligni, poiché la domanda spontanea sarebbe quella di capire da dove vengano quei suggerimenti.

2.2 Motivazioni reali

La motivazione più forte che ha indotto questo studio è certamente quella di cercare di garantire un certo equilibrio tra invasione della privacy e utilità della localizzazione stessa.

Sicuramente la localizzazione fornisce uno strumento molto utile all'utente finale, garantendo la stessa funzionalità di un navigatore satellitare su un semplice smartphone dotato di una connessione internet. Per non parlare dell'abbattimento dei costi sugli abbonamenti, considerando che le mappe utilizzate comprendono l'intero pianeta e sono completamente libere.

D'altro canto, fornire costantemente la posizione a entità sconosciute, le quali nessuno sa che uso ne facciano, è comprensibilmente motivo di fastidio e malcontenti tra gli utenti che potrebbero vedere come un'inosservanza delle regole sulla privacy.

Tuttavia, l'utilizzo dei servizi di localizzazione necessita di un permesso specifico da parte dell'utente che può decidere se attivare o meno, ciò significherebbe che i dati relativi alla posizione del dispositivo diventerebbero inaccessibili se l'utente non ne consente l'accesso. Se invece si prendono in considerazione altri servizi presenti su un dispositivo, come ad esempio i servizi di monitoraggio esterno (accelerometro, magnetometro, giroscopio), l'utente rischia di incorrere in un maggiore rischio in quanto i dati dei sensori non necessitano di nessun permesso da parte dell'utente. Chiaramente, il tracciamento attraverso i sensori diventerebbe molto più impreciso e instabile, a causa della forte influenza di fattori esterni (traffico, pianta della città, stile di guida).

Questo argomento è il tema principale e quello su cui si basa tutto l'elaborato, condurre test che mettono in luce quanto sarebbe a rischio di intercettazione un utente a sua insaputa.

2.3 Metriche

Le metriche utilizzate per questo studio si basano sui dati rilevati dai sensori di uno smartphone, poiché l'obiettivo dell'esperimento è quello di capire se esistono metodi alternativi al GPS che consentano la tracciabilità di un utente da parte di malintenzionati. L'idea è quella di installare un'applicazione su un dispositivo bersaglio e accedere ai dati di magnetometro e accelerometro che, come già emesso da altri studi condotti su questa tematica, non necessitano di permessi rilasciati dall'utente.

A causa di mancanza di risorse, si è reso necessario simulare la raccolta dei dati attraverso un algoritmo che genera dei percorsi reali prendendo due nodi dal grafo della città che si desidera analizzare, questo algoritmo estrae i gradi di curvatura e le lunghezze di ogni arco con una specifica funzione di libreria e le memorizza su una lista apposita.

Ciò che vuole simulare l'algoritmo è la raccolta di dati attraverso un'applicazione apposita installata sullo smartphone di una possibile vittima, sfruttando l'accelerometro per determinare l'inizio e la fine di una strada e il magnetometro per rilevare l'orientamento del dispositivo che darà luce al grado di curva effettuato ad un incrocio.

Le città sono state analizzate sia con l'uso dei soli gradi di curvatura, sia con l'uso di entrambe le metriche, facendo emergere risultati notevolmente diversi che verranno discussi in seguito.

2.4 Modello

Per testare e sperimentare lo scenario descritto in precedenza è stato sviluppato uno script in Python, servendosi di alcune librerie ad hoc per il problema in questione.

L'approccio iniziale consiste nella costruzione di un grafo diretto e non orientato per ogni città da analizzare, contenente informazioni su strade e incroci, come lunghezza delle strade e curvatura. Per realizzare i grafi in questione è stata utilizzata una libreria molto utile, *OSMnx* [12] che utilizza le mappe di OpenStreetMap per la costruzione dei grafi e l'acquisizione delle informazioni.

Successivamente, si esplora il grafo e si costruiscono diversi percorsi su di esso attraverso *NetworkX* [13], libreria utile per la costruzione e la ricerca di percorsi nel grafo sfruttando le informazioni di archi e nodi. Ciò serve a simulare i possibili percorsi effettuati dagli utenti e cercare di capire quanto siano riconoscibili attraverso metodi alternativi al GPS.

Il passo successivo alla costruzione dei percorsi è quello della ricerca degli stessi nel grafo della città, esplorando ogni nodo.

Questo approccio è descritto da due differenti algoritmi, la visita del grafo e la ricerca dei percorsi con conseguente estrapolazione delle curvature degli archi all'interno di esso.

Algoritmo 2.1 Algoritmo di visita per trovare le curve

```
def getRandomPathBearing(){
        found = false
        while not found{
                node1 = random_choice(g.nodes)
                node2 = random_choice(g.nodes)
                route = shortest_path(g, node1, node2)
                if 1 < len(route) < 25 then found = true
        }
        foreach bearing in route{
                path.append(bearing)
        }
        return path

}
```

L'algoritmo di visita non prende dati in input, poiché utilizza variabili globali, mentre restituisce in output un array contenente i gradi di curvatura di ogni arco presente nel percorso analizzato. La funzione scansiona un dizionario globale in cui sono memorizzati gli archi con le relative curvature e salva il dato all'interno dell'array da dare in output come lista ordinata. Per simulare uno scenario verosimile sono stati presi in considerazione percorsi di al massimo 25 incroci attraversati, questa scelta è stata attuata perché più i percorsi sono lunghi, più alta sarà la probabilità che il percorso sia unico e quindi riconoscibile.

Il test viene ripetuto 2000 volte, ottenendo così 2000 percorsi differenti da poter dare in input all'algoritmo di ricerca.

Lo stesso algoritmo è stato riadattato per sfruttare una metrica diversa di analisi, quella delle lunghezze degli archi, che si aggiunge ai gradi di curvatura per confrontarne l'andamento e l'accuratezza.

Algoritmo 2.2 Algoritmo di visita per trovare le lunghezze

```
def getRandomPathLength(random1, random2){
        while not found{
                route = shortest_path(g, random1, random2)

        }
        foreach length in route{
                path.append(bearing)
        }
        return path

}
```

Il secondo algoritmo principale è quello che riguarda la ricerca dei percorsi generati nel grafo della città in input, si tratta di un algoritmo di visita in ampiezza sui grafi (BFS) modificato opportunamente per il caso specifico.

Questo algoritmo prende in input il grafo della città, il nodo sorgente, l'array delle curve e quello delle lunghezze (all'occorrenza), inizializza le strutture dati adatte per la visita di un grafo, cioè un array di booleani per tenere traccia dei nodi visitati e di una coda come struttura ausiliaria. Per ogni elemento presente nell'array delle curve si effettua un ciclo sui nodi adiacenti a quello estratto dalla coda, dentro al ciclo viene fatto un controllo sulla corrispondenza tra la curva e/o la lunghezza dell'arco esaminato e quelle degli array in input, sul controllo viene aggiunto un range di rumore passato come argomento al momento dell'esecuzione dello script per simulare una raccolta di dati reale tramite un'applicazione installata su uno smartphone. Se l'arco esaminato viene riconosciuto perché ricade nell'intervallo, allora si aggiunge il nodo corrente ad un array che verrà restituito in output e si marca il nodo come visitato, l'array in output è una lista ordinata di nodi che formano un possibile percorso sulla mappa.

Le funzioni utilizzate per recuperare gli attributi degli archi sono funzioni della libreria OSMnx, mentre le funzioni usate per scorrere i nodi adiacenti a quello corrente e per verificare che il percorso sia reale sulla mappa, fanno parte della libreria NetworkX.

Algoritmo 2.3 Algoritmo di ricerca

```
def search(graph, bearing, source, length){
        inizializzo la coda e il vettore di booleani
        foreach item in (bearing, length){
                u = queue.pop()
                foreach node in u.adj() {
                        if node.bearing <= noise {
                                if node.length <= noise {
                                        path.append(node)
                                        queue.append(node)
                                }
                        }
                }
        }
        controllo che il percorso sia valido se non lo è return 0
        return path
}
```

L'algoritmo viene richiamato all'interno di un ciclo in cui si scorrono tutti i nodi del grafo, che vengono passati in input all'algoritmo stesso per essere utilizzati come nodo sorgente, ciò significa che ogni nodo viene analizzato 2000 volte, poiché è il numero dei percorsi simulati tramite l'algoritmo precedente, in questo modo la complessità dell'algoritmo aumenta esponenzialmente a causa dei diversi cicli annidati.

Ogniqualvolta, l'algoritmo ritorna un percorso trovato, lo script principale aumenta il valore di una variabile contatore che rappresenta un percorso riconosciuto. Successivamente, scrive una riga su un file CSV in cui descrive:

- la città analizzata
- il noise utilizzato
- il numero di svolte effettuate in quel percorso
- il numero di percorsi riconosciuti, nonché valore del contatore
- il numero di nodi del grafo

I file CSV scritti contengono tutte le informazioni necessarie per svolgere un'analisi sull'andamento e la previsione ottenendo i risultati che serviranno per trarre le conclusioni della tesi in seguito.

Capitolo 3

Analisi e risultati

In questo capitolo si descriveranno le analisi effettuate sui dati raccolti dalle simulazioni e se ne discuteranno i risultati, mettendo in luce differenze con gli altri studi menzionati in precedenza commentandole.

L'analisi è stata effettuata utilizzando i file CSV generati dalle simulazioni, creando grafici che mostrano gli andamenti dei valori in base alle due diverse metriche descritte prima.

Per le analisi sono state scelte 10 città per 3 macrogruppi, 5 città italiane, 5 europee e 5 città statunitensi. Le scelte si sono basate sulla grandezza delle mappe e sul tipo di pianta che le distingue,

- per l'Italia sono state prese: Bologna, Palermo, Napoli, Torino e Milano
- per l'Europa: Nizza, Ginevra, Siviglia, Manchester e Colonia
- per gli Stati Uniti: Manhattan, Miami, Downtown Los Angeles, San Francisco e Boston

3.1 Analisi secondo le curve

Per poter ottenere un'analisi che si avvicina il più possibile alla realtà, sono stati introdotti diversi scaglioni di rumore sui dati, facendo emergere andamenti molto diversi. Ovviamente, più è alto il valore del rumore più sono i percorsi che combaciano con le curve date in input. I valori del rumore vanno da 5 a 20 sia in positivo che in negativo a scaglioni di 5, in questo modo si possono simulare gli smartphone più datati con valori elevati di rumore e quelli più recenti e completi con un valore di rumore minimo.

3.1.1 Città degli Stati Uniti

Una prima analisi introduttiva mostra l'andamento delle medie delle città USA ordinate per numero di nodi con i diversi scaglioni di rumore :

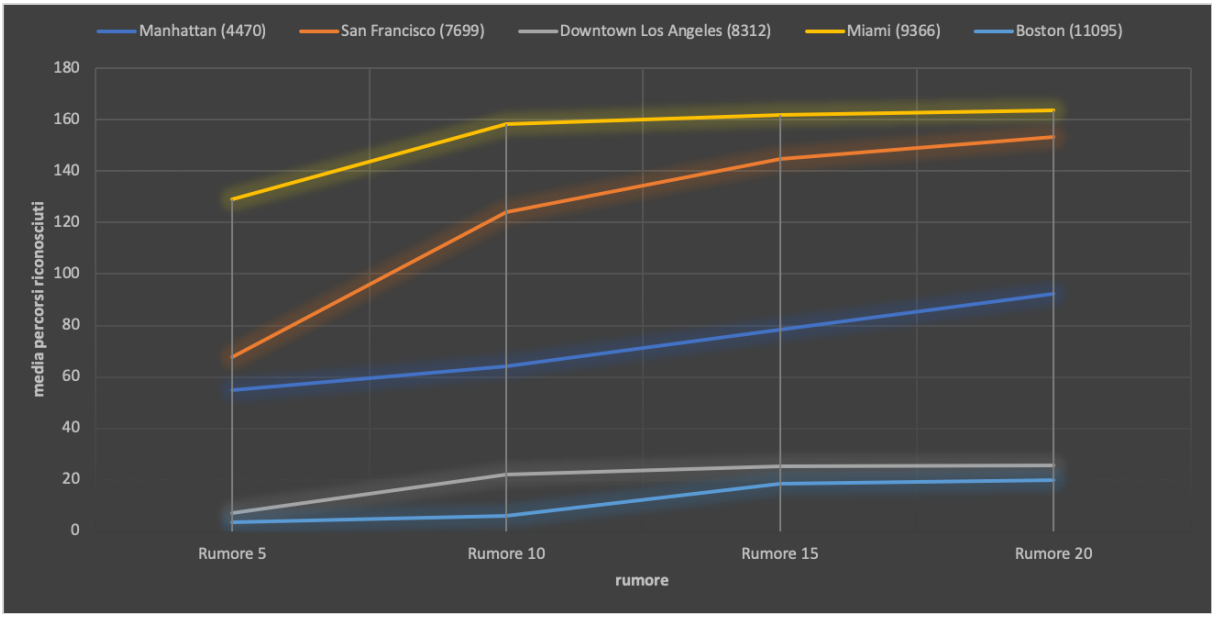


Figura 3.1: Andamento delle medie delle città statunitensi

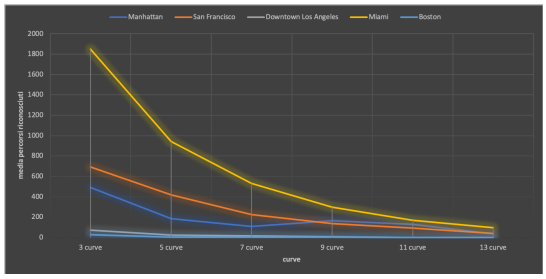
Da questo grafico si può notare che la media non è proporzionale rispetto al numero dei nodi, infatti Boston risulta avere la media più bassa, nonostante sia la città con il maggior numero di nodi. Ciò significa che il fattore di maggiore influenza sui percorsi riconosciuti non è la grandezza della città, ma bensì la pianta della rispettiva mappa. La maggior parte delle città degli Stati Uniti hanno una pianta a scacchiera, cioè con numerosi incroci perpendicolari l'uno all'altro, questa distribuzione uniforme delle intersezioni causa una difficoltà nel riconoscimento.

Interessante è la differenza tra l'andamento con rumore +/- 5 e quello con rumore +/- 10, poiché alcuni valori tendono a raddoppiare o triplicare, mentre altri subiscono una variazione di poche decine. Le variazioni più sostenute rimangono quelle di Boston con un raddoppio e Downtown Los Angeles con un valore più che triplo rispetto al precedente. Si noti la particolarità della mappa di Boston, che rispecchia maggiormente le città europee, per questo motivo, partendo da bassi valori, le variazioni risultano più elevate.

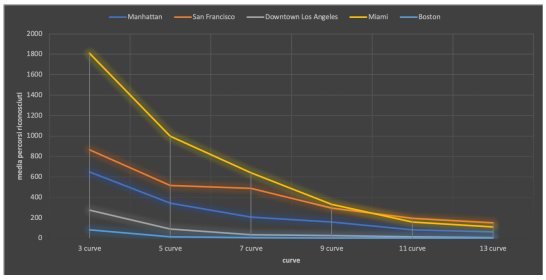
Con un rumore +/- 15, il valore della media di Boston risulta triplicato, mentre gli altri valori subiscono una crescita di poche decine, e anche passando all'ultimo scaglione di rumore i risultati restano per lo più stazionari. Questo avviene perché, pur aumentan-

do il rumore, i percorsi riconoscibili rimangono gli stessi, in quanto agli incroci le svolte possibili saranno sempre le stesse.

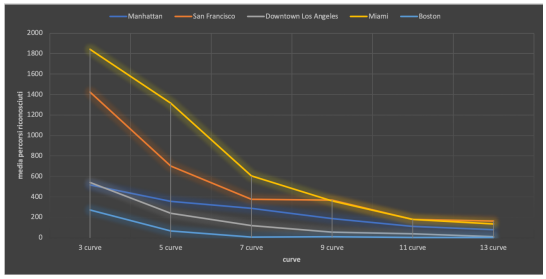
I valori delle medie vanno però valutati con cautela, poiché possono risultare elevati a causa di pochi casi in cui sono stati riconosciuti parecchi percorsi simili, questi casi sono quelli in cui il numero di svolte effettuate dal guidatore sono contenute. I grafici seguenti mostrano come vengono influenzate le medie generali tramite la divisione in cluster delle medie per il numero di curve effettuate.



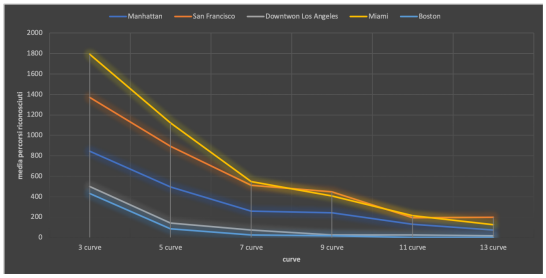
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.2: Media dei percorsi riconosciuti per cluster di curve effettuate

Dai grafici emergono alcuni dati interessanti, infatti come già accennato la media generale è parecchio influenzata dai valori dei cluster iniziali in qualunque scaglione di rumore, ad eccezione di Boston e Downtown Los Angeles che appaiono come funzioni lineari. I valori massimi vengono raggiunti da Miami fin dal rumore più contenuto, in questo caso particolare i valori dei cluster non sono proporzionati all'aumento del valore di disturbo dei dati, infatti soprattutto i primi cluster si mantengono su valori superiori a 1000, mentre gli ultimi hanno valori comunque superiori al centinaio. Questo dato rende il riconoscimento dei percorsi parecchio difficile nella maggior parte delle città americane, eccetto quelle con piante riconducibili a quelle europee.

Le figure sottostanti mostrano queste differenze.

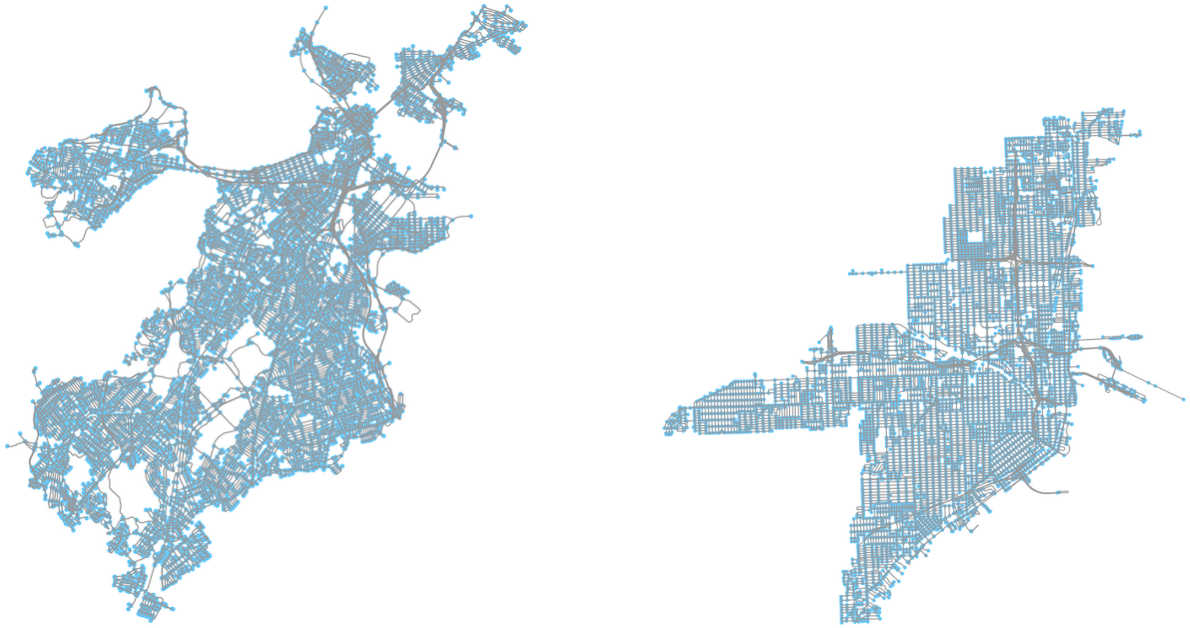
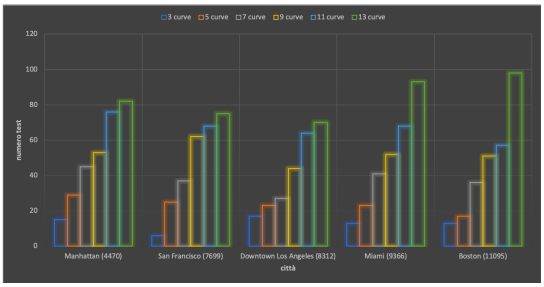


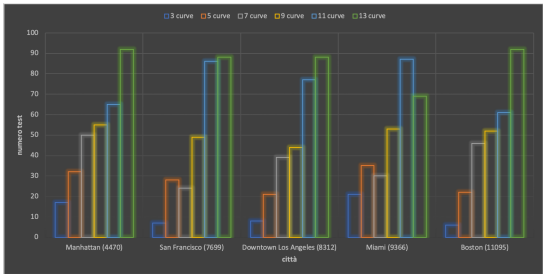
Figura 3.3: Pianta di Boston a sinistra, a destra quella di Miami

Le piante in figura 3.3 presentano differenze sostanziali, la pianta di Miami risulta prevalentemente a scacchiera, il che comporta un aumento dei percorsi riconosciuti specialmente su poche curve effettuate. In una pianta del genere, un percorso più breve corrisponde ad avere risultati simili tra loro. In una pianta come quella di Boston, prevalentemente radiocentrica e irregolare, anche se i percorsi ricercati sono brevi sono maggiormente distinguibili fra loro, in questo modo i valori medi si abbassano notevolmente.

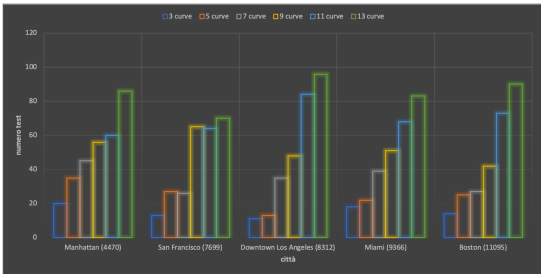
A completamento di quanto detto sull'analisi condotta finora, vengono mostrati i grafico che raccolgono il numero di volte in cui si ricerca un percorso con un certo numero di curve per ogni città.



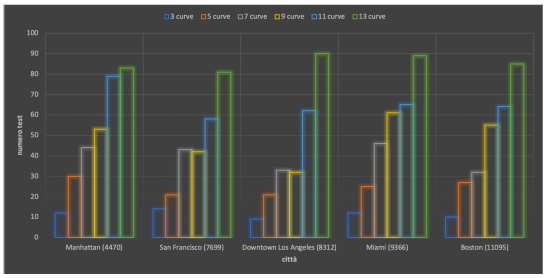
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.4: Numero di volte che viene ricercato un percorso con n curve nei vari scaglioni

I grafici evidenziano che, per Boston e Miami, si ha il maggior numero di volte in cui sono stati riconosciuti percorsi con un numero di curve superiore a 10, questi dati avranno per lo più valori vicini ad uno, ciò significa che sono i valori responsabili del decremento delle medie. Nonostante questo la media generale di Miami rimane molto alta, ciò significa che sono stati riconosciuti molti più percorsi con poche curve anche se ne sono stati ricercati meno. L'analisi dimostra che i risultati non sono falsati dal numero di percorsi ricercati con poche curve, poiché il fattore che influisce maggiormente è il riconoscimento degli stessi.

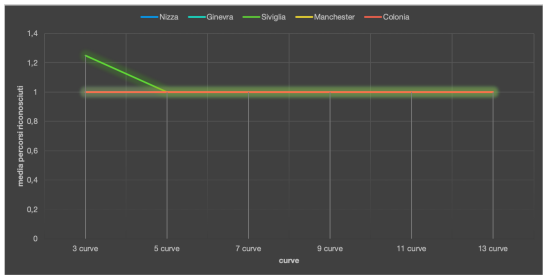
3.1.2 Città europee

In questa sezione si discuteranno le differenze di comportamento dei valori medi per i 4 scaglioni di rumore usati, per iniziare vengono mostrate le medie generali.

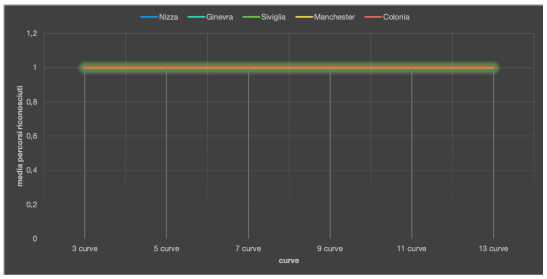
Figura 3.5: Andamento delle medie generali ordinate per numero di nodi

Dalla figura 3.5 si nota come l'andamento abbia un comportamento lineare, tendente sempre a valori vicini ad 1, ciò evidenzia una notevole differenza con le città statunitensi che presentavano medie piuttosto elevate. Questo significa che la metrica applicata a questa analisi è abbastanza rigida e i percorsi cittadini risultano per la maggior parte unici, con incroci dalle angolazioni molto differenti tra loro.

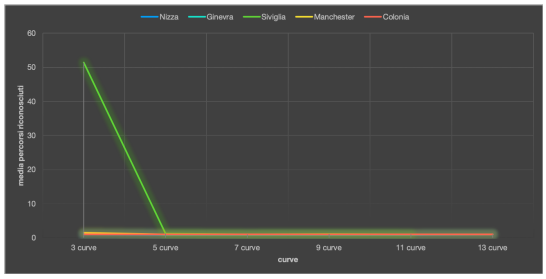
I grafici seguenti mostrano come vengono influenzate le medie generali.



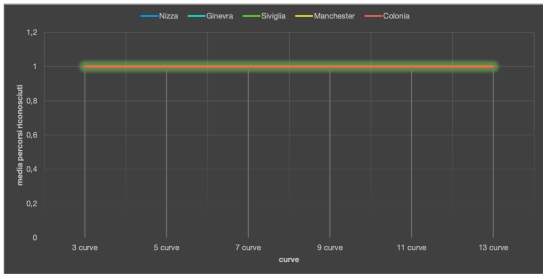
Rumore 5



Rumore 10



Rumore 15



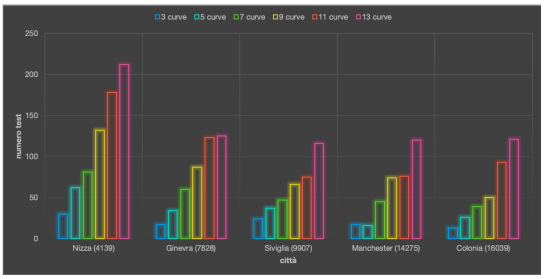
Rumore 20

Figura 3.6: Andamento delle medie per cluster di curve

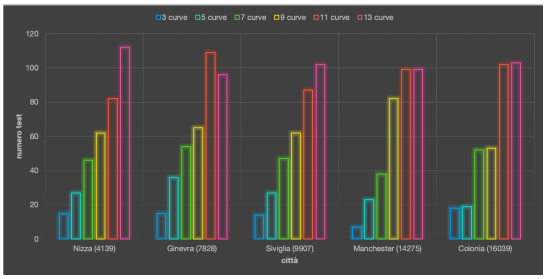
Qui si può vedere che anche la divisione in cluster non evidenzia particolari differenze, gli andamenti rimangono sempre lineari e in alcuni scaglioni di rumore addirittura delle rette stazionarie sul valore 1.

Ciò significa che le città prese in esame, nonostante siano tutte di stati diversi, possiedono tutte lo stesso stile di architettura e gli stessi tipi di pianta, il che comporta una notevole riduzione del rischio di un possibile attacco alla privacy.

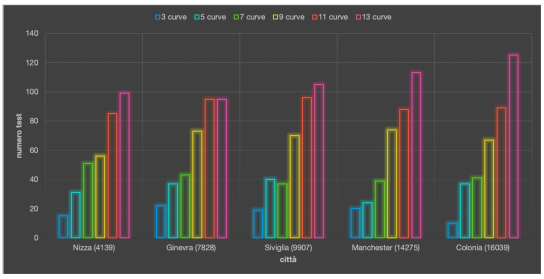
L'unica città che si distingue, ma solo per il primo cluster, è Siviglia con una media che supera 50 per il primo cluster con rumore +/- 15.



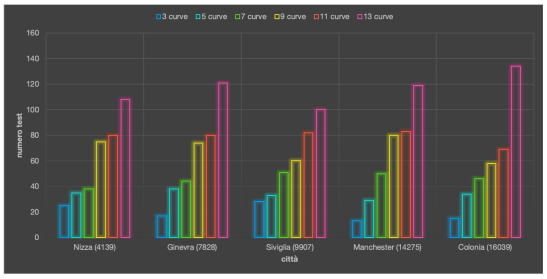
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.7: Numero di test effettuati per ogni cluster

Da questi grafici si può concludere che il numero di test effettuati per ogni cluster rimane proporzionato alle medie ottenute per tutti gli scaglioni di rumore, questo significa che non vi è un vizio sui valori medi causato dal numero di test. La città con un maggior numero di test è Nizza nel primo scaglione, la quale però mantiene sempre una media lineare e uniforme.

3.1.3 Città italiane

Per poter confrontare i risultati tra diversi tipi di città e diversi tipi di architetture, sono state prese in considerazione alcune città italiane, in base sia all'area geografica di appartenenza (nord, sud), sia in base al numero di nodi della mappa. Come prima viene mostrato inizialmente un andamento generale evidenziando i diversi scaglioni di disturbo sui dati.

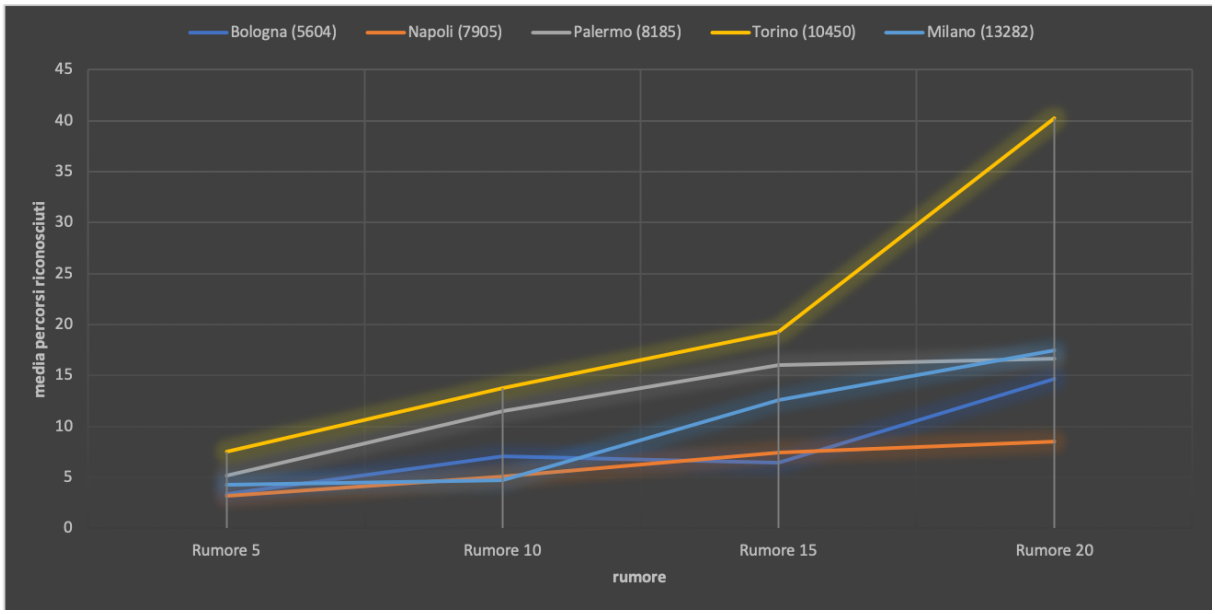


Figura 3.8: Andamento delle medie per le città italiane

La figura 3.8 suggerisce che la città con il valore medio più alto è Torino, nonostante non sia la città più grande per numero di nodi. Ancora una volta il motivo è molto chiaro, la pianta della città di Torino è a scacchiera e somiglia molto allo stile statunitense. A seguire si trova subito Palermo, essa presenta una pianta mista, con zone di stile antico in prevalenza a scacchiera ed altre in prevalenza radiocentrica e irregolare. Le città rimanenti hanno valori tra loro abbastanza vicini, soprattutto Bologna e Napoli, dove Napoli è leggermente più grande nel numero di nodi.

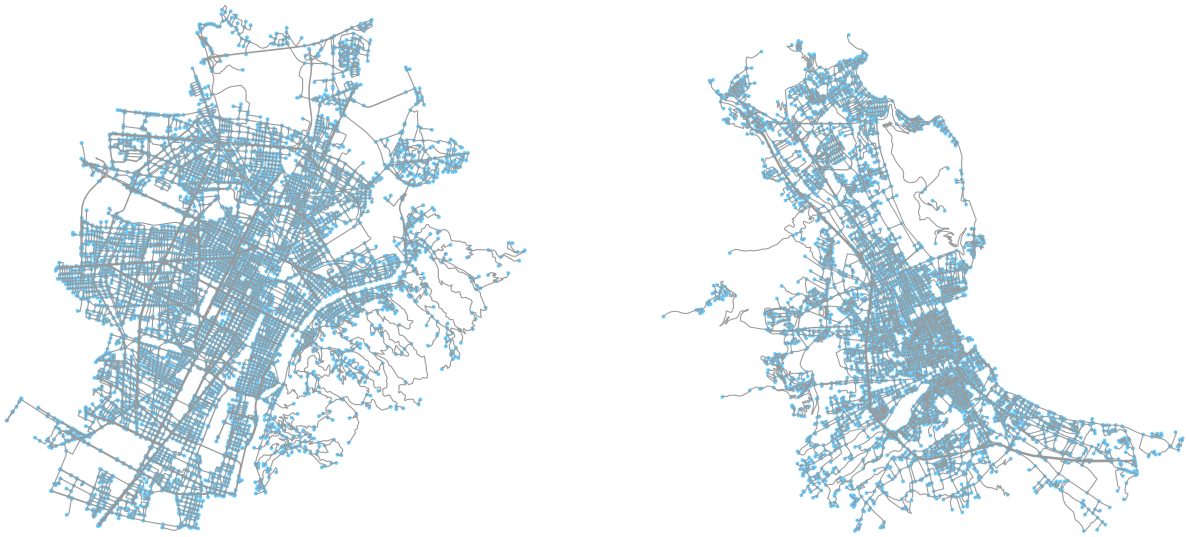
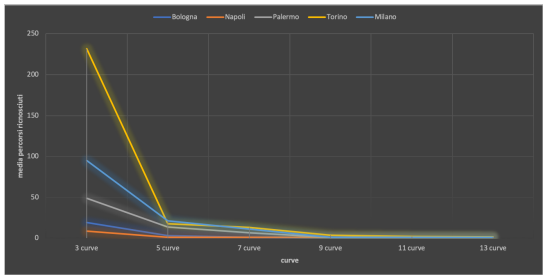


Figura 3.9: Pianta di Torino a sinistra, quella di Palermo a destra

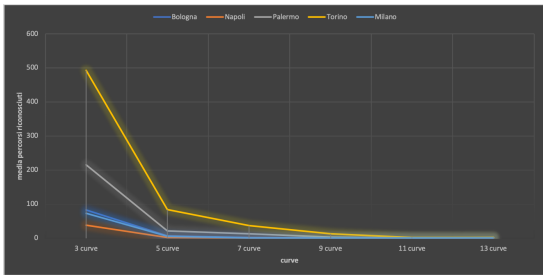
Come si può notare dalla figura 3.9, Palermo ha una pianta in prevalenza a scacchiera nella parte del centro storico e in quella adiacente alle acque, mentre nelle zone periferiche presenta percorsi meno lineari e questo influenza i valori dei risultati, abbassando la media generale rispetto a Torino.

La città con il valore più basso, Napoli, presenta una pianta totalmente irregolare con conseguente maggiore probabilità di trovare percorsi unici, specialmente con un ridotto tasso di rumore sulle rilevazioni.

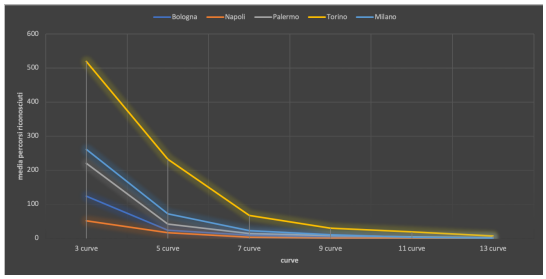
Come prima, viene mostrato un grafico che mette in evidenza i valori medi per cluster di curve, confermando anche per le città italiane che il minor numero di curve combacia con i valori più alti.



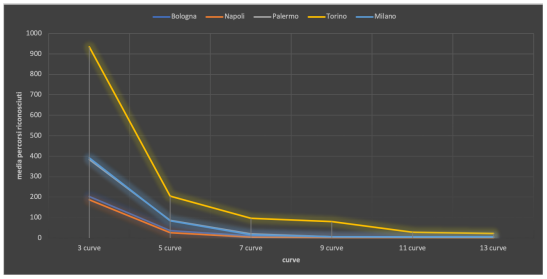
Rumore 5



Rumore 10



Rumore 15



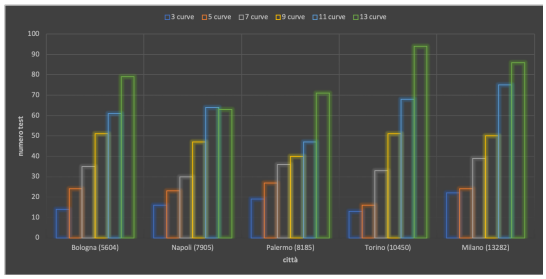
Rumore 20

Figura 3.10: Media dei percorsi per cluster di curve

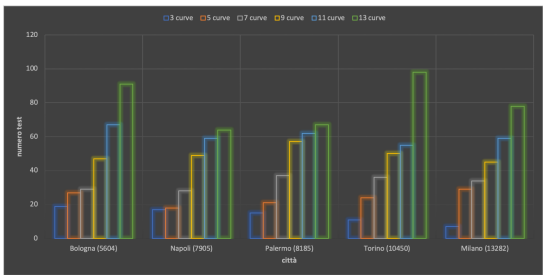
La figura 3.10 giustifica e completa i valori delle medie generali, infatti l'andamento di ogni città si può considerare come una funzione decrescente, man mano che le curve di un percorso aumentano, diventa sempre più facile riconoscerlo unicamente. Attraverso i grafici, si possono notare le differenze tra Torino e Napoli che confermano quanto detto prima sulle medie generali, infatti i valori di Napoli somigliano più ad una funzione lineare e regolare nei primi 3 scaglioni di rumore.

Nei vari scaglioni, l'andamento delle funzioni rimane molto simile con ovvi slittamenti dei valori delle ordinate in crescendo, ma si nota una comportamento particolare, più il rumore aumenta e più le funzioni tendono a unificarsi e avvicinarsi. Questo dimostra che, per dati molto disturbati, la pianta della città diventa poco influente, anche se i valori delle città con piante simili si avvicinano a tal punto da risultare quasi uguali.

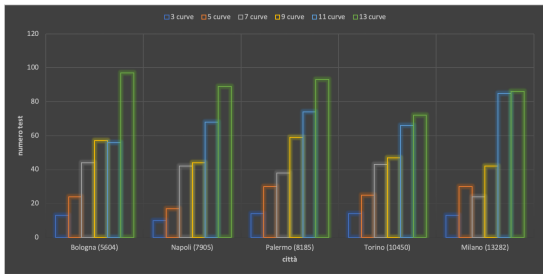
E' interessante vedere come sull'asse delle ordinate, tendendo all'infinito, i valori di ogni città sembrano fondersi in un unica funzione, questo accade perchè le piante delle città non sono in relazione alle curve effettuate in ogni percorso.



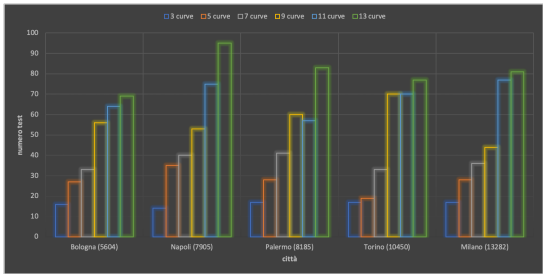
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.11: Occorrenze di percorsi ricercati con n curve effettuate nei diversi scaglioni

La figura 3.11 completa l'analisi per le città italiane, evidenzia che i percorsi ricercati con poche curve sono sempre meno rispetto a quelli con molte curve, questo conferma che bastano pochi tentativi per accorgersi che riconoscere un breve percorso con l'utilizzo di una sola metrica è assai difficile. Al contrario, per i lunghi percorsi, pur facendo svariati tentativi, si mantengono hanno ugualmente buoni risultati di riconoscimento.

Il primo risultato che si può notare dalle analisi svolte è che, in generale, i percorsi effettuati nelle città americane sono notevolmente più difficili da riconoscere rispetto a quelli effettuati nelle città italiane. Infatti, anche con dati poco disturbati, il valore massimo delle medie delle città italiane è 7,5135 di Torino, mentre il valore massimo per quelle americane è 129,1995 di Miami, cioè il 1700% in più rispetto a Torino. In poche parole, riconoscere un percorso a Miami senza l'ausilio di sistemi di localizzazione è pressoché impossibile.

Rimane comunque l'eccezione di Boston, che essendo una città in stile molto europeo, mostra gli stessi risultati dei valori minimi italiani con una media di 3,2875 contro il 3,2125 di Napoli.

3.2 Una metrica in più: la lunghezza dei segmenti

Fino ad ora, l'analisi fatta si è basata esclusivamente sulle curve effettuate in ogni percorso, mostrando risultati positivi per quanto riguarda il riconoscimento, ma non abbastanza per avere una certezza assoluta anche con un rumore minimo. Per questo motivo è stata

introdotta una metrica aggiuntiva per il riconoscimento dei percorsi, cioè la lunghezza di ogni segmento. Questa metrica simula i dati che verrebbero rilevati dall'accelerometro al variare dell'accelerazione del veicolo, anche se servirebbe comunque un metodo ausiliario per dedurre la posizione di partenza.

3.2.1 Città americane

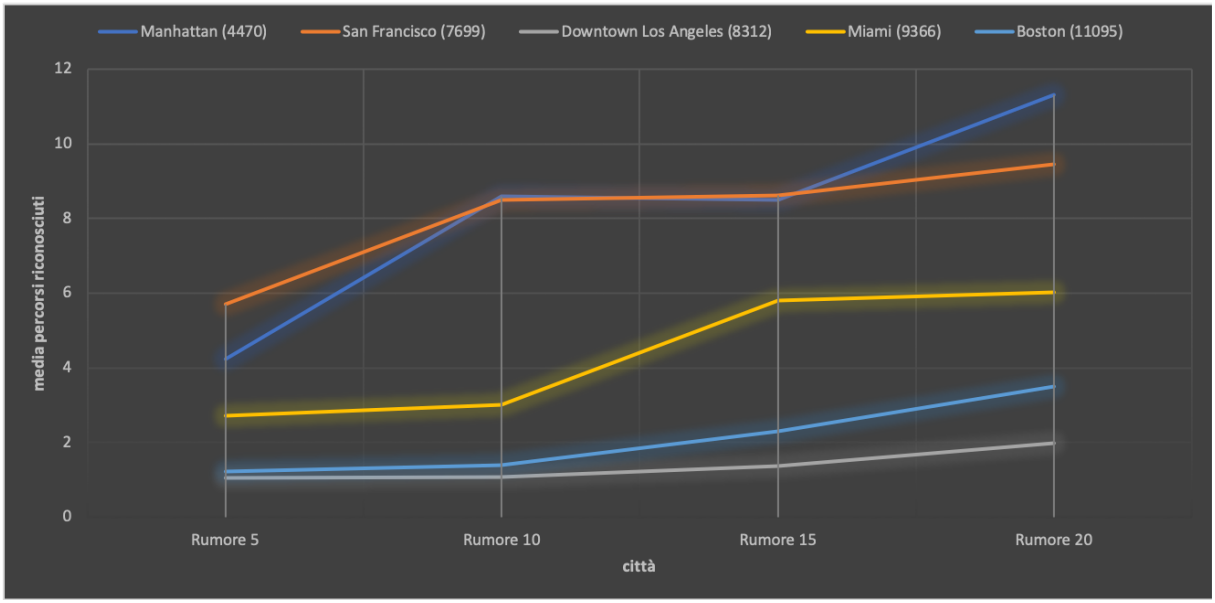


Figura 3.12: Andamento delle medie generali in ordine crescente di rumore

La figura 3.12 da uno sguardo d'insieme a come cambia il valore medio generale di ogni città, le città sono ordinate in modo crescente nel numero dei nodi.

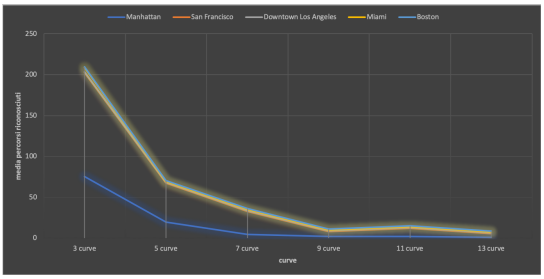
A differenza di quanto accadeva effettuando le analisi con un unica metrica, in questa analisi non è Miami a presentare un valore più alto, si alternano invece le città di San Francisco e Manhattan. Questo significa che le due città hanno molte strade con lunghezze simili tra loro, il che complica il riconoscimento dei percorsi anche a ridotti tassi di rumore sui dati.

Sorprendentemente, il valore più basso lo detiene Downtown Los Angeles che mostra una crescita molto contenuta se paragonata alla crescita del rumore, mentre Boston subisce una variazione del 300% nell'ultimo scaglione di rumore. Il dato riguardante Downtown Los Angeles evidenzia una diversificazione molto accentuata della rete stradale.

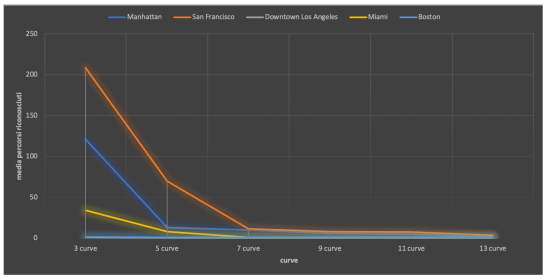
Figura 3.13: A sinistra la pianta di Downtown Los Angeles, a destra quella di Manhattan

La figura 3.13 mette a confronto la pianta di Manhattan con quella di Downtown Los Angeles che risulta essere una mappa molto particolare, poiché a prima vista potrebbe sembrare potrebbe sembrare una normale pianta a scacchiera, ma in realtà presenta molte differenze di grandezza degli isolati e di orientamento. In questa mappa, ogni isolato ha una grandezza diversa e alcuni blocchi sono orientati diversamente rispetto agli altri, ciò significa che la lunghezza delle strade è molto diversa tra loro, motivo per cui Downtown Los Angeles detiene il valore più basso, come accennato in precedenza.

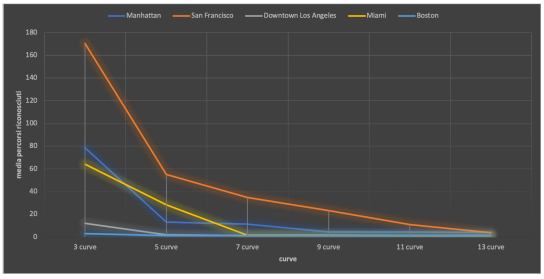
Un'analisi significativa, che può evidenziare al meglio come vengono influenzate le medie generali, è quella delle medie per cluster di curve che saranno discusse di seguito.



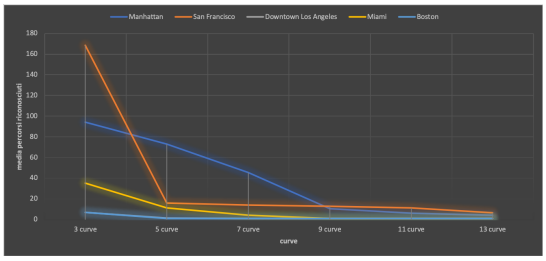
Rumore 5



Rumore 10



Rumore 15

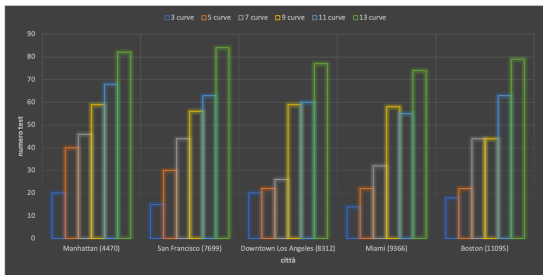


Rumore 20

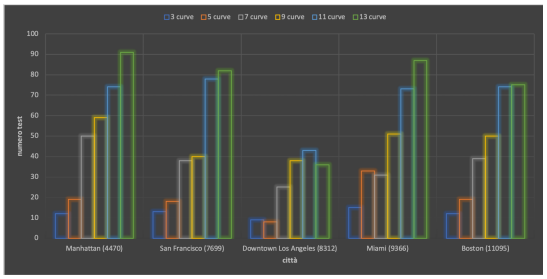
Figura 3.14: Medie di percorsi riconosciuti per cluster di curve

La figura 3.14 mette in risalto come vengono influenzate le medie generali, a ridotti tassi di rumore si può notare come gli andamenti siano uniformi per tutte le città eccetto che per Downtown Los Angeles, che pur mantenendo lo stesso andamento parte da valori leggermente più bassi. Il valore più alto che denota la media dei percorsi riconosciuti aventi 3 curve con un rumore di +/- 5 è quello di San Francisco, 208,4666 è un valore molto elevato che però ha risentito notevolmente dell'aggiunta delle lunghezze dei segmenti come metrica, poiché la stessa analisi utilizzando solo una metrica aveva un valore più che triplicato. Nonostante questo, aumentando il numero di curve tutti i valori precipitano tendendo sempre più ad 1.

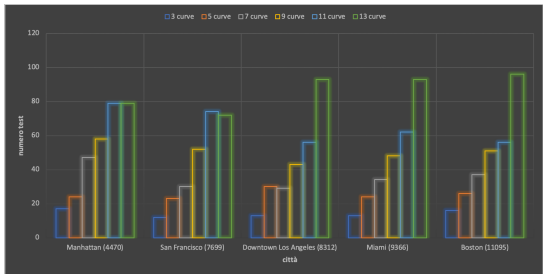
Più particolari sono i grafici dove viene utilizzato rumore +/- 15 e +/- 20, presentano un andamento completamente irregolare, soprattutto per Manhattan e San Francisco. Per quanto riguarda Manhattan, nel terzo grafico presenta un andamento irregolare e intermittente, ma che comunque tende sempre a valori più bassi fino a rasentare il valore 1. Nell'ultimo grafico, invece mostra un andamento più armonioso ma che mantiene valori più alti fino a quando si considerano percorsi di massimo 7 curve, successivamente il decremento è molto accentuato. Questo accade perché, considerando un grado di rumore elevato in una pianta come quella di Manhattan, è più probabile che ci siano percorsi molto simili tra loro anche con un numero di curve sostenuto.



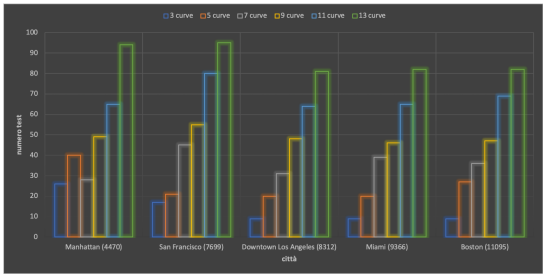
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.15: Occorrenze di percorsi ricercati con n curve effettuate

I grafici mostrati in figura 3.15 giustificano quanto affermato prima, nell'analisi fatta su San Francisco, per tutti gli scaglioni di rumore, il numero di percorsi ricercati è sempre basso (vicino a 10), questo vuol dire che il numero di percorsi trovati con poche curve è ridotto rispetto alle altre città. Il fatto che i percorsi con poche curve siano più facilmente riconoscibili con rumore +/- 20 lascia presumere che, in generale, le strade di San Francisco siano più lunghe.

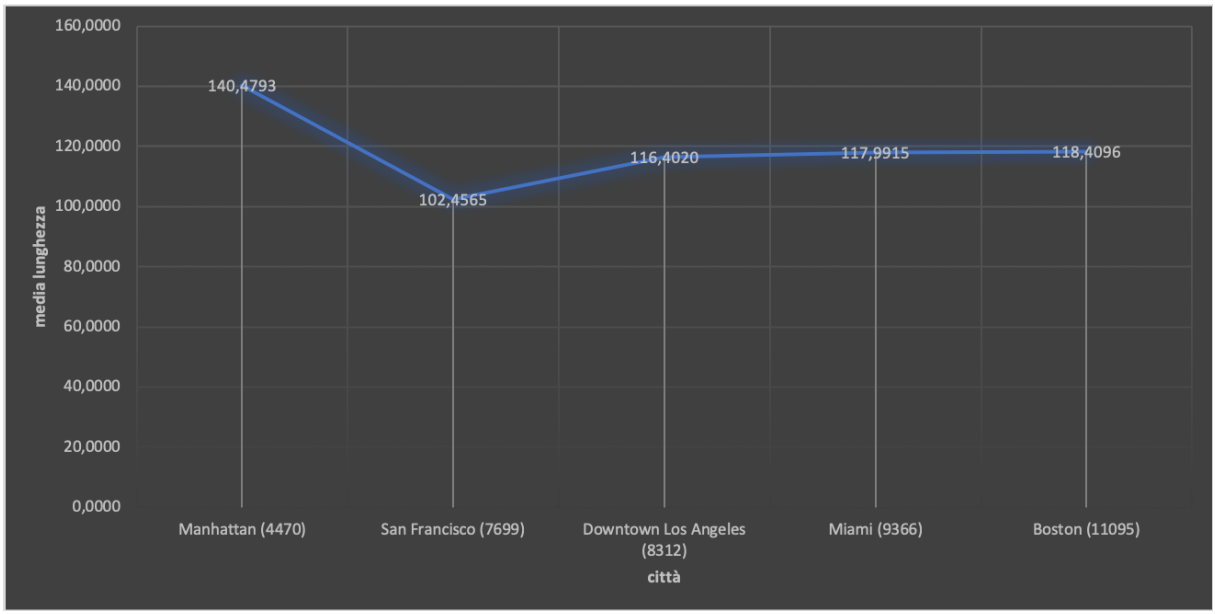


Figura 3.16: Media delle lunghezze dei segmenti ordinate secondo il numero di nodi

Questo grafico conferma che San Francisco e Downtown Los Angeles siano le città con la media di lunghezza dei segmenti più alta, con valori di 118,4096 e 140,4793 rispettivamente, infatti anche Dowtown Los Angeles presenta un andamento simile a quello di San Francisco con rumore +/- 20 (mostrato in figura 3.14).

3.2.2 Città europee

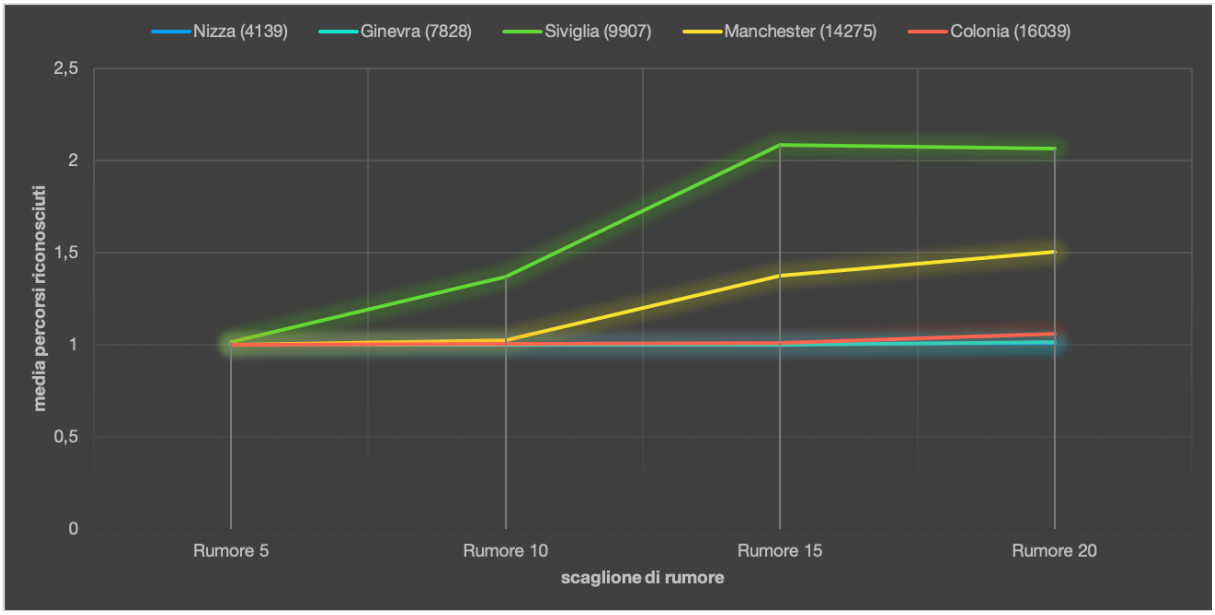
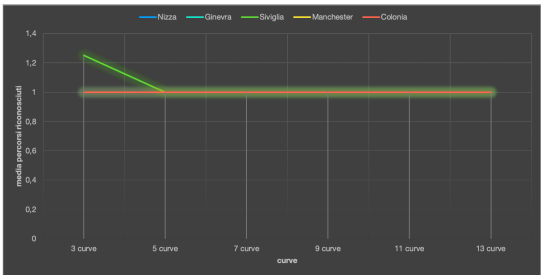


Figura 3.17: Andamento delle medie generali per le città europee

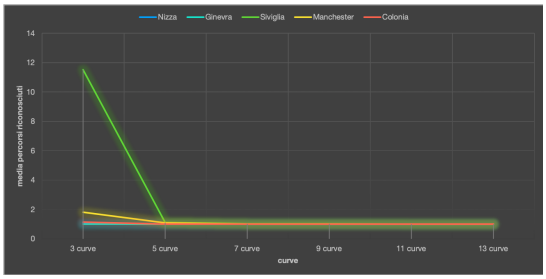
La figura 3.17 mostra l'andamento delle città europee prese in esame, si notano alcune differenze con le città americane, l'andamento è strettamente diverso così come la scala. Le medie rilevate per queste città si mantengono molto vicine al valore 1 e l'unica che supera il valore 2 è Siviglia. Una differenza particolare con le altre analisi effettuate, soprattutto se confrontata con quella sulle città americane, è che aggiungendo la lunghezza dei segmenti come metrica si ottengono risultati più alti, questa è la chiara evidenza che ciò che incide maggiormente sul riconoscimento dei percorsi per le città europee è la diversità di lunghezza tra le strade.

In questo caso, il riconoscimento dei percorsi diventa difficile, poiché nelle architetture europee si riscontra molta diversità nelle strade e nella pianta in sé. Come nel caso delle americane, non risultano correlazioni tra il numero dei nodi e il numero di percorsi riconosciuti.

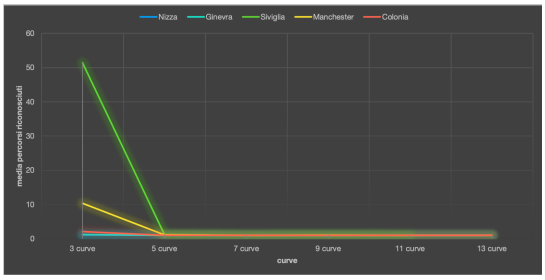
Andando più nel dettagli si osservano le medie di ogni cluster di curve effettuate.



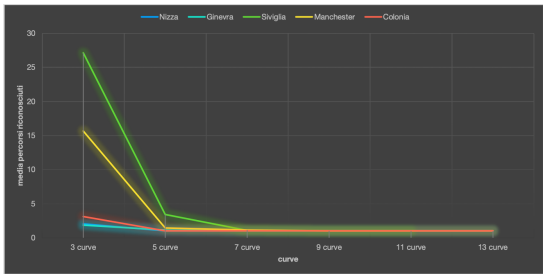
Rumore 5



Rumore 10



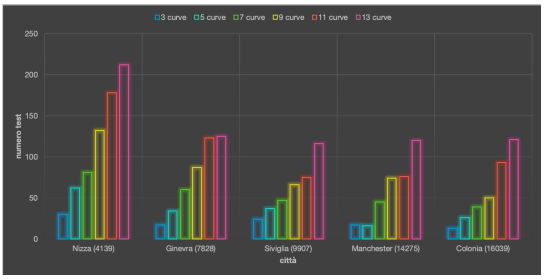
Rumore 15



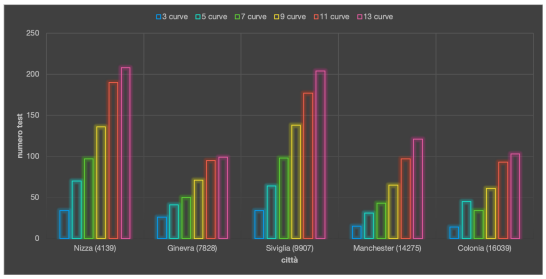
Rumore 20

Figura 3.18: Medie suddivise per cluster di curve effettuate

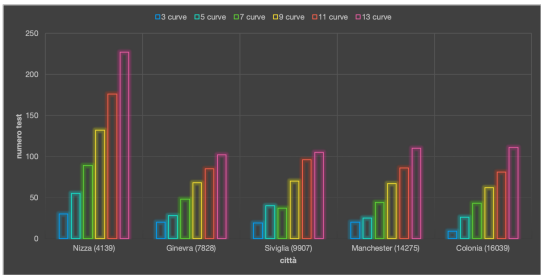
La figura 3.18 mette in risalto proprio la differenza accennata prima, poiché le medie dei percorsi trovati considerando anche le lunghezze sono maggiori, soprattutto nei primi cluster di curve. Questo conferma le assunzioni fatte precedentemente riguardo alla metrica che più influenza i risultati.



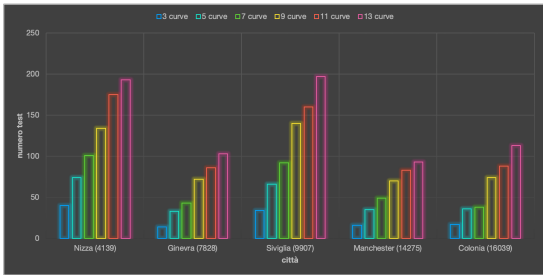
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.19: Numero di test effettuati per ogni cluster

Anche qui il numero di test rimane piuttosto proporzionato, ad eccezione di Siviglia e Nizza che mantengono un numero più alto in quasi tutti gli scaglioni, nonostante questo la media di Siviglia rimane la più alta, questo dimostra che i test effettuati hanno prodotto risultati con un numero elevato di percorsi riconosciuti.

Per completare l'analisi, viene presentato il grafico della lunghezza media dei segmenti delle città europee.

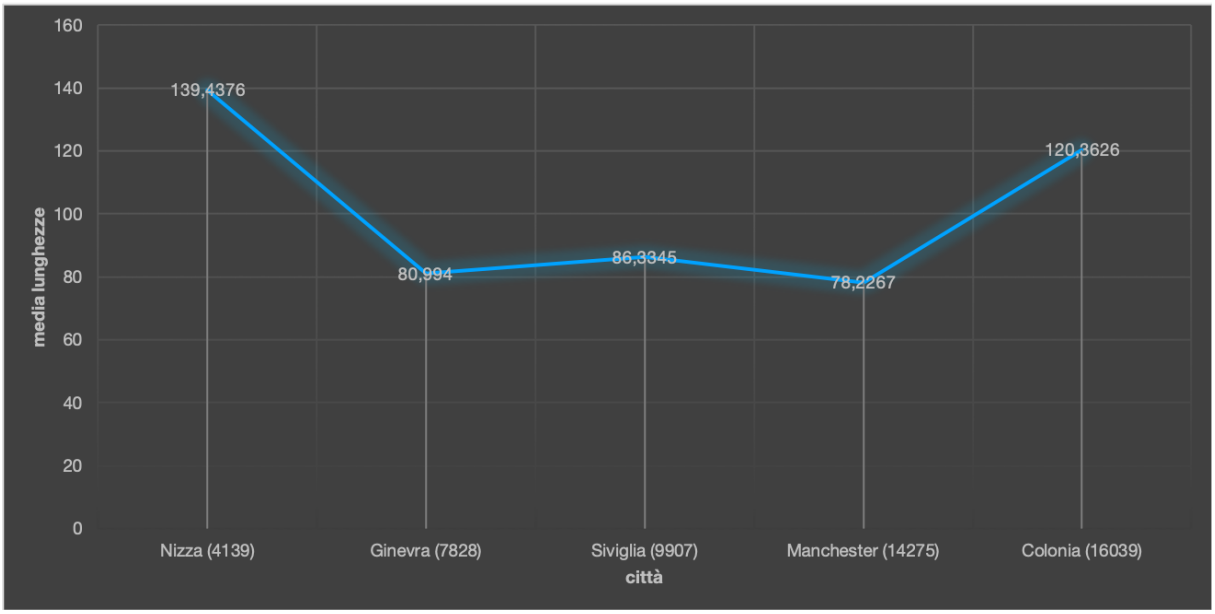


Figura 3.20: Lunghezza media dei segmenti di ogni città ordinate per numero di nodi

Il grafico suggerisce che anche per la lunghezza media dei segmenti non vi è relazione con il numero di nodi della città. Infatti, Nizza risulta avere la lunghezza media più alta, ma mantiene una media di riconoscimento dei percorsi piuttosto bassa, da qui si può intuire che le strade più lunghe hanno maggiore probabilità di risultare unici. Il caso di Siviglia può confermare l'intuizione, poiché possedendo una media più bassa di lunghezze dei segmenti, presenta una maggiore difficoltà di riconoscimento.

3.2.3 Città italiane

Per poter effettuare un confronto ed uno studio più approfondito, l'analisi con l'aggiunta della seconda metrica è stata applicata alle stesse città italiane già precedentemente passate in rassegna. Inizialmente viene presentato l'andamento generale dei valori medi per valutare i cambiamenti subiti dopo l'inserimento della metrica aggiuntiva.

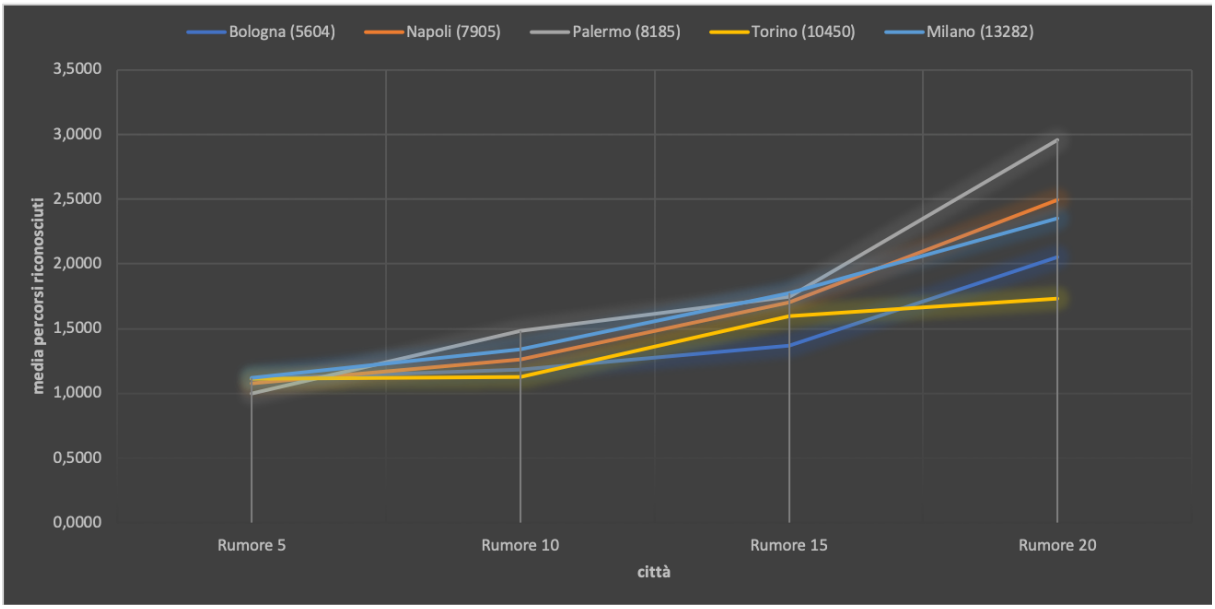


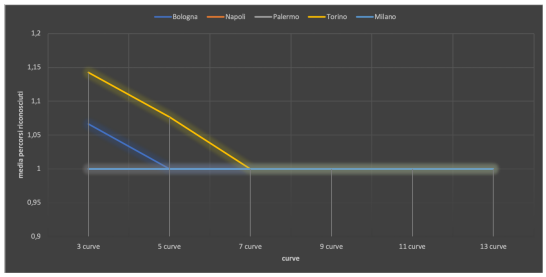
Figura 3.21: Andamento delle medie generali con i 4 scaglioni di rumore

I grafici in figura 3.21 fanno emergere risultati molto differenti da quelli delle città americane analizzate precedentemente e anche dalle stesse con una sola metrica. Con l'introduzione della nuova metrica si ha un notevole decremento dei valori medi, sia con tasso di rumore contenuto che non, poiché anche per i valori disturbati da un rumore di +/- 20, il valore massimo risulta essere 2,96 di Palermo contro il 40,2270 di Torino sull'analisi iniziale (riduzione di più del 90%). Anche in questa analisi si evince che la grandezza delle città non è in relazione con il numero di percorsi riconosciuti.

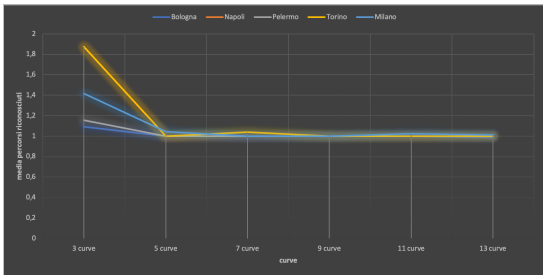
Interessante è anche il valore minimo considerando il minimo rumore, 1,0024 di Palermo da confrontare con 3,2125 di Napoli, anche qui si può notare una riduzione di più del 60%. Curioso è il caso specifico di Palermo, che dal valore minimo con il minimo rumore, passa a quello massimo con il massimo del rumore. Questo evidenzia che, con ogni probabilità, Palermo ha un elevato numero di strade con lunghezza simile uscente da ogni nodo.

Questo significa, in generale, la topologia delle piante delle città italiane è più che altro irregolare, con una distribuzione delle lunghezze delle strade eterogenea.

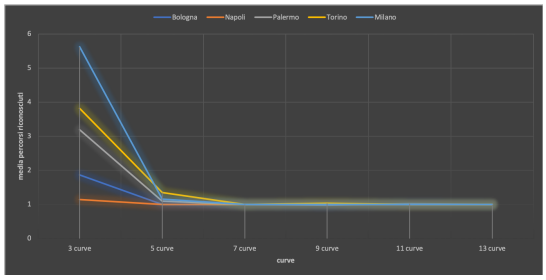
Per illustrare al meglio come vengono influenzate le medie generali, come già svolto per le altre analisi, vengono presentati i grafici delle medie organizzate per cluster di percorsi con determinate curve.



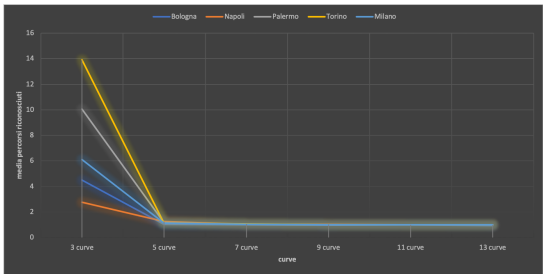
Rumore 5



Rumore 10



Rumore 15

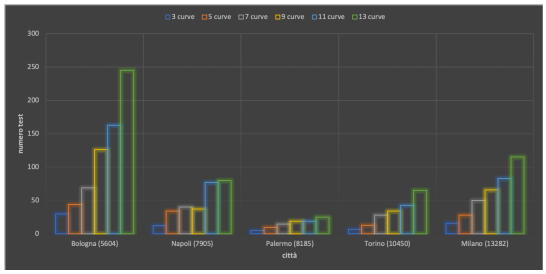


Rumore 20

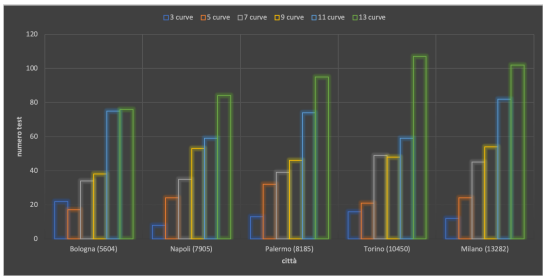
Figura 3.22: Andamento delle medie per cluster di curve effettuate

L'andamento delle medie suddivise in cluster risulta lineare e uniforme, a differenza di quanto visto con le città americane, i valori più alti vengono rilevati solo nel primo cluster, il quale non basta ad influenzare positivamente la media generale. Infatti, il primo cluster nei primi due scaglioni di rumore presenta comunque valori molto contenuti avvicinandosi a 2. Restringendo la ricerca con valori di rumore abbastanza contenuti, in mappe con un maggior numero di strade simili, alcune medie tendono ad avere valori poco molto vicini ad 1.

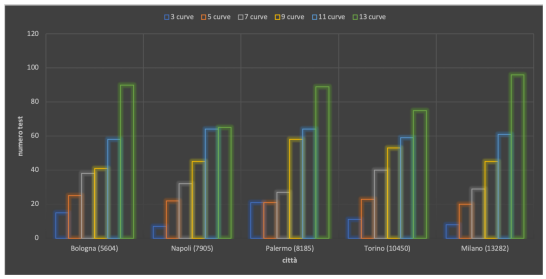
Nella figura 3.22 non si ha omogeneità sui valori massimi (registrati nel primo cluster), infatti in base ai diversi rumori, il valore massimo passa da una città all'altra. Per quanto riguarda i valori minimi, invece si ha una distribuzione uniforme, poiché per tutti gli altri cluster la media rimane compresa tra 1 e 2.



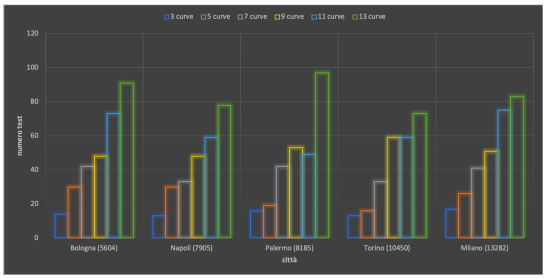
Rumore 5



Rumore 10



Rumore 15



Rumore 20

Figura 3.23: Numero di percorsi ricercati con n curve

Nel primo grafico in figura 3.23 si nota che, in generale, i percorsi ricercati su Palermo per tutti i cluster di curve sono meno rispetto agli altri. Questo comportamento giustifica il valore minimo registrato su Palermo per tutte le medie dei cluster di curve, soprattutto sull'ultimo (13 curve) dove si trova una media di 1. Per il resto degli scaglioni di rumore, la distribuzione dei percorsi ricercati risulta regolare, con qualche picco su quelli da 13 curve. Tuttavia gli andamenti delle medie per cluster restano sempre tendenti ad 1, il che vale a dire che la maggior parte dei percorsi trovati è sempre 1 o 2, un risultato ottimo se confrontato a quello delle città statunitensi.

Per completezza, viene mostrato il grafico della media della lunghezza degli archi delle diverse città.

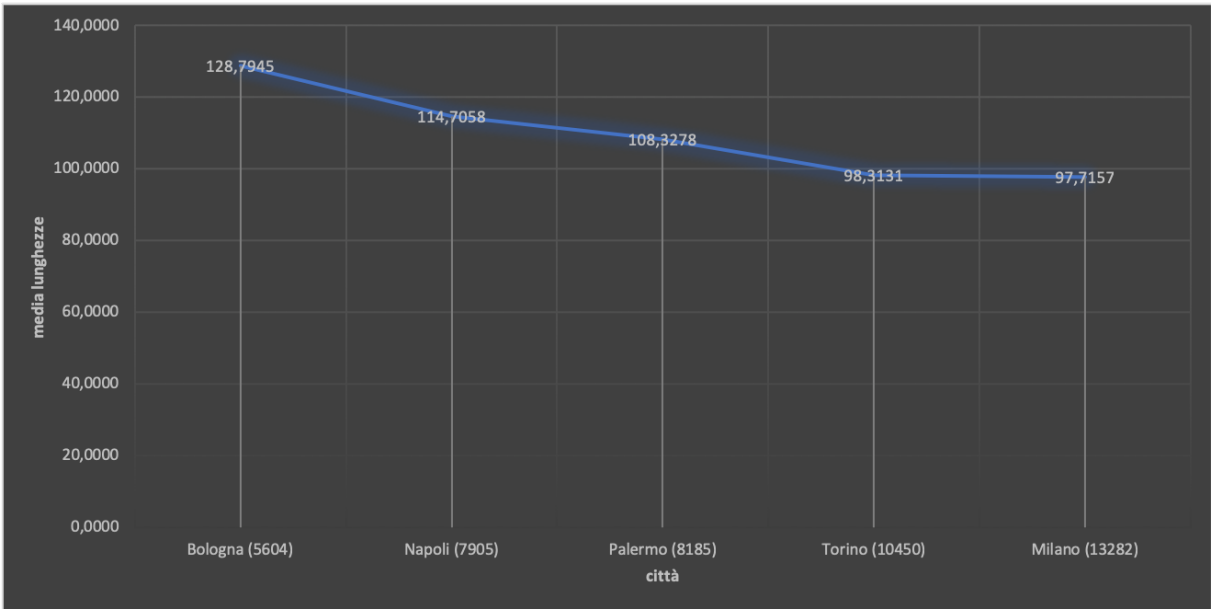


Figura 3.24: Media delle lunghezze degli archi ordinate per numero di nodi

Nonostante Bologna sia la città più piccola, detiene la media di lunghezza delle strade più alta, ciò non significa che la diversità tra i vari archi sia omogenea, infatti nelle medie generali Bologna risulta tra i valori più bassi in tutti gli scaglioni di rumore. Questo indica un'elevata diversità, sia per quanto riguarda le curve di ogni nodo che per le lunghezze degli archi. Il resto delle medie di lunghezza rimane lineare, con un decremento progressivo in relazione al numero dei nodi delle diverse città.

Conclusioni

Alla luce di quanto ottenuto attraverso le diverse analisi dei dati raccolti dalle simulazioni, si possono trarre alcune importanti conclusioni riguardanti la tematica in generale.

Per rendere la simulazione il più vicina possibile ad un'applicazione reale, sono stati introdotti 4 scaglioni di rumore durante la raccolta dei dati che rispecchiano i reali dispositivi, distinguendoli tra quelli più datati e quelli più recenti. Sicuramente un disturbo contenuto sui dati, corrisponderà all'utilizzo di uno smartphone di ultima generazione, mentre un rumore elevato simulerebbe un dispositivo meno recente e performante.

I risultati ottenuti possono servire a capire meglio come si comporterebbe un soggetto intenzionato a recuperare informazioni sulla localizzazione degli utenti per motivi illeciti. Ciò che si nota dalle analisi effettuate sono le enormi differenze tra una possibile applicazione in Italia, Europa in generale e Stati Uniti, estendendo lo studio anche ad altre nazioni, queste differenze potrebbero portare quindi ad una diffusione del fenomeno in uno stato piuttosto che in un altro.

Nello specifico, i risultati di questo studio mostrano un rischio elevato di attacco per la maggior parte delle città italiane, poiché è diffusa principalmente l'architettura a pianta radiocentrica. Questo tipo di architettura consente un facile riconoscimento attraverso l'uso del magnetometro di un dispositivo abbastanza recente(svolte effettuate agli incroci), perché le strade uscenti da ogni incrocio hanno inclinazioni differenti l'una con l'altra e questo garantisce una maggiore unicità dei percorsi.

Se invece si discutono i risultati ottenuti dall'analisi delle città americane, il rischio diventa minimo anche con lo smartphone più preciso, poiché la maggioranza delle città possiede una pianta a scacchiera. Essa rende molto difficile il riconoscimento valutando soltanto i dati raccolti dalle curve effettuate, questo avviene perché, molti incroci (nodi) della mappa avranno archi (strade) uscenti con le medesime quattro angolazioni (0, 90, 180, 270). Come di consueto, esistono le eccezioni, ad esempio la città di Boston si distingue dalle altre per l'architettura "europea", infatti nell'analisi i valori minimi sono stati registrati in questa città.

La situazione sulla sicurezza si aggrava se ai dati raccolti dal magnetometro si aggiungono quelli dell'accelerometro, questo procedimento porterebbe ad una precisione massima per il riconoscimento dei percorsi, anche utilizzando smartphone datati e poco precisi. Come si può dedurre dalle discussioni fatte finora, le città italiane sono quelle

che ne risentono maggiormente, questo sempre a causa dell'architettura delle città stesse che, oltre ad avere incroci molto diversi tra loro, hanno anche eterogeneità nella lunghezza delle strade. Questa affermazione viene confermata infatti dai valori minimo e massimo dell'analisi sulle città italiane con disturbo minimo sui dati, 1,0024 (Napoli) e 1,1198 (Milano) rispettivamente. Invece, il massimo e il minimo con disturbo massimo sui dati sono 2,96 (Palermo) e 1,731 (Torino), dati abbastanza preoccupanti parlando di sicurezza sulla privacy.

Questa volta i risultati riportati dall'analisi sulle città americane risultano più simili a quelli delle città italiane, facendo crollare drasticamente i valori medi sia ad elevati livelli di rumore che a quelli più bassi. Infatti, in questo caso il valore massimo registrato con il massimo del rumore è 11,31 (Manhattan) contro 162,6225 (Miami) dell'analisi fatta considerando solo i dati dell'accelerometro. Questo significa che, nonostante i percorsi siano simili per quanto riguarda le curve, non lo sono riguardo alla lunghezza delle strade, specialmente per città come Boston e Downtown Los Angeles.

Resta particolare il caso europeo, che detiene la maggioranza di valori medi tendenti ad 1, sia a tassi di rumore bassi che a quelli più alti. Inoltre, a differenza di Italia e Stati Uniti, in Europa la metrica determinate è quella della lunghezza dei segmenti, che in uno scenario reale diventerebbe la meno accurata. La causa di ciò è da imputare agli innumerevoli fattori di disturbo e influenza sulle rilevazioni dell'accelerometro

Come già anticipato, l'accuratezza della localizzazione degli utenti, qualsiasi sia la nazione o la città considerata, è parecchio influenzata dal dispositivo utilizzato per la raccolta dei dati. Uno smartphone top di gamma di ultima generazione, garantirà risultati molto precisi e di conseguenza esporrà il proprietario a rischi più elevati.

Ciò che invece accomuna le città delle diverse nazioni è la mancanza di relazioni tra la facilità o difficoltà di localizzazione e la grandezza della città in cui si trova la vittima.

In generale, le città degli Stati Uniti creeranno non pochi problemi ad un possibile attaccante, mentre le città europee si presteranno ad essere un bersaglio più facile e raggiungibile.

Bibliografia

[1] S. Narain, T. D. Vo-Huu, K. Block and G. Noubir, "Inferring User Routes and Locations Using Zero-Permission Mobile Sensors," 2016 IEEE Symposium on Security and Privacy (SP), San Jose, C 2016, pp. 397-413.

[2] OpenStreetMap, "OpenStreetMap Project", https://www.openstreetmap.org/.

[3] Android SDK, "Step detection," http://developer.android. com/reference/android/hardware/Sensor.html#TYPE STEP DETECTOR.

[4] Collin R. Mulliner, "Dynamic Dalvik Instrumentation Framework for Android," https://github.com/crmulliner/ddi.

[5] M. Backes, S. Bugiel, C. Hammer, O. Schranz, and P. von Styp-Rekowsky, "Boxify: Full-fledged app sandboxing for stock android," in 24th USENIX Security Symposium (USENIX Security 15). Washington, D.C.: USENIX

[6] K. Fawaz and K. G. Shin, "Location privacy protection for smartphone users," in Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, ser. CCS '14. ACM, 2014.

[7] D. F. Kune, J. Koelndorfer, N. Hopper, and Y. Kim, "Location leaks over the GSM air interface," in Proceedings of the 19th Annual Network & Distributed System Security Symposium, Feb. 2012.

[8] L. Bindschaedler, M. Jadliwala, I. Bilogrevic, I. Aad, P. Ginzboorg, V. Niemi, and J.-P. Hubaux, "Track me if you can: On the effectiveness of context-based identifier changes in deployed mobile networks." in NDSS. The Internet Society, 2012.

[9] S.-W. Lee and K. Mase, "Activity and location recognition using wearable sensors," Pervasive Computing, IEEE, vol. 1, no. 3, pp. 24–32, July 2002.

[10] Python 3, https://www.python.org.

[11] Google Maps, Google Inc., https://www.google.it/maps/

[12] Boeing, G. 2017. “OSMnx: New Methods for Acquiring, Constructing, Analyzing, and Visualizing Complex Street Networks.” Computers, Environment and Urban Systems. 65, 126-139. doi:10.1016/j.compenvurbsys.2017.05.004

[13] NetworkX, «Software for complex network», https://networkx.github.io/

[14] Jun Han, E. Owusu, L. T. Nguyen, A. Perrig and J. Zhang, "ACComplice: Location inference using accelerometers on smartphones," 2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012), Bangalore, 2012, pp. 1-9.

[15] Joshua Greenfield. Matching GPS observations to locations on a digital map. Proc. 81st Annual Meeting of the Transportation Research Board, 2002.

[16] S. Nawaz and C. Mascolo, “Mining users’ significant driving routes with low-power sensors,” in Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems, ser. SenSys ’14. ACM, 2014, pp. 236– 250.

[17] X. Zhou, S. Demetriou, D. He, M. Naveed, X. Pan, X. Wang, C. A. Gunter, and K. Nahrstedt, “Identity, location, disease and more: Inferring your secrets from android public resources,” in Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, ser. CCS ’13. ACM, 2013, pp. 1017– 1028.

[18] Y. Michalevsky, A. Schulman, G. A. Veerapandian, D. Boneh, and G. Nakibly, “Powerspy: Location tracking using mobile device power analysis,” in Proceedings of the 24th USENIX Conference on Security Symposium. Washington, D.C.: USENIX Association, Aug. 2015, pp. 785–800.

[19] Facebook Inc, https://www.facebook.com/.

[20] Microsoft Skype Division, https://www.skype.com/it/