

UNIVERSITÀ POLITECNICA DELLE MARCHE

FACOLTÀ DI INGEGNERIA



*Corso di Laurea Triennale in
Ingegneria Informatica e dell'Automazione*

***Integrazione di dati AIS ed immagini SAR da Sentinel-1 a
supporto dell'identificazione di attività di pesca illegali***

*Integration of AIS data and SAR images from Sentinel-1 to support the identification
of illegal fishing activities*

Relatore: Chiar.mo
DOTT. MANCINI ADRIANO

Correlatore: Chiar.mo
DOTT. GALDELLI ALESSANDRO

Tesi di Laurea di:
GIACCAGLINI ALESSANDRO

ANNO ACCADEMICO 2019-2020

Indice

1	Introduzione	5
1.1	Stato dell'arte	5
1.2	Il progetto	6
1.3	Struttura della tesi	7
2	Strumenti e tecnologie utilizzate	9
2.1	Sentinel 1 e SAR	9
2.1.1	Carico utile del satellite	10
2.1.2	Hub Copernicus e catalogo ONDA	12
2.2	Sistema AIS	13
2.3	Sentinel 1 Toolbox	15
2.3.1	SNAP	15
2.4	SUMO	16
2.4.1	Confronto con SNAP	17
2.5	PostgreSQL	17
2.5.1	PostGIS	18
2.6	QGIS	19
2.7	Python e le sue librerie	19
2.7.1	Pandas	20
2.7.2	Psycopg2	21
2.7.3	Matplotlib e Seaborn	21
3	Sviluppo progetto	23
3.1	Download immagini SAR	23
3.2	Elaborazione immagini	24
3.3	Script Python	26
3.3.1	Interrogazione database	27
3.3.2	Associazione Punto-Punto	28
3.3.3	Associazione Punto-Linea	29
3.3.4	Previsione traiettoria nel periodo di blackout e heatmap	30
4	Analisi Risultati	33
4.1	Risultati Tabelle	33
4.2	Analisi dati ottenuti	35
4.2.1	Confronto risultati su QGIS	36

5 Conclusioni	39
5.1 Obiettivi raggiunti	39
Bibliografia	41
Elenco delle figure	43
Elenco delle tabelle	45

Capitolo 1

Introduzione

La pesca illegale è una minaccia complessa e dilagante per gli stock ittici mondiali e per le comunità che dipendono dagli stessi. Ogni anno, in tutto il mondo vengono pescate illegalmente tra 11 e 26 tonnellate di pesce con perdite annuali totali tra i 10 e i 23,5 miliardi di dollari. Le stime indicano che le catture INN (pesca Illegale, Non dichiarata e Non regolamentata) globali corrispondano a un valore che oscilla tra il 13 e 31% della produzione ittica dichiarata. In alcune regioni, questa percentuale arriva addirittura al 40%. Le disparità e la debolezza dei controlli sulle importazioni in alcuni dei maggiori Stati membri dell'Unione Europea fanno sì che le catture illegali riescano ancora a infilarsi tra le maglie del sistema e a entrare nella filiera UE.[1]

1.1 Stato dell'arte

Negli anni si è iniziato a parlare di *Big Data* [2], ovvero una raccolta di dati informativi così estesa in termini di volume, velocità e varietà da richiedere tecnologie e metodi analitici specifici per l'estrazione di valore o conoscenza. Il termine è utilizzato dunque in riferimento alla capacità (propria della scienza dei dati) di analizzare ovvero estrapolare e mettere in relazione un'enorme mole di dati eterogenei, strutturati e non strutturati (grazie a sofisticati metodi statistici e informatici di elaborazione), allo scopo di scoprire i legami tra fenomeni diversi e prevedere quelli futuri. In questo caso, lo studio della grande mole di dati è finalizzato a limitare le frodi da parte dei pescatori e per fare ciò vengono utilizzati strumenti e tecnologie che danno la possibilità di monitorare i mari.

1.2. La spiegazione puntuale di ognuna di esse, con le relative sotto-fasi, viene riportata nel terzo capitolo, mentre quello a cui si vuole portare l'attenzione ora è la struttura della *pipeline* seguita per poter rilevare le imbarcazioni.

- In primis è necessario scaricare un'immagine su cui ricavare dei dati.
- Il passo successivo è quello di elaborarle attraverso i *Toolbox* messi a disposizione dall'ESA. Come output si otterrà un file csv in cui vengono riportati tutti i target rilevati.
- Infine, preso in input il csv con i target, sarà necessario processarlo, secondo diversi approcci, con uno script in linguaggio Python per poter avere come risultato dei dati che verranno poi presi in analisi.

1.3 Struttura della tesi

La tesi viene strutturata nella seguente modalità:

- Nel secondo capitolo vengono illustrate le varie tecnologie e strumenti utilizzati per la realizzazione di questo progetto, alcuni dei quali sono stati citati precedentemente. Viene illustrato anche il linguaggio di programmazione Python attraverso il quale è stato scritto il codice per elaborare i dati ottenuti dalle immagini.
- Nel terzo capitolo viene spiegato in dettaglio come è stato sviluppato il meccanismo alla base di questo progetto.
- Nel quarto capitolo sono riportati i dati ottenuti dalle varie elaborazioni, seguiti da un'approfondita analisi su di essi.
- Nel quinto capitolo vengono riportate le conclusioni in cui sono anche espresse considerazioni riguardanti i limiti strumentali riscontrati.

Capitolo 2

Strumenti e tecnologie utilizzate

In questo capitolo verranno analizzate le varie soluzioni software adottate per poter far fronte ai *Big Data*, estrapolandoli e mettendoli in relazione tra loro così da avere a disposizione le informazioni specifiche che aiutino a monitorare le attività di pesca.

Nella parte iniziale vengono mostrate quali sono le tipologie di dati di interesse per questa indagine, mentre poi in secondo luogo si parlerà degli strumenti necessari per poterli elaborare e ricavare i risultati necessari per trarre delle conclusioni.

2.1 Sentinel 1 e SAR

Sentinel-1 [4] è la prima delle cinque missioni che l'ESA ha sviluppato per l'iniziativa *Copernicus*. Essa include l'*imaging* in banda C operante in quattro modalità di *imaging* esclusive con diversa risoluzione (fino a 5 m) e copertura (fino a 400 km). Fornisce capacità di doppia polarizzazione, tempi di revisione molto brevi e consegna rapida del prodotto. Per ogni osservazione sono disponibili misurazioni precise della posizione e dell'assetto del veicolo spaziale.

Il radar ad apertura sintetica (SAR) ha il vantaggio di funzionare a lunghezze d'onda non ostacolate dalla copertura nuvolosa o dalla mancanza di illuminazione e può acquisire dati su un sito durante il giorno o la notte in tutte le condizioni meteorologiche. *Sentinel-1*, mostrato in figura 2.1, con il suo strumento C-SAR può offrire un monitoraggio affidabile e ripetuto di un'ampia area.

La missione è composta da una costellazione di due satelliti, *Sentinel-1A* e *Sentinel-1B*, che condividono lo stesso piano orbitale. Questi satelliti sono progettati per funzionare in una modalità operativa pre-programmata e priva di conflitti. La missione fornisce una capacità operativa indipendente per la mappatura radar continua della Terra. E' stata progettata per fornire una maggiore frequenza di revisione, copertura, tempestività e affidabilità per servizi operativi e applicazioni, come la creazione di un archivio dati coerente a lungo termine, che richiedono lunghe serie temporali.

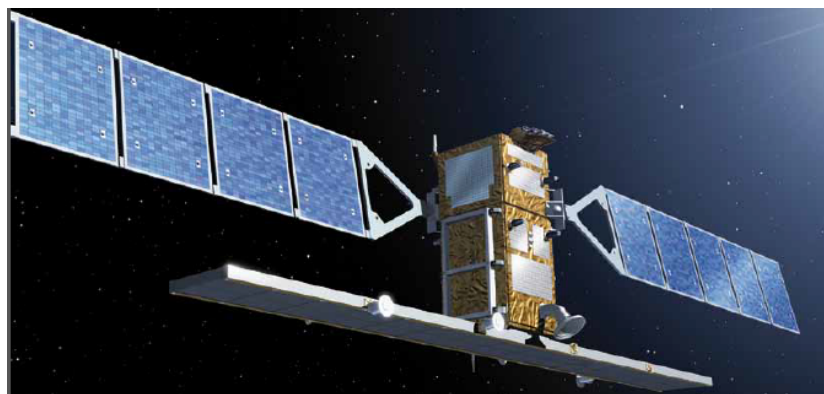


Figura 2.1 – Satellite Sentinel 1. [4]

La missione fornirà una capacità di interferometria operativa attraverso requisiti rigorosi imposti sulla precisione dell'assetto, sulla conoscenza dell'assetto e dell'orbita e sulla precisione dei tempi di acquisizione dei dati. La costellazione coprirà le masse terrestri di tutto il mondo su base bisettimanale, le zone di ghiaccio marino, le zone costiere europee e le rotte marittime su base giornaliera e aprirà continuamente l'oceano grazie alle immagini delle onde. *Sentinel 1* e il breve tempo di revisione miglioreranno notevolmente le capacità degli utenti e forniranno dati quotidianamente e sistematicamente per il monitoraggio marittimo e terrestre, la risposta alle emergenze, i cambiamenti climatici e la sicurezza. Ognuno di questi satelliti dovrebbe trasmettere i dati di osservazione della Terra per almeno 7 anni e avere carburante a bordo per 12 anni [5].

2.1.1 Carico utile del satellite

Il *Sentinel 1* trasporta un singolo strumento radar ad apertura sintetica in banda C che opera a una frequenza centrale di 5,405 GHz. Comprende un'antenna ad allineamento di fase attiva orientata verso destra che fornisce una scansione rapida in elevazione e azimuth, capacità di archiviazione dati di 1.410 Gb e capacità di downlink in banda X di 520 Mbit / s. Supporta il funzionamento in doppia polarizzazione (HH + HV, VV + VH) implementato attraverso una catena di trasmissione (commutabile su H o V) e due catene di ricezione parallele per la polarizzazione H e V. I dati sulla doppia polarizzazione sono utili per la classificazione della copertura del suolo e per le applicazioni con ghiaccio marino. *Sentinel 1* opera in quattro modalità di acquisizione esclusive [6], le quali sono mostrate in figura 2.2. Esse sono:

- *Stripmap* (SM), la quale fornisce una copertura con una risoluzione di 5 m per 5 m su una larghezza di andana stretta di 80 km. È possibile selezionare una delle sei strisce di imaging modificando l'angolo di incidenza del raggio e l'ampiezza del raggio di elevazione.
- *Interferometric Wide swath* (IW), che consente di combinare un'ampia larghezza di andana (250 km) con una risoluzione geometrica moderata (5 m per 20 m). La modalità IW mostra tre sottoandane utilizzando l'osservazione del terreno con SAR a scansione progressiva (TOPSAR). Con

la tecnica TOPSAR, oltre a guidare il raggio nel raggio d'azione come in SCANSAR, il raggio viene anche guidato elettronicamente da indietro in avanti nella direzione dell'azimut per ogni raffica, evitando la smeratura e risultando in un'immagine di qualità superiore. L'interferometria è assicurata da una sufficiente sovrapposizione dello spettro Doppler (nel dominio azimutale) e dello spettro del numero d'onda (nel dominio dell'elevazione). La tecnica TOPSAR garantisce una qualità dell'immagine omogenea in tutta l'andana.

- *Extra-Wide swath* (EW), modalità destinata ai servizi operativi in zone marittime, ghiacciate e polari in cui sono richiesti un'ampia copertura e tempi di revisione brevi. La modalità EW funziona in modo simile alla modalità IW che impiega una tecnica TOPSAR utilizzando cinque sottoandane invece di tre, con una risoluzione inferiore (20 m per 40 m). La modalità EW può essere utilizzata anche per l'interferometria come con la modalità IW.
- *Wave mode* (WV), la quale, insieme ai modelli globali di onde oceaniche, può aiutare a determinare la direzione, la lunghezza d'onda e l'altezza delle onde negli oceani aperti. Le acquisizioni in modalità *Wave* sono composte da immagini *stripmap* di 20 km per 20 km, acquisite alternativamente su due diversi angoli di incidenza. Le immagini delle onde vengono acquisite ogni 100 km, con le immagini sullo stesso angolo di incidenza separate da 200 km.

Le principali modalità senza conflitti sono IW su terra e WV su oceano aperto.

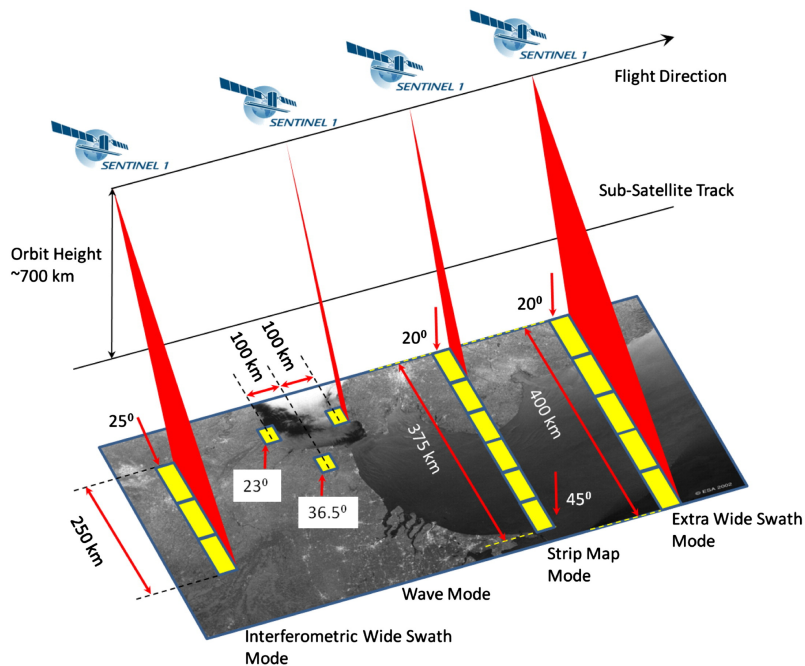


Figura 2.2 – Modalità di acquisizione Sentinel 1. [6]

2.1.2 Hub Copernicus e catalogo ONDA

L'ESA mette a disposizione dei cataloghi attraverso i quali, con apposite interrogazioni in base alla tipologia di informazione ricercata, è possibile scaricare l'immagine desiderata per poter poi effettuare le opportune elaborazioni. Questi cataloghi offrono moltissime risorse, una molto rilevante è quella di poter avere accesso ad immagini immagazzinate nell'archivio a lungo termine, dopo essere state rimosse da quello online. Basterà ordinarle e dopo un determinato intervallo di tempo queste torneranno ad essere disponibili per il *download*.

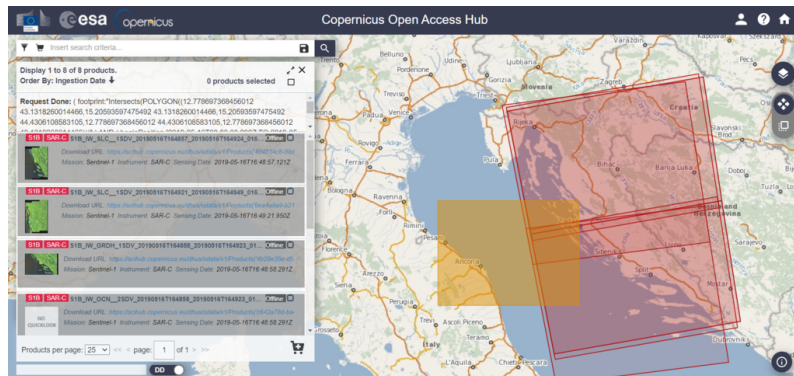


Figura 2.3 – Hub Copernicus [7]

Per la realizzazione di questo progetto sono stati utilizzati due cataloghi, l'Hub Copernicus [7], insieme al catalogo di ONDA [8], riportato in figura 2.4, che praticamente fornisce le stesse funzioni del primo, con l'unica differenza nel ripristinare le immagini offline con maggior efficienza e rapidità. Per poter ricercare ciò di cui si ha bisogno i passi da seguire sono:

- Selezionare nella scheda dei filtri la data a cui si è interessati.
- Selezionare il satellite *Sentinel* desiderato, con i relativi parametri.
- Infine bisognerà disegnare attraverso appositi strumenti l'area di cui si vuole ottenere l'immagine.
- Come risultato si otterranno delle aree che intersecano la regione indicata precedentemente, come viene mostrato dalla figura 2.3. In questo modo si avranno risultati che si avvicinano a ciò a cui si era interessati.
- Infine basterà scegliere l'immagine più adatta e scaricarla, oppure ordinarla nel caso sia offline.

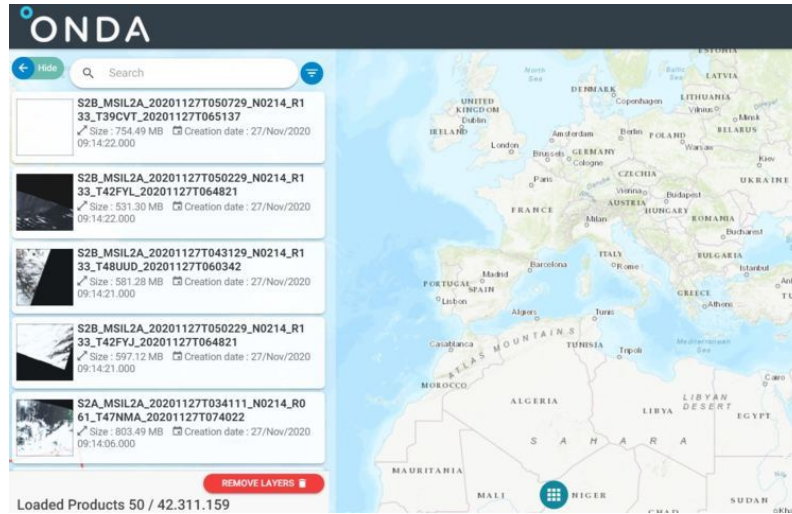


Figura 2.4 – Catalogo ONDA [8]

2.2 Sistema AIS

Il sistema di identificazione automatica (AIS) [9], è un sistema automatico di tracciamento utilizzato in ambito navale. Il suo scopo è quello di fornire una serie di informazioni sulle navi presenti nelle immediate vicinanze. Queste comprendono principalmente l'identificazione, il tipo di unità, la posizione, la rotta, la velocità. Queste informazioni possono essere visualizzate su schermo e usate dal sistema di pilotaggio per dare un allarme in caso di rotta di collisione. I dati trasmessi consentono inoltre alle autorità marittime di monitorare i movimenti delle navi. L'AIS è formato da un ricetrasmittitore VHF collegato a un sistema di posizionamento come un LORAN o un ricevitore GPS, con altri sensori elettronici di navigazione, quali ad esempio una girobussola. La trasmissione avviene su due canali (87B e 88B) della banda destinata alle trasmissioni radio fra mezzi navali. I dati vengono trasmessi periodicamente utilizzando degli algoritmi che consentono di evitare la collisione fra le trasmissioni delle varie unità. La frequenza con cui vengono aggiornate le informazioni è variabile e dipende dall'attività dell'unità, può andare da una trasmissione ogni 3 minuti per i mezzi all'ancora o ormeggiati fino a una ripetizione ogni 2 secondi per unità in manovra o in rapido movimento. Date le caratteristiche delle frequenze radio utilizzate, ricevibili solo in condizioni di portata ottica, il segnale copre una distanza relativamente ridotta, dipendente dall'altezza delle antenne, in genere attorno le 20 miglia nautiche. E' però possibile far rimbalzare il segnale tra un trasponder di una nave a uno di un'altra se quest'ultima è nella sua portata, fino a raggiungere l'antenna a terra. Le navi al di fuori della copertura radio dell'AIS possono essere monitorate con il sistema di identificazione e tracciamento a lungo raggio (LRIT). L'Organizzazione marittima internazionale (IMO) e la Convenzione internazionale per la salvaguardia della vita umana in mare (SOLAS) richiedono che l'AIS sia montato a bordo di tutte le navi con stazza lorda pari o superiore a 300 tonnellate, e su tutte le navi passeggeri, indipendente-

mente dalle dimensioni. Esistono dei ricevitori AIS che possono essere installati anche su unità di piccole dimensioni e che consentono la sola ricezione e visualizzazione dei dati utilizzando in genere il display di un plotter cartografico e del radar. I dati vengono trasmessi tramite un sistema di tracciamento che utilizza un datalink auto-organizzato ad accesso multiplo a divisione di tempo (SOTDMA).

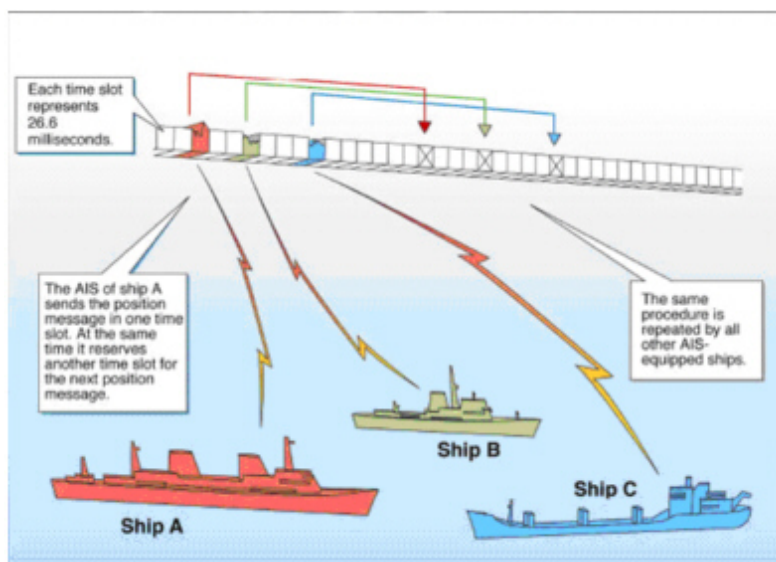


Figura 2.5 – Meccanismo di funzionamento AIS [9]

Lo standard AIS comprende diversi sub-standard chiamati "tipi" che specificano i singoli tipi di prodotto ed i principali sono il classe A e B:

- Nel primo, il ricetrasmittitore AIS montato sull'imbarcazione funziona utilizzando SOTDMA. Destinato alle grandi navi commerciali, SOTDMA richiede un ricetrasmittitore per mantenere una mappa degli slot costantemente aggiornata nella sua memoria in modo tale da avere una conoscenza preliminare degli slot disponibili per la trasmissione. I ricetrasmittitori SOTDMA annunceranno quindi la loro trasmissione, riservando di fatto il loro slot di trasmissione, come viene mostrato in figura 2.5.
- Per il secondo esistono due specifiche, un sistema CSTDMA (*carrier-sense* a divisione di tempo) ad accesso multiplo e un sistema che utilizza SOTDMA (come nella Classe A):
 - Nel sistema originale basato su CSTDMA, i ricetrasmittitori ascoltano la mappa degli slot immediatamente prima della trasmissione e cercano uno slot in cui il rumore nello slot è lo stesso (o simile) al rumore di fondo, indicando così che lo slot non è in uso da un altro dispositivo AIS. La velocità di trasmissione predefinita è normalmente ogni trenta secondi, ma può essere variata in base alla velocità dell'imbarcazione o alle istruzioni delle stazioni base.

- Il più recente sistema SOTDMA Classe B "SO", sfrutta lo stesso algoritmo di ricerca delle fasce orarie della Classe A e ha la stessa priorità di trasmissione dei trasmettitori di Classe A, contribuendo a garantire che sarà sempre in grado di trasmettere.

I dati AIS saranno poi necessari per poter effettuare le varie associazioni con i target rilevati nelle immagini SAR.

2.3 Sentinel 1 Toolbox

Il *Sentinel-1 Toolbox* (S1TBX) [10] consiste in una raccolta di strumenti di elaborazione e un'applicazione di visualizzazione e analisi per supportare l'ampio archivio di dati delle missioni SAR dell'ESA. Il *Toolbox* include strumenti per la calibrazione, il filtraggio delle macchie, la coregistrazione, l'ortorettifica, il mosaico, la conversione dei dati, la polarimetria e l'interferometria. Essi sono fondamentali per poter processare le immagini SAR e ricavare da esse le informazioni sulle imbarcazioni rilevate.

2.3.1 SNAP

L'architettura comune per tutti i *Sentinel Toolbox* è denominata *Sentinel Application Platform* (SNAP) [11], la cui interfaccia è mostrata in figura 2.6. Essa è ideale per l'elaborazione e l'analisi dell'osservazione della Terra grazie alle seguenti innovazioni tecnologiche: estensibilità, portabilità, piattaforma *Rich Client* modulare, astrazione dati EO generica, gestione della memoria piastrellata e *framework* di elaborazione di grafici.

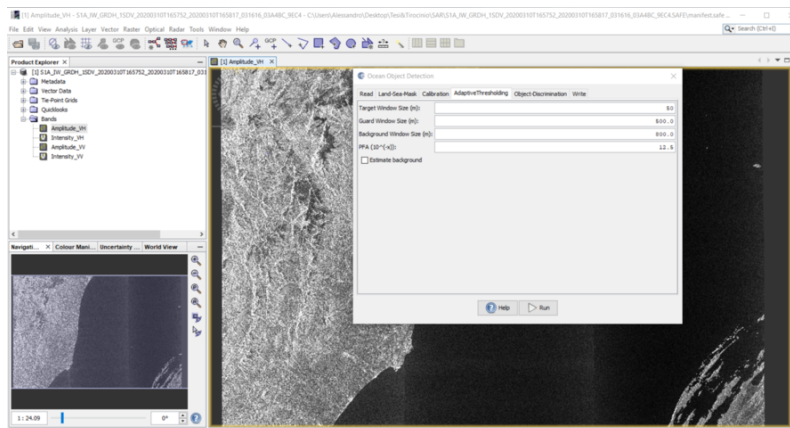


Figura 2.6 – Sentinel Application Platform

Le sue caratteristiche principali sono:

- Architettura comune per tutti i *Toolbox*
- Visualizzazione delle immagini molto veloce e navigazione anche di immagini giga-pixel

- *Graph Processing Framework* (GPF): per la creazione di catene di elaborazione definite dall'utente.
- La gestione avanzata dei livelli consente l'aggiunta e la manipolazione di nuovi *overlay* come immagini di altre bande, immagini da server *WMS* o *shapefile ESRI*
- Definizioni ricche di regioni di interesse per statistiche e grafici vari
- Facile definizione e sovrapposizione di maschere di bit
- Aritmetica a banda flessibile che utilizza espressioni matematiche arbitrarie
- Riproiezione accurata e orto-rettifica a proiezioni cartografiche comuni
- Geo-codifica e rettifica mediante punti di controllo a terra
- Download automatico SRTM DEM e selezione delle tessere
- Libreria di prodotti per la scansione e la catalogazione efficiente di archivi di grandi dimensioni
- Supporto multithreading e multi-core
- Visualizzazione WorldWind integrata

Nella figura 2.7 è mostrata la catena di elaborazione, creata con il GPF, per poter effettuare il rilevamento delle imbarcazioni.

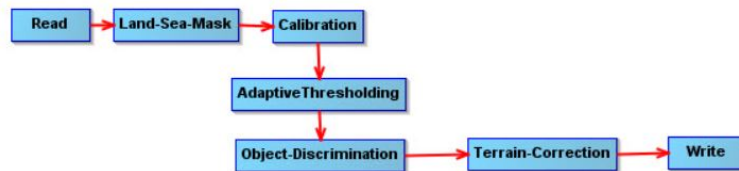


Figura 2.7 – Graph Builder per rilevamento navi

2.4 SUMO

SUMO [12] è un pacchetto software per il rilevamento semi o completamente automatico delle navi nelle immagini satellitari SAR. Può lavorare sulle immagini della maggior parte dei *Sentinel* recenti e contemporanei, su una gamma di frequenze radar e risoluzioni. È stato sviluppato presso il Centro comune di ricerca della Commissione Europea (JRC) negli ultimi 15 anni ed è stato ampiamente testato, dimostrato e convalidato. Il rilevamento delle navi dalle immagini satellitari viene utilizzato per la pianificazione dello spazio marittimo, il controllo della pesca, dell'inquinamento e delle frontiere marittime, lotta alla pirateria e altre applicazioni nel campo della sicurezza marittima, protezione e sostenibilità. In figura 2.8 è mostrata l'interfaccia di SUMO.



Figura 2.9 – Logo PostgreSQL [13]

Ha guadagnato una solida reputazione per la sua comprovata architettura, affidabilità, integrità dei dati, robusto set di funzionalità, estensibilità e la dedizione della comunità open source dietro il software per fornire costantemente soluzioni performanti e innovative. PostgreSQL funziona su tutti i principali sistemi operativi e dispone di potenti componenti aggiuntivi come PostGIS che vedremo in seguito.

In esso verranno immagazzinate le tabelle con tutti i dati AIS a disposizione, in maniera tale da avere accesso a specifiche informazioni con delle semplici interrogazioni al database.

2.5.1 PostGIS

PostGIS è un'estensione spaziale per database relazionali a oggetti PostgreSQL. Aggiunge il supporto per gli oggetti geografici consentendo l'esecuzione di *query* sulla posizione in SQL. Oltre al riconoscimento della posizione di base, PostGIS offre molte funzionalità che raramente si trovano in altri database spaziali [14]. Esso aggiunge tipi extra (geometria, geografia, raster e altri) al database PostgreSQL. Aggiunge anche funzioni, operatori e miglioramenti dell'indice che si applicano a questi tipi spaziali. Queste funzioni, operatori, associazioni di indici e tipi aggiuntivi aumentano la potenza del DBMS PostgreSQL di base, rendendolo un sistema di gestione del database spaziale veloce, ricco di funzionalità e robusto. Il suo utilizzo è fondamentale per poter correlare i dati AIS con i risultati dell'elaborazione delle immagini SAR. Infatti, per esempio, permette di calcolare distanze, di creare dei buffer attorno a dei punti e creare delle linee dati due estremi, le quali sono funzioni necessari per il raggiungimento dell'obiettivo.

2.6 QGIS

QGIS è un Sistema di Informazione Geografica (GIS) Open Source facile da usare, rilasciato sotto la GNU (General Public License) [15]. Esso permette di analizzare ed editare dati spaziali e di generare cartografia. QGIS supporta sia dati vettoriali che raster oltre che i principali database spaziali come PostgreSQL/PostGIS. La forza di QGIS, inoltre, è che integra al suo interno gli algoritmi di processing di altri progetti open source in una interfaccia intuitiva, come si vede in figura 2.10.



Figura 2.10 – Interfaccia QGIS [15]

E' possibile aggiungere dei *layer*, per esempio importando dei file *csv*, così da avere un riscontro visivo di ciò che è stato elaborato attraverso gli script e questo risulterà molto utile in fase di verifica ed analisi dei dati ottenuti.

2.7 Python e le sue librerie

Python è un linguaggio di programmazione di più "alto livello" rispetto alla maggior parte degli altri linguaggi, orientato a oggetti, adatto, tra gli altri usi, a sviluppare applicazioni distribuite, scripting, computazione numerica e *system testing* [16].



Figura 2.11 – Logo Python

Le sue caratteristiche principali [17] consistono in:

- Il linguaggio e' interpretato, ovvero tradotto in linguaggio macchina una istruzione per volta, tipo la shell di Unix, e' quindi un linguaggio che può essere usato in modo interattivo, ma inefficiente dal punto di vista del calcolo. Un programma Python può anche essere messo in un file, ed eseguito, come un'unica procedura; durante la traduzione viene prodotto un "byte-code", ovvero una versione del programma semi compilata, posta in un file con estensione ".pyc". Se il programma viene eseguito una seconda volta, senza modifiche, si utilizza direttamente il bytecode, con un vantaggio in termini di prestazioni.
- La sintassi e' semplice, ci sono pochi costrutti e non c'è bisogno di dichiarare il tipo delle variabili. Non ci sono puntatori, anche se tutte le variabili sono implementate con puntatori. Il Python si occupa di individuare il tipo di ogni variabile quando viene definita, e fa conversioni automatiche quando occorre.
- Ha un modo semplice ma efficace di definire e trattare classi, e praticamente tutti i tipi di dati sono oggetti. C'è anche la possibilità di definire "moduli": insieme di funzioni Python predefinite, conservate in un file, che possono essere riutilizzate, inserendole in programmi diversi, senza che ci sia confusione fra le variabili del programma e quelle dei moduli stessi.
- Comprende una implementazione di strutture complesse, di uso comune in programmazione, come: liste, dizionari, tuple, stringhe. In Python queste sono trattate come tipi di dati e ci sono già funzioni per agire su di esse, come funzioni di ordinamento, di ricerca etc.
- Arricchito di un elevato numero di librerie di terze parti facilmente integrabili, mediante l'utilizzo di un *package manager* (come ad esempio *pip*)

Sono elencate in seguito le principali librerie che sono state utilizzate per lo sviluppo del progetto, integrate attraverso il *package manager*, come descritto dall'ultimo punto precedente.

2.7.1 Pandas

Pandas fornisce strutture dati veloci, flessibili ed espressive progettate per rendere il lavoro con dati "relazionali" o "etichettati" sia facile che intuitivo [18]. Mira a essere l'elemento fondamentale di alto livello per eseguire analisi pratiche e reali dei dati in Python. Inoltre, ha l'obiettivo più ampio di diventare lo strumento di analisi/manipolazione dei dati open source più potente e flessibile disponibile.

Le sue strutture dati principali sono le Series e i Dataframe, attraverso i quali diventa davvero semplice manipolare dataset e estrarre ciò che si desidera. In questo caso verranno usati per poter lavorare con i dati dei csv ottenuti dalle elaborazioni delle immagini, rendendo possibili comparazioni o inserimenti di dati in maniera veloce e potente.

2.7.2 Psycopg2

Psycopg è l'adattatore per database PostgreSQL più popolare per il linguaggio di programmazione Python. Le sue caratteristiche principali sono l'implementazione completa della specifica Python DB API 2.0 e la *thread safety* (diversi thread possono condividere la stessa connessione). È stato progettato per applicazioni fortemente multi-thread che creano e distruggono molti cursori e fanno un gran numero di "INSERT" o "UPDATE" simultanei [19].

Psycopg 2 è implementato principalmente in C come *wrapper libpq*, risultando sia efficiente che sicuro. Presenta cursori lato client e lato server, comunicazioni e notifiche asincrone, supporto "COPIA A / COPIA DA". Molti tipi di Python sono supportati immediatamente e adattati alla corrispondenza con i tipi di dati PostgreSQL; l'adattamento può essere esteso e personalizzato grazie ad un sistema flessibile di adattamento degli oggetti.

2.7.3 Matplotlib e Seaborn

Matplotlib è un modulo per la generazione di grafici 2D, completamente sviluppata in Python ed utilizza il modulo *Numpy* per la rappresentazione di grandi *array* [20]. E' costituita da tre componenti:

- La matplotlib API (interfaccia ad oggetti), utilizzata direttamente per inserire funzionalità per la creazione di grafici nei propri script Python
- Il modulo Pyplot è un'interfaccia procedurale alla matplotlib API ed è progettato per emulare, in ambiente Python, i più comuni comandi grafici di Matlab; dunque è pensato prevalentemente per un utilizzo interattivo.
- Output back-end: per l'output dei grafici su varie tipologie di GUI e su diversi formati file

Seaborn è una libreria per creare grafici statistici in Python. È costruito su matplotlib e strettamente integrato con le strutture dati di pandas [21]. Ecco alcune delle funzionalità offerte da Seaborn:

- Un'API orientata al set di dati per esaminare le relazioni tra più variabili
- Supporto specializzato per l'utilizzo di variabili categoriali per mostrare osservazioni o statistiche aggregate
- Stima automatica e rappresentazione grafica di modelli di regressione lineare per variabili dipendenti di diversi tipi
- Comode viste sulla struttura complessiva di set di dati complessi
- Astrazioni di alto livello per strutturare griglie multi-trama che consentono di creare facilmente visualizzazioni complesse
- Controllo conciso sullo stile delle figure matplotlib con diversi temi incorporati

- Strumenti per la scelta delle tavolozze dei colori che rivelano fedelmente i modelli nei dati

Seaborn mira a rendere la visualizzazione una parte centrale dell'esplorazione e della comprensione dei dati. Le sue funzioni di plottaggio orientate ai dataset operano su dataframe e array contenenti interi dataset ed eseguono internamente la mappatura semantica e l'aggregazione statistica necessarie per produrre grafici informativi. Tramite queste due librerie è stato possibile creare delle mappe di calore (heatmap) per poter trarre delle conclusioni dai dati ottenuti con gli script.

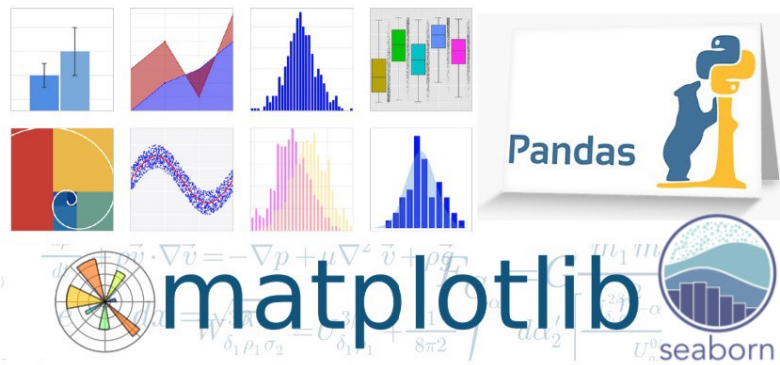


Figura 2.12 – Python Data Visualization

Capitolo 3

Sviluppo progetto

Dopo aver spiegato in dettaglio gli strumenti utilizzati e aver descritto il contesto in cui ci si trova, in questo capitolo viene spiegato come è stato realizzato il progetto, descrivendo attentamente ogni procedimento che ha portato alla creazione di un meccanismo robusto e coerente che sia in grado di identificare eventuali imbarcazioni praticanti l'attività di pesca illegale.

Basandosi su i dati AIS disponibili, si è scelto di analizzare la settimana più popolosa a livello di informazioni, in maniera tale da incrementare le possibilità di ottenere riscontri. Quindi verificando il database sono stati scelti i giorni che vanno dal 19 Luglio al 26 Luglio del 2019, prendendo solamente cinque giorni, in quanto per tre di essi non erano disponibili le foto satellitari. In più è stata anche scelta la data del 9 Marzo nei tre anni "2018-19-20". L'elaborazione di queste immagini è stata effettuata seguendo i passi elencati in seguito.

3.1 Download immagini SAR

Una volta decise le date su cui effettuare l'analisi, si procede con la richiesta nei cataloghi dell'ESA delle immagini scattate dal Sentinel-1 SAR. E' stata seguita la linea guida descritta nel secondo capitolo, e per ogni data è stata presa una foto scattata alle 5.00 o alle 17.00, in base al moto orbitale del satellite, facendo attenzione alla zona scelta, verificabile dalla preview di ogni foto, come mostrato in figura 3.1. Molte di esse erano offline in quanto ormai datate, quindi è stato necessario attendere che il catalogo le rendesse di nuovo disponibili per poi procedere con il download.

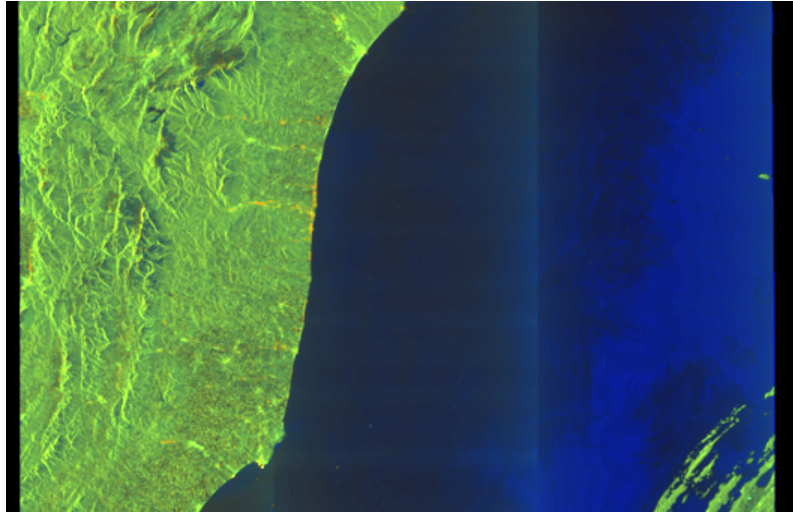


Figura 3.1 – Preview immagine SAR

3.2 Elaborazione immagini

Una volta in possesso delle foto scattate dal satellite nei giorni desiderati, è necessario effettuare un'elaborazione tramite SUMO, in maniera tale da identificare le macchioline luminose che rappresentano le imbarcazioni presenti in mare quel giorno e avere informazioni utili su di esse, come ad esempio la loro latitudine e longitudine.

- Il primo passo, una volta lanciato il programma, è quello di caricare l'immagine su di esso per poi procedere con i vari strumenti di analisi. Fondamentale è l'utilizzo della *landmask*, la quale permette di mascherare la terra e quindi evitare che vengano riscontrate imbarcazioni che non siano in acqua, poichè altrimenti si potrebbe confondere la luminosità della zona costiera, dovuta a tutte le strutture metalliche presenti, con delle navi. Questa è costituita da uno *Shapefile*, un popolare formato vettoriale per sistemi informativi geografici. Il formato è stato sviluppato e regolato da Esri, allo scopo di accrescere l'interoperabilità fra i sistemi ESRI e altri GIS. Di fatto è diventato uno standard per il dato vettoriale spaziale, e viene usato da una grande varietà di sistemi GIS. Con "shapefile" si indica di norma un insieme di file con estensione .shp, .dbf, .shx, altre che hanno in comune il prefisso dei nomi (per es. "laghi.*"). Spesso con shapefile si indica però solo i file ".shp". Tuttavia questo file da solo è incompleto poiché interpretazione ed utilizzo dipendono dagli altri file. In caso di assenza (o corruzione) dei file DBF, SHX è comunque possibile caricare la sola geometria .SHP mediante un collegamento FDO in ambiente Autocad cartografico [22]. Gli shapefile descrivono spazialmente punti, polilinee e poligoni, utilizzabili nel nostro caso per creare una geometria che mascheri la terra, ovvero la linea mostrata nella figura 3.2.



Figura 3.2 – Landmask SUMO

- Il secondo passo è quello di selezionare i parametri per l'analisi che porterà al rilevamento delle imbarcazioni nell'apposito menù. Infatti basterà selezionare la voce *VDS Analysis* per poter scegliere il giusto settaggio, andando a specificare quale landmask è stata utilizzata e quali sono i valori della soglia di rilevamento, uno per ogni tipo di polarizzazione in cui è stata scattata la foto (VV o VH). Dopo alcuni minuti, tempistica decisamente migliore rispetto a quella che si avrebbe avuto con l'ambiente SNAP, si ottiene l'analisi desiderata, mostrata in figura 3.3. Per la banda di polarizzazione VV sono usati dei rettangoli rossi mentre per la banda VH vengono presi dei triangoli verdi e alla fine vengono combinati i due risultati sotto forma di croci gialle. Per questo progetto è stato deciso di utilizzare la banda VH in quanto risultava più precisa rispetto alle altre nel rilevare le navi.

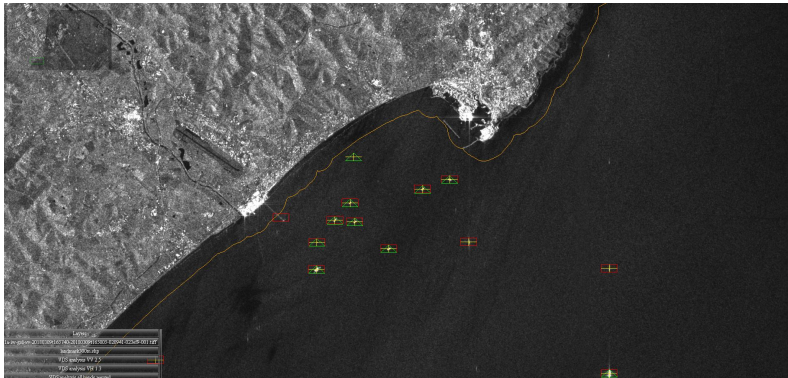


Figura 3.3 – Analisi SUMO

- Una volta ottenuta l'elaborazione finale, questa viene salvata in un file in formato CSV, contenente tutte le informazioni delle navi trovate, così da poter essere analizzate con gli script in Python. Osservando il risultato dopo l'analisi si è in grado di verificare anche la presenza di falsi positivi che verranno poi eliminati nelle fase successive in maniera tale da lavorare con dei dati più veritieri possibili. In figura 3.4 per esempio si può vedere come la scia di una nave è stata considerata un target.

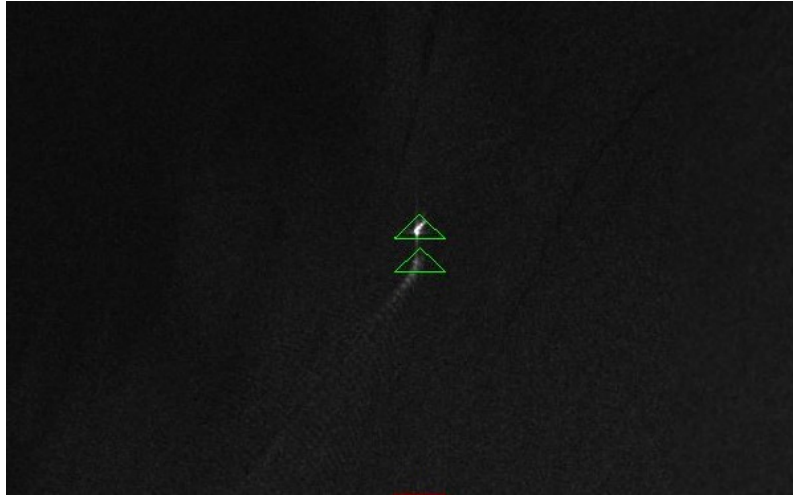


Figura 3.4 – Falso positivo

3.3 Script Python

Il termine script [23], in informatica, designa un tipo particolare di programma, scritto in una particolare classe di linguaggi di programmazione, detti linguaggi di scripting. Una classe specifica di tali programmi sono i cosiddetti shell script, ossia script concepiti per essere eseguiti all'interno di una shell di un sistema operativo. La distinzione tra un programma normale ed uno script non è netta, ma generalmente negli script si possono individuare le seguenti caratteristiche:

- complessità relativamente bassa
- utilizzo di un linguaggio interpretato
- una certa linearità (uno script può anche accettare input dall'utente, ma solitamente input diversi non modificano sostanzialmente la struttura del diagramma a blocchi che descrive il comportamento dello script)
- mancanza di una propria interfaccia grafica
- richiamo di altri programmi per svolgere operazioni più sofisticate

Per poter costruire un metodo robusto di analisi e comparazione dei dati ottenuti è stato necessario servirsi del linguaggio di programmazione Python in maniera da creare sette script per il raggiungimento dell'obiettivo:

1. *buffer55nm-Ancona.py*
2. *false-positive.py*
3. *ShipWithoutPlatforms.py*
4. *sarais.py*
5. *unk-segment.py*

6. *p500-1nm.py*

7. *heatmap.py*

- Il primo ha il compito di prendere tutti i target del file csv e verificare quali di essi ricadano in un buffer di 55 miglia dall'antenna situata ad Ancona eliminando poi quelli che non rispettano questa condizione.
- Attraverso il secondo si è in grado di eliminare, dal risultato del primo script, i falsi positivi descritti precedentemente.
- Con il terzo si va a comparare le coordinate di tutte le piattaforme note nel mare Adriatico con i target rimasti dopo i primi due filtri. Una volta effettuata questa analisi si va a salvare in un csv i target che corrispondono alle piattaforme e i restanti che quindi saranno delle imbarcazioni andranno in un altro file csv.

Questi primi tre script erano comuni ad entrambi gli approcci utilizzati per il progetto, mentre gli ultimi quattro verranno analizzati successivamente in quanto specifici dell'associazione punto-punto e punto-linea, metodi che saranno trattati nel dettaglio in seguito. Fondamentale però per poter comparare i dati AIS con i target SAR è stata la possibilità di stabilire una connessione con il database in cui erano immagazzinati i vari dati AIS.

3.3.1 Interrogazione database

Il progetto si basa sul confronto tra le informazioni SAR e i dati AIS, ma per poter aver accesso a quest'ultimi è necessario stabilire una connessione al database PostgreSQL, così da poter estrarre ciò di cui si ha bisogno con apposite *query*. Inoltre grazie all'utilizzo dell'estensione Postgis, con delle interrogazioni è stato possibile effettuare dei confronti spaziali tra le due tipologie di dati, come misurare distanze o creare dei buffer. Per poter fare tutto ciò bisogna importare la libreria di Python denominata Psycopg2 e scrivere una serie di istruzioni fondamentali qui elencate:

```
1 import psycopg2 as psy
2 try:
3     # Stabilisce connessione passando i parametri del db.
4     conn=psy.connect(host="localhost", database="CovidAIS", user="postgres",
5                       password="0000")
6
7     # Crea cursor object.
8     cur = conn.cursor()
9     if(conn):
10        print("Connessione stabilita...")
11
12    # Query al db
13    cur.execute("select * from full_tab")
14    result=cur.fetchall()
15
16    # In caso di errore avverte.
17    except (Exception, psy.Error) as error :
18        print ("Errore", error)
19
20    # Chiude la connessione.
21 finally:
```

```

20     if(conn):
21         cur.close()
22         conn.close()
23         print("Connessione PostgreSQL chiusa")

```

Da questo estratto di codice si può vedere come venga all'inizio stabilita la connessione passandogli i parametri necessari, poi viene creato un cursore sui cui salvare le informazioni ottenute dalle interrogazioni. Fondamentale poi è chiudere la connessione anche in caso di errore, reso possibile grazie al costrutto *try* e *finally*.

3.3.2 Associazione Punto-Punto

Il primo approccio utilizzato è denominato punto-punto, il cui nome deriva dal fatto che ad un target SAR viene associato un ping AIS, così da identificare a quale nave corrispondesse quella macchiolina luminosa nella foto satellitare. Per fare ciò è stato utilizzato il quarto script precedentemente citato, "*sarais.py*", nel quale per prima cosa si importano subito le librerie necessarie e si stabilisce la connessione con il database. In seguito per avere un'associazione il più reale possibile si è seguita questa scaletta:

- Una volta processati i dati provenienti da SUMO con i primi tre script, viene salvato il csv contenente le sole imbarcazioni, ottenuto con "*ShipWithoutPlatforms.py*", all'interno di un Dataframe, struttura dati alla base della libreria pandas, così da riuscire a manipolare tali dati in maniera facile. La sintassi è mostrata qui di seguito.

```

1     # Salva nel dataframe i target SAR del csv.
2     df = pd.read_csv(r'csv\script\26_07_2019\onlyShip26_07_2019.csv')
3

```

- Successivamente viene eseguita una *query* al database per poter estrarre i dati AIS desiderati, i quali devono rispettare alcune condizioni. In primis vengono scelti i dati compresi nel periodo di tempo che va da venti minuti prima e venti minuti dopo la data in cui è stata scattata la foto SAR, ma in seguito si va a verificare se questi rientrano in un buffer di due chilometri, centrato nel target SAR presente nel csv. Alla fine quindi si avrà un nuovo Dataframe contenente, per ogni target SAR, tutti i ping che rientrano nel proprio buffer, i quali possono essere più di uno per SAR e per questo necessitano di ulteriore filtraggio.
- In seguito, viene calcolata sia la distanza effettiva tra il ping e il target, attraverso l'utilizzo di PostGis, che la distanza potenziale che l'AIS può aver percorso da quando è stata scattata la foto SAR, servendosi della velocità a cui la nave stava andando e la differenza di tempo tra SAR e AIS, dati che vengono presi dal Dataframe e poi moltiplicati per ottenere lo spazio percorso, come viene mostrato dal codice seguente:

```

1     # Calcolo lo spazio percorso da ciascun ping e lo salvo in una
2     # nuova colonna del dataframe.
3     print("Calcolo spazio percorso...")
4     space=[]
5     for vel,temp in zip(df2['msec'],df2['datetime']):

```

```

5         dtt=time.mktime(temp.timetuple())
6         t=abs(dtSar-dtt)
7         sp = vel * t
8         space.append(sp)
9         df2['space(m)']= space
10

```

- Nel caso in cui ad un target sono associati più ping, il secondo livello di filtraggio si basa sulla vicinanza spaziale, calcolata precedentemente.

In questo modo si ottiene un'associazione univoca tra il dato AIS e il target SAR, la quale segue i criteri di vicinanza temporale e spaziale per renderla la più veritiera possibile.

3.3.3 Associazione Punto-Linea

Il secondo approccio è denominato punto-linea, poichè questa volta il target SAR non sarà più associato ad un ping AIS, ma si cercherà una corrispondenza con una linea di spegnimento. Queste linee non sono altro che dei periodi di blackout in cui le imbarcazioni hanno spento e poi riacceso il trasmettitore AIS, e così ,essendo noti solamente i due ping prima e dopo lo spegnimento, viene tracciata una linea tra questi due punti supponendo che la nave abbia percorso quella traiettoria. Successivamente si andrà ad indagare se ha veramente seguito quel tragitto o se abbia effettuato manovre per finire in aree vietate. Per questa associazione è stato utilizzato il quinto script, "*unk-segment.py*", in cui anche qui viene subito stabilita la connessione con il database per poter effettuare delle *query* spaziali. Le operazioni successive consistono in:

- Salvare in un Dataframe tutte le linee di spegnimento che si hanno a disposizione
- Effettuare un primo filtraggio, andando a selezionare solamente le linee il cui spegnimento rientri in un intervallo che va dai trenta minuti alle due ore e mezza.
- In seguito viene effettuato un secondo filtraggio sulle linee derivate dal risultato precedente, in cui vengono prese solamente le linee che comprendono la data della foto SAR desiderata tra la propria data di inizio e fine.
- Successivamente per ogni linea viene calcolata la sua distanza da ogni target SAR e le viene associato solamente quello la cui distanza risulta minima, a meno che non sia già stato assegnato ad un'altra linea ed in tal caso verrà considerato il secondo più vicino.
- Come ultimo passo vengono poi eliminate tutte le linee che non rientrano nella zona di 55 miglia dall'antenna situata ad Ancona.

Come nel caso punto-punto, si arriva ad ottenere un'associazione univoca tra il SAR e la linea di spegnimento.

3.3.4 Previsione traiettoria nel periodo di blackout e heatmap

Per monitorare l'attività di pesca, quello che si cerca di fare, è verificare se le navi durante il loro periodo di blackout sono andate a pescare in zone protette, come la zona di cinquecento metri intorno ad una piattaforma. Ciò è reso possibile grazie all'utilizzo degli ultimi due script, "*p500-1nm.py*" e "*heatmap.py*", con i quali si controllerà se le imbarcazioni durante lo spegnimento hanno seguito veramente una linea retta che le impedisca di finire in aree vietate o se il loro percorso si discosta molto da essa. Innanzitutto con il primo dei due codici si va a filtrare le linee ottenute con l'elaborazione di "*unk-segment.py*", prendendo solamente quelle che intersecano un buffer di cinquecento metri o un miglio da una piattaforma, in quanto sono le più probabile ad avere commesso un'attività illegale.

```

1 print('Filtraggio p1nm e p500m...')
2 i=0
3 lst=[]
4 for ind,ind1 in zip(df['p1nm'],df['p500m']):
5     if ind=='Y' or ind1=='Y':
6         lst.append(df.loc[i])
7     i+=1
8 df1 = pd.DataFrame(lst)

```

In seguito con il secondo codice si andrà ad analizzare proprio quest'ultime, creando una *heatmap*, ovvero una rappresentazione grafica dei dati in cui viene utilizzato uno spettro di colori per rappresentare valori diversi [24], che mostrerà graficamente il risultato di questa analisi. La linea guida per studiare le traiettorie delle navi è la seguente:

- Per ogni linea viene calcolata la distanza tra essa e le piattaforme, associandole poi solamente le tre più vicine.
- In seguito viene calcolata la distanza minima per cui l'imbarcazione possa intersecare un buffer di cinquecento metri attorno ad una piattaforma, considerando gli estremi della linea e il punto del buffer più vicino ad essa.
- Viene anche calcolata la distanza potenzialmente percorsa dalla nave, essendo noto per quanto tempo è stato spento il dispositivo AIS e la velocità a cui stava andando, stimata in base alle ultime due velocità note.

Prevedere il tragitto effettuato dall'imbarcazione durante lo spegnimento, è reso possibile quindi grazie al rapporto tra la distanza potenziale e quella minima, in quanto se esso sarà all'incirca pari all'unità, vorrà dire che la traiettoria coincide con la linea retta, mentre nel caso in cui risulti molto maggiore di uno, significa che la distanza percorsa risulta superiore a quella che bastava per finire in una zona protetta. Il risultato di questi rapporti può essere visualizzato nella heatmap mostrata in figura 3.5.

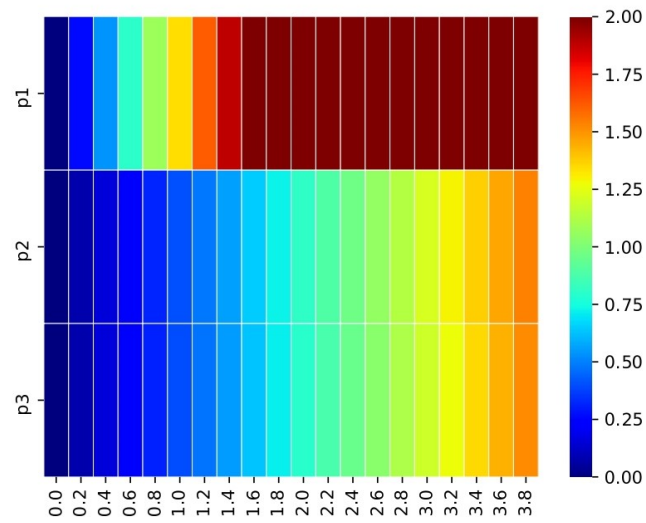


Figura 3.5 – Heatmap

Nell'ascissa è riportato un range di velocità centrato nel valore a cui era stata stimata la velocità della nave durante lo spegnimento mentre nell'ordinata è mostrato a quale piattaforma sono associati i rapporti. Si può notare che a destra è riportata la legenda che indica a quale colore corrisponde il valore del rapporto. Quindi, per questo particolare esempio, se l'imbarcazione fosse andata a una velocità di due nodi, probabilmente non entrerà nelle zone protette delle ultime due piattaforme in quanto la colorazione indica un rapporto pari ad uno, ma sicuramente entrerà nell'area vietata della prima piattaforma.

Capitolo 4

Analisi Risultati

In questo capitolo verrà approfondita l'interpretazione dei dati ottenuti dalle varie elaborazioni, in quanto risulta fondamentale dare un significato e un contesto ai risultati ricavati.

4.1 Risultati Tabelle

Di seguito vengono riportate le tabelle contenenti l'intera mole di dati derivante dagli script, le quali saranno poi in dettaglio spiegate.

Tabella 4.1 – Dati ottenuti dai primi tre script

DATA	SUMO(55nm*)	FP*	P*
19/07/2019 05:10:31	253 (117)	1	28
20/07/2019 16:57:09	128 (37)	0	14
24/07/2019 05:18:43	222 (57)	2	10
25/07/2019 05:11:14	395 (185)	2	28
26/07/2019 16:57:52	144 (39)	0	17
09/03/2018 16:57:40	147 (39)	0	17
09/03/2019 05:10:32	228 (116)	2	36
09/03/2020 05:11:14	321 (187)	2	28

*55nm = entro i 55 miglia dall'antenna di Ancona, *FP = falsi positivi,

*P = Match Piattaforme

Nella tabella 4.1 oltre alla data in cui è stata scattata la foto SAR sono presenti altre tre colonne che rappresentano:

- Il numero totale di target identificati da SUMO e tra parentesi quali di essi ricadano nei 55 miglia da Ancona, ovvero la distanza massima coperta dall'antenna AIS lì installata.

- La seconda colonna mostra il numero di falsi positivi rilevati
- Mentre la terza indica il numero di target SAR che sono risultati coincidere con le piattaforme.

Tabella 4.2 – Associazione punto-punto

DATA	AIS(C)*	SA*	MMSI*
19/07/2019 05:10:31	156 (79)	9	43
20/07/2019 16:57:09	97 (42)	5	36
24/07/2019 05:18:43	491 (112)	10	101
25/07/2019 05:11:14	476 (144)	25	103
26/07/2019 16:57:52	150 (60)	4	43
09/03/2018 16:57:40	210 (112)	5	47
09/03/2019 05:10:32	254 (139)	9	50
09/03/2020 05:11:14	511 (199)	19	93

*AIS(C)=ping disponibili nei 55 miglia (ping nella costa), *SA= match SAR-AIS, *MMSI= quantità di identificativi delle imbarcazioni

Nella tabella 4.2 sono mostrati i dati ottenuti durante il processo di associazione punto-punto. In essa infatti si possono notare:

- Il numero di ping AIS totali nei 55 miglia da Ancona e tra parentesi il numero di essi che è situato in prossimità della costa.
- I match riscontrati tra target SAR e ping AIS
- Infine nell'ultima colonna sono mostrati quanti MMSI, ovvero l'identificativo di ogni nave, diversi erano presenti in quell'ora.

Tabella 4.3 – Associazione punto-linea

Data	L(55nm)*	LP(55nm)*
19/07/2019 05:10:31	22 (3)	0 (0)
20/07/2019 16:57:09	46 (4)	5 (2)
24/07/2019 05:18:43	37 (7)	4 (1)
25/07/2019 05:11:14	33 (9)	3 (1)
26/07/2019 16:57:52	31 (5)	3 (1)
09/03/2018 16:57:40	5 (5)	0 (0)
09/03/2019 05:10:32	4 (2)	0 (0)
09/03/2020 05:11:14	6 (6)	0 (0)

*L = Linee rilevate, *LP = Linee che intersecano buffer piattaforma.

Infine nella tabella 4.3 sono presenti i risultati ottenuti dall'approccio punto-linea. In essa si può notare:

- La quantità di linee di spegnimento rilevate, che rispettino le condizioni descritte nel capitolo precedente, e tra parentesi quante di esse siano nella zona di Ancona.
- Nell'ultima colonna è presente il numero di linee che intersecano un buffer intorno ad una piattaforma, e quindi potenziali candidate ad aver commesso attività illecite.

4.2 Analisi dati ottenuti

La prima cosa che si nota nella tabella 4.1 è la grande differenza, tra le varie date, di target rilevati della zona di Ancona, fatto che dipende dall'angolazione in cui è stata scattata la foto SAR in quanto a volte la zona del buffer di cinquantacinque miglia potrebbe risultare minore in alcune immagini rispetto ad altre. Sapendo che nella zona di questo buffer sono presenti in totale 42 piattaforme, si nota come in alcuni giorni il grado di accuratezza non sia molto elevato, cosa che dipende dallo stesso motivo descritto precedentemente. Riguardo ai falsi positivi, in quanto la quantità rilevata risulta essere decisamente minima, si può concludere che l'algoritmo di rilevamento SUMO è molto preciso.

La tabella 4.2 necessita di un'approfondita spiegazione, poichè i risultati non sono di facile interpretazione. Infatti da essa evince che molti dei ping AIS presenti nei vari giorni non abbiano trovato nessuno riscontro, dato che il match con il SAR è stato riscontrato per pochi di essi. Innanzitutto bisogna notare quante imbarcazioni erano presenti quel giorno a quell'ora osservando l'ultima colonna, dato che ci si aspetterebbe che quel numero corrispondesse con i match totali, però ci sono alcune considerazioni da fare:

1. Per prima cosa molti di quei ping emessi risultano vicini alla costa e quindi quegli MMSI potrebbero appartenere a delle navi che stavano rientrando in porto per cui non abbiamo target SAR da associare ad essi
2. Un'altra considerazione importante sta nel fatto che nella grande quantità di target SAR derivanti da SUMO potrebbero esserci imbarcazioni di dimensioni inferiori ai 15 metri e quindi non dotate di dispositivi AIS, motivo per cui non si è trovato un match e molti target non sono stati associati

Infine nella tabella 4.3 viene riportata la quantità di linee di spegnimento rilevata per ogni giorno, incluso il filtraggio che mostra quante di esse era inclusa nella zona di Ancona. Nella colonna finale invece è presente il numero di esse che rispetti la condizione di intersecare un buffer di cinquecento metri o un miglio intorno ad una piattaforma.

4.2.1 Confronto risultati su QGIS

L'utilizzo dell'ambiente QGIS è stato fondamentale per riuscire ad interpretare i dati ottenuti e dare un riscontro visivo di ciò che si era andato fare. Nella figura 4.1 viene mostrato uno screen in cui si riassume l'intera analisi effettuata e il quale mostra con precisione ogni aspetto precedentemente descritto.

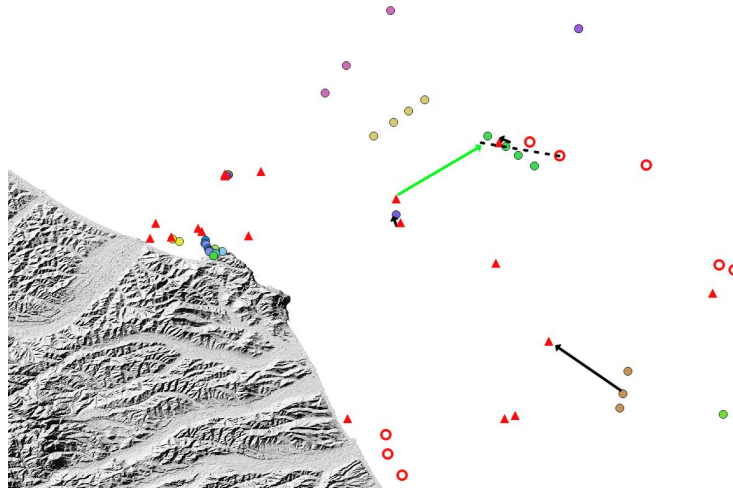


Figura 4.1 – Screen QGIS

La figura rappresenta l'analisi del 20/07/2019 ed in essa sono presenti:

- I ping AIS rappresentati dai pallini, categorizzati in base all'MMSI, quindi ognuno con il proprio colore
- I target SAR costituiti dai triangoli rossi
- La linea di spegnimento è tratteggiata in nero
- Le piattaforme rappresentate dai cerchi rossi
- L'associazione punto-punto è costituita dalle frecce nere
- L'associazione punto-linea dalla freccia verde

Si nota subito come la prima considerazione fatta nell'analisi precedente sia spiegata da quest'immagine in quanto nella zona del porto sono presenti innumerevoli ping AIS i quali non sono associati a nessun target, poichè non rientrano nei due chilometri da nessun triangolo rosso. Si vede anche la presenza di alcuni triangoli non associati a nulla il che può essere spiegato dalla seconda considerazione precedentemente fatta. Per mostrare in maniera migliore le associazioni derivanti dai due approcci utilizzati nella figura 4.2 è riportato uno zoom dell'immagine precedente che mostra in dettaglio i match riscontrati.

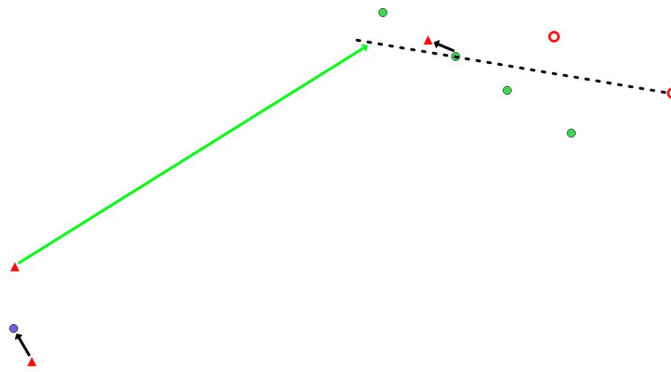


Figura 4.2 – Zoom Screen QGIS

L'associazione punto-punto rispetta i criteri di vicinanza spaziale e temporale, infatti nel caso dei ping verdi è stato scelto quello emesso alle 16:57, ora in cui è stata anche scattata la foto SAR mentre nel caso dei due triangoli rossi vicini al ping blu, è stato associato il triangolo spazialmente più vicino. Di conseguenza l'ultimo triangolo rimasto è stato associato alla linea di spegnimento, che risulta essere molto vicina alle due piattaforme in figura e per questo potrebbe aver probabilmente effettuato manovre che l'abbiano portate in aree protette.

Capitolo 5

Conclusioni

Le nostre acque sono costantemente monitorate e si hanno a disposizione molti strumenti per sventare attività illegali che potrebbero danneggiare l'ambiente e causare danni economici. Infatti con lo sviluppo di tecnologie sempre più avanzate, in tutti i campi si cerca proprio di contrastare problematiche una volta complesse da contenere ma ormai gestibili più facilmente grazie all'utilizzo di strumenti all'avanguardia. Il progresso risulta infatti costante, così da permettere negli anni di migliorare notevolmente il campo informatico, il quale al giorno d'oggi presenta ancora alcuni limiti. Alcuni di essi sono stati inevitabilmente riscontrati nello sviluppo di questo progetto, come ad esempio:

- I satelliti non sono in grado di ottenere una mappatura del globo di ogni ora rendendo indisponibili foto che sarebbero state utili per effettuare confronti ed ulteriori analisi
- Un ambiente come SNAP che offre innumerevoli strumenti e possibilità di analisi purtroppo richiede elaboratori potenti senza i quali si avrebbero tempistiche insoddisfacenti
- I dispositivi AIS perdono una grande quantità di dati a causa di condizioni atmosferiche sfavorevoli rendendo più difficile l'associazione

5.1 Obiettivi raggiunti

Nonostante i vari limiti precedentemente descritti, i quali possono aver creato un lieve margine di errore nei risultati ottenuti, si può affermare che l'implementazione di un meccanismo di monitoraggio di attività di pesca illegale è stata completata con successo. Partendo da strumenti messi a disposizione dall'ESA, si è poi stati in grado di elaborare una pipeline da seguire per poter rilevare le imbarcazioni e attraverso apposite librerie del linguaggio di programmazione Python si è potuto analizzarle per trovare eventuali navi che avessero commesso manovre illecite.

Bibliografia

- [1] WWF. Pesca eccessiva e pesca illegale. https://www.wwf.it/il_pianeta/impatti_ambientali/pesca_eccessiva_e_illegale.
- [2] Wikipedia contributors. Big data. https://it.wikipedia.org/wiki/Big_data.
- [3] Copernicus. https://www.copernicus.eu/sites/default/files/documents/Brochure/Copernicus_Brochure_IT_WEB.pdf.
- [4] ESA. Missione sentinel 1. <https://sentinel.esa.int/web/sentinel/missions/sentinel-1>.
- [5] ESA. Obiettivi sentinel 1. <https://sentinel.esa.int/web/sentinel/missions/sentinel-1/mission-objectives>.
- [6] ESA. Carico utile satellite. <https://sentinel.esa.int/web/sentinel/missions/sentinel-1/instrument-payload>.
- [7] Hub copernicus. <https://scihub.copernicus.eu/dhus/#/home>.
- [8] Catalogo onda. <https://catalogue.onda-dias.eu/catalogue/>.
- [9] Wikipedia contributors. Dispositivo ais. https://en.wikipedia.org/wiki/Automatic_identification_system.
- [10] ESA. Sentinel-1 toolbox. <https://sentinel.esa.int/web/sentinel/toolboxes/sentinel-1>.
- [11] ESA. Sentinel application platform. <http://step.esa.int/main/toolboxes/snap>.
- [12] European Commission. Sumo ship detection software. <https://ec.europa.eu/jrc/en/publication/eur-scientific-and-technical-research-reports/sumo-ship-detection-software-satellite-radar-images-short-installation-and-user-guide>.
- [13] PostgreSQL Developers. PostgreSQL. <https://www.postgresql.org/about>.
- [14] PostGIS Developers. Postgis. <https://postgis.net>.
- [15] QGIS Developers. Qgis. <https://www.qgis.org/it/site/about/index.html>.
- [16] Wikipedia contributors. Python. <https://it.wikipedia.org/wiki/Python>.
- [17] Caratteristiche python. http://www.helldragon.eu/marcello/galli_python/03-Us0.html.
- [18] Pandas Developers. Libreria pandas. https://pandas.pydata.org/pandas-docs/stable/getting_started/overview.html.
- [19] Python Developers. Libreria pycopg2. <https://pypi.org/project/pycopg2>.
- [20] Libreria matplotlib. https://hpc-forge.cineca.it/files/CoursesDev/public/2014/Python_for_computational_science/Rome/02-matplotlib.pdf.

- [21] Python Developers. Libreria seaborn. <https://pypi.org/project/seaborn>.
- [22] Wikipedia contributors. Shapefile. <https://it.wikipedia.org/wiki/Shapefile>.
- [23] Wikipedia contributors. Script. <https://it.wikipedia.org/wiki/Script>.
- [24] Dizionario Collins. Heatmap. <https://www.collinsdictionary.com/it/dizionario/inglese/heat-map>.

Elenco delle figure

1.1	Logo comune Big Data. [2]	6
1.2	Diagramma macro-fasi del progetto	6
2.1	Satellite Sentinel 1. [4]	10
2.2	Modalità di acquisizione Sentinel 1. [6]	11
2.3	Hub Copernicus [7]	12
2.4	Catalogo ONDA [8]	13
2.5	Meccanismo di funzionamento AIS [9]	14
2.6	Sentinel Application Platform	15
2.7	Graph Builder per rilevamento navi	16
2.8	Interfaccia SUMO	17
2.9	Logo PostgreSQL [13]	18
2.10	Interfaccia QGIS [15]	19
2.11	Logo Python	19
2.12	Python Data Visualization	22
3.1	Preview immagine SAR	24
3.2	Landmask SUMO	25
3.3	Analisi SUMO	25
3.4	Falso positivo	26
3.5	Heatmap	31
4.1	Screen QGIS	36
4.2	Zoom Screen QGIS	37

Elenco delle tabelle

4.1	Dati ottenuti dai primi tre script	33
4.2	Associazione punto-punto	34
4.3	Associazione punto-linea	34

Ringraziamenti

Per concludere ci tengo a dedicare un pensiero a tutti quelli che mi sono stati accanto in questo lungo percorso.

In primis, voglio ringraziare il mio relatore, il Dott. Adriano Mancini, che si è reso disponibile ogni qual volta ne avessi bisogno. Un ringraziamento anche al mio correlatore, il Dott. Alessandro Galdelli, che mi ha seguito per tutto il mio tirocinio, aiutandomi ad affrontare le numerose difficoltà incontrate, nonostante i suoi molti impegni.

Voglio ringraziare la mia famiglia e tutti i miei parenti, senza i quali non so se sarei giunto fino a qui. Infatti, avendo incontrato moltissimi ostacoli e insidie in questi anni, sia in ambito universitario che personale, penso che se ne sono uscito è stato proprio grazie alla costante presenza di mio padre e mia madre che mi hanno incoraggiato a non mollare mai. Lo stesso per mio fratello Nicola che ci è sempre stato, sia quando c'era bisogno di scherzare, sia quando qualcosa non andava, perchè mi conosce meglio di chiunque altro e sa come tirarmi su.

Poi per ultimo, voglio ringraziare il mio gruppo di amici, una seconda famiglia per me, con i quali ho condiviso una miriade di esperienze e momenti indimenticabili. Con alcuni ci conosciamo da una vita e con alcuni meno, ma praticamente si può dire che siamo cresciuti insieme, condividendo un'adolescenza fantastica ed indelebile. Ci sono sempre stati, sia nei momenti belli che in quelli brutti e per questo non smetterò mai di ringraziarli, sapendo che me li porterò dietro per sempre.

