Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE

Corso di Laurea in Informatica

APPLICAZIONE DI METODOLOGIE AI DI SEGMENTAZIONE SEMANTICA SU IMMAGINI SATELLITARI NELLA PIANURA ALLUVIONALE MESOPOTAMICA

Relatore:
Chiar.mo Prof.
Marco Roccetti

Presentata da:
Alessandro Pistola

Sessione IV
Anno Accademico 2022/2023

Sommario

Il problema che questo studio si propone di affrontare riguarda l'integrazione di vecchie immagini satellitari, denominate Corona, con un modello di machine learning esistente, una specifica rete CNN, per migliorare l'identificazione di siti archeologici nella pianura alluvionale mesopotamica, nello specifico nel distretto di Abu Ghraib.
Il lavoro propone un metodo che include uno studio di conferma sui precedenti lavori, il riaddestramento del modello sulla nuova area di AbuGhraib e la generazione di output utili agli archeologi come le mappe di calore.
Durante lo studio di conferma si replicano e verificano i risultati ottenuti nelle precedenti pubblicazioni, a seguire, si addestra la rete neurale su immagini satellitari Bing e Corona sperimentando due tipologie di fine-tuning (sull'intero modello e a 2 fasi) per poi effettuare le attività di predizione/individuazione di eventuali siti scomparsi dalle attuali mappe.
Si conclude con l'introduzione di una nuova classe d'errore, i dubbi positivi, utili per modellare le specifiche casistiche d'errore in cui Tell non più visibili vengono segnalati come falsi positivi, per poi essere successivamente confermati attraverso verifiche sul campo da parte degli archeologi.

Indice

Elenco delle figure

Elenco delle tabelle

Capitolo 1

Introduzione

1.1 Contesto: Deep learning a supporto dell'archeologia

In questo primo capitolo si introduce il contesto in cui tale lavoro si è sviluppato, i concetti alla base del funzionamento dei metodi e delle tecniche che si descriveranno approfonditamente nei capitoli successivi e soprattutto si analizzeranno i vantaggi dell'intersezione tra le due principali aree tematiche trattate: l'informatica e l'archeologia.

1.1.1 Pianura alluvionale mesopotamica

L'archeologia, come molte altre discipline, sta sperimentando una trasformazione digitale [1], [2]. Il deep learning, una sottocategoria dell'intelligenza artificiale (AI), sta diventando uno strumento sempre più prezioso per gli archeologi, infatti, l'utilizzo di tecniche di deep learning offre nuove opportunità per l'interpretazione e l'analisi dei dati archeologici, in particolare quando si trattano immagini satellitari [3].
Il lavoro di ricerca proposto si inserisce in un contesto di sviluppi negli anni che si sta sempre più approfondendo [4], [5], [6]. Nello specifico, il tema principale riguarda la localizzazione e l'identificazione di siti archeologici co-

nosciuti come "Tell" attraverso l'analisi di immagini satellitari.
Un "Tell" è un tipo di sito archeologico caratteristico delle regioni pianeggianti, come la pianura alluvionale mesopotamica, e consiste in una collina artificiale formata dal sovrapporsi delle rovine di insediamenti umani successivi nel corso dei secoli.
Nel contesto dello studio del paesaggio nella pianura alluvionale mesopotamica, il rilevamento dei Tell è di grande importanza per diversi motivi:

- Studio del cambiamento del paesaggio nel tempo: l'analisi delle immagini satellitari può fornire informazioni sulla distribuzione spaziale dei Tell nella pianura alluvionale mesopotamica nel corso del tempo. Questo può aiutare gli studiosi a comprendere i modelli di insediamento umano, le dinamiche sociali ed economiche e i cambiamenti ambientali che hanno influenzato la regione nel corso dei millenni.

- Pianificazione della ricerca archeologica: Il rilevamento dei Tell tramite immagini satellitari può guidare la pianificazione delle ricerche archeologiche sul campo, consentendo agli archeologi di selezionare le aree più promettenti per ulteriori sopralluoghi.

Sebbene per anni ciò sia avvenuto "manualmente", le nuove tecniche di acquisizione remota di immagini ed i progressi innovativi nel campo dell'intelligenza artificiale stanno modificando il flusso di lavoro dei ricercatori.

1.1.2 Remote-sensing

Il remote-sensing ormai da anni rappresenta una risorsa fondamentale per i ricercatori, essendo un metodo non invasivo che contribuisce alla conservazione e alla ricerca del patrimonio culturale.
Il punto di forza principale nel suo utilizzo risiede nel fatto che consente di esaminare ampie aree di terreno in breve tempo, consentendo quindi agli archeologi di individuare potenziali siti o caratteristiche del paesaggio che altrimenti potrebbero essere difficili da individuare.
Tuttavia, in assenza di un processo automatizzato, a seconda del campo di

studio e delle dimensioni dell'oggetto archeologico investigato, il lavoro richiesto al ricercatore può essere considerevole, specialmente in termini di tempo. In questo contesto si inseriscono i modelli di deep learning e le tecniche di segmentazione semantica.

1.1.3 Deep learning e segmentazione semantica

Il campo del deep learning è una sottodisciplina dell'intelligenza artificiale ispirata al funzionamento del cervello umano, si basa su reti neurali artificiali composte da strati multipli di neuroni interconnessi, capaci di apprendere automaticamente rappresentazioni complesse dei dati.
Negli ultimi anni l'avvento di potenti hardware grafici, l'abbondanza di dati e l'innovazione degli algoritmi hanno contribuito a rendere il deep learning una delle tecnologie più promettenti del ventunesimo secolo soprattutto nell'ambito della computer vision.
Tuttavia, nonostante i suoi notevoli successi, il deep learning presenta ancora sfide significative, tra cui la necessità di grandi quantità di dati di addestramento e per porre rimedio a tale problematica sono state sfruttate tecniche di transfer learning [7]. Queste tecniche consistono nel partire da un modello pre-addestrato su un grande e generale dataset (ad esempio, ImageNet [8]) e adattarlo per risolvere problemi specifici (come i problemi nel campo dell'archeologia). Questo approccio ha dimostrato di essere particolarmente efficace per l'analisi delle immagini satellitari.
Inoltre, nei dati sono presenti molti siti che non sono più visibili sulle mappe attuali, a causa dello sviluppo urbano degli ultimi decenni. Parte del lavoro ha infatti come prerogativa l'analisi delle variazioni dei modelli sviluppati quando come input si forniscono oltre alle immagini satellitari attuali le immagini Corona [9]. Queste immagini, scattate durante le missioni spaziali Corona negli anni '60, rappresentano una risorsa inestimabile per gli archeologi.
Vi sono molti esempi recenti di deep learning applicati con successo al rilevamento di siti in molteplici scenari diversi [10] e [11]

Più nel dettaglio, tale rilevamento avviene applicando specifiche configurazioni di reti CNN come Unet [12] MaNet [13].
Una CNN, o Convolutional Neural Network, è un tipo di rete neurale artificiale specializzata nell'elaborazione di dati strutturati, in particolare immagini. Le CNN sono composte da diversi strati, inclusi strati di convoluzione, di pooling e di classificazione, che lavorano insieme per estrarre progressivamente le caratteristiche salienti dell'immagine e classificarle correttamente.
Il task assegnato a queste reti convoluzionali è detto "segmentazione semantica" e consiste nell'assegnare un'etichetta di classe a ciascun pixel di un'immagine. In altre parole, si tratta di capire a quale categoria appartiene ogni pixel dell'immagine (ad esempio, auto, edificio, persona, ecc.).
Nonostante i risultati dell'utilizzo di tali tecniche siano sempre più promettenti, un loro utilizzo non supervisionato non rappresenta una soluzione fattibile, resta quindi di vitale importanza adottare un approccio "Human in the Loop".
Grazie all'approccio "Human in the Loop", gli archeologi possono collaborare con l'AI per analizzare queste immagini e identificare caratteristiche di interesse [14].

1.2 Organizzazione e divisione del lavoro

In questa trattazione, verranno innanzitutto fornite le informazioni necessarie alla comprensione del contesto di ricerca attorno al quale il presente lavoro si colloca, ovvero verrà presentato il problema e descritta la soluzione proposta, ovviamente analizzando la letteratura scientifica inerente.
Presentati i concetti basilari si passerà alla prima parte del lavoro che essenzialmente consta di uno studio di conferma e quindi di verifica dei precedenti lavori sul quale l'attuale progetto si fonda.
Terminata la trattazione dello studio di conferma si passerà alla specifica soluzione proposta e l'analisi dei relativi risultati sperimentali.
Infine, nell'ultimo capitolo, si discutono i risultati traendo le conclusioni dello

studio riportando eventuali limiti e lavori futuri.
La struttura del documento è la seguente:

- **Informazioni preliminari**: descrizione del problema, riepilogo contesto di ricerca e presentazione della soluzione proposta.
- **Metodo proposto**: diviso in due parti, la prima riguardante uno studio di conferma volto a verificare la riproducibilità dei lavori precedenti, la seconda riguardante l'applicazione del modello nel distretto di Abu-Ghraib dall'acquisizione dei dati fino alla presentazione delle relative metriche di performance.
- **Risultati sperimentali**: in questo capitolo si riporta tutta le procedura per la creazione degli artefatti e due test sulla zona di Maysan e dell'Uzbekistan.
- **Discussione e conclusioni**: riassunto e discussione dei risultati ottenuti. Si riportano inoltre limiti e possibili sviluppi futuri.

Capitolo 2

Informazioni preliminari

2.1 Descrizione del problema

Il problema che questo studio si propone di affrontare riguarda l'integrazione di vecchie immagini satellitari, denominate Corona [9], con un modello di machine learning esistente, una specifica rete CNN, per migliorare l'identificazione di siti archeologici nella pianura alluvionale mesopotamica.
In particolare, i professori Marco Roccetti e Nicolò Marchetti dell'Università di Bologna hanno dimostrato l'efficacia dell'uso di modelli di deep learning potenziati con meccanismi di segmentazione e auto-attenzione nell'individuazione di nuovi siti archeologici [6]. Tuttavia, la sfida attuale è quella di integrare queste tecniche con le immagini Corona.

Il distretto mesopotamico, un'area di grande interesse archeologico, ha subito nel tempo significative modifiche a causa di lavori agricoli e di urbanizzazione. Questo ha portato alla perdita di visibilità di molti siti potenziali sulle mappe moderne. Le immagini Corona, risalenti a decenni fa, potrebbero rendere nuovamente "visibili" al sistema neurale una quantità significativa di siti (circa il 50%) che sono ormai irreperibili.

Questo studio, quindi, esplora l'applicazione di metodologie di intelligenza

artificiale, ossia modelli di deep learning per la segmentazione semantica su immagini satellitari nella pianura alluvionale mesopotamica, nello specifico nel distretto di AbuGhraib.
Utilizzando tecniche di deep learning e remote sensing, si mira a migliorare il supporto all'archeologia.
Il lavoro propone un metodo che include uno studio di conferma sui precedenti lavori, il riaddestramento del modello sulla nuova area di AbuGhraib e la generazione di mappe di calore (una rappresentazione grafica delle immagini dove i valori dei singoli pixel, che rappresentano la probabilità che vi sia o meno un "Tell" , sono rappresentati da colori).
Durante lo studio di conferma si andranno a replicare e verificare i risultati ottenuti nei precedenti lavori [4], [5], [6], a seguire, è previsto un nuovo addestramento della rete neurale su immagini Corona con conseguente attività di predizione/individuazione di eventuali siti scomparsi dalle attuali mappe. Qualora i risultati fossero positivi, tale progetto avrebbe un impatto significativo sulla preservazione delle evidenze archeologiche in via di sparizione.

Nel paragrafo seguente verranno ripercorse le metodologie di lavoro ed i risultati ottenuti nelle precedenti pubblicazioni per comprendere a fondo il contesto di ricerca nel quale l'attuale progetto si colloca.
Si introdurranno inoltre i principali modelli e tecniche di *deep learning* e segmentazione semantica utilizzati che rappresentano lo stato dell'arte nel settore della *computer vision*.

2.2 Revisione della letteratura

Il primo dei quattro lavori che costituiscono le fondamenta dell'attuale studio [6] si prefissò come obiettivo il verificare qualora una collezione di immagini satellitari con siti archeologici degli di nota potesse essere abbastanza informativa dal riuscire a istruire o meno un modello di *deep learning*, così che quest'ultimo possa essere usato per rilevarne di nuovi.

Nello specifico, constatando il grande tempo dedicato alle attività di ricerca manuale di punti di interesse ed il fatto che quest'ultima possa portare a dei frequenti errori umani, l'idea principale è stata quella di impiegare modelli di rete neurale per assistere l'attività degli archeologi, evidenziando possibili siti di interesse nella mappa, limitando così le azioni sul campo.
Tale intuizione è stata poi concretamente sviluppata nel contesto degli studi di ricerca che vengono portati avanti dall'Università di Bologna nella regione di Qadisiyah in Iraq.
Il primo step ha riguardato l'automatizzazione del processo di *remote sensing* dell'area di interesse, anche in questo progetto come nell'attuale e nei successivi, la specifica tipologia di sito archeologico da rilevare è chiamata "*Tell*" [15].

L'area di interesse è stata suddivisa in *tile* corrispondenti a immagini di 299x299 pixel, inoltre nel processo di campionatura, per non perdere informazione sono stati campionati anche degli "intermediate tile" ottenuti mediante lo *shifting* del *tile* di metà finestra.
Al fine di risolvere problemi di dataset sbilanciato e immagini vero-positive con percentuali di *Tell* inferiori al 10% è stato effettuato un processo di filtraggio in cui sono state selezionate circa 2000 immagini vero-positive e altrettante vero-negative scelte casualmente da *tile* che non presentano intersezioni con siti.

Sebbene 4000 immagini possano rappresentare un ampio set di immagini archeologiche, nel campo del *deep learning* non è sufficiente, proprio per questo, si è scelto di ricorrere al *transfer learning* [16]. Come modello pre addestrato è stato scelto *Google Inception V3* con *Inception* come *feature-extractor*. I pesi sono stati congelati, rimossa la testa del modello (utilizzata per la classificazione) e rimpiazzata con un nuovo *layer* addestrato per 10 epoche.
Nel tentativo di aiutare il modello a evitare errori sui casi positivi è stato specificato un peso (*class weights*) a sfavore dei *tile non-site* ed utilizzato un

processo di *data augmentation* (sole trasformazioni geometriche) per evitare l'*overfitting* del modello.

Per quanto riguarda lo studio di comparazione, sono stati addestrati quattro modelli, i quali differiscono solamente per la distrubuzione delle classi del dataset utilizzate per l'addestramento. In termini di *AUC accuracy* il modello con configurazione sperimentale nr.3 (quello con rapporto 1:1 tra numero di siti e non siti) è quello che ottiene la miglior performance.
Sebbene i risultati della metrica non siano stati così promettenti il team di ricerca ha utilizzato il modello per generare previsioni e sovrapporle a una mappa geografica su QGIS con l'obiettivo di verificare se le previsioni potessero indirizzare l'utente nella giusta direzione, evidenziando un punto particolare sulla mappa (la figura 2.1 riporta una delle *heatmap* generate).

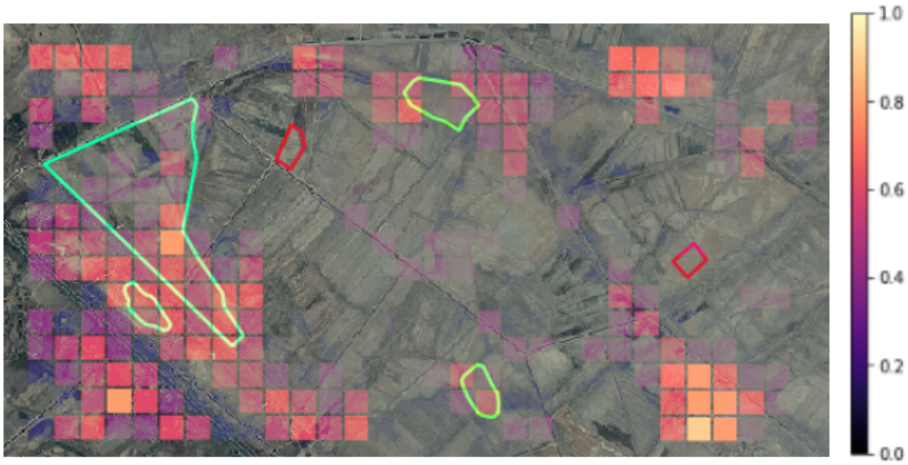


Figura 2.1: Esempio overlay della mappa di calore con l'immagine satellitare dell'area Qadis [6]

Tuttavia, è emerso un numero considerevole di falsi positivi che avrebbe potuto ostacolare la potenziale utilità del modello, proprio per questo il secondo approccio ha cercato di risolvere la scarsa accuratezza del primo modello provando ad aggiungere informazioni contestuali.

Come modifica è stato aggiunto un intero *branch* (uno schema del modello è riportato in fig. 2.2): uno per il *tile* ed uno per il *context*, il quale è costituito da 8 immagini rappresentanti gli 8 *tile* che che circondano il *tile* originale.

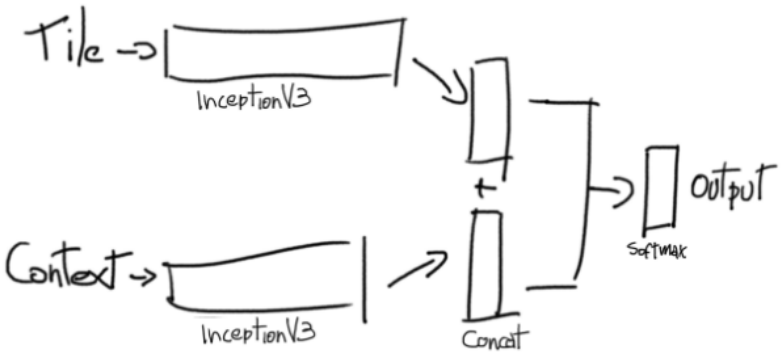


Figura 2.2: Modello con un secondo branch per le informazioni contestuali [6]

Con questa nuova configurazione unita ad un'intensa attivita di *data augmentation* si raggiunge uno score AUC-ROC di 0.71. Sebbene si possa notare un miglioramento notevole, si sottolinea il fatto che ad una maggiore prestazione nella classificazione dei singoli *tile*, non corrisponde automaticamente una migliore prestazione nel riconoscere i veri siti archeologici nella loro interezza. Infatti, i *tile* creati sono solamente piccole porzioni di siti più grandi nella mappa.

Una considerazione che poi si ripresenterà nei futuri lavori come in quello attuale è che, sebbene da un certo punto di vista si stia affrontando un task impegnativo riguardante l'implementazione di un processo di *image recognition* puramente informatico/tecnico dall'altro non si deve dimenticare la natura "archeologica" dell'immagini e quindi dei risultati che si ottengono. La creazione degli artefatti quali particolari *shapefile* o mappe di calore lasciano aperto il problema dell'interpretazione del risultato ottenuto che è

fortemente legato alla comprensione dello specifico contesto che solo un archeologo possiede.

Il secondo lavoro di ricerca [17] affronta la questione dell'uso dell'intelligenza artificiale (IA) e del *deep learning* (DL) nell'archeologia discutendone i principali problemi ed analizzando le soluzioni presenti in letteratura.
Presentando come esempio l'area di Qadis, gli autori si chiedono se l'impiego di modelli *deep learning* rappresenti la soluzione più adatta e se esistano metodi tradizionali meno complessi che siano in grado di raggiungere gli stessi risultati.
Come ultima analisi introducono il concetto di *barrier of meaning* definito come il gap tra la conoscenza dell'esperto di dominio e la conoscenza codificata nel modello.
Tale gap, infatti, risiede sostanzialmente nell'aver a disposizione una maggiore quantità di informazioni contestuali, cosa che difficilmente si riesce a codificare in una rete CNN o un modello *deep learning* in generale.
Va ricordato infatti che l'affidabilità dei modelli di *deep learning* (DL) è direttamente proporzionale alla qualità dei dati su cui sono stati addestrati.
Posizionare gli esseri umani, sia esperti che utenti, al centro della progettazione dei modelli di intelligenza artificiale, in una sorta di ciclo di apprendimento automatico centrato sull'uomo, potrebbe aiutare a superare la "barriera del significato", o almeno rappresentare un passo nella giusta direzione: si vedrà poi come nei successivi lavori tale concetto venga implementato.

Il terzo lavoro di ricerca [5], rappresenta lo sviluppo di due delle proposte/lavori futuri descritte nei lavori precedenti [6], [17].
In questo specifico articolo si esplora la possibilità di utilizzare modelli di segmentazione semantica per rilevare ed evidenziare la presenza di siti archeologici presenti nella pianura alluvionale mesopotamica attraverso mappe disponibili e accessibili gratuitamente online, le mappe Bing.
Il sito archeologico oggetto di rilevazione è anche in questo caso il *Tell* e il

processo automatizzato che si mira ad implementare non è un processo che mira a sostituire completamente l'attività umana ma prevede una collaborazione e uno scambio di informazioni con esperti del dominio, in questo caso rappresentati da archeologi.
La differenza principale, in questo caso, oltre all'area geografica di interesse, riguarda la tipologia di task, infatti, si è scelto di trattare il problema come un task di segmentazione semantica.
La segmentazione semantica è un processo di *computer vision* che classifica ogni pixel in un'immagine come appartenente a una determinata classe o categoria. In termini matematici, se I è un'immagine e C è l'insieme di classi possibili, la segmentazione semantica è una funzione $f : I \rightarrow C$ che assegna a ogni pixel un'etichetta di classe.
Più precisamente, si parla di segmentazione semantica binaria, un caso particolare di segmentazione semantica in cui le classi da assegnare sono due (*site*, *non-site*). Questo processo è fondamentale in molte applicazioni di *computer vision*, come il rilevamento di oggetti e la segmentazione di immagini mediche.
Il principale obiettivo di ricerca di tale lavoro è stato quindi duplice, da una parte si è indagata la fattibilità dell'approccio di segmentazione per questo particolare compito di rilevamento e dall'altra si è potuto confrontare il risultato dell'approccio basato sul metodo della classificazione del precedente lavoro con un modello testato operativamente.
La procedura di campionamento è stata modificata, in quanto l'area mesopotamica e la tipologia del problema lo hanno reso necessario. Grazie al progetto *FloodPlains* [18] è stato possibile ottenere gli *shapefile* relativi a 5000 siti censiti da diversi team in diversi periodi. A questo punto è stato possibile in QGIS campionare i siti attraverso le mappe Bing, le rispettive maschere attraverso i *shapefile*, creando una collezione di immagini 1000x1000 centrate nei siti.
Dopo aver aggiunto immagini senza *Tell* per bilanciare il dataset ed aver effettuato uno *splitting* del dataset in *training*, *validation* con proporzioni 80:20

si sono implementate delle tecniche di *data augmentation*. Tali tecniche sono risultate di vitale importanza in quanto senza quest'ultime il modello avrebbe potuto apprendere un bias riguardante la posizione sempre centrale dei *Tell* nelle immagini da valutare, in quanto avrebbe rappresentato una certezza.
Oltre alle precedenti operazioni è stata effettuata anche un'operazione di filtraggio delle immagini, rimuovendo quelle che non sarebbero state utili ai fini dell'addestramento, ad esempio immagini con *Tell* non più visibili o con piccole porzioni di questi ai lati.
Per quanto riguarda i modelli sperimentati, sono state scelte due particolari tipologie di reti neurali convoluzionali profonde.

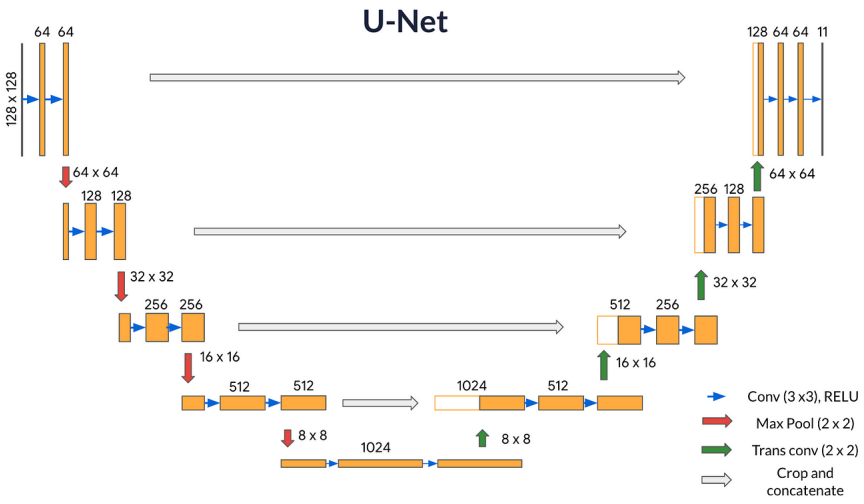


Figura 2.3: Architettura di una rete U-net [12]

In generale, la maggior parte di queste reti DCNN (*Deep Convolutional Neural Networks*) condividono la stessa struttura architettonica di base, sono infatti composte da un *encoder* ed un *decoder*.
L'*encoder* è responsabile per la *feature extraction* dall'immagine di input. Questo viene fatto attraverso una serie di strati convoluzionali e di *pooling* che riducono progressivamente la dimensione spaziale dell'immagine mentre aumentano la profondità della *feature map*. Questo processo permette

all'*encoder* di catturare le informazioni visive importanti presenti nell'immagine e di rappresentarle in un formato più compatto e gestibile.
Il *decoder*, d'altra parte, è responsabile della ricostruzione dell'immagine a partire dalle *feature* estratte dall'*encoder*. Questo viene fatto attraverso una serie di strati di *upsampling* e convoluzionali che aumentano progressivamente la dimensione spaziale della *feature map* mentre riducono la sua profondità. Il *decoder* utilizza le informazioni fornite dall'*encoder* per creare un'immagine di output che corrisponde all'obiettivo desiderato, come una mappa di segmentazione semantica.
Un esempio significativo di DCNN, primo modello implementato, è l'architettura U-Net [12], ampiamente utilizzata per la segmentazione semantica. U-Net è un'architettura encoder-decoder che utilizza connessioni residue (o "skip connections") per preservare le informazioni spaziali durante la fase di decodifica. Questo permette di ottenere mappe di segmentazione più dettagliate e nitide.
Il secondo modello sperimentato è MANet (Multi-scale Attentional Network), che introduce un meccanismo di attenzione multi-scala per catturare le caratteristiche a diverse scale. Questo permette al modello di concentrarsi su aree specifiche dell'immagine durante l'addestramento, migliorando così la precisione della segmentazione.

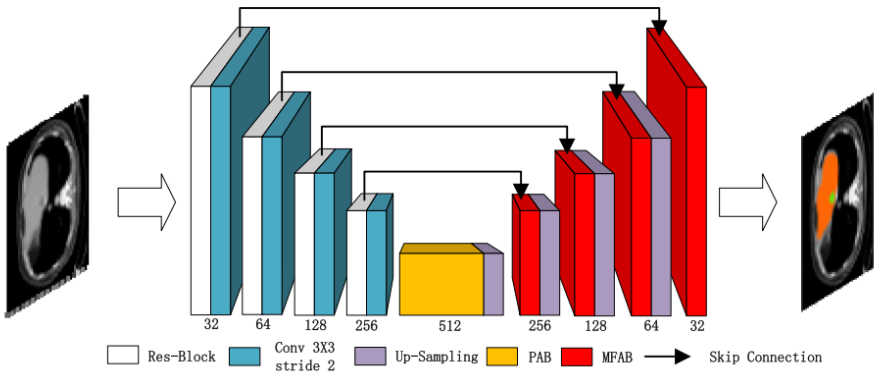


Figura 2.4: Architettura di una rete MANet [13]

Le figure 2.3 e 2.4 mostrano l'architettura delle due particolari reti.

Proseguendo l'analisi, l'impianto sperimentale prevede l'utilizzo di due diverse tipologie di *encoder*, ResNet-18 [19] e EfficientNet-B3 [20], la prima introdotta nel 2015 è diventata popolare per aver implementato il concetto di "*skip connections*", mentre la seconda sviluppata da Google Brain, rappresenta uno dei modelli più efficienti.
Congiuntamente alle due tipologie di architetture ed *encoder* sono state testate tre diverse tipologie di *loss function*:

- **IoU (Intersection over Union)**:

$$IoU = \frac{|A \cap B|}{|A \cup B|} == \frac{TP}{TP + FP + FN}$$

  dove A e B sono, rispettivamente, l'insieme dei pixel dell'oggetto di interesse e l'insieme dei pixel predetti come appartenenti all'oggetto.

- **Dice Loss**:

$$DiceLoss = 1 - \frac{2TP}{2TP + FP + FN}$$

- **Focal Loss**:

$$FocalLoss = -\alpha(1-p)^{\gamma}\log(p)$$

  dove p è la probabilità predetta per la classe di interesse, α è un peso di bilanciamento e γ è un parametro di focalizzazione.

Le differenze tra queste funzioni di costo risiedono principalmente nel modo in cui gestiscono i disallineamenti tra la previsione e la maschera di verità. La IoU e la Dice Loss sono entrambe misure di sovrapposizione tra due insiemi e sono particolarmente utili per problemi di segmentazione, ma possono essere meno sensibili ai falsi positivi. D'altra parte, la Focal Loss è stata progettata per affrontare il problema dello squilibrio di classe nelle attività di classificazione, dando più peso alle classi minoritarie e/o alle previsioni errate.

A seguito di un addestramento di 10 e 20 epoche MANet non sembra fornire alcun vantaggio significativo rispetto a U-net, hanno entrambi previsioni molto simili nella maggior parte dei casi e punteggi che differiscono per punti decimali. Tuttavia, all'aumentare del numero di epoche di addestramento MANet sembra riesca ad ottenere uno score IoU leggermente maggiore (2 punti percentuali).
Allo stesso modo, la Dice Loss e la Focal Loss ottengono score estremamente simili, con la differenza maggiore che resiede nell'output prodotto. Infatti, la Dice Loss a creare maschere pulite con contorni netti metre le maschere generate con la Focal Loss sono smussate, le probabilità sono distribuite in modo più uniforme sull'intervallo [0.5-1].
La figura 2.5 mostra le due tipologie di maschere generate con le due diverse funzioni.
Passando all'analisi dell'*encoder* ResNet-18 ha ottenuto sempre risultati peggiori, sebbene lo scarto in termini di IoU sia contenuto, quest'ultimo sembra generare quasi sistematicamente degli errori importanti come tralasciare alcune parti dei siti.

Figura 2.5: Esempio di maschere generate con Focal e Dice Loss [5]

I risultati ottenuti sono stati soddisfacenti, lasciando molto spazio a futuri sviluppi. In termini assoluti uno score IoU del 70% circa non è un grande risultato, tuttavia, va tenuto conto del particolare dominio di applicazione e dell'obiettivo di ricerca.
In campo archeologico le caratteristiche visive non sono così evidenti come in altri task di segmentazione e anche gli archeologici riscontrano difficoltà nel tracciare precisamente la forma di un sito.
Nelle conclusioni del lavoro, vengono riportati degli aspetti molto importanti:

- l'ottenimento di un IoU del 100% è pressoché impossibile. Questo è facilmente dimostrabile in quanto il dataset presenta maschere non precise e addirittura alcuni siti non sono più visibili;

- da alcune interazioni preliminari con gli archeologi si è concluso che anche una maschera parziale nel giusto spot è sufficiente a guidare l'archeologo e, inoltre, si è riscontrato che spesso gli errori commessi dal modello rientrano nella gamma degli errori che anche l'archeologo avrebbe potuto compiere;

- le performance omogenee dei vari livelli indicherebbero come sviluppo ulteriore di lavorare sul dataset e sulla sua gestione, infatti le modifiche dell'architettura non generano differenze e/o benefici rilevanti.

L'ultimo dei quattro lavori revisionati [4] non differisce dal precedente nè per area geografica di interesse nè per modelli adottati.
Ciò che viene proposto è un processo di collaborazione che cerca di integrare e rendere lo scambio informativo con gli esperti di archeologia più vantaggioso (processo iterativo *Human in the loop* con esperti di dominio).
Le principali differenze riguardanti la creazione del dataset sono:

1. Creazione di un dataset con immagini Corona come complemento, costruito scartando i siti non più visibili dopo il 1970 (studio preliminare di integrazione);

2. Aumento delle dimensioni delle immagini campionate, si testano due configurazioni, 1000x1000 e 2000x2000;

3. Le immagini *non-site*, che non contengono *Tell* vengono campionate da località suggerite dagli archeologi.

Sebbene l'architettura del modello rimanga invariata, ora si valutano 2 diverse performance, una a livello pixel (*pixel-wise*) ovvero la segmentazione semantica vera e propria e una a livello di rilevazione dei *Tell*, ovvero il task di *object detection*.
La valutazione delle performance per il task di segmentazione semantica avviene attraverso la metrica IoU come nel lavoro precedente, mentre le perfomance per il task di *detection* avviene avvalendosi delle seguenti metriche:

- **Accuracy**:
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
- **Recall** (o Sensitivity o True Positive Rate):
$$Recall = \frac{TP}{TP + FN}$$
- **Precision** (o Positive Predictive Value):
$$Precision = \frac{TP}{TP + FP}$$
- **MCC (Matthews Correlation Coefficient)**:
$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Dove:

- TP = True Positives
- TN = True Negatives
- FP = False Positives

- FN = False Negatives

Al fine di automatizzare e rendere più agevole l'intervento e la valutazione degli archeologi vengono generati degli artefatti e output intermedi utili nel processo di lavoro *human in the loop*.
Partendo dalla predizione della rete si possono generare due output:

- **Mappa di calore** o *heatmap*, semplicemente vengono assegnati dei colori ai vari valori di probabilità dei singoli pixel;
- **Shapefile**, tali file rappresentano delle vere e proprie maschere di verità.

Entrambi gli output seppur con scopi diversi vengono geo-referenziati. Le *heatmap* vengono utilizzate direttamente come *layer* su QGIS così da rendere la consultazione agli archeologi più agevole, mentre, gli *shapefile* oltre a poter essere importati su QGIS vengono utilizzati per compiere una prima valutazione automatica del processi di rilevamento. Ovvero, vengono confrontati uno ad uno alla ricerca di un'intersezione con gli *shapefile* originali. Inoltre, essendo l'output della rete una maschera di probabilità, per poter generare gli *shapefile* è necessario applicare una soglia con cui si discretizzano i valori dei pixel, le soglie utilizzate sono di 0.2 e 0.5.
Un esempio di *shapefile* e *heatmap* generati come output, sono riportati in figura 2.6.

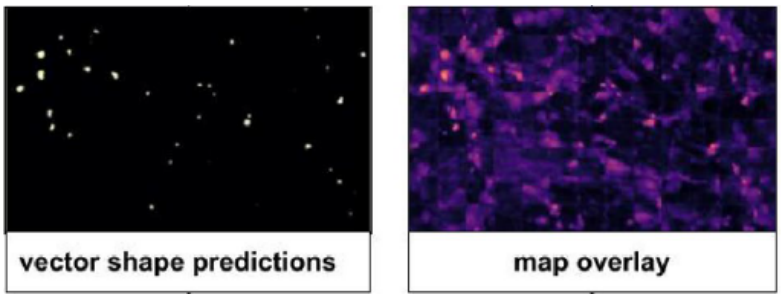


Figura 2.6: Shapefile e mappa di calore generati come output [4]

Analizzando i risultati dei test, effettuati più volte così da approssimare il fattore randomico delle operazioni di *crop*, come prima cosa, si è notato un netto miglioramento delle performance all'aumentare delle dimensioni dell'input, questo si ipotizza derivi dalle maggiori informazioni contestuali a disposizione della rete.
L'impiego delle immagini Corona è alquanto discutibile. Per dimensioni più piccole (1k), non sembra offrire vantaggi significativi e si può ipotizzare che ciò derivi dalla bassa risoluzione delle stesse.
Al contrario, con dimensioni maggiori (2k) sembra che possano fornire un miglioramento delle prestazioni, forse ancora una volta a causa del contesto più ampio. Ispezionando le singole previsioni si è rilevata l'assenza di una evidente differenza, il che potrebbe significare che il più alto valore di IoU registrato sia il risultato di contorni leggermente più precisi.
Seconda parte dell'analisi delle performance riguarda la *detection accuracy* che avviene mediante gli *shapefile*.
Attraverso la valutazione automatica (rilevamento intersezione *shapefile* e maschera di verità) si ottengono dei buoni risultati seppur non eccezionali, *accuracy* intorno al 60%. Tuttavia, come riportato nei lavori già analizzati attraverso il flusso di lavoro *human in the loop*, il modello deve guidare l'archeologo, ed un modello in grado di rilevare 2 siti su 3 fornisce un buon punto di partenza per procedere con l'analisi manuale da parte dell'archeologo.
Vi sono tuttavia delle considerazioni da riportare:

1. Molti siti non sono più visibili dalle mappe attuali ma non sono stati rimossi dal dataset, ogni immagine contenente siti non più visibili dovrebbe essere considerata un vero negativo se il modello non rileva niente, al contrario, nel processo automatizzato si considera come un falso negativo;

2. Molte previsioni sono considerate false positive poiché il modello rileva un sito nelle vicinanze invece di quello che sta venendo testato, in questo caso la previsione può essere considerata un vero positivo sia nel caso il sito che sta venendo testato non fosse più visibile sia nel caso fosse

ancora visibile, infatti il modello sta indicando una zona con potenziali siti che l'archeologo può esaminare;

3. Alcune previsioni sono effettivamente presenti negli output ma troppo deboli per la soglia limite che si è scelta.

Vista l'impossibilità di modificare manualmente il dataset e le maschere, il lavoro di ricerca propone un possibile approccio per il flusso di lavoro, ovvero l'utilizzo delle previsioni come *layer* sovrapposto alla mappa così da poterla analizzare manualmente.
Ricalcolando i risultati della valutazione automatica a seguito della valutazione umana (atta ad evidenziare i siti non visibili) si ottiene un punteggio di *accuracy* dell'80%.

Come ulteriori analisi sono stati implementati due test, il test sull'area di Maysan, in cui si è testato il modello su un'area operativa con alta percentuale di veri negativi ed un test su una regione geografica Uzbeka.
Il primo test ha restituito un esito positivo in quanto il modello ha rilevato correttamente 17 siti sui 20 totali ed ha rilevato altrettanti siti come falsi positivi, tuttavia, analizzando le previsioni errate si è notato come queste non fossero utili e potessero essere scartate velocemente da un occhio esperto.
Il secondo test aveva come obiettivo l'analisi dell'adattabilità del modello a diversi contesti geografici, il test sfortunatamente ha dato esito negativo in quanto il modello è riuscito a rilevare correttamente solamente il 25%/30% dei siti presenti nell'area in analisi. La ragione principale del drastico calo di performance risiede nella differente natura dello specifico paesaggio, più urbanizzato in generale e con aree di vegetazione più fitta.

2.3 Descrizione della soluzione proposta

Nella revisione della letteratura, presente nel paragrafo precedente, risulta evidente che tale lavoro di Tesi si debba sviluppare su più fronti, ponendosi diversi obiettivi.
Di seguito, vengono riportate tutte le domande di ricerca (*research questions*) che tale progetto si propone di rispondere:

- **RQ1**: come primo passo, si effettua uno studio di conferma, riproducendo il lavoro svolto nell'ultimo lavoro di ricerca, al fine di poter avere un punto di partenza revisionabile ed accessibile da cui poi testare gli ulteriori sviluppi di ricerca;

- **RQ2**: Il secondo obiettivo del lavoro di Tesi rientra nel campo del *transfer learning*. Viene cioè testato il processo di *fine tuning* per determinare se tale processo possa rappresentare una soluzione ai dubbi sollevati dal test sull'area geografica Uzbeka in merito alla versatilità del modello. E' utile effettuare *fine tuning* o il modello è in grado di adattarsi e di essere applicato efficacemente a una varietà di scenari e compiti diversi senza la necessità di modifiche significative e ridimensionamenti?
  Nell'ambito di tale obiettivo si procede ulteriormente testando due diverse tipologie di *fine tuning*, una sull'interno modello e una a 2 fasi (*2 stage fine-tuning*);

- **RQ3**: Come terzo obiettivo si propone un'analisi dettagliata sull'utilizzo delle immagini Corona, singolarmente o in modo complementare alle immagini Bing, cercando di indagare il loro effettivo beneficio.
  Tra le operazioni del processo di analisi sull'utilizzo delle immagini Corona, in particolar modo durante l'analisi della matrice di confusione e quindi degli errori che il modello commette in questa particolare configurazione, si cercherà di catalogarli mettendo in evidenza i limiti della classica matrice di confusione quando viene utilizzata in questo specifico ambito archeologico.

Capitolo 3

Metodo proposto

3.1 Studio di conferma

Uno studio di conferma, noto anche come studio di validazione, è un elemento cruciale nel processo di ricerca scientifica e tecnologica. Questo tipo di studio è progettato per verificare i risultati ottenuti in precedenti esperimenti o analisi, e per testare la riproducibilità dei metodi e dei processi utilizzati. La riproducibilità è un pilastro fondamentale della scienza e dell'ingegneria, in quanto garantisce che i risultati ottenuti siano affidabili e non siano il prodotto di errori casuali o di circostanze uniche. Inoltre, uno studio di conferma può aiutare a identificare eventuali limitazioni o problemi con i metodi originali, consentendo così di apportare miglioramenti. In informatica, dove gli algoritmi e i modelli possono essere influenzati da una vasta gamma di fattori, gli studi di conferma sono particolarmente importanti per garantire che le soluzioni proposte siano robuste, efficaci e applicabili in diversi contesti o scenari. Pertanto, la realizzazione di uno studio di conferma rappresenta una pratica standard nell'ambito della ricerca informatica, contribuendo significativamente all'avanzamento della disciplina.

Nel campo del *machine learning*, la necessità di studi di conferma assume un'importanza ancora maggiore. I modelli di apprendimento automatico so-

no spesso complessi e possono essere influenzati da una moltitudine di fattori, tra cui la qualità dei dati di addestramento, la scelta degli iperparametri, e l'architettura del modello stesso. Pertanto, è fondamentale verificare che i risultati ottenuti siano riproducibili e non siano il frutto di *overfitting*, bias nei dati o altre anomalie.

Nei successivi paragrafi, si procederà descrivendo i passaggi salienti del processo di verifica utilizzando il flusso di lavoro dell'ultima pubblicazione [4], accentuando le modifiche che sono state proposte, discutendone le motivazioni. Si procede illustrando il processo relativo ai 2 modelli considerati migliori, il modello facente uso delle immagini Bing con risoluzione 2000x2000 campionate da dataset filtrato e quello che incorpora le immagini contestuali agguntive Corona.

3.1.1 Dataset

Lo studio utilizza come sorgente dati un insieme di forme vettoriali (*shapefile*) georeferenziate, corrispondenti ai contorni dei siti (*Tell*) noti nell'area di indagine del progetto FloodPlains [18].
Tale area come già detto è situata nella pianura alluvionale mesopotamica orientale, nell'attuale Iraq. L'insieme di forme vettoriali è sviluppato, mantenuto e reso disponibile dall'Università di Bologna e contiene 4934 forme vettoriali.

Filtraggio forme vettoriali

Essendo il dataset stato redatto come fonte di informazione per gli archeologi e non specificamente per addestrare modelli di *deep learning*, si è resa necessaria l'applicazione di una procedura di filtraggio dei siti non idonei all'addestramento della rete (ovvero di quelli che avrebbero potuto effettivamente compromettere il processo di apprendimento).
Il dataset è stato ordinato per grandezza, area del *Tell* in m^2 e sono state rimosse le prime 200 istanze, in quanto notevolmente più grandi rispetto al

resto del dataset (area dell *Tell* che si estende più dell'area circoscritta da una singola immagine di input passata alla rete).
Inoltre, si sono filtrati 684 siti che presentavano un'area troppo piccola per essere un *Tell* o che erano stati segnalati dagli archeologi come distrutti (la descrizione nelle annotazioni includeva una delle seguenti parole chiave: "unlocated", "unclear" o "not visible").
Di conseguenza, la dimensione finale del dataset utilizzato è di 4050 siti.
La figura 3.1 illustra l'area di indagine del progetto FloodPlains in cui sono riportati in verde i siti rilevati ed in rosso i luoghi di non interesse come città e laghi.

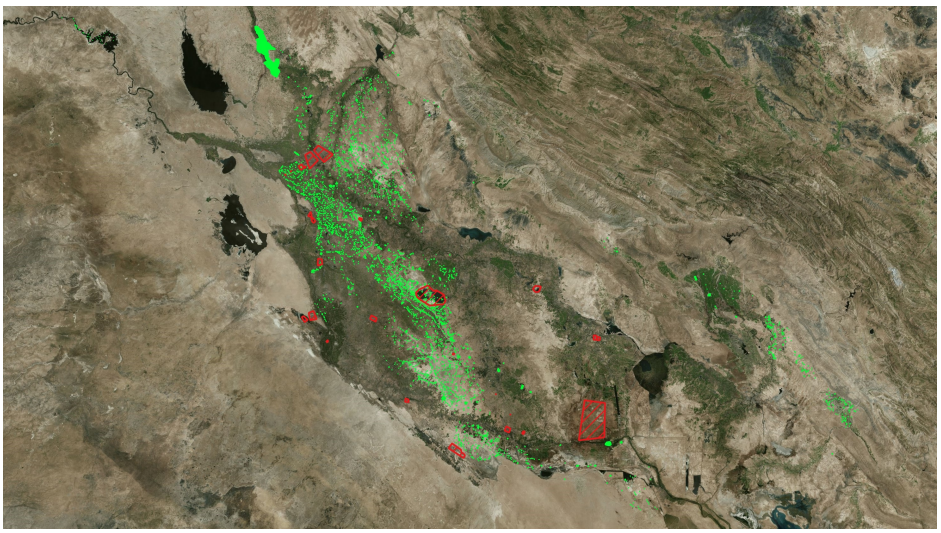


Figura 3.1: Progetto FloodPlains: area di indagine

Generazione immagini

Partendo dal dataset di forme vettoriali georeferenziate, il processo per generare le immagini Bing, le immagini Corona e le maschere di verità è rimasto invariato, avviene tramite il software QGIS ed è costituito dalle seguenti operazioni:

1. Caricamento delle forme vettoriali georeferenziate, delle mappe Bing e delle mappe Corona in QGIS;

2. Per ogni forma vettoriale, si calcolano i centroidi e le rispettive coordinate (cx, cy) degli stessi;

3. Si esportano le coordinate (cx, cy) e l'identificativo di ogni *Tell* in un file csv;

4. Per ogni tipologia di output da generare, si attivano/disattivano i layer di interesse e si procede con la campionatura del layer desiderato (Bing, Corona, Shapefile per la maschera di verità) mediante uno script Python che salva dei quadrati di grandezza fissa utilizzando le coordinate presenti all'interno del file csv.
   Nel caso specifico i *tile* generati sono di grandezza 2000x2000 metri con risoluzione 2048x2048 centrati sul *Tell*. Come riportato nel capitolo 1, si eseguiranno delle operazioni, contestualmente a quelle di *data augmentation*, per scongiurare che il modello apprenda come rilevare tali siti sfruttando il fatto che quest'ultimi siano sempre collocati al centro dell'immagine.

Generate le immagini "vero positive", al fine di bilanciare il dataset e permettere al modello di apprendere come riconoscere l'assenza di siti archeologici, sono state aggiunte 1155 immagini "vero negative". Tali immagini sono state campionate da zone indicate dagli archeologi, zone di particolare interesse poiché includono aree altamente urbanizzate, aree agricole, luoghi soggetti a inondazioni e colline e montagne rocciose.
Con tale operazione, termina la fase di acquisizione del dataset che risulta

così composto:

Classe	Dimensione	Percentuale
Positivi	4050	77,81%
Negativi	1155	22,19%
Totale	5205	100%

Tabella 3.1: Composizione dataset Floodplains

Operazioni di data augmentation

A partire dall'immagine quadrata iniziale, ad ogni immagine si applica un ritaglio casuale di una porzione di 1024x1024 pixel e si applicano delle tecniche di *data augmentation* in modo diverso e casuale ad ogni iterazione di addestramento.
Le operazioni includono:

- Un capovolgimento
- Una rotazione di 90°
- Una modifica della luminosità e del contrasto

Per concludere, prima di creare i vari batch, ogni immagine viene ridimensionata a 512x512 pixel. Le tecniche riportate vengono utilizzate simultaneamente sulle relative immagini Corona e sulle maschere di verità.
In figura 3.2 si riporta un esempio del processo di campionatura concatenato alle operazioni di *data augmentation*.

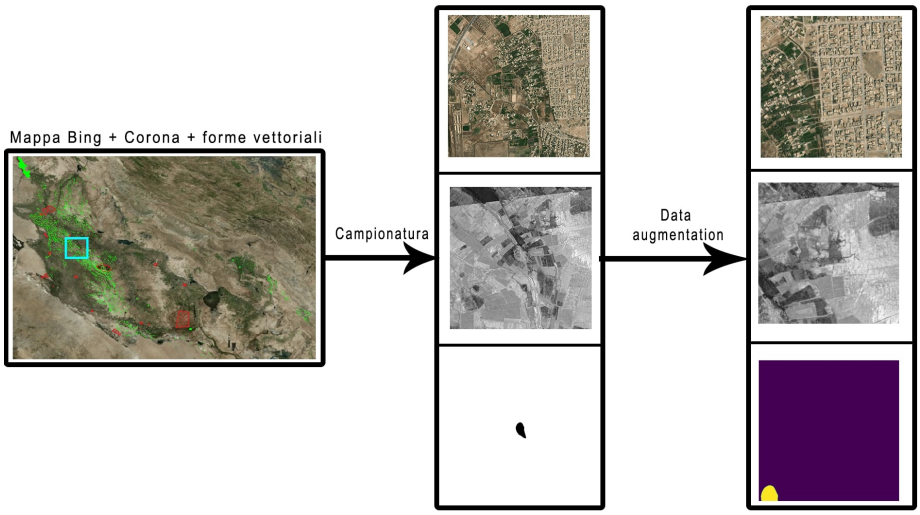


Figura 3.2: Processo di creazione del dataset

3.1.2 Modello di segmentazione semantica

Il modello utilizzato è lo stesso descritto nel capitolo 2.2 ovvero come architettura di segmentazione è stata utilizzata MANet [13] ed EfficientNet [20] come encoder, più nello specifico EfficientNet-B2, variante con 9.2 milioni di parametri.
Per inizializzare i pesi del modello sono stati scelti i pesi dei rispettivi modelli addestrati su ImageNet [8].

Come funzione di loss si è scelta la Focal Loss in quanto nasce per risolvere il problema di sbilanciamento delle classi durante l'addestramento, specificamente per il task di rilevamento di oggetti [21]. Tale funzione di costo si concentra sugli esempi che il modello classifica erroneamente piuttosto che su quelli che riesce e prevedere con sicurezza, cercando di garantire che le previsioni sugli esempi più difficili migliorino nel tempo anziché diventare eccessivamente fiducioso con quelle che classifica correttamente. Infatti, la Focal Loss, applicando il *down weighting*, riesce a ridurre l'influenza delle classificazioni "facili".

Per l'effettiva implementazione si è utilizzata la libreria python Segmentation Models [22] la quale permette di assemblare il proprio modello di segmentazione semantica (astraendone la complessità costruttiva che avviene attraverso moduli Pytorch, PyTorch nn.Module), scegliendo tra 9 architetture e più di 500 tipologie di encoder, il tutto attraverso un'API di alto livello.
Segmentation Models, oltre ad offrire architetture ed encoder, fornisce un'implementazione delle funzioni di costo più utilizzate nella *computer vision* così come delle metriche più diffuse.
In figura 3.3 si riporta lo schema finale del modello implementato (MANet, EfficientNet, SegmentationHead).

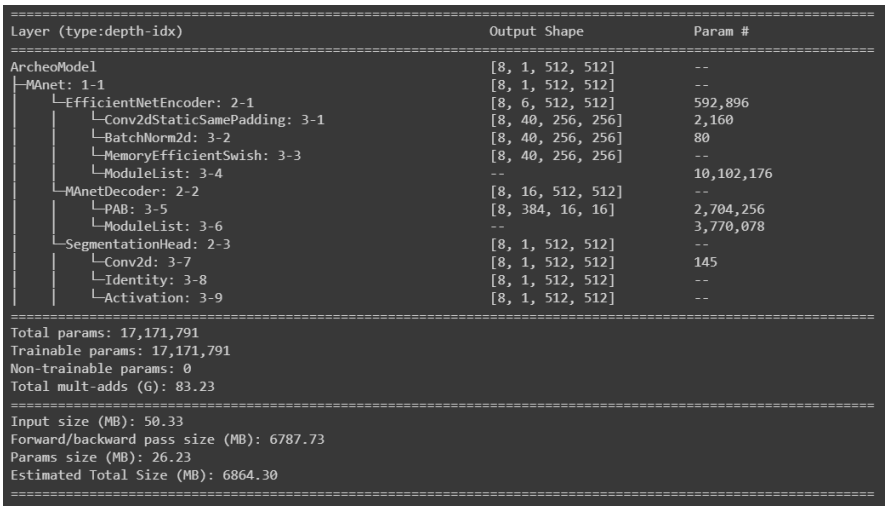


Figura 3.3: Schema della rete CNN utilizzata

3.1.3 Training

Con uno sguardo ai futuri lavori, in questa fase, sono state apportate delle leggere modifiche al processo proposto nella pubblicazione originale.
Infatti, oltre ai due modelli analizzati nel lavoro originale, si è deciso di testare una nuova configurazione che fa uso delle sole immagini Corona. Le configurazioni proposte sono quindi le seguenti:

1. Modello addestrato su immagini Bing, presente nel lavoro originario;
2. Modello addestrato su immagini Bing e Corona, presente nel lavoro originario;
3. Modello addestrato sulle sole immagini Corona, nuova proposta.

Inoltre, va menzionato il fatto che a causa delle limitate risorse computazionali a disposizione, per non saturare la ram della GPU si è reso necessario limitare le dimensioni del batch a 8.
Seguendo le precedenti sperimentazioni, si è optato per mantenere il numero delle epoche invariato (*max epochs = 20*).

3.1.4 Test

I risultati, sono riportati in termini di *Intersection over Union*, descritta nel paragrafo 2.2, durante la disamina della più recente pubblicazione.
Sebbene si fosse preventivamente ipotizzata una leggera diminuzione delle performance in termini di IoU dovuta alla diminuzione del batch a cui non è seguita nessuna modifica compensativa (come un possibile aumento delle epoche di addestramento), i risultati pur rispecchiando i lavori precedenti presentano un leggero miglioramento.
La tabella 3.2 mostra i risultati in termini di IoU per il modello addestrato per 20 epoche sulle 3 configurazioni del dataset proposte (dataset filtrato, risoluzione 2048x2048).

Modello	**Bing**	**Corona**	**IoU (%)**	**St. dev**
Modello 1	✓	×	81.3	0.40
Modello 2	✓	✓	**84.6**	0.37
Modello 3	×	✓	78.9	0.30

Tabella 3.2: Studio di conferma: performance segmentazione semantica

Le conclusioni verificano i risultati del precedente lavoro di ricerca, infatti, le performance delle varie configurazioni dei 3 modelli addestrati non ottengono variazioni significative in termini di IoU e rispecchiano in valori assoluti i risultati del lavoro originale.
Per quanto riguarda i risultati ottenuti nel task di Tell detection, la tabella 3.3 mostra gli score di ogni configurazione in termini di *Accuracy* e *Recall*, mentre la tabella 3.4 in termini di TP, TN, FP, FN (*True positive*, *True negative*, *False positive*, *False negative*). Si è deciso di includere anche gli score ottenuti impostando come soglia di cutoff 0.2.

Modello	**Accuracy@0.5**	**Recall@0.5**	**Accuracy@0.2**	**Recall@0.2**
Bing	61.42	61.47	75.24	88.06
BingCorona	**67.56**	74.77	**75.43**	90.88
Corona	37.43	25.53	50.10	67.84

Tabella 3.3: Studio di conferma: performance object detection

Modello	**TP**	**TN**	**FP**	**FN**
Bing@0.2	295	97	89	40
Bing@0.5	217	103	65	136
BingCorona@0.2	299	94	98	30
BingCorona@0.5	249	103	85	84
Corona@0.2	192	69	169	91
Corona@0.5	96	99	46	280

Tabella 3.4: Studio di conferma: matrice di confusione

Quello che si nota è un generale livello delle performance in linea con le attese, tuttavia vanno evidenziati alcuni comportamenti:

1. Il modello che ottiene performance migliori (*accuracy*) è, contrariamente a quanto si presupponeva, il modello addestrato su mappe Bing e Corona, sia con soglia di cut-off 0.5 che con soglia cut-off 0.2.

2. Per ogni modello, vi è un aumento dell'*accuracy* passando dalle soglie 0.5 a 0.2. A tale aumento va riportato il conseguente aumento di richiamo, indicando che appunto il modello si sbilancia maggiormente, tuttavia, ogni modello presenta delle peculiarità in questo sbilanciamento:

   - Il modello addestrato sulle sole mappe Bing ed il modello che include le mappe Corona (Bing, BingCorona) presentano dei comportamenti simili, infatti, passando dalla soglia di cut-off 0.5 allo soglia di cut-off 0.2, ciò che si nota è una drastica diminuzione dei falsi negativi ed un aumento cospicuo dei veri positivi indicando che la maggior parte delle previsioni scartate dalla soglia di 0.5 (troppo stringente), risultino poi dei vero positivi, seppur, di contro, vi è una leggera diminuzione dei veri negativi ed un leggero aumento dei falsi positivi.
   - Il modello addestrato sulle sole immagini Corona con soglia di cut-off pari a 0.5 presenta un altissimo numero di falsi negativi, stando ad indicare una soglia troppo stringente. Tuttavia, alla riduzione della soglia di cut-off a 0.2 ciò che si nota è sì una diminuzione del 60% dei falsi positivi ma più del 60% degli stessi si rivelano poi essere dei falsi positivi.

3. Ultimo importante aspetto che si evince dal confronto fra il modello Bing e il modello BingCorona, risiede nel fatto che l'inclusione di tali mappe permette al modello di ridurre (per le soglie di cut-off 0.2 e 0.5) il numero di falsi negativi in favore di un aumento di veri positivi

(con conseguente aumento di falsi positivi, seppur ridotto, infatti il calo del 38% dei falsi negativi del modello Bing@0.5 rispetto il modello BingCorona@0.5, è attribuito per il valore di 62% alla classe TP e per il valore di 38% alla classe FP).
Ciò che avviene è quindi un aumento di precisione nella classificazione della classe positiva. Nell'ambito dello studio che si presenterà nel prossimo paragrafo, riguardante l'addestramento del modello nella zona di AbuGhraib, si tenterà, seppure la quantità di dati a disposizione sia limitata, di dimostrare tale tendenza ricercando lo stesso effetto sul nuovo dataset.

3.2 Riaddestramento su AbuGhraib

Nel seguente paragrafo, si costituiscono le basi che porteranno poi alla discussione del secondo e terzo obiettivo di ricerca.
In tale paragrafo, infatti, si procederà illustrando i processi di acquisizione dei dati, che poco si discostano dal processo del lavoro precedente nonostante l'area di interesse risulti diversa, si illustreranno poi le tecniche di *data augmentation*, le quali vengono sfruttante in maniera maggiore, in quanto il dataset è di dimensione veramente ridotta, ed infine si descriveranno le configurazioni dei modelli e le due tipologie di *training* proposte (completo e a 2 fasi).
Descritto l'apparato sperimentale messo in atto, nel successivo capitolo si discuteranno i risultati che tale processo produce, verificando o meno gli obiettivi di ricerca.

3.2.1 Acquisizione dati

La prima fase progettuale riguarda, ovviamente, la generazione del dataset di immagini.
La differenza fondamentale risiede sicuramente nella zona. In particolare, la

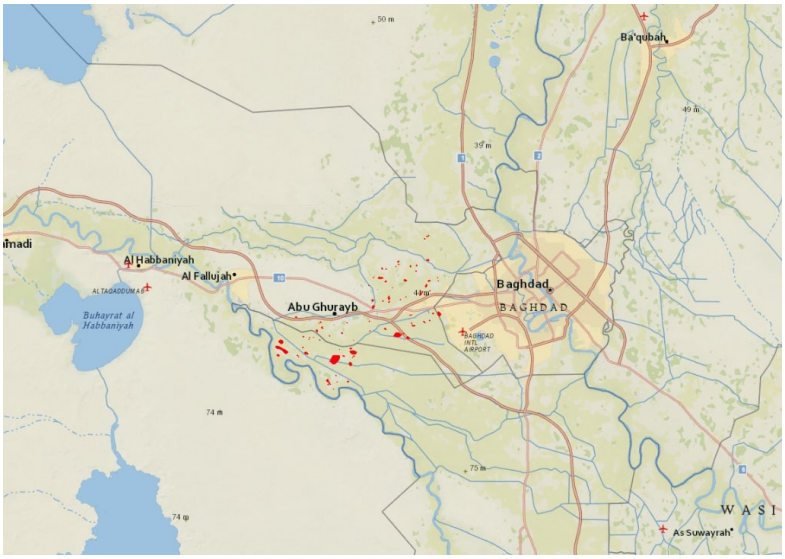
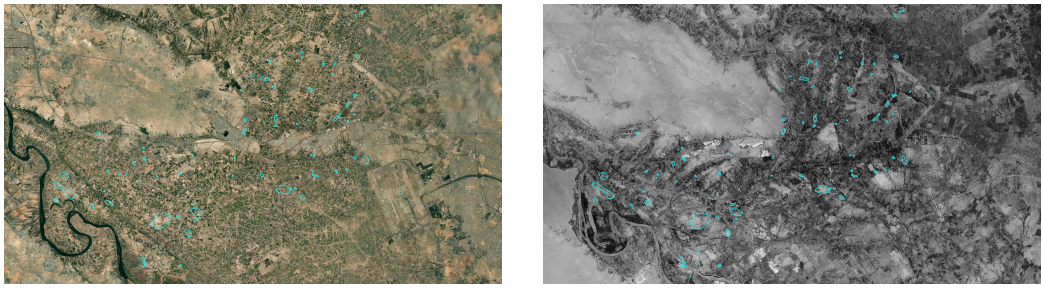


Figura 3.4: Area di interesse: visualizzazione di contesto

zona di interesse del seguente studio è il distretto di Baghdad denominato Abu Ghraib (in figura 3.4 si mostra la zona oggetto di studio da una prospettiva più alta).
Dalle annotazioni degli archeologici, nell'area di interesse, risultano 89 *Tell*, in figura 3.5 è riportata una schermata di esempio in cui sono evidenziati i siti su mappe Bing e su mappe Corona.

Sebbene nei lavori precedenti si procedesse con delle operazioni di filtraggio



(a) (b)

Figura 3.5: Visualizzazione dei siti nel distretto di AbuGhraib su mappe Bing (3.5a) e su mappe Corona (3.5b)

massive, la ridotta dimensione del dataset in questione rende l'operazione controproducente.
La fase di acquisizione dei dati coinvolge tre tipologie di immagini che vengono utilizzate nella fase di addestramento, validazione e testing.

Mappe Bing

Per quanto riguarda le immagini campionate da mappa Bing, il procedimento di campionamento è rimasto invariato, attraverso il plugin Quick Map Services [23] si campionano immagini 2048x2048 centrate sui *Tell* (seguendo il procedimento descritto nel paragrafo precedente).

Immagini Corona

Le mappe Corona sono un insieme di fotografie satellitari create dal progetto Corona [9], un programma di ricognizione spaziale avviato nel 1958 dalla DARPA. Queste mappe sono state utilizzate per la sorveglianza e l'analisi geografica durante la Guerra Fredda.
Le immagini dei *Tell* su mappe Corona, vengono campionate seguendo lo stesso procedimento, ovvero, centrate sui *Tell* con una risoluzione di 2048x2048 pixel.

Immagini negative

In tale progetto, le immagini negative (veri negativi), risultano avere duplice scopo. Risultano infatti di maggiore rilevanza servendo sia per bilanciare il dataset sia per aumentarne le dimensioni. Tuttavia, nella zona di interesse, non sono state selezionate in collaborazione con il team di archeologi delle zone da cui poter campionare delle immagini significative, vista la ridotta estensione della stessa.
Per ottenere tali immagini, è stato prima selezionato un intorno (poligono) della zona del distretto di AbuGhraib (seguendo delle indicazioni generali ricevute dal team di ricerca di archeologia), è stato poi creato un layer a

cui sono state sottratte le aree in cui sono presenti i siti e poi, attraverso la funzione QGIS "Punti casuali nei poligoni", sono stati generati 120 punti, ognuno con distanza minima di 2km dall'altro.
In figura 3.6 è presente la visualizzazione del processo di campionamento delle immagini negative appena descritto.

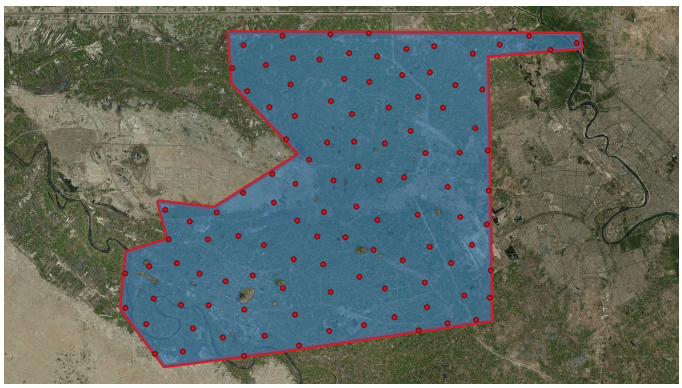


Figura 3.6: Processo di campionamento delle immagini negative

Al termine del processo di acquisizione dati si ottiene un dataset composto come riassunto nella tabella 3.5:

Classe	Dimensione	Percentuale
Positivi	89	42,58%
Negativi	120	57,42%
Totale	209	100%

Tabella 3.5: Composizione dataset AbuGhraib

3.2.2 Data augmentation

A primo impatto, l'utilizzo di tecniche di *data augmentation* può sembrare banale, tuttavia, l'applicazione di tali tecniche risulta di vitale importanza nella seguente trattazione.

Durante l'addestramento, in genere, si desidera applicare tecniche con una probabilità inferiore al 100% poiché è necessario poter disporre anche delle immagini originali nella pipeline di addestramento. Inoltre, è utile poter controllare l'entità della modifica dell'immagine. Se il dataset a disposizione presenta dimensioni ridotte come nel seguente caso, è necessario agire in modo aggressivo con l'utilizzo di più tecniche per prevenire l'*overfitting* della rete.
A tal proposito, si è scelto di utilizzare la libreria Albumentations [24] in quanto permette di applicare trasformazioni all'immagine di input, maschera di output ed è possibile specificare ulteriori target su cui applicare gli stessi parametri (le immagini Corona).
Sono stati definiti 3 set di trasformazioni:

1. Trasformazioni **geometriche** quali distorsione e mescolamento di parti dell'immagine (previa trasformazione in griglia);

2. Trasformazioni sullo **spazio dei colori** tra cui mescolamento canali, modifica casuale di luminosità, contrasto, saturazione, tinta ed equalizzazione adattiva dell'istogramma;

3. Trasformazioni **basate su maschere** (*kernel filters*) come aggiunta di sfocatura, rumore gaussiano e sfocatura laterale.

Attraverso il costrutto "OneOf" è stato poi possibile specificare che ad ogni immagine nel dataset dovesse venir applicata una trasformazione per gruppo. Oltre alle operazioni di ritaglio casuale e di ridimensionamento che vengono applicate di default sono state inserite anche un'operazione di capovolgimento, una di rotazione e un filtro di nitidezza, tutte con probabilità di applicazione del 50%.
In figura 3.7 è mostrato un esempio di applicazione della *pipeline* di *data augmentation* proposta.

Figura 3.7: Esempi dell'applicazione delle tecniche di data augmentation

3.2.3 Configurazioni dei modelli

I modelli proposti rispettano l'impianto sperimentale dei lavori su cui tale sperimentazione poggia.
Durante lo studio di conferma si è aggiunta una configurazione del modello addestrato sulle sole immagini Corona, questo proprio per poterla usare in questo step.
Si propongono quindi 3 configurazioni simmetriche nei modelli, ovvero per ogni modello proposto si utilizza il corrispettivo pre-addestrato del progetto precedente. I modelli sorgente sono quindi quelli già analizzati e discussi nello studio di conferma, il modello Bing, il modello BingCorona ed il modello Corona.
Specularmente, ognuno di essi, viene addestrato sulle relative mappe, riassumendo:

- **Bing_Bing**: modello sorgente Bing addestrato sulle sole mappe Bing;
- **BingCorona_BingCorona**: modello sorgente BingCorona addestrato sulle mappe Bing e sulle mappe Corona;
- **Corona_Corona**: modello sorgente Corona addestrato sulle sole mappe Corona,

3.2.4 Addestramento: fine-tuning

Ogni progetto di *machine learning* prevede una fase di *training* ed una di *testing*, tuttavia, nella seguente trattazione, non si procede con un classico

training del modello.
Generalmente, la fase di *training*, nel contesto del *deep learning* e delle reti neurali, si riferisce alla fase in cui un nuovo modello apprende da un set di dati. Durante questa fase, il modello adegua i propri pesi in base ai dati di input e all'output corrispondente, spesso utilizzando specifici *layer* e funzioni di attivazione.
Ciò che attualmente avviene è, invece, definito *fine-tuning*. Contrariamente al processo di addestramento iniziale, le tecniche di *fine-tuning* coinvolgono un ulteriore addestramento di un modello pre-addestrato su un set di dati più ristretto e specifico per un'attività particolare. L'obiettivo è capitalizzare la conoscenza acquisita dal modello durante il suo addestramento iniziale e adattarla per risolvere un compito più specifico (fig. 3.8). Questo approccio diventa particolarmente vantaggioso quando il nuovo set di dati è limitato per il nuovo compito, in quanto un addestramento da zero potrebbe portare all'*overfitting*.

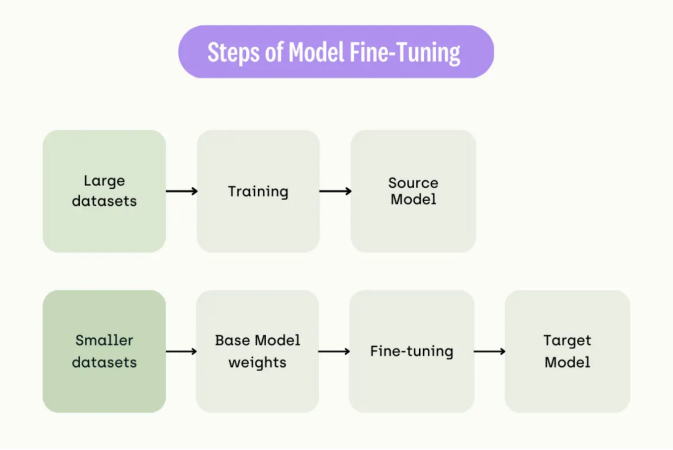


Figura 3.8: Schema del processo di finetuning
fonte: kili-technology.com

Nel contesto dell'attuale studio, si ha a disposizione un dataset estremamente ridotto, tuttavia, fortunatamente, il task su cui il modello va raffinato è molto simile a quello su cui il modello precedente è stato addestrato. Infatti, seppur la zona geografica differisca e con essa le caratteristiche delle zone

su cui vi è evidenza di siti, queste rappresentano problematiche riguardanti il dominio dell'archeologia, muovendosi nella sfera informatica del *machine learning* e gli oggetti da rilevare seppur archeologicamente diversi, possono essere considerati estremamente sovrapponibili.
Se il nuovo task è simile a quello originale, l'applicazione di tecniche di *transfer-learning* e dei procedimenti di *fine-tuning* sui modelli CNN, sono per lo stato dell'arte il modo più semplice che restituisce risultati migliori [25], [26].
Seguendo gli approcci comunemente utilizzati sono state sperimentate due tipologie base di fine-tuning, una sull'intero modello e una che si avvale di due fasi di addestramento separate, uno sulla sola Segmentation Head e una sull'intero modello con *learning rate* dimezzato (configurazione sperimentale aggiunta dopo la consultazione con Isaac Corley, coautore di Torchgeo [27] e attuale manutentore della repository Torchseg [28], fork attivo della libreria Python Segmentation Models [22], utilizzata per lo sviluppo del modello).

Full model fine-tuning

La prima tecnica di *fine-tuning* consta di un'addestramento completo del modello. Una delle problematiche dell'effettuare *fine-tuning* su tutto il modello (senza congelare i pesi di nessun *layer*), come già detto, riguarda il rischio di *overfitting*. Tale rischio si accentua quando il dataset target presenta dimensioni ridotte, come in questo caso.
Per scongiurare tale ipotesi, si sono utilizzate due *callback*, la *early stopping* per fermare l'addestramento nel momento opportuno e il *model checkpoint* per tener traccia del miglior modello.
Sapendo che una divergenza di tendenza tra il valore della *validation* e della *training loss* indica un possibile *overfitting*, l'*early stopping* monitora la *loss* calcolata sul set di validazione e qualora questa non diminuisca entro 15 epoche (parametro *patience*) il processo di addestramento viene bloccato.
Tuttavia, non sapendo se l'ultima iterazione abbia o meno generato un miglioramento nelle performance del modello, si è scelto di tener traccia, ad

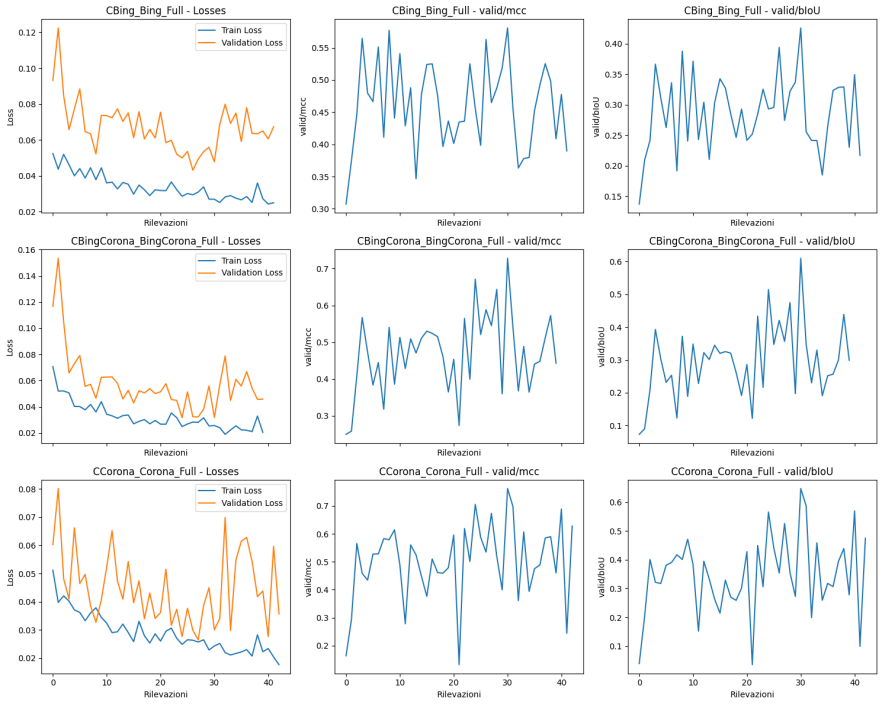


Figura 3.9: Diagrammi delle metriche monitorate a seguito del fine-tuning sull'intero modello per i modelli proposti

ogni iterazione, del *checkpoint* del modello che ottiene performance migliori (metrica MCC).
In figura 3.9 sono mostrati i valori per ogni metrica monitorata (train e valid loss, valid mcc e bIoU) per i tre modelli proposti.
Il *learning rate* rimane invariato e si procede con l'addestramento finché non si rileva una stagnazione della *loss* sul set di *validation*, indicando un possibile *overfitting*.

2 Stage fine-tuning

Questo secondo approccio, diviso in 2 fasi di addestramento separate, prevede di congelare i pesi dei layer profondi, così da non essere aggiornati durante la backpropagation e e lasciare che gli ultimi strati della rete vengano invece aggiornati.

Se il nuovo set di dati è piccolo, il motivo per limitare l'addestramento ai livelli più esterni è evitare di incorrere in *overfitting*. L'intera rete contiene circa 17 milioni di parametri, se addestrata su un dataset ridotto, vi è una maggiore possibilità che quest'ultima trovi una soluzione che si adatti al set di addestramento ma non generalizzi bene.
L'idea alla base del *transfer learning* è che la rete originale (i vari modelli Bing, BingCorona e Corona) abbia appreso una rappresentazione interna che funzionerà bene anche per il nuovo task. Addestrando solo gli ultimi *layer*, semplicemente si mantiene quella rappresentazione interna e la rete impara ad elaborarla per il nuovo compito. Poiché i *layer* finali contengono meno parametri dell'intera rete, c'è meno rischio di *overfitting*.
Prendendo atto dell'architettura del modello in uso, si è scelto di "congelare" tutti i pesi dei layer eccetto per quelli che costituiscono la "SegmentationHead" (layer finali, in quanto la rete è costituita macroscopicamente da Encoder EfficientNet, decoder MANet e SegmentationHead). A questo punto si effettua un primo addestramento con *learning rate* di default. Anche in questo caso si fa uso delle due *callback Pytorch* per *early stopping* e *model checkpointing*. Terminata questa prima fase, si procede "scongelando" tutti i *layer* e procedendo con la seconda fase dell'addestramento, stavolta con *learning rate* ridotto di 10 volte.
Come per la precedente casistica sono state monitorate le metriche d'interesse durante l'intero processo, le quali sono mostrate in figura 3.10.

3.2.5 Testing

Il processo di testing per il task di segmentazione semantica è identico a quello proposto nel lavoro precedente, ovvero si effettuano 10 test randomici. Tuttavia, come dimostrato, la metrica IoU presenta delle criticità se utilizzata per valutare i modelli in comparazione a questo stadio progettuale.
Infatti, le differenze percentuali dei vari valori IoU non indicano una maggioreminore capacità di rilevazione dei *Tell*, tuttavia, è dimostrabile che modelli con IoU maggiori tendano a produrre previsioni in generale più "precise"

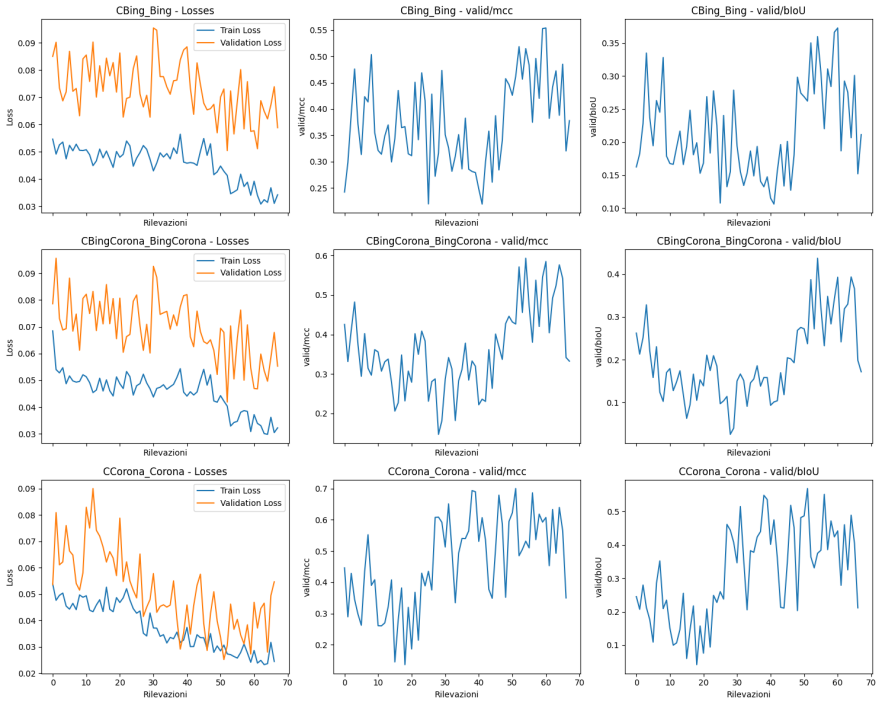


Figura 3.10: Diagrammi delle metriche monitorate a seguito del fine-tuning a 2 fasi per i modelli proposti

dove con il termine precise si intende che rispecchiano maggiormente il reale contorno delle maschere di verità dei *Tell* (oppure, in generale, l'aumento di IoU può derivare da una minore propensione alla rilevazione della classe positiva, infatti essendo calcolata come divisione tra intersezione ed unione, un modello restio nelle previsioni tenderà ad avere IoU tendenzialmente maggiore).

Nell'ambito della segmentazione semantica, ciò ha senso, tuttavia, l'utilizzo finale del modello riguarda l'assistenza dell'archeologo nel processo di rilevamento, ovvero, risulta di maggiore importanza fornire più evidenze possibili di eventuali *Tell*, piuttosto che poche evidenze dai contorni precisi. Oltretutto, molte volte, le maschere di verità risultano traslate di qualche pixel o addirittura, non essendo ancora stati indagati tutti i *Tell* sul campo, ve ne sono alcuni che non presentano maschera di verità.

Ulteriore considerazione che va fatta riguarda lo sbilanciamento del dataset,

sebbene il dataset nell'interezza non rappresenti un problema, si ricorda che l'attuale problema rientra nella classe della segmentazione semantica binaria e per queste metriche *pixel-wise* va controllata la percentuale di pixel di classe positiva rispetto quelli di classe negativa. In quest'ottica, il problema che si sta affrontando è altamente sbilanciato, in quanto, mediante, su un batch, la percentuale di pixel positivi è mediamente del 10%.
Al fine di rimpiazzare la metrica IoU è stata implementata una nuova metrica custom non presente su Pytorch Lightning. Prendendo spunto dalla classica Binary Intersection Over Union, si è sviluppata una versione batch-pixel-wise, che viene applicata solamente per la classe positiva, calcolata come segue:

$$IOU = \frac{true\_positives}{true\_positives + false\_positives + false\_negatives}$$

Oltre all'introduzione della bIoU (*Binary Intersection over Union*), di maggiore importanza, riguarda la scelta progettuale di valutare i modelli attraverso un'ulteriore metrica che prima veniva utilizzata solamente per valutare il task di rilevamento, il coefficiente di correlazione Matthews (*Matthews correlation coefficient*), il quale prende in considerazione tutti e quattro i valori nella matrice di confusione, e un valore alto (vicino a 1) significa che entrambe le classi sono previste bene, anche se una classe è sotto (o sovra) rappresentata in modo sproporzionato.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Alcune proprietà interessanti del MCC possono essere facilmente derivate dalla formula: quando il classificatore è perfetto (FP = FN = 0), il valore del MCC è 1, indicando una correlazione positiva perfetta. Al contrario, quando il classificatore sbaglia sempre (TP = TN = 0), otteniamo un valore di -1, che rappresenta una correlazione negativa perfetta (in questo caso, si può semplicemente invertire l'esito del classificatore per ottenere il classificatore ideale). Infatti, il valore del MCC è sempre compreso tra -1 e 1.

Sebbene la metrica IoU rappresenti senza dubbio una metrica consolidata per la comparazione di modelli di diversi studi nel campo della segmentazione semantica, va evidenziato come sempre più studi dimostrino l'importanza nell'utilizzo della metrica MCC come metrica di valutazione per la classificazione binaria [29], proprio per questo, si è scelto di utilizzarla anche a livello di pixel, per il task di segmentazione semantica.

Per ogni modello proposto e per entrambe le modalità di *fine-tuning*, nella tabella 3.6 sono riportate schematicamente le performance in termini di IoU, MCC e bIoU. Con il suffisso "_Full" si indica la configurazione su cui è stato effettuato un *fine-tuning* completo.

Modello	**IoU**	**MCC**	**bIoU**
Bing_Bing	86.12	34.04	21.53
BingCorona_BingCorona	85.77	**55.63**	**39.23**
Corona_Corona	85.09	47.27	33.19
Bing_Bing_Full	86.44	35.17	21.73
BingCorona_BingCorona_Full	**87.30**	55.31	38.26
Corona_Corona_Full	84.88	39.61	26.75

Tabella 3.6: Abu Ghraib: performance di segmentazione semantica

Dall'analisi della tabella 3.6 si può notare che tutti le configurazioni proposte raggiungono un buon livello di IoU. Tuttavia, come spiegato nel precedente paragrafo, un alto valore di IoU non corrisponde automaticamente ad una capacità di rilevamento.
In generale, i modelli addestrati utilizzando il *fine-tuning* a due stadi ottengono performance migliori, solamente il modello Bing differisce da questo comportamento.
Analizzando le performance delle metriche aggiunte, il modello "BingCorona_BingCorona" è il modello che più emerge, in quanto presente il valore più alto sia per la metrica MCC che per la metrica bIoU.

Fine-tuning test effect

Al fine di dimostrare analiticamente l'efficacia e quantificare l'importanza del processo di *fine-tuning* (obiettivo di ricerca nr.2 definito al paragrafo 2.3), seguendo lo stesso procedimento proposto per i modelli sviluppati, si sono testati i modelli definiti sorgente sul test set della zona di AbuGhraib, al fine di comparare le performance con i modelli proposti.
La tabella 3.7 mostra i risultati dei test effettuati sui modelli sorgenti comparate ai rispettivi modelli su cui è stato effettuato *fine-tuning*.

Modello	**IoU**	**MCC**	**bIoU**
Bing	82.24	**35.24**	**22.7**
Bing_Bing_Full	**86.44**	35.17	21.73
BingCorona	84.30	45.76	28.80
BingCorona_BingCorona	**85.77**	**55.63**	**39.23**
Corona	83.54	31.98	18.80
Corona_Corona	**85.09**	**47.27**	**33.19**

Tabella 3.7: Studio di comparazione con modelli baseline

Si può quindi concludere affermando che, nell'ambito della segmentazione semantica, il processo di *fine-tuning* ricopre un ruolo importante, in quanto su 2 modelli dei 3 proposti è possibile notare un aumento delle performance in termini di MCC nell'ordine di 12 punti percentuali ed in termini di bIoU nell'ordine di 10 punti percentuali. Per affermare che il modello senza *fine-tuning* non è in grado di adattarsi e di essere applicato efficacemente e che la soluzione sia il processo di *fine-tuning*, è necessario valutare il task di rilevamento, infatti, i soli risultati riguardanti il task di segmentazione semantica indicano solamente una tendenza.
Nel prossimo capitolo si procederà generando gli artefatti che consentiranno di valutare il processo di rilevamento così da poter trarre le conclusioni finali.

Capitolo 4

Risultati sperimentali

In questo capitolo, diviso in due parti distinte, si introdurrà nel primo e secondo paragrafo il processo che porta alla generazione dei rilevamenti, quindi si descriveranno le azioni che portano dalla generazione dell'output del modello alla rilevazione dei *Tell*, includendo anche i processi di generazione di tutti gli artefatti intermedi che risultano di particolare interesse ai ricercatori archeologici (es. le mappe di calore).
Nella seconda parte si discuteranno i risultati ottenuti applicando l'intero processo "Human in the loop" nella zona di Abu Ghraib, si analizzeranno quindi tutte le fasi di utilizzo del modello in collaborazione con il team di ricerca di archeologia e si presenteranno dei risultati derivanti dal concreto utilizzo del modello sul campo.

4.1 Creazione artefatti

Come discusso nel precedete capitolo, le valutazioni sul task di segmentazione semantica, da sole, non forniscono una visione completa sull'effettiva capacità di rilevazione del modello.
Al fine di valutare analiticamente il task di rilevamento, si utilizza la stessa metodologia adottata nei lavori precedenti.
L'output del modello rappresenta la stessa area dell'immagine in input tra-

sformata e ritagliata dove ogni pixel ha valore nell'intervallo $[0, 1]$ e valori prossimi a 1 indicano una maggiore probabilità che tale pixel appartenga all'area del *Tell*, al contrario, valori prossimi allo 0 indicano una bassa probabilità che il pixel appartenga al *Tell*. Proprio per questo, come prima cosa si ricostruisce l'immagine originale, applicando inversamente le trasformazioni iniziali. Dopodiché, in ordine, partendo dalla maschera di probabilità si applica un filtro gaussiano per preservare le forme riducendo il rumore e si applica la soglia di cut-off in modo da discretizzare la mappa generando dei poligoni, i quali rappresentano le previsioni dei *Tell*. Le soglie di cut-off usate sono due, 0.2 e 0.5, l'utilizzo di due soglie distinte fa si che per ogni input vengano generati due immagini Tiff (formato di file di grafica raster) con livelli di soglia diversi, uno più stringente dell'altro.

Generate le immagini Tiff, le previsioni raster sono state trasformate in forma vettoriale utilizzando la libreria GDAL (Geospatial Data Abstraction Library). Utilizzando gli shapefile, è poi possibile mediante il confronto con le annotazioni originali dei siti calcolare per ogni predizione la corretta classificazione.
Le possibili casistiche sono le seguenti:

- Se vi è completa o parziale intersezione tra previsione e maschera di verità, allora la previsione è classificata come TP (*true positive*);
- Se il modello genera una previsione vuota e non sono presenti siti, allora la previsione è classificata come TN (*true negative*);
- Se il modello predice una forma che non presenta intersezione con la maschera di verità allora la previsione è classificata come FP (*false positive*);
- Se è presente un sito e la previsione non rileva niente allora è classificata come FN (*false negative*).

Tale processo di confronto avviene per le maschere con soglia di cut-off 0.2 e 0.5, infatti, con diverse soglie di troncamento si producono previsioni diverse.

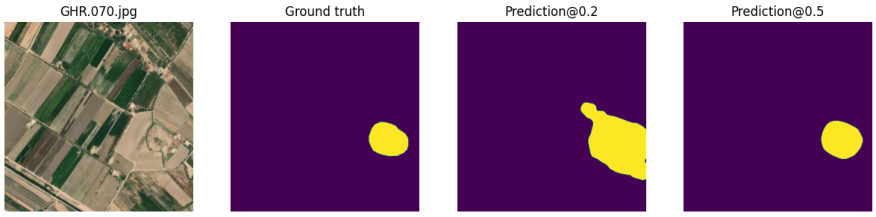


Figura 4.1: Variazione della previsione al variare della soglia di troncamento

In figura 4.1 si riporta un esempio in cui si mostrano le diverse maschere che si generano al variare della soglia di troncamento.
In figura 4.2 è mostrata una delle due casistiche che più vanno analizzate durante la scelta della soglia di troncamento, infatti come è possibile notare molte volte, scegliendo una soglia minore la previsione che emerge è corretta, rendendo quest'ultima un *true positive*.
Tuttavia, ciò che sperimentalmente è possibile dimostrare accada è che scegliendo 0.2 come soglia di cut-off, le previsioni siano troppo sbilanciate verso la classe positiva, creando un problema di falsi positivi, come si può osservare in figura 4.3, questo comportamento è comune a tutti i modelli.
Terminato il confronto per le immagini di test, il risultato viene salvato in un file GeoJson, che viene poi utilizzano per generare la matrice di confusione e calcolare i rispettivi valori di *accuracy* e *recall*.

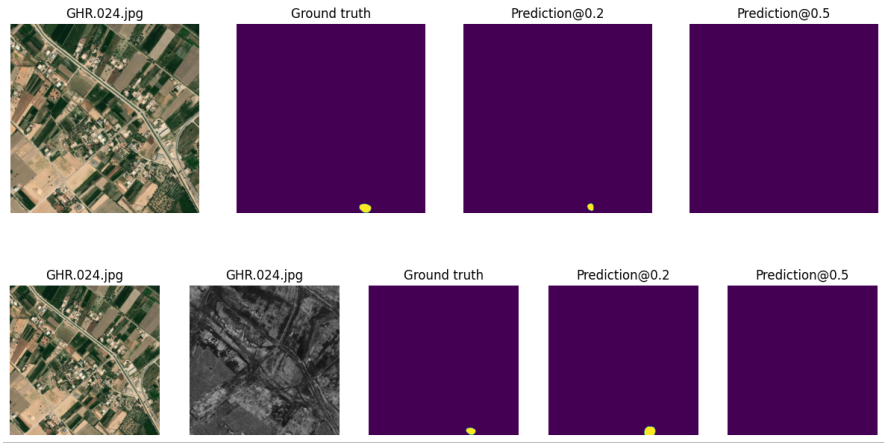


Figura 4.2: Previsioni al variare della soglia: emersione di un vero positivo

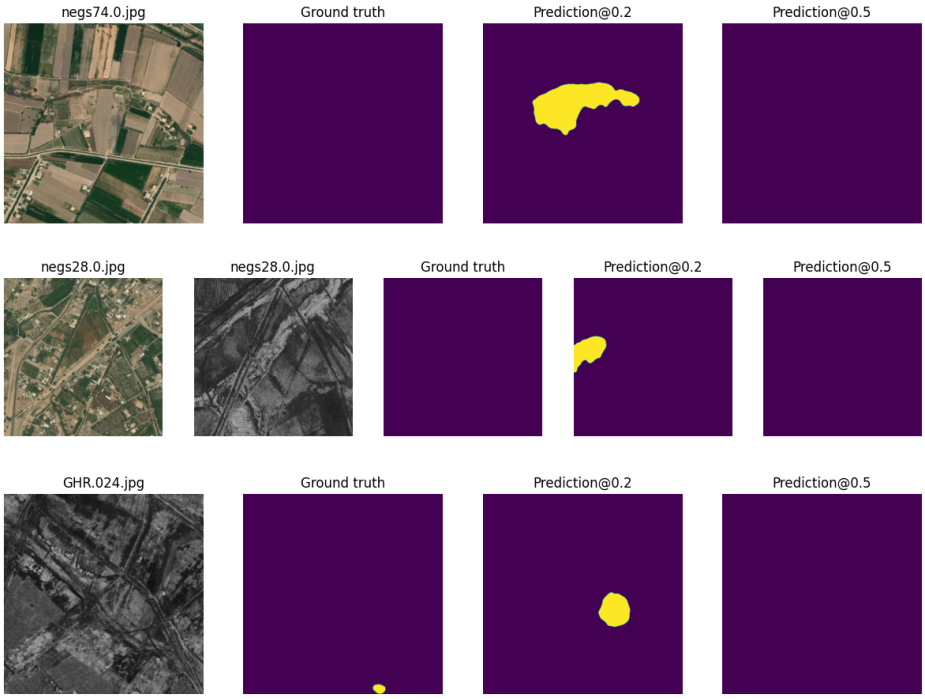


Figura 4.3: Previsioni al variare della soglia: emersione di un falso positivo

4.1.1 Generazione heatmap

Prima di passare alla discussione dei risultati ottenuti applicando il processo *human in the loop* nell'area di Abu Ghraib, in questo paragrafo si mostra il processo per la generazione delle *heatmap*, utili ai ricercatori archeologi in quanto, le previsioni discretizzate, mostrano solamente le aree per cui il modello è più o meno (dipendentemente dalla soglia utilizzata) sicuro della previsione. Utilizzare una mappa di calore, mostra tutto lo spettro di probabilità, guidando l'occhio dell'archeologo riesce a fornire un'indicazione, seppur meno precisa, di dove il modello prevede possa trovarsi un *Tell*.
Come prima cosa per generare la mappa di calore è suddividere l'area di interesse in rettangoli di dimensione uguale a quella dell'input della rete CNN sviluppata.
Nell'attuale studio riguardante la zona di Abu Ghraib, per comprendere tutti i siti archeologici derivanti dalle annotazioni si è dovuto dividere la zona di

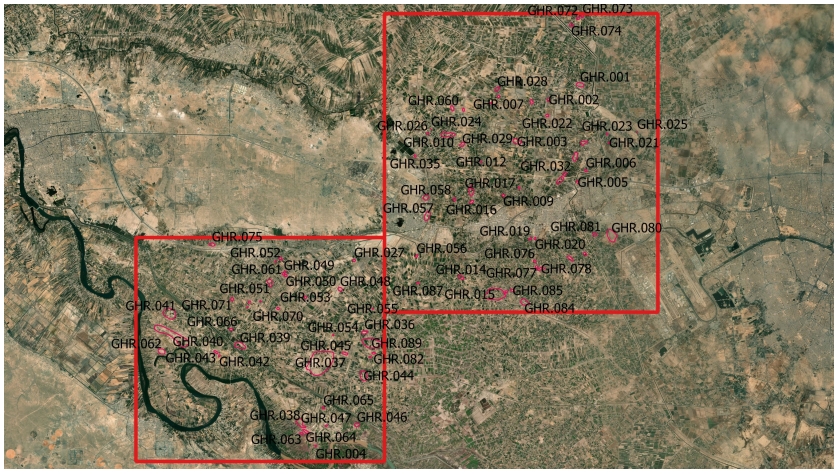


Figura 4.4: Divisione zona di Abu Ghraib in due sotto-aree

interesse in due sotto-aree, come riportato in figura 4.4.
Suddivisa la zona in sotto-aree, si campiona la mappa Bing e la mappa Corona, in modo da generare *n tile* di dimensione 1000x1000 pixel, i quali vengono passati in input al modello.
Generate le previsioni, si procede ri-assemblando le tessere in modo da generare un'unica immagine Tiff da poter importare sul software QGIS. Si riporta un esempio in figura 4.5.

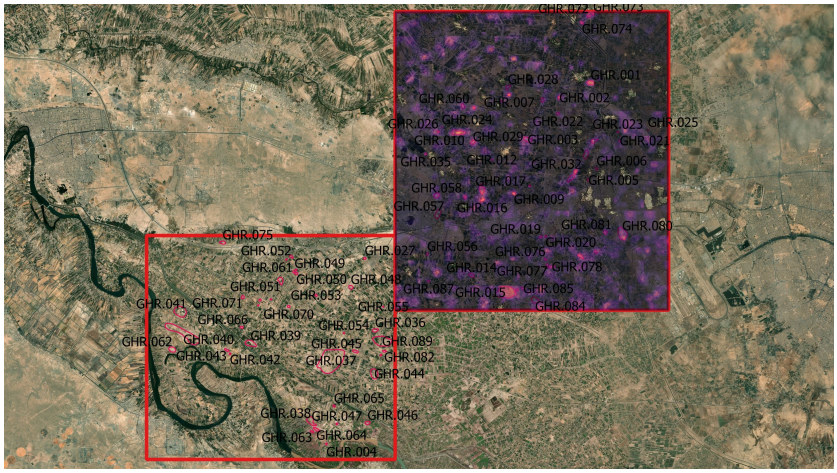


Figura 4.5: Abu Ghraib: visualizzazione di una mappa di calore

4.2 Risultati del test di rilevamento per la zona di Abu Ghraib

Nella fattispecie, per la zona di Abu Ghraib, avendo a disposizione una limitata quantità di siti, è stato possibile valutare il modello su un test set di 20 immagini. Le tabelle 4.1 e 4.2 riassumo i risultati rispettivamente con soglia di troncamento a 0.2 e 0.5.

Modello	**Accuracy**	**Recall**	**TP**	**TN**	**FP**	**FN**
Bing_Bing_Full	0.70	0.86	7	7	5	1
BingCorona_BingCorona	0.50	1	8	2	10	0
Corona_Corona	**0.75**	**1**	6	9	5	0

Tabella 4.1: Abu Ghraib: performance di rilevamento con soglia 0.2

Modello	**Accuracy**	**Recall**	**TP**	**TN**	**FP**	**FN**
Bing_Bing_Full	0.75	0.50	4	11	1	4
BingCorona_BingCorona	**0.90**	**0.88**	7	11	1	1
Corona_Corona	0.70	0.67	4	10	4	2

Tabella 4.2: Abu Ghraib: performance di rilevamento con soglia 0.5

Dall'analisi delle tabelle riassuntive, è evidente che utilizzando una soglia di troncamento troppo bassa (tabella 4.1) il numero di falsi positivi aumenta eccessivamente, addirittura, per il modello BingCorona quest'ultimi rappresentano il 50% delle previsioni. Tuttavia è interessante notare come con soglia 0.2 il modello che ottiene performance migliori di *accuracy* è il modello addestrato sulle sole immagini Corona.
Sebbene un suo utilizzo pratico non sia fattibile poiché non vi è certezza che per ogni area siano disponibili le mappe Corona, e spesso, come riportato nei precedenti lavori, queste presentino una risoluzione scadente, dovuta anche

a fattori ambientali quali la presenza di nuvole, si dimostra che per alcune aree come la zona di interesse di Abu Ghraib, le immagini derivanti dalle mappe Corona forniscono una inestimabile fonte di informazioni contestuali che vanno fornite alla rete CNN.
La tabella 4.2 dimostra empiricamente le precedenti supposizioni, infatti, il modello BingCorona ottiene un valore di *accuracy* pari a 0.90, inoltre, ulteriore fattore di estrema rilevanza è l'abbassamento dei valori di falsi positivi. Va evidenziato il fatto che seppur il modello BingCorona avesse ottenuto performance più basse rispetto al modello senza mappe Corona, questo non avrebbe escluso il possibile utilizzo in parallelo dei due modelli, o persino lo sviluppo di un modello che avesse fatto uso dei due classificatori in modo indipendente, ad esempio, un modello *ensemble*.

4.3 Test d'uso: applicazione del processo human in the loop

Dopo aver introdotto il contesto applicativo e bibliografico dell'attuale lavoro, aver descritto tutte le fasi sperimentali e aver presentato e discusso i risultati teorici del progetto riguardante la zona geografica del distretto di Abu Ghraib, si è avuta la possibilità di testare concretamente l'impianto sperimentale fin qui trattato attraverso la collaborazione con il team di ricerca di archeologia.
L'ultimo paragrafo di questo lavoro riguarda quindi l'analisi e la discussione dell'applicazione del processo *human in the loop* e dell'identificazione e discussione dei suoi punti di forza e criticità all'atto pratico.

4.3.1 L'AI come strumento di supporto, non come sostituto

Lo sviluppo e l'implementazione dei modelli proposti nel seguente lavoro ha sempre avuto come obiettivo ultimo l'individuamento delle aree potenzial-

mente contenenti siti archeologici, fornendo così un'indicazione agli esperti del settore, rimane poi compito dell'esperto tracciare la forma precisa del sito archeologico in base alla sua esperienza e competenza. Pertanto, il modello è concepito per offrire un supporto agli archeologi anziché sostituirsi a essi. Attualmente, gli esperti del settore, in questo caso gli archeologici, utilizzano le mappe predittive generate dal modello come ulteriore informazione contestuale, ciò che avviene quindi non è una sostituzione dell'esperto di dominio, ma l'utilizzo dell'AI come strumento per generare informazioni contestuali aggiuntive, sotto la supervisione dell'esperto, utili per fornire sempre più punti di vista all'archeologo.
Questo approccio consente agli archeologi di confrontare simultaneamente più mappe all'interno di un qualunque software GIS, scegliendo di usare e ponderare maggiormente determinate informazioni in base alla propria esperienza nell'ottica di generare indicazioni e annotazioni il più accurate possibili.

4.3.2 Oltre l'AI: L'importanza dell'esperto di dominio, la verifica sul campo

Dopo aver generato le previsioni del modello sulla zona di interesse, normalmente queste vengono confrontate con le annotazioni archeologiche, tuttavia, non sempre tali annotazioni sono aggiornate o derivano da verifiche sul campo.
Grazie al processo di collaborazione, è stato possibile fornire tale previsioni al team di ricerca prima che quest'ultimo potesse partire in missione e quindi verificare sul campo le annotazioni. Ciò si è rivelato altamente utile principalmente per due motivi, con rilevanza archeologica e informatica:

- Verifica delle annotazioni: la maggior parte delle annotazioni stilate dagli archeologi comprendono evidenze che derivano da verifiche sul campo di precedenti missioni, tuttavia, seppur con minor frequenza, tali annotazioni contengono evidenze generate da remote sensing. Utilizzare la mappa predittiva generata dal modello come overlay su un

qualunque software GIS permette in tempo ridotto di ottenere ulteriori confermeinformazioni su tali evidenze;

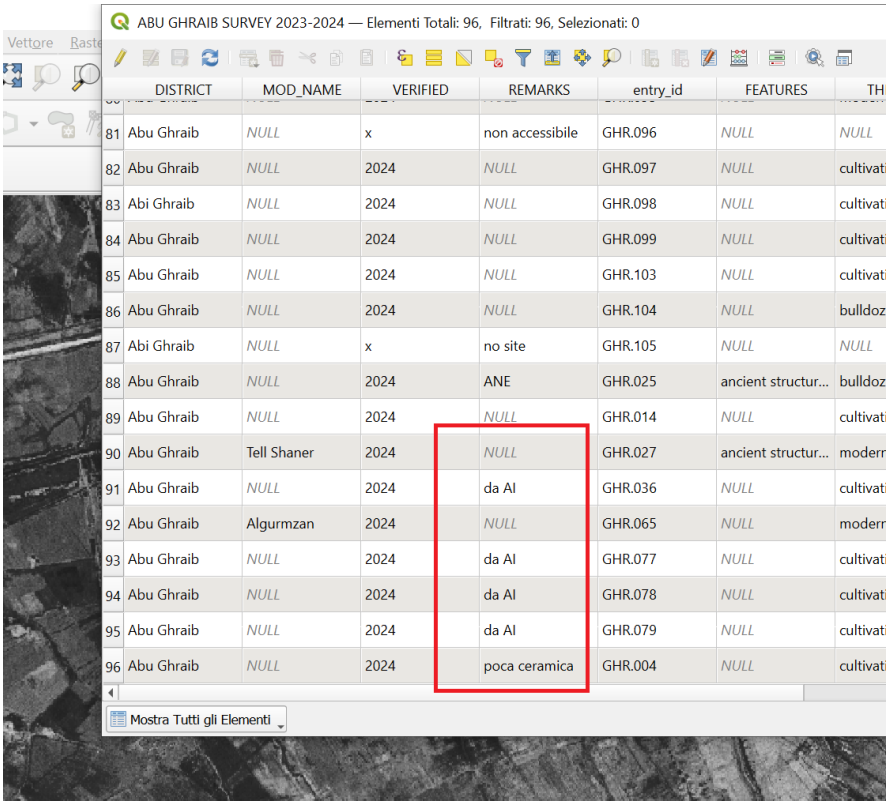
- Verifica delle proposte del modello: da una verifica preliminare della mappa predittiva, sono emerse alcune previsioni in aree *fringe* di scarso interesse per gli archeologi essendo state identificate come aree con bassa probabilità di presenza di un Tell.
  Sulla base non solo della probabilità di verosimiglianza ma anche sulla prossimità ai percorsi prestabiliti per effettuare le verifiche ad altri siti, sono stati selezionati 8 siti tra quelli proposti dal modello.

4.3.3 Analisi dei siti rilevati da AI

Dalla verifica sul campo, è emerso che, delle 8 zone proposte dal modello, 4 si sono rivelate dei siti confermati mentre le altre 4 non lo sono state. Sebbene a primo impatto possa sembrare un dato non del tutto positivo, in quanto il 50% di precisione non lo è, va ricordato che tali previsioni, in tutto l'impianto sperimentale dei capitoli precedenti, sono state classificate come dei falsi positivi.
Tutti i precedenti studi, hanno avuto tra gli obiettivi quello di non lasciar troppa libertà di previsione al modello, in quanto, un elevato numero di falsi positivi risulterebbe in una perdita di affidabilità nelle previsioni, soprattutto in un contesto di applicazione reale, dove la frequenza di riscontro di un *Tell* è ridotta e l'analisi di un numero elevato di zone (proposte dal modello) risulterebbe in un utilizzo inefficiente delle risorse, rappresentate dal tempo dedicato dagli archeologi.
Il modello è perciò stato penalizzato nell'effettuare tali previsioni, tuttavia, l'esito di tale verifica sul campo ha dimostrato che dal punto di vista informatico è possibile ipotizzare sia presente una buona percentuale di falsi positivi che in realtà sono dei veri positivi (oltre a quelli classificati come falsi positivi a causa di annotazioni errate) e dal punto di vista archeologico, è possibile concludere che il modello abbia avuto una precisione del 100%.

Infatti, se nei precedenti lavori sono state differenziate le tipologie di errore del modello, tra quelle che avrebbe commesso anche un esperto del dominio, come un archeologo, e quelli che invece sarebbe riuscito ad evitare, si deve far notare che le 8 previsioni proposte dal modello sono previsioni in aree che gli archeologi avrebbero scartato, in questo senso, si ha un tasso di successo del 100%.

Il processo *human in the loop* si conclude con l'aggiornamento delle annotazioni a seguito delle verifiche sul campo, oltre a quelle effettuate in autonomia dal team di ricerca di archeologia, si dimostra come l'utilizzo del modello sia ormai integrato nei processi di lavoro degli archeologi con l'inserimento nelle nuove annotazioni della dicitura "da AI" per identificare la provenienza del rilevamento (riportate in figura 4.6).



	DISTRICT	MOD_NAME	VERIFIED	REMARKS	entry_id	FEATURES	TH
81	Abu Ghraib	NULL	x	non accessibile	GHR.096	NULL	NULL
82	Abu Ghraib	NULL	2024	NULL	GHR.097	NULL	cultivati
83	Abi Ghraib	NULL	2024	NULL	GHR.098	NULL	cultivati
84	Abu Ghraib	NULL	2024	NULL	GHR.099	NULL	cultivati
85	Abu Ghraib	NULL	2024	NULL	GHR.103	NULL	cultivati
86	Abu Ghraib	NULL	2024	NULL	GHR.104	NULL	bulldoz
87	Abi Ghraib	NULL	x	no site	GHR.105	NULL	NULL
88	Abu Ghraib	NULL	2024	ANE	GHR.025	ancient structur...	bulldoz
89	Abu Ghraib	NULL	2024	NULL	GHR.014	NULL	cultivati
90	Abu Ghraib	Tell Shaner	2024	NULL	GHR.027	ancient structur...	moderr
91	Abu Ghraib	NULL	2024	da AI	GHR.036	NULL	cultivati
92	Abu Ghraib	Algurmzan	2024	NULL	GHR.065	NULL	moderr
93	Abu Ghraib	NULL	2024	da AI	GHR.077	NULL	cultivati
94	Abu Ghraib	NULL	2024	da AI	GHR.078	NULL	cultivati
95	Abu Ghraib	NULL	2024	da AI	GHR.079	NULL	cultivati
96	Abu Ghraib	NULL	2024	poca ceramica	GHR.004	NULL	cultivati

Figura 4.6: Schermata in cui sono visibili le annotazioni archeologiche

4.3.4 Una nuova classe di errore: il dubbio positivo

L'intelligenza artificiale ha rivoluzionato numerosi settori, una tendenza sempre più comune è l'utilizzo di modelli pre-addestrati in modo trasversale in diversi settori. Questi modelli, sebbene potenti, spesso producono risultati che non sono applicabili a situazioni reali questo perché manca il coinvolgimento degli esperti di settore, che possiedono una comprensione profonda del dominio di applicazione.
Tale tendenza si scontra con la cosiddetta "barriera del significato" [17]. Per garantire un significato ai risultati generati dai modelli di AI, è necessario comprendere il dominio di applicazione. Senza questa comprensione, i risultati possono essere inutili o addirittura fuorvianti. Questo sottolinea l'importanza del coinvolgimento degli esperti di dominio nella progettazione e nell'implementazione dei modelli di AI. Nel contesto dell'attuale studio il ruolo di collaborazione dell'archeologo deve essere costante (all'interno del loop), è proprio grazie ad esso che problemi come l'overdiagnosis possano essere interpretati e contestualizzati.
Nel campo medico, l'overdiagnosis è un problema ben noto. Ad esempio, nel rilevamento dei tumori, un overdiagnosis può portare a trattamenti inutili e potenzialmente dannosi. Tuttavia, nel contesto dell'attuale studio, l'overdiagnosis assume un aspetto completamente diverso.
Infatti, mentre la maggior parte dei lavori precedenti si è concentrata sulla riduzione del numero di falsi positivi, in questo studio si riconosce che oltre alle cause sopra citate come *label* errate o siti non ancora verificati sul campo, molti dei falsi positivi possono rivelarsi veri positivi attraverso l'intervento e il parere di un archeologo.
Nell'ambito di questo studio, è possibile quindi introdurre una nuova classe di errore, il "**dubbio positivo**". Sebbene un eccessivo numero di falsi positivi sia una situazione in genere da evitare, valori alti potrebbero star ad indicare una buona capacità di rilevamento da parte del modello, tuttavia, per l'effettiva valutazione è necessario l'intervento dell'esperto di dominio.
Le figure 4.7, 4.8, 4.9 (in ordine, mappa Bing, mappa Corona e mappa di

calore) mostrano visivamente due siti che, prima dell'ultima verifica sul campo, appartenevano alla classe dei "dubbi positivi", ovvero il modello rilevava la loro presenza, tuttavia, le annotazioni archeologiche non li includevano facendoli risultare dei falsi positivi. A seguito dell'ultima verifica sul campo, le previsioni sono state confermate e censite con gli identificativi "GHR.078" e "GHR.079" rendendoli dei vero positivi.
In ultima analisi, ciò che risulta di particolar rilievo è l'analisi delle singole previsioni. Infatti, come si può notare, entrambi i *Tell* non sono in alcun modo visibili dalle mappe Bing bensì dalle sole mappe Corona, ciò fa concludere che, in questo specifico caso, il modello sia stato in grado di utilizzare entrambe le informazioni contestuali correttamente.

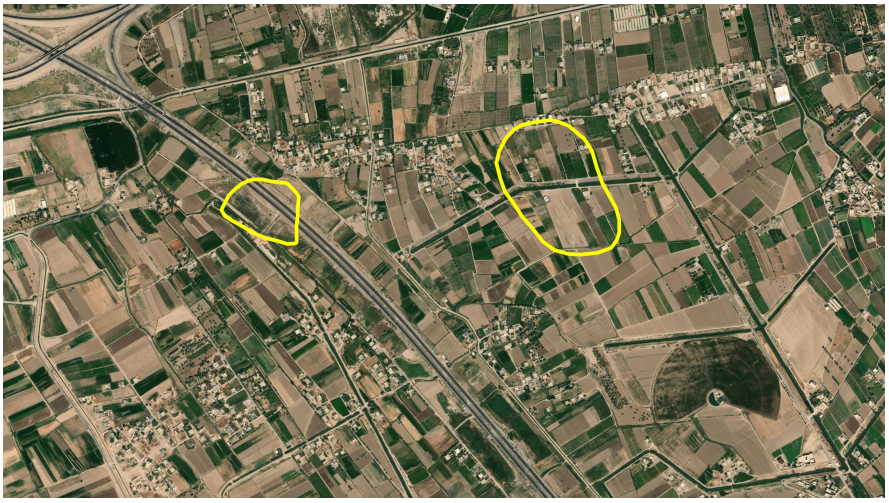


Figura 4.7: Caso esemplare: mappa Bing

Figura 4.8: Caso esemplare: mappa Corona

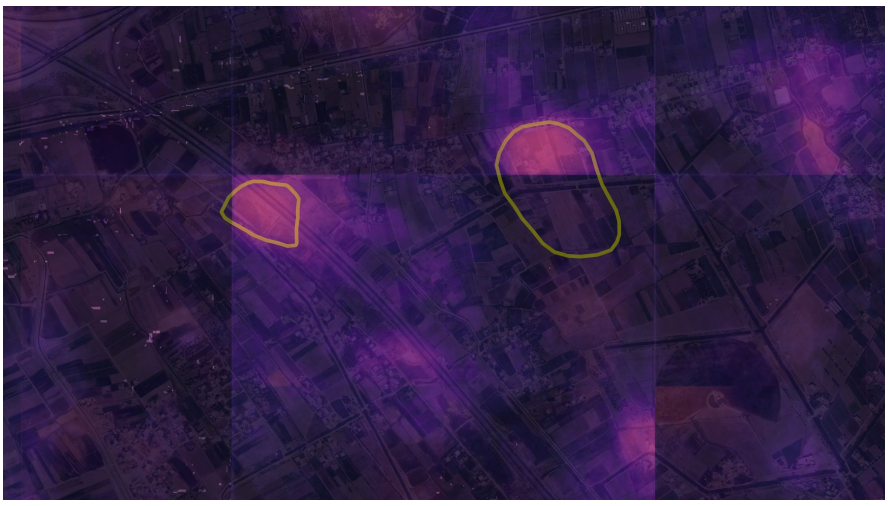


Figura 4.9: Caso esemplare: mappa di calore

Conclusioni

4.4 Limiti e sviluppi futuri

Nel presente lavoro di tesi si è presentato un percorso di ricerca che ha combinato archeologia e intelligenza artificiale, in particolare il deep learning, per affrontare una sfida specifica: la rilevazione di siti archeologici nella pianura alluvionale mesopotamica attraverso l'analisi di immagini satellitari e mappe Corona.
Si è iniziato con una breve introduzione al contesto e alla motivazione di questo lavoro, esplorando l'importanza del remote sensing e della segmentazione semantica nel campo dell'archeologia. Successivamente, si è presentata una revisione della letteratura e delineata la soluzione proposta per affrontare i vari obiettivi di ricerca prefissati.

Il cuore di questo lavoro ha riguardato lo studio di verifica e riproducibilità dei lavori precedenti, così da poter discutere l'importanza dei processi di *fine-tuning* del modello su nuove aree come quella di Abu Ghraib.

I risultati sperimentali hanno mostrato l'efficacia del metodo proposto. Si sono creati artefatti utili come le heatmap ottenendo risultati promettenti nel test di rilevamento per la zona di Abu Ghraib. Tuttavia, forse l'aspetto più importante di questo lavoro è stato l'approccio *human in the loop* che si è adottato. Si è sottolineata l'importanza dell'AI come strumento di supporto, non come sostituto, per l'esperto di dominio, introducendo anche una

nuova classe di errore, il "**dubbio positivo**", per sottolineare l'importanza del processo della verifica sul campo.

Questo lavoro apre diverse prospettive future. Senza dubbio alcuni dei possibili sviluppi futuri potrebbero essere:

1. Sviluppo di un modello *ensemble* che possa utilizzare indipendentemente due o più sotto-modelli addestrati su mappe geografiche differenti;
2. Effettuare uno studio per trovare la miglior configurazione di tecniche di data augmentation;
3. Implementare una metodologia adattiva per la scelta della soglia di troncamento;
4. Integrazione di annotazioni riguardanti i corsi d'acqua, infatti in molte rilevazione errate si rilevano dei paleoalvei.

4.5 Conclusioni

Ripercorrendo le tappe del seguente lavoro, in merito agli obiettivi di ricerca prefissati nel paragrafo 2.3, in ordine, si può concludere:

- **RQ1**: lo studio di conferma ha dato esito positivo, i risultati ottenuti sono in linea con quelli ottenuti nell'articolo Casini L., Marchetti N., Montanucci A. et al. [4]. Si è quindi verificato l'effettivo lavoro svolto e la riproducibilità dello stesso;
- **RQ2**: utilizzando tale studio di conferma come base di partenza si sono implementate due differenti strategie per effettuare *fine-tuning* nell'area del distretto di Baghdad denominato Abu Ghraib. Attraverso uno studio comparativo tra i 3 modelli baseline dello studio di conferma e 6 configurazioni totali derivanti dalle 2 strategie di *fine-tuning* applicate a 3 modelli proposti, si è dimostrato empiricamente che per la zona

distrettuale di Abu Ghraib, il processo di ri-addestramento produca degli effettivi miglioramenti rendendo i modelli più precisi in termini di IoU (per i modelli Corona e BingCorona, vi è un significativo aumento anche in termini di MCC e bIoU);

- **RQ3**: nell'ambito dello studio del ruolo delle mappe Corona (terzo e ultimo obiettivo di ricerca) è stato possibile constatare che nel task di segmentazione semantica applicata nella zona di Abu Ghraib (dataset con dimensione ridotta) il modello che ottiene performance migliori è il modello BingCorona, il quale include le immagini Corona.
  In generale, l'inclusione delle mappe Corona sembra generare contorni delle previsioni più accurate permettendo di aumentare i valori delle metriche pixel-wise. Per il task di rilevazione ciò che si nota analizzando il modello Bing ed il modello BingCorona è la diminuzione dei falsi negativi in favore di veri positivi.
  Si conclude valorizzando l'importanza dell'esperto di dominio, introducendo la classe di errore dei **dubbi positivi** come soluzione alla classificazione degli errori del modello. Durante la presentazione di un caso esemplare per rappresentare la classe dei dubbi positivi, si utilizza un esempio dimostrando come il modello BingCorona sia riuscito ad utilizzare correttamente le informazioni aggiuntive contenute nelle mappe Corona.

In conclusione, gli obiettivi del progetto possono essere considerati raggiunti, si spera inoltre che il presente lavoro possa contribuire a promuovere ulteriori ricerche nell'intersezione tra l'archeologia e l'intelligenza artificiale e che sia di aiuto nel sottolineare l'importanza del coinvolgimento degli esperti di dominio in questo processo.

4.6 Disponibilità del codice e dei dati

Oltre alle informazioni specifiche fornite all'interno del documento, tutto il codice, i dati e le varie risorse sono disponibili su GitHub (link alla reposi-

tory). Per quanto riguarda i dati geografici, tutti i dati visualizzati rientrano nelle condizioni di corretto utilizzo dei dati geografici per scopi accademici.
La creazione delle mappe è avvenuta rispettando i termini d'uso delle API di Microsoft Bing Maps, la relativa visualizzazione delle mappe è avvenuta tramite software open source regolamentata dalle licenze GNU di QGIS e QuickMapsServices.
L'elaborazione finale delle mappe è ottenuta attraverso il software sviluppato e disponibile su Github

Bibliografia

1. Bickler, S. H. Machine learning arrives in archaeology. *Advances in Archaeological Practice* **9,** 186–191 (2021).
2. Mantovan, L. & Nanni, L. The computerization of archaeology: Survey on artificial intelligence techniques. *SN Computer Science* **1,** 1–32 (2020).
3. Orengo, H. A. *et al.* Automated detection of archaeological mounds using machine-learning classification of multisensor and multitemporal satellite data. *Proceedings of the National Academy of Sciences* **117,** 18240–18250 (2020).
4. Casini, L., Marchetti, N., Montanucci, A., Orrù, V. & Roccetti, M. A human–AI collaboration workflow for archaeological sites detection. *Scientific Reports* **13,** 8699 (2023).
5. Casini, L., Orrù, V., Roccetti, M. & Marchetti, N. *When machines find sites for the archaeologists: A preliminary study with semantic segmentation applied on satellite imagery of the Mesopotamian floodplain* in *Proceedings of the 2022 ACM Conference on Information Technology for Social Good* (2022), 378–383.
6. Roccetti, M., Casini, L., Delnevo, G., Orrù, V. & Marchetti, N. *Potential and limitations of designing a deep learning model for discovering new archaeological sites: A case with the Mesopotamian floodplain* in *Proceedings of the 6th EAI International Conference on Smart Objects and Technologies for Social Good* (2020), 216–221.

7. Torrey, L. & Shavlik, J. in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques* 242–264 (IGI global, 2010).

8. Deng, J. *et al. Imagenet: A large-scale hierarchical image database* in *2009 IEEE conference on computer vision and pattern recognition* (2009), 248–255.

9. Nasa. Corona imagery. *https://data.nasa.gov/dataset/CORONA-Satellite-Photography/4rni-qjx7/about_data* (1959-1972).

10. Guyot, A., Lennon, M., Lorho, T. & Hubert-Moy, L. Combined detection and segmentation of archeological structures from LiDAR data using a deep learning approach. *Journal of Computer Applications in Archaeology* **4,** 1 (2021).

11. Argyrou, A. & Agapiou, A. A Review of Artificial Intelligence and Remote Sensing for Archaeological Research. *Remote Sensing* **14,** 6000 (2022).

12. Ronneberger, O., Fischer, P. & Brox, T. *U-net: Convolutional networks for biomedical image segmentation* in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18* (2015), 234–241.

13. Li, R. *et al.* Multiattention network for semantic segmentation of fine-resolution remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing* **60,** 1–13 (2021).

14. Sech, G. *et al. Tranfer Learning of Semantic Segmentation Methods for Identifying Buried Archaeological Structures on Lidar Data* in *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium* (2023), 6987–6990.

15. Marchetti, N. *et al.* The rise of urbanized landscapes in Mesopotamia: The QADIS integrated survey results and the interpretation of multilayered historical landscapes. *Zeitschrift für Assyriologie und vorderasiatische Archäologie* **109,** 214–237 (2019).

16. Ribani, R. & Marengoni, M. *A Survey of Transfer Learning for Convolutional Neural Networks* in *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)* (2019), 47–57.

17. Casini, L., Roccetti, M., Delnevo, G., Marchetti, N. & Orrù, V. The Barrier of meaning in archaeological data science. *arXiv preprint arXiv:2102.06022* (2021).

18. Marchetti, N. FloodPlains Project. *The FloodPlains Project has been developed in the framework of the European Union Project EDUU – Education and Cultural Heritage Enhancement for Social Cohesion in Iraq, EuropeAid CSOLA/2016/382-631, coordinated by Nicolò Marchetti* (2020).

19. He, K., Zhang, X., Ren, S. & Sun, J. *Deep residual learning for image recognition* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), 770–778.

20. Tan, M. & Le, Q. *Efficientnet: Rethinking model scaling for convolutional neural networks* in *International conference on machine learning* (2019), 6105–6114.

21. Lin, T.-Y., Goyal, P., Girshick, R., He, K. & Dollár, P. *Focal loss for dense object detection* in *Proceedings of the IEEE international conference on computer vision* (2017), 2980–2988.

22. Iakubovskii, P. Segmentation Models Pytorch. *https://smp.readthedocs.io/en/latest/*.

23. plugin, Q. P. QuickMapServices. *https://plugins.qgis.org/plugins/quick_map_services/* (2024).

24. Buslaev, A. *et al.* Albumentations: Fast and Flexible Image Augmentations. *Information* **11.** ISSN: 2078-2489. https://www.mdpi.com/2078-2489/11/2/125 (2020).

25. Poojary, R. & Pai, A. *Comparative Study of Model Optimization Techniques in Fine-Tuned CNN Models* in *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)* (2019), 1–4.

26. Yin, X., Chen, W., Wu, X. & Yue, H. *Fine-tuning and visualization of convolutional neural networks* in *2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA)* (2017), 1310–1315.

27. Stewart, A. J. *et al. TorchGeo: Deep Learning With Geospatial Data* 2022. arXiv: 2111.08872 [cs.CV].

28. Corley, I. Torchseg. *https://github.com/isaaccorley/torchseg*.

29. Chicco, D. & Jurman, G. The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification. *BioData Mining* **16,** 1–23 (2023).

Ringraziamenti

Giunto al termine di questo impegnativo e stimolante percorso di studi, sento di dover ringraziare tutte le persone che mi sono state vicine durante la mia carriera universitaria, soprattutto, un ringraziamento particolare va al Professor Marco Roccetti relatore di questa tesi.
Ringrazio inoltre la Dott.ssa Valentina Orrù ed il Professor Nicolò Marchetti per la disponibilità ed il loro contributo nello svolgimento di questo progetto.
Un doveroso e sentito ringraziamento è riservato a mio fratello Mirco, che è sempre stato al mio fianco, ai miei genitori, che mi hanno supportato in questi anni e agli amici del Caffè dell'Accademia e del FCM con cui ho condiviso momenti indimenticabili.