From ScanDDM to ART: DDM-DINO

Salvatore Ferrara 65927A

Abstract—In questo documento viene presentato un modello per la generazione di fissazioni umane su immagini, guidate da frasi descrittive che indirizzano l'attenzione degli osservatori. Questo compito è già svolto dal modello di predizione incrementale `ART` [1]. Il modello qui descritto, chiamato DDM-DINO, è una modifica di `ScanDDM`, un modello *zero-shot* per la predizione generalizzata dell'attenzione guidata da un obiettivo [2].

1 Introduzione

La problematica che questo progetto si prefigge di risolvere è la stessa trattata dal modello predittivo chiamato ART (*Attention in Referral Transformer*). Si tratta di prevedere in modo incrementale l'attenzione umana mentre un individuo osserva un'immagine e ascolta un'espressione di riferimento che definisce l'oggetto nella scena su cui dovrebbe posare lo sguardo. Precisamente, il suo scopo è predire i percorsi di scansione relativi allo sguardo (*scanpath*) [1]. Per marcare meglio il campo di applicazione possiamo fare un esempio e quindi immaginare di avere un presentatore che mostra una diapositiva contenente una fotografia di un ufficio. All'interno dell'immagine sono presenti vari elementi: due scrivanie, due sedie e un computer. Poniamo il caso che durante il discorso il presentatore pronunci la frase: "Osserviamo il computer presente sulla scrivania in legno a sinistra". Subito ogni osservatore, partendo dalla prima parola, inizierebbe a effettuare una serie di fissazioni che, unite, producono uno scanpath. Ecco, il modello proposto in questo documento, così come anche ART, dovrebbero idealmente cercare di generare anch'essi uno scanpath che non sia distinguibile da quello di un osservatore umano. Il modello in questione si chiama DDM-DINO ed è il risultato delle modifiche apportate a ScanDDM (*Scanpath Drift Diffusion Model*), un modello di generazione di scanpath innovativo che consente la predizione *zero-shot* generalizzata dell'attenzione guidata da obiettivi [2]. Un problema secondario era appunto individuare le mancanze di ScanDDM rispetto ad ART e colmarle per riuscire a far svolgere al nuovo modello lo stesso compito svolto da ART.

Importanza del problema: la capacità di dirigere l'attenzione dell'ascoltatore è sfruttata assiduamente nelle interazioni tra esseri umani; questo ci rende capaci di cooperare, allineando le nostre menti verso un contesto comune. Se quindi il nostro obiettivo è quello di rendere l'interazione uomo-macchina sempre più naturale possibile, questa è una tappa fondamentale da attraversare, soprattutto se si vorranno in futuro avere luoghi di lavoro dove ci sia una collaborazione tra umani e macchine. Per fare un breve esempio, basta immaginare un sistema robotico autonomo che coopera con un tecnico addetto alla riparazione di autoveicoli. Per svolgere anche il più semplice dei compiti, come reperire un determinato strumento oppure fornire supporto nella riparazione di un veicolo, la sua attenzione deve poter essere guidata allo stesso modo di quella di un assistente umano. Inoltre, una macchina che ha conoscenza di come l'attenzione umana funzioni può permetterle di guidare a sua volta l'attenzione di un uomo aiutandolo magari a svolgere un compito in cui è inesperto.

Approccio seguito: Inizialmente, è stata effettuata un analisi approfondita del paper di ART e del suo codice, così da comprenderne appieno il suo funzionamento. All'interno del paper vengono descritte chiaramente tutte le sue varie componenti, delle quali è stato visionato il codice. Poi, si è passati a ScanDDM, effettuando su di esso le stesse analisi. Si è potuto quindi constatare che i due modelli eseguivano compiti molto simili, ma utilizzando approcci completamente diversi. Successivamente, sono state prelevate delle immagini dal dataset *RefCOCO-Gaze* [1] utilizzato per addestrare ART, con il loro task testuale e il relativo scanpath ideale. Sono quindi stati effettuati dei test su ScanDDM fornendogli in input le immagini e il task; questo ha permesso di evidenziare quali erano le mancanze di questo modello rispetto ad ART. Nella Fig.1 è presente uno scanpath che è stato generato da ScanDDM avendo in input l'immagine e una descrizione dell'elemento da cercare, in questo caso *"small sheep on right front"*. Osservando il risultato, si può immediatamente notare che il modello non ha individuato l'animale descritto. Questo evidenzia delle criticità nella comprensione del testo e nell'individuazione dell'entità richiesta. Per effettuare questo, ART si serve di un *Trasformer Encoder* multimodale che apprende congiuntamente la previsione dello sguardo e gli obiettivi di *grounding* degli oggetti. Inoltre, integra un *Trasformer Decoder* autoregressivo che sfrutta le fissazioni passate per prevedere meglio quelle successive, corrispondenti a ciascuna parola presentata

S. Ferrara, Corso di Affective Computing e Natural Interaction, A/A 2024-2025, Universitá degli studi di Milano, via Celoria 18, Milano, Italy
E-mail: salvatore.ferrara1@studenti.unimi.it

in sequenza dall'espressione di riferimento [1]. ScanDDM, d'altra parte, si basa su un modello di diffusione della deriva (DDM), che inquadra le dinamiche dello sguardo come processo decisionale che incapsula sia la durata della fissazione che l'esecuzione della saccade. Ciò consente di implementare un processo di accumulo di evidenze basato sul valore, simile ai meccanismi neurobiologici che si presume siano alla base del processo decisionale percettivo. Un processo che però viene influenzato dalla *saliency map* generata in base al prompt testuale, infatti, le aree con maggior salienza accumulano evidenza più velocemente rispetto alle altre, rendendo più probabile che una fissazione venga generata all'interno di quelle aree [2]. La saliency map è il punto di partenza della predizione, la base su cui il modello DDM è eseguito e, dopo approfondite analisi, si è potuto constatare che, alla luce di prompt testuali complessi, la saliency map generata era errata, per esempio, nel caso dell'immagine raffigurante le pecore, la "piccola pecora obiettivo" aveva una salienza trascurabile rispetto alle altre, rendendo pressoché impossibile il fatto che il modello generasse delle fissazioni su di essa. La saliency map veniva generata dal modello *ClipSeg*, che esegue la segmentazione di immagini zero-shot in base a un prompt testuale. Durante i vari test, dove principalmente veniva stampata la saliency map generata, si è appurato che il modello di segmentazione non "comprendeva" le indicazioni direzionali ("a destra", "il più a sinistra") che normalmente vengono usate per aiutare l'osservatore a discriminare l'oggetto indicato dagli altri. Era quindi necessario andare a sostituire ClipSeg con un modello in grado di comprendere frasi complesse, serviva un modello che facesse object-grounding. In DDM-Dino viene utilizzato Grounding-DINO, un *object detector open-set* in grado di rilevare oggetti arbitrari con input umani. Ovviamente, siccome ScanDDM utilizza le saliency map e invece Grounding-DINO fornisce delle box attorno agli oggetti ricercati, è stato necessario adattare l'output fornito trasformandolo in una saliency map, andando a inserire una maschera ellittica all'interno della box dell'oggetto rilevato a cui poi viene applicata un filtro gaussiano. La frase in input viene suddivisa in pezzi, ognuno con una parola nuova, fino ad arrivare alla frase completa. Ogni pezzo viene dato in input a Grounding-DINO; dalla box otteniamo una saliency map che si va a sommare a quelle precedenti e su cui poi verrà di volta in volta generata una fissazione tramite il DDM. In questa maniera, ScanDDM riesce a emulare il comportamento di ART. Successivamente, per avere la certezza che il modello performi correttamente, si è provveduto a effettuare dei test sulle immagini e i prompt del dataset RefCOCO-Gaze, su cui poi sono state calcolate le metriche di similarità tra scanpath umani e quelli generati dal modello.

Contributi Durante la fase iniziale dello sviluppo del progetto, è stato necessario fare numerosi test su ScanDDM per capire a fondo quali erano i suoi limiti. ScanDDM differisce rispetto ad ART per la sua semplicità; è

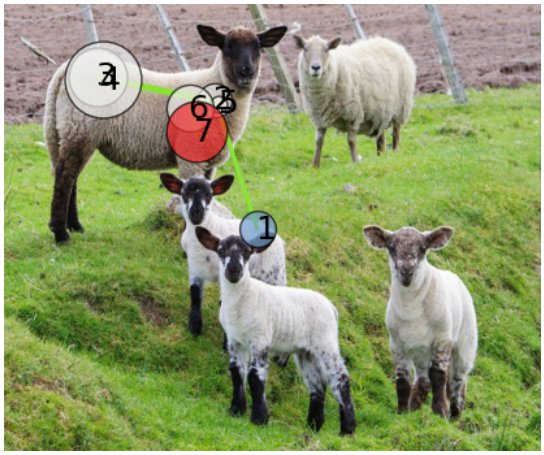

Fig. 1: Scanpath generato da ScanDDM, in input la frase: *"small sheep on right front"*

in grado di unificare elegantemente le dinamiche dello sguardo sotto una singola teoria consolidata del processo decisionale. Al contrario, ART impiega un'architettura più complessa basata su transformer con componenti separati per codificare le informazioni visive e linguistiche e per prevedere le fissazioni. Inoltre, include una fase di pre-formazione con obiettivi di localizzazione degli oggetti e previsione della categoria target, oltre alla principale formazione sulla previsione dello sguardo. Un modello che quindi avesse come base ScanDDM sarebbe stato meno oneroso a livello computazionale e di complessità, ed è questo il motivo per cui ci si è subito concentrati su di esso. Durante le prove sono stati generati output aggiuntivi per avere la conoscenza dello stato del programma. É stata stampata anche la saliency map che non è altro che la *segmentation map* fornita dal modello ClipSeg. Proprio analizzando le rappresentazioni delle saliency map è stato possibile constatare che il "collo di bottiglia" era il modello di segmentazione. Infatti, utilizzava un modello di *image-segmentation pre-trained* che non era stato addestrato ad avere una comprensione intrinseca e spaziale precisa. L'obiettivo da raggiungere nell'addestramento di quel tipo di modelli è apprendere l'associazione tra concetti visivi globali e le loro descrizioni testuali, trascurando quindi le relazioni spaziali. É stato quindi necessario affidarsi ad un altro campo, quello dell'object grounding. Grounding DINO ha permesso una piena comprensione del prompt testuale. Fin dai primi test ci si è accorti che qualsiasi frase in input veniva compresa appieno, soprattutto le indicazioni direzionali; in alcuni test, quando si avevano nell'immagine entità della stessa tipologia, veniva scelta esattamente quella nella posizione descritta senza errori. Inoltre, implementando all'interno del codice solo il modello pre-addestrato, è stato possibile mantenere uno dei punti di forza di ScannDDM, cioè il fatto di essere zero-shot, senza quindi aver bisogno di *Fine-tuning* che potrebbe portare a dell'*overfitting*.

2 ANALISI DELLO STATO DELL'ARTE

Il campo in cui ci muoviamo è la previsione incrementale dell'attenzione visiva, ma siamo in una branca ancora più specifica in quanto l'attenzione visiva è influenzata dallo scopo da conseguire. In letteratura, ART è un contributo significativo per il settore di nicchia di cui si occupa. Si discosta dai molteplici modelli che si concentrano principalmente sulla previsione dello sguardo in scenari di *free-viewing*. Riesce a catturare le complesse dinamiche che legano linguaggio e attenzione visiva, basandosi su un dataset creato appositamente per questo compito. Costituisce una solida base di partenza essendo un modello performante, lo si evince dalla tabella di comparazione che si può trovare all'interno del paper, le metriche calcolate su ART vengono confrontate con le metriche di vari modelli:

- *Chen at al.*: un modello progettato per prevedere sia il comportamento di VQA (*Visual Question Answering*) sia il comportamento di ricerca visiva.
- *Gazeformer-ref*: un modello Trasformer multimodale progettato per prevedere dove un osservatore guarderà durante un compito di ricerca visiva.
- *Gazeformer-cat*: un'altra variante del Gazeformer.
- *OFA (One-For-All)*: modello generale di visione e linguaggio che, come si può evincere da suo nome, può essere utilizzato per svolgere un'ampia varietà di compiti.

Sebbene questi modelli si concentrino su compiti leggermente diversi, ART li supera in capacità di catturare efficacemente le interazioni tra linguaggio e attenzione visiva. Oltre a registrare delle ottime performance, rappresenta uno dei lavori più completi e documentati che questo ambito può fornire.

3 MODELLO TEORICO

DDM-DINO, essendo un miglioramento di ScanDDM ha ereditato alcune sue caratteristiche, per la precisione il fatto di servirsi del *Drift Diffusion Model* le cui componenti verranno formalizzate a livello matematico nella sezione sottostante. La componente secondaria alla base di DDM-DINO è Grounding DINO, di cui tratteremo il processo di *Language-Guided Query Selection*.

3.1 ScanDDM

Nell' Eq.1 è possibile osservare il valore V_p, il *Prior Value* che rappresenta l'importanza di una patch p per la task attuale. Quest'ultimo, viene calcolato come la probabilità che la patch p appartenga a un oggetto d'interesse $\mathbb{O}_\mathbb{G}$ in base all'obbiettivo corrente $\mathbb{G}$. Prima delle modifiche effettuate, i Prior value dipendevano dal modello *ClipSeg*; successivamente, i valori di priorità vengono generati artificiosamente all'interno della box fornita dall'*object detector*.

$$V_p = \rho(p \in \mathbb{O}_\mathbb{G}) \tag{1}$$

In DDM-DINO è rimasto invariato quello che è considerato il nucleo di ScanDDM, cioè il Drift Diffusion Model, responsabile della simulazione del processo decisionale relativo alla successiva fissazione. La variazione dell'evidenza accumulata nel tempo è rappresentata da un'equazione differenziale stocastica che si può osservare nel Eq.2

$$dq(\mathbf{t}) = Id\mathbf{t} + \sigma dW(\mathbf{t}) \tag{2}$$

dove $dq(\mathbf{t})$ rappresenta la variazione infinitesimale nell'accumulo di evidenza (q) in un piccolo intervallo di tempo (dt) al tempo t, $q(t)$ è il quantitativo di evidenza che il modello ha "raccolto" a favore di una determinata posizione nell'immagine come punto di fissazione. Invece, $Id\mathbf{t}$ rappresenta la variazione deterministica in $q(t)$, in cui I è il *drift rate*, cioè la velocità media con cui l'evidenza viene accumulata. Con un I grande avremo un rapido accumulo, mentre se avesse un valore piccolo verrebbe accumulata lentamente. Il suo valore è influenzato dalla *saliency* di una posizione nell'immagine. $\sigma dW(\mathbf{t})$ rappresenta la variazione stocastica in $q(t)$, in cui σ è la deviazione standard del rumore, in parole povere, controlla quanto sia "rumoroso" l'accumulo di evidenza. Per quanto riguarda invece $dW(\mathbf{t})$, si tratta di un processo Weiner, che si occupa di modellare il rumore casuale e che segue una distribuzione normale con media zero e varianza uguale a dt. L'equazione quindi formalizza l'incertezza e la gradualità con cui l'uomo prende decisioni, che si rispecchiano anche nell'attenzione visiva umana.

Nel caso in cui però ci fossero diverse patch da osservare e quindi si avrebbero a disposizione più scelte, viene utilizzato un *Race Model*, con un'equazione differenziale stocastica per ciascuna patch, aspetto formalizzato nell' Eq.3.

$$dq_p(t) = I_p dt + \sigma dW(t), \qquad p = 1, \cdots, N_P. \tag{3}$$

In questo caso però sorge una problematica, in caso si utilizzi come discriminante della scelta "l'attrazione" assoluta. Infatti, se una patch è solo leggermente più interessante delle altre, potrebbe non essere sufficiente per indurre un cambio di sguardo. Calcolando per ogni patch il *Relative Decision Value* (RDV), presente nell' Eq.3, è possibile misurare quanto sia più "attraente" una patch rispetto alla patch più interessante in quel momento.

$$RDV_p(t) = q_p(t) - max[q_k(t)], \qquad \forall k \neq p \tag{4}$$

L'$RDV_p(t)$ è il valore relativo alla patch p al tempo t, $q_p(t)$ è l'evidenza accumulata per la patch p, mentre $max[q_k(t)]$ è l'evidenza accumulata per la patch più interessante in quel momento tra tutte le k patch. Il modello decide di soffermarsi su una patch, quindi porre una fissazione, quando, continuando a calcolare gli RDV, uno di essi supera una certa soglia a.

Nell'Eq.5 viene trattato il *drift rate* (I_p), calcolato per ciascuna patch, dipende dal *prior value* (V_p) della patch e da quello della patch attualmente fissata V_{p^*}.

$$I_p = \eta \cdot \frac{\psi(p, p^*)}{1 + \log_2(1 + V_{p^*})} \tag{5}$$

All'interno dell'equazione η è una costante per il tasso di accumulo di evidenza di base, mentre $\psi(p, p^*)$ è la *gazing function* definita nell' Eq.6.

$$\psi(p, p^*) = V_p \cdot A(p, p^*) \cdot D(p, p^*) \tag{6}$$

Non è altro che un prodotto di tre termini: il primo è il *prior value* della patch p. Il secondo, più complesso, è la funzione di visibilità $A(p, p^*)$. Una distribuzione di Cauchy che assegna un valore più alto alle patch vicine alla patch corrente p^* e più basso a quelle lontane.

$$A(p, p^*) = \frac{1}{\pi}\left[\frac{\gamma}{(x - x_0)^2 + \gamma^2}\right] \tag{7}$$

All'interno dell'Eq.7 $(x - x_0)^2$ rappresenta la distanza tra le patch, mentre γ è un parametro che controlla quanto velocemente diminuisce la visibilità al variare della distanza. La terza componente della gazing funtion è la funzione che modella la direzione ($D(p, p^*)$); si basa su fatto che l'uomo tende a fare saccadi più spesso in orizzontale che in verticale. Ha la seguente forma:

$$D(p, p^*) = \frac{\cos(\alpha \cdot \theta(p, p^*)) + 1)}{2} \tag{8}$$

Nell'equazione è presente il $\theta(p, p^*)$ e rappresenta l'angolo tra la patch p e la patch corrente p^*. Il valore di a controlla la forza della tendenza ad attuare questo comportamento. Per quanto riguarda il coseno, è utilizzato per la sua particolarità di raggiungere il suo valore massimo quando l'angolo è di 0 gradi (orizzontale a destra) o 180 (orizzontale a sinistra) e minimo quando l'angolo è di 90 gradi.

3.2 Grounding DINO

L'Eq.9 è un modulo chiave di Grounding DINO, serve a "concentrare l'attenzione" del modello sulle parti più rilevanti dell'immagine in base a ciò che viene descritto nel testo in input. Questo perché solitamente le immagini contengono molte informazioni, e non tutte sono rilevanti per una specifica richiesta. Il fatto di concentrarsi solo sulle parti rilevanti rende il processo di rilevamento oggetti più efficiente. Nell' Eq.9 si può osservare com'è formalizzato questo processo.

$$\mathbf{I}_{Nq} = \text{Top}N_q(\text{Max}^{(-1)}(\mathbf{X}_I\mathbf{X}_T^{\intercal})) \tag{9}$$

Dove con $(\mathbf{X}_I\mathbf{X}_T^{\intercal})$ otteniamo la matrice di affinità, che rappresenta quanto siano "simili" le diverse parti dell'immagine alle diverse parti di testo, se un area ha un valore alto allora possiamo affermare che una certa parte dell'immagine è molto rilevante per una certa parte di testo. Attraverso $(\text{Max}^{(-1)}(...))$ si trova la massima affinità per ogni parte dell'immagine; un numero predefinito di parti dell'immagine (dette *query*) vengono selezionate, attraverso l'ottenimento degli indici $((\text{Top}_{N_q}(...)))$. Successivamente, gli indici ottenuti saranno utilizzati per inizializzare il *decoder* del modello.

4 Simulazione e esperimenti

Dopo la fase di sviluppo, DDM-DINO è stato sottoposto a vari test, utilizzando come input le stesse immagini con i relativi prompt che erano stati riportati nel paper di ART. Questo per cercare di avere un riscontro visivo, visto che nel paper sono ben documentati. Come già affermato in precedenza, la *saliency map* generata da ScanDDM, nella maggior parte dei prompt di ART risultava essere errata, non tenendo minimamente in considerazione dell'entità obiettivo. Come si può notare, invece, osservando la Fig.2, l'area di interesse evidenziata è esattamente sovrapponibile alla pecora descritta nel prompt. Con una saliency map corretta, il DDM performa secondo quanto atteso; facendo un confronto con la Fig.1 mostrata nell'introduzione è possibile notare gli enormi passi avanti compiuti, con una mappa di salienza corretta. Nella Fig.4 è mostrato lo *scanpath* generato da ART, quello che idealmente doveva essere conseguito. Si può osservare come entrambi i due scanpath abbiano le stesse due fasi, una fase dove il modello dispone ancora di poche informazioni ed è quindi completamente fuori obiettivo ("sono state udite solo le prime parole della frase"), per passare poi all'intuizione dell'obiettivo richiesto, come si può evincere dalle fissazioni molto ravvicinate sulla "pecora bersaglio".

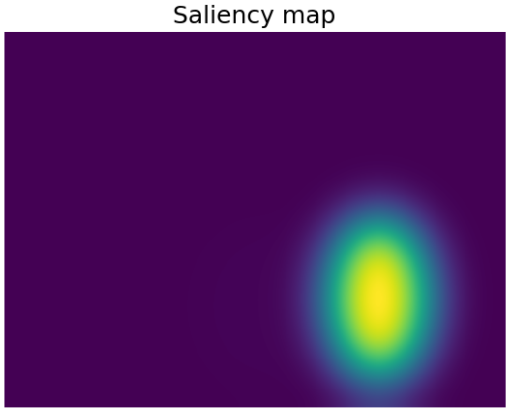

Fig. 2: Saliency map generata da DDM-DINO, in input la frase: *"small sheep on right front"*

4.1 Dataset

RefCOCO-Gaze anche se indirettamente, ha fornito un apporto cruciale per la riuscita di questo progetto. Indirettamente, in quanto DDM-DINO non necessita di addestramento. Però, RefCOCO-Gaze ha fornito le immagini per poter svolgere i vari test, ha consentito anche di calcolare le metriche che hanno permesso di avere una stima della qualità delle predizioni generate. RefCOCO-Gaze consiste in un dataset di discrete dimensioni creato appositamente per studiare il comportamento dello sguardo umano nel compito di riferimento incrementale degli oggetti. É stato prodotto catturando i movimenti

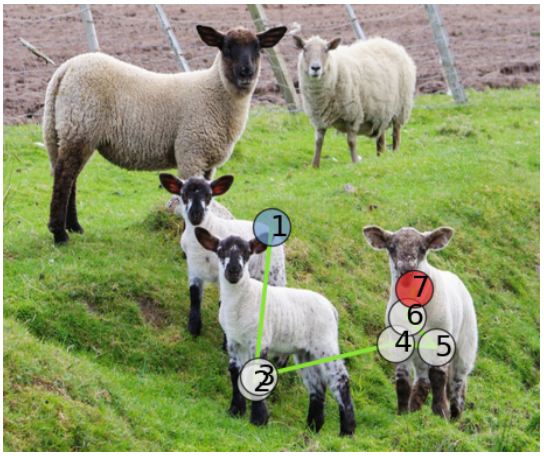

Fig. 3: Scanpath generato da DDM-DINO, in input la frase: *"small sheep on right front"*

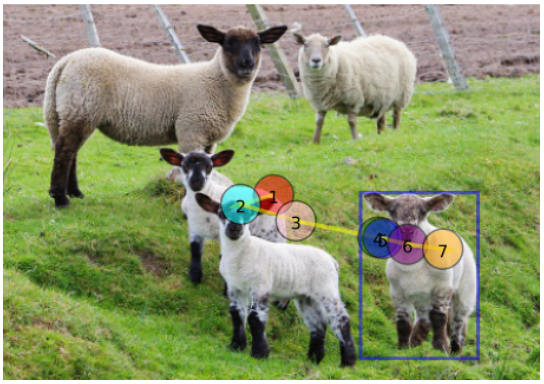

Fig. 4: Scanpath generato da ART, in input la frase: *"small sheep on right front"*

oculari di una serie di osservatori umani, mentre visionavano un'immagine ascoltando delle espressioni di riferimento che descrivevano gli oggetti target. Parlando di numeri, questo dataset contiene 19.738 *scanpath* umani. Raggruppati in 2094 coppie immagine-espressione. Sono dati che sono stati prodotti da 220 partecipanti. Per la sua completezza, RefCOCO-Gaze costituisce uno dei contributi più significativi nel suo settore di ricerca, ha infatti permesso lo sviluppo di modelli innovativi come ART.

4.2 Architettura del sistema

L'architettura di DDM-DINO è rappresentata all'interno nella Fig.5. Come si può osservare, la struttura riceve in input un Text prompt e una Query image. Il text prompt scandisce l'iterazione dell'intero blocco, in quanto il testo viene frammentato in pezzi con numero di parole incrementali, questo a simulare il processo di ascolto dell'osservatore umano, che sente gradualmente la frase. Successivamente, l'input testuale attuale e l'immagine vengono passati a Grounding-DINO, per la precisione, inizialmente vengono inseriti rispettivamente nella *Text Backbone* e l'*Image Backbone* che si occupano di estrarre le caratteristiche inserite e affinate dal *Feature Enhancer*. Le features verranno poi filtrate in base al testo, per poi combinare informazioni visive e testuali per produrre delle previsioni. Le previsioni effettuate consistono nella creazione di una box attorno alla componente dell'immagine descritta dal testo attuale, dove verrà effettuato il *post-processing* che restituirà in output una saliency map comprensibile per il *Multialternative DDM*. Dopo "l'ascolto" della frase completa verrà fornito in output lo *scanpath*

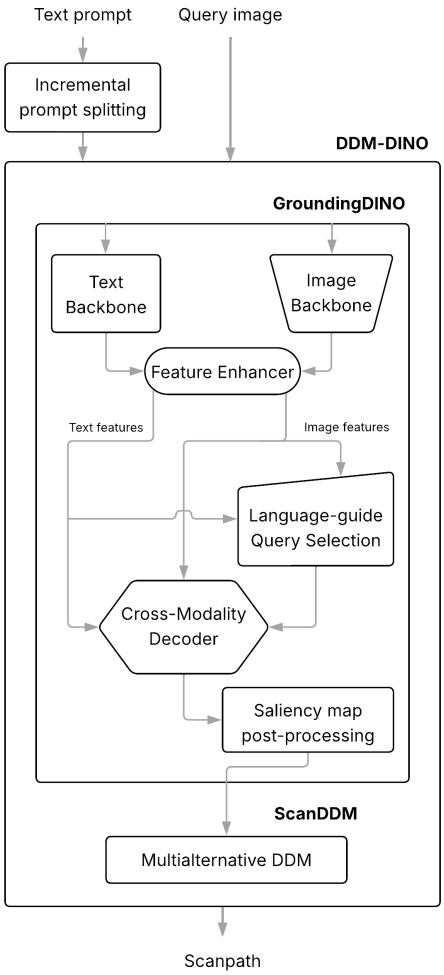

Fig. 5: Architettura funzionale del sistema

4.3 Dettagli implementativi

Durante i numerosi esperimenti effettuati, dovendo elaborare matrici di discrete dimensioni, si è sviluppata la consapevolezza dei limiti computazionali imposti dall'utilizzo esclusivo della CPU. Questo a causa del fatto che le CPU sono progettate per eseguire un numero relativamente piccolo di task complessi in modo sequenziale o in alternativa in parallelo, ma con un esiguo numero di core. Al contrario, le GPU hanno un'architettura parallela composta da migliaia di core più piccoli e meno potenti, progettati specificatamente per eseguire la stessa istruzione su grandi quantità di dati contemporaneamente. L'utilizzo della scheda grafica

è il metodo migliore per operare su delle matrici. Utilizzando *CUDA*, è possibile servirsi dela propria GPU *NVIDIA* per l'esecuzione di calcoli generici, velocizzando notevolmente i tempi di esecuzione. Tramite la riga di codice sottostante è possibile accertarsi della presenza della piattaforma CUDA, altrimenti verrà utilizzata semplicemente la CPU.

```
device = "cuda" if torch.cuda.is_available() else "cpu"
```

5 Risultati ottenuti

Nella Tab.1 sono presentate le metriche calcolate (prelevate da FixaTons [3]) con i relativi risultati ottenuti. Le metriche sono state calcolate tramite il confronto con circa 920 scanpath umani suddivisi in 92 immagini correlate da un prompt testuale che sono state visionate da 10 osservatori umani. I valori riportati sono la media di tutte le misurazioni effettuate per ogni scanpath. Le metriche riportate sono:

- *Euclidean Distance*: Misura la distanza spaziale media tra le fissazioni corrispondenti nei due scanpath. Per poter effettuare il calcolo, è necessario che i due scanpath abbiano uguale lunghezza (la stessa quantità di fissazioni), per questo motivo in alcuni casi è stato necessario diminuire la lunghezza, in presenza di differenze, dello scanpath più lungo utilizzando il *Dynamic Time Wrapping* (DTW). Minore è il valore relativo all'Euclidean distance, maggiore è la similarità tra scanpath umano e simulato.
- *String Edit Distance*: Anche detta *Levenshtein Distance*, viene calcolata discretizzando gli scanpath in sequenze di celle su una griglia. Il valore prodotto rappresenta il numero minimo di operazioni di editing (inserimento, eliminazione o sostituzione) necessarie per trasformare la sequenza di celle dello scanpath simulato nella squenza di celle dello scanpath umano. Un valore basso indica una maggiore somiglianza nell'ordine e nella posizione delle regioni fissate.
- *ScanMatch*: Metrica che confronta gli scanpath in base alla sovrapposizione spaziale delle fissazioni in finestre temporali consecutive. Per ogni fissazione nello scanpath più corto, si cerca la fissazione più vicina nello scanpath più lungo entro una certa finestra temporale. Il valore di similarità è calcolato in base al numero di match trovati, un valore alto indica una maggiore sovrapposizione spaziale delle fissazioni.
- *Sequence Score(SS)*: utilizza il *clustering* delle fissazioni umane per una data immagine, che vengono poi raggruppate utilizzando l'algoritmo di *Mean-Shift*. Il quale permette di identificare i cluster di fissazioni salienti per l'immagine. Ogni scanpath umano viene convertito in una sequenza di *cluster ID*, secondo l'appartenenza di ciascuna fissazione a un cluster. La stessa cosa è fatta con quello simulato utilizzando lo stesso modello. Avendo ora le due sequenze, è possibile confrontarle, osservando se lo scanpath simulato visita gli stessi cluster (anche nello stesso ordine) degli scanpath umani. Un valore alto rappresenta una maggiore somiglianza delle sequenze.

TABLE 1: Metriche

	ED ↓	SED ↓	SM ↑	SS ↑
ScanDDM	0.185	8.804	0.325	0.432
DDM-DINO	0.155	8.635	0.365	0.539
ART	0.102	8.791	0.459	0.585

Nella Tab.1 è possibile constatare che DDM-DINO abbia ottenuto un risultato intermedio tra le performance di partenza e quelle obiettivo.

6 Commenti conclusivi

Osservando le metriche prodotte dal modello sviluppato e confrontandole con ScamDDM, è possibile notare che un miglioramento c'è stato, ma guardando invece i risultati registrati da ART, si può constatare che non è stato abbastanza. Un aspetto, in parte, giustificabile, visto che i test sono stati effettuati proprio utilizzando come input i dati su cui ART è stato addestrato. Questo è dovuto al fatto che probabilmente non esista un altro dataset simile a quello usato, a causa della specificità dei compiti svolti. Uno scopo che però è stato conseguito con successo, è il mantenimento di un codice semplice è che non richieda addestramento; infatti, DDM-DINO dopo l'installazione delle librerie può essere subito eseguito senza bisogno di alcun addestramento. Quello che però spaventa è la sua onerosità a livello computazionale, infatti per produrre uno scanpath è necessario attendere un periodo che abbastanza ambio rispetto agli standard a cui si è abituati. Ma, ovviamente, i processi che il modello svolge sono onerosi e vengono eseguiti su variabili di discrete dimensioni. In conclusione, leggendo questo documento è osservandone i risultati, dalle immagini alle metriche, è possibile constatare che sono stati fatti dei passi in avanti rispetto alla situazione di partenza. Sviluppando DDM-DINO si è acquisita consapevolezza del fatto che per progredire significativamente in questo campo è necessario lo sviluppo di modelli con complessità maggiore e onerosi a livello computazionale, anche perché si sta cercando di emulare i meccanismi biologici del cervello, un organo di una complessità sbalorditiva, capace di imprese che ancora oggi lasciano sconcertati.

References

[1] S. Mondal, S. Ahn, Z. Yang, N. Balasubramanian, D. Samaras, G. Zelinsky, and M. Hoai, "Look hear: Gaze prediction for speech-directed human attention," in *European Conference on Computer Vision*. Springer, 2024, pp. 236–255.

[2] A. D'Amelio, M. Lucchi, and G. Boccignone, "ScanDDM: Generalised Zero-Shot Neuro-Dynamical Modelling of Goal-Directed Attention," in *Proceedings of the European Conference on Computer Vision*, 2024.

[3] D. Zanca, V. Serchi, P. Piu, F. Rosini, and A. Rufa, "Fixatons: A collection of human fixations datasets and metrics for scanpath similarity," 2018. [Online]. Available: https://arxiv.org/abs/1802.02534