ModelRAIL: navigazione autonoma di UAV commerciali per il monitoraggio ferroviario

Andrea Maria Lingua[1], Filiberto Chiabrando[2], Edward Borgogno[2], Chiara Graziani[1]

[1] Politecnico di Torino, Dipartimento di Ingegneria per l'Ambiente, il Territorio e le Infrastrutture, Corso Duca degli Abruzzi 24, 10129 Torino, Italia
andrea.lingua@polito.it, chiara.graziani@polito.it
[2] Politecnico di Torino, Dipartimento di Architettura e Design, Viale Pier Andrea Mattioli 39, 10125 Torino, Italia
filiberto.chiabrando@polito.it, edward.borgogno@polito.it

Abstract. Per il monitoraggio ferroviario con UAV, ModelRAIL integra segmentazione delle rotaie e navigazione visiva. Un dataset annotato in CVAT permette di confrontare YOLO11n-seg e YOLO26n-seg. Dalle maschere si ricavano la mezzeria e i riferimenti di guida, simulati fuori linea mediante telemetria GNSS e un modello cinematico semplice. La prova sul radiocomando, con la camera del drone rivolta a un video preregistrato, documenta il riconoscimento e prepara la validazione in campo.

Parole chiave: Monitoraggio ferroviario, UAV, segmentazione di istanza, navigazione visiva, telemetria GNSS.

1 Introduzione

Per descrivere lo stato di una linea ferroviaria occorre rilevare con continuità il binario e lo spazio che lo circonda. Accanto alle rotaie interessano la massicciata, la vegetazione, le opere e i versanti, le cui variazioni acquistano significato solo se le acquisizioni sono confrontabili nel tempo. La fotogrammetria da Uncrewed Aerial Vehicle (UAV) permette di osservare tratti estesi, ma l'utilità metrica dei prodotti dipende dalla geometria di presa e dalla qualità della georeferenziazione [1]. Cuypers et al. [2] hanno esaminato l'estrazione planimetrica delle rotaie da immagini aeree; Ghassoun et al. [3] hanno verificato un procedimento fotogrammetrico sui binari di una gru portuale. In entrambi i casi, al riconoscimento dell'elemento si affianca il problema della sua localizzazione affidabile.

ModelRAIL affronta questi problemi nel quadro di un gemello digitale dell'infrastruttura, alimentato da rilievi ripetuti e modelli tridimensionali. Affinchè le immagini raccolte siano efficaci lungo la tratta, il drone deve mantenere il binario nel campo di vista anche dove il tracciato curva o le rotaie sono parzialmente coperte. Ombre, prospettiva e oggetti lineari estranei possono rendere instabile la segmentazione [4, 5]. La questione di ricerca riguarda quindi il passaggio dalla maschera di una rotaia, valutata su un singolo fotogramma, a una mezzeria coerente nel tempo e utilizzabile

per la guida. Il controllo visivo fornisce l'impostazione metodologica [6]; la sua applicazione richiede però che gli errori percettivi siano riconosciuti prima di trasformarsi in comandi.

Il lavoro descrive la catena sperimentale, dalla scelta della configurazione DJI Mini 3 Pro–DJI RC Pro alla segmentazione delle rotaie, alla stima della mezzeria e alla generazione dei riferimenti di guida in modalità virtual stick. I comandi sono propagati fuori linea con un modello di moto semplificato, inizializzato con telemetria GNSS preregistrata.

2 Progetto preliminare del sistema

Nel progetto preliminare viene scelto il drone DJI Mini 3 Pro che acquisisce immagini e telemetria, mentre il DJI RC Pro ospita l'ambiente applicativo previsto per Mobile SDK v5. L'abbinamento fra i due dispositivi è riportato nella documentazione DJI [7]. La catena è articolata in acquisizione, segmentazione, ricostruzione della geometria e calcolo dei riferimenti di guida (Fig. 1); stabilizzazione e attuazione restano affidate al controllore di volo. Le elaborazioni descritte nelle sezioni seguenti sono state sviluppate su dati registrati. La schermata acquisita sul radiocomando documenta una prova della parte percettiva, distinta dall'impiego dei comandi per muovere il velivolo.

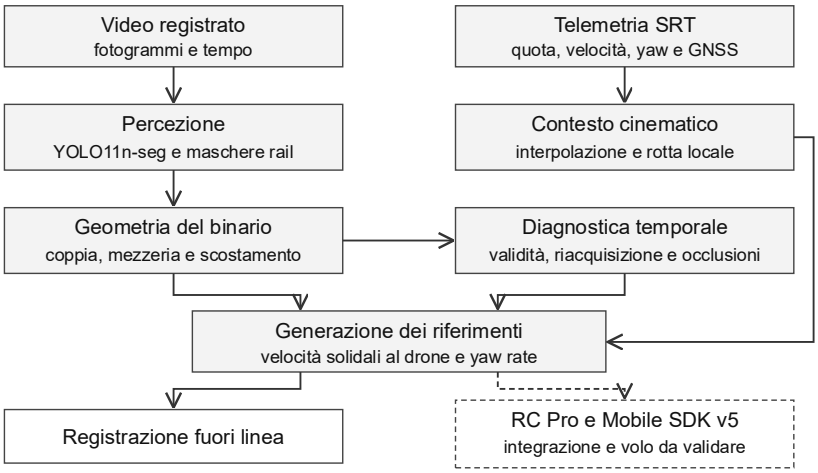

Fig. 1. Architettura ModelRAIL. Il ramo continuo rappresenta la generazione dei riferimenti fuori linea; il collegamento tratteggiato individua l'integrazione con Mobile SDK v5 da verificare per l'attuazione.

I riferimenti calcolati dal prototipo sono espressi secondo le convenzioni della modalità virtual stick dell'SDK [7]. Prima dell'attuazione occorrerà verificare sul radiocomando segni, sistemi di riferimento, cadenza dei comandi e gestione delle interruzioni del riconoscimento. La compatibilità con l'SDK non implica la disponibilità di tutte le funzioni di volo: nell'elenco DJI dei velivoli con evitamento degli ostacoli in modalità

virtual stick avanzata non compare il Mini 3 Pro [7], l'architettura non attribuisce pertanto tale funzione alla configurazione studiata. Si progetta un volo che stia entro la sagoma del treno, sotto i cavi elettrici a circa 2-3 m di altezza dai binari.

3 Annotazione e produzione del dataset

Per proporre un algoritmo di riconoscimento automatico dei binari, si definisce un dataset composto da 5 sequenze con punti di vista differenti su 2 tratte differenti. I fotogrammi obliqui dei 4 video indicati in tabella 1 sono destinati all'addestramento; quelli più nadirali di DJI_0150.MP4 alla validazione, per valutare il modello su una sequenza indipendente da quelle utilizzate per l'addestramento e con una geometria differente.
Il campionamento procede con un passo di tre fotogrammi e produce i conteggi riassunti nella Tabella 1. Si usano nel complesso oltre 2200 fotogrammi per l'addestramento e 241 per la validazione.

Tabella 1. Composizione del dataset secondo i conteggi della campagna di annotazione.

Caratteristica	Addestramento	Test
Video	DJI_0070.MP4 DJI_20250325175730_0002_W.MP4 DJI_20231111163709_0001_W.MP4 DJI_20240611114108_0445_S.MP4	DJI_0150.MP4
Vista	Obliqua	Nadirale
Intervallo dei frame	300–1500 390-1800 4000-7200 90-3450	30–750
Passo di campionamento	3 frame	3 frame
Frame dichiarati	400 451 718 1104	241
Annotazioni manuali	100 129 118 109	70
Altre annotazioni Interpolate	300 322 600 996	171

L'annotazione descrive la singola rotaia con la classe rail, senza distinguere sinistra e destra. In CVAT sono state tracciate manualmente polilinee su fotogrammi chiave di ciascuna sequenza, utilizzando la modalità Track per interpolare le annotazioni nei fotogrammi intermedi ([8]). Le annotazioni così ottenute sono state verificate visivamente.

4 Addestramento e verifica dei segmentatori

4.1 Impostazione del confronto

L'esplorazione iniziale ha incluso Rail-Net, che localizza le rotaie lungo righe dell'immagine [5], e un procedimento classico fondato sulla ricerca di minimi e sulla stima robusta delle linee. Rail-Net, applicato direttamente alle riprese oblique disponibili, non ha restituito una geometria utilizzabile con continuità. Il confronto principale riguarda perciò due reti di segmentazione di istanza in configurazione nano per potenziale compatibilità con le prestazioni del RC Pro, YOLO11n-seg [9] e YOLO26n-seg [10], entrambe adattate alla sola classe *rail*. Ciascuna fornisce una maschera e una confidenza per ogni rotaia candidata. Per entrambi gli addestramenti la migliore epoca riportata è la 84.

Tabella 2. Confronto delle metriche delle maschere e dei tempi per immagine. F1 è ricalcolato da precisione e richiamo; i tempi derivano da due misure per modello.

Indicatore	YOLO11n-seg	YOLO26n-seg
Migliore epoca	84	84
Precisione maschere	0,963	0,949
Richiamo maschere	0,971	0,965
F1 ricalcolato	0,967	0,957
mAP50 riportata	0,945	0,957
mAP50–95 riportata	0,351	0,903
Tempo totale (ms)	19,0–19,2	19,9–20,9
Immagini/s teoriche	52,1–52,6	47,8–50,3

La Tabella 2 riporta precisione e richiamo delle maschere elevati per entrambe le reti. Le mAP indicate sono maggiori per YOLO26n-seg, soprattutto nell'intervallo di soglie IoU 0,50–0,95. Nella scena illustrata in Fig. 2, tuttavia, YOLO26n-seg individua una rotaia, mentre YOLO11n-seg ne individua due: il caso rende visibile un errore rilevante per la mezzeria, senza dire quanto spesso si verifichi.

4.2 Prestazioni temporali e scelta

Sull'ingresso di 544 × 960 pixel i tempi complessivi di preparazione, inferenza e post-elaborazione sono 19,0–19,2 ms per YOLO11n-seg e 19,9–20,9 ms per YOLO26n-seg. I reciproci corrispondono a circa 52 e 48–50 immagini al secondo, realizzate su DELL Precision 5490 con CPU Ultra 9 185 H 2500 MHz, 64 GB RAM, NVIDIA RTX 3000 ADA 8GB dedicati con CUDA driver. YOLO11n-seg è stato adottato nel prototipo per i tempi di elaborazione leggermente inferiori e perché, nelle prove condotte, YOLO26n-seg individuava talvolta una sola delle due rotaie, impedendo la stima diretta della mezzeria.

La Fig. 2 mostra un esempio in cui YOLO11n-seg individua entrambe le rotaie necessarie alla navigazione (immagine a sinistra), mentre YOLO26n-seg non rileva la rotaia sinistra del binario (immagine a destra). La scelta riguarda questo impiego: le mAP riportate suggeriscono un diverso ordinamento e la continuità della mezzeria dovrà essere quantificata su sequenze indipendenti.

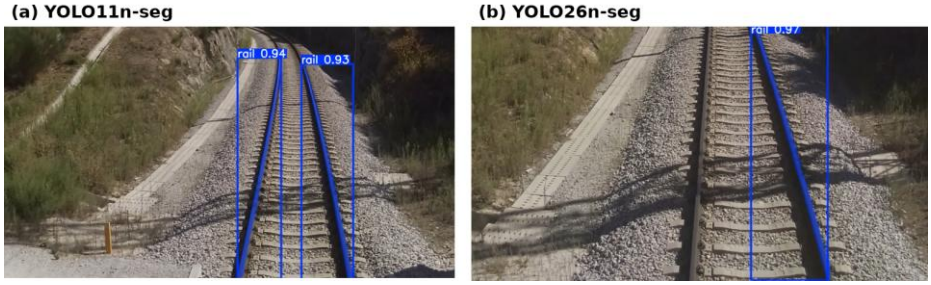

Fig. 2. Esempi di segmentazione della classe rail nello stesso contesto ferroviario: (a) YOLO11n-seg individua due rotaie; (b) YOLO26n-seg ne individua una.

5 Ricostruzione del binario e navigazione

5.1 Selezione delle rotaie e stima della mezzeria

Nelle prove fuori linea YOLO11n-seg riaddestrato viene utilizzato per segmentare i video con una soglia di confidenza di 0,60. La ricerca è concentrata nel 70% inferiore dell'immagine, dove le rotaie sono più vicine alla camera. Area, estensione verticale e posizione delle maschere escludono i candidati troppo piccoli o lontani dalla zona utile; la distanza fra i due assi e la concordanza con il fotogramma precedente orientano poi la scelta della coppia. Questi vincoli riducono il rischio di seguire un elemento lineare estraneo al binario. La qualifica di rotaia sinistra o destra compare soltanto a questo punto, ordinando le due curve secondo la loro posizione nella parte bassa del fotogramma. Quando un tratto è occluso, la geometria precedente può sostenere una previsione breve, esplicitamente distinta da una nuova osservazione.

Per ogni ordinata y comune alle due maschere, siano $x_L(y)$ e $x_R(y)$ le ascisse dei rispettivi assi. La loro media definisce la mezzeria nell'immagine:

$$x_c(y) = \frac{x_L(y) + x_R(y)}{2} \quad (1)$$

Si considerino una sezione y_n vicina al bordo inferiore e una sezione intermedia y_m. Lo scostamento rispetto al centro del fotogramma di larghezza W è $epx = x_c(y_n) - W/2$; la variazione della mezzeria fra le due sezioni fornisce l'orientamento apparente $\theta_{img} = atan2[x_c(y_n) - x_c(y_m), y_n - y_m]$. Quest'ultimo è un angolo nel piano immagine, non l'azimut del binario. Assumendo per il prototipo una distanza d = 1,505 m fra gli assi delle teste delle rotaie, la separazione apparente $g(y_n)$ dà una scala locale $e_m \approx epx\ d/g(y_n)$. La relazione è utile per il riferimento di guida, ma risente della prospettiva, dell'assetto della camera e della definizione delle maschere: non va letta come una misura metrica validata.

5.2 Riferimenti virtual stick e modello di moto

Dallo scostamento laterale e dall'orientamento apparente, integrati con la velocità disponibile e con lo stato della soluzione, il prototipo ricava i segnali normalizzati ROLL, PITCH, YAW e THR. Un controllo proporzionale filtrato li trasforma in riferimenti di velocità laterale, longitudinale e verticale e in una velocità d'imbardata, secondo le convenzioni dell'SDK [7]. Il filtraggio e i limiti di variazione attenuano i salti fra fotogrammi. Se la coppia di rotaie perde affidabilità, il modulo diagnostico evita di trattare la previsione temporale come una nuova misura; guadagni e durata ammissibile della previsione dovranno essere tarati sulla risposta effettiva del velivolo.

La propagazione dei riferimenti è descritta con un modello cinematico discreto. Si indichino con v_f e v_r le velocità comandate in avanti e verso destra nel sistema solidale al drone, con v_u la velocità verticale, con ψ l'azimut misurato in senso orario dal Nord e con ω la velocità d'imbardata. In coordinate locali Est–Nord, al passo Δt valgono:

$$\begin{aligned} E_{k+1} &= E_k + \Delta t(vf \sin \psi_k + vr \cos \psi_k) \\ N_{k+1} &= N_k + \Delta t(vf \cos \psi_k - vr \sin \psi_k) \\ h_{k+1} &= h_k + \Delta t\ vu\ e\ \psi_{k+1} = \psi_k + \Delta t\ \omega. \end{aligned} \quad (2)$$

Il modello integra i set-point associati agli stick simulati; non riproduce inerzia, vento o risposta del controllore di bordo.

6 Simulazione fuori linea e dimostrazione sul radiocomando

La riproduzione fuori linea associa i fotogrammi di DJI_0070.MP4 ai dati del file DJI_0070.srt che contengono i parametri di posizione e assetto angolare del drone con una frequenza pari a 1 Hz. Le grandezze telemetriche vengono interpolate ai tempi dei fotogrammi: posizione GNSS, quota, velocità e imbardata servono a inizializzare e contestualizzare il moto simulato. L'interpolazione non migliora la risoluzione temporale delle osservazioni originarie. Anche rotta e imbardata hanno significati diversi: la prima deriva dallo spostamento tra posizioni successive, la seconda dall'orientamento del drone. La traiettoria GNSS registrata descrive il volo da cui proviene il video, non una misura indipendente dell'errore della guida simulata. Visto

che i comandi calcolati non modificano le immagini già acquisite, questa prova verifica la catena di elaborazione senza chiudere l'anello fra moto e osservazione.



Fig. 3. Tracking fuori linea: rotaie e mezzeria su un fotogramma registrato. Stato OK; scostamento −83 pixel, corrispondenti a −0,36 m nella stima locale. I campi SRT e il tempo di elaborazione non sono disponibili in questo fotogramma.

Una verifica preliminare della percezione è stata svolta sul radiocomando implementando il modello YOLO11n-seg riaddestrato con la funzione di predizione in ambiente Android mediante Android Studio. Quindi, la funzionalità è stata verificata dirigendo la camera del Mini 3 Pro verso uno schermo che riproduce il video ferroviario preregistrato. Nell'interfaccia compaiono due rilevamenti rail (Fig. 4). La schermata documenta il riconoscimento in tale configurazione ottica; la latenza visualizzata, pari a 689 ms, si riferisce a un singolo istante e l'indicatore di frequenza mostra 0,0. Non se ne può ricavare una frequenza di esercizio. Il passaggio alla guida sul campo richiederà la misura continuativa dei tempi, la verifica dell'invio dei riferimenti virtual stick e la valutazione della traiettoria ottenuta.

7 Conclusioni

La sperimentazione ha collegato la segmentazione delle rotaie alla selezione geometrica della coppia, alla stima della mezzeria e alla simulazione dei riferimenti di guida. YOLO11n-seg è stato impiegato nel prototipo per la capacità di restituire entrambe le rotaie nella scena esaminata e per i tempi misurati leggermente inferiori Rispetto a Yolo26n-seg; La prova sul radiocomando aggiunge una verifica del riconoscimento sul flusso ottico. Le successive misure dell'intera catena e dell'errore di traiettoria sul

campo permetteranno di valutare se la guida visiva fornisca acquisizioni abbastanza regolari per il monitoraggio ripetuto della linea.

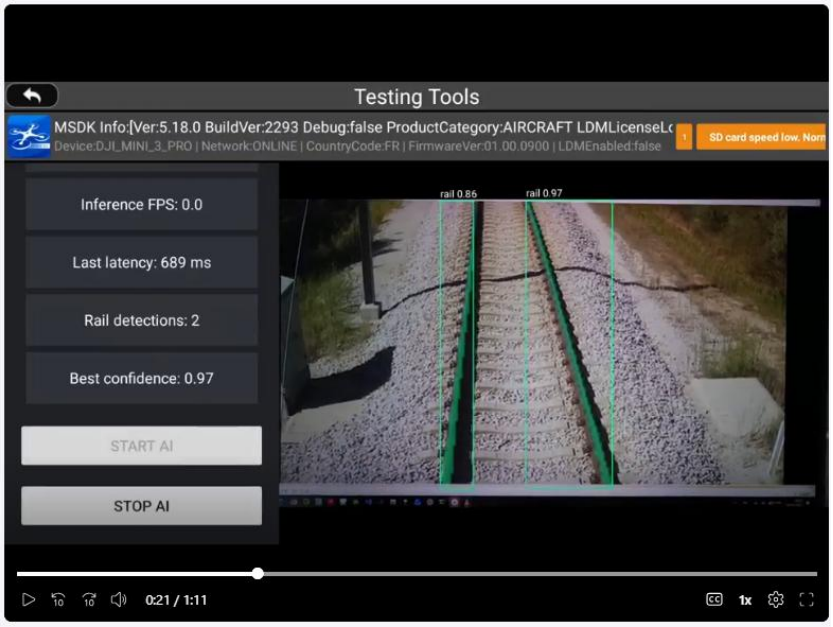

Fig. 4. Prova di riconoscimento sul radiocomando. La camera del drone riprende un video preregistrato riprodotto su schermo; l'interfaccia mostra due rilevamenti della classe rail. Immagine fornita dagli autori.

Ringraziamenti

Lo studio è stato finanziato dalla Regione Piemonte nell'ambito dei progetti SWITCH. Gli autori ringraziano Zirak, partner di progetto, e i colleghi del gruppo di Rappresentazione del DISEG per la collaborazione.

Riferimenti bibliografici

1. Colomina, I., Molina, P.: Unmanned aerial systems for photogrammetry and remote sensing: A review. ISPRS Journal of Photogrammetry and Remote Sensing 92, 79–97 (2014). https://doi.org/10.1016/j.isprsjprs.2014.02.013
2. Cuypers, S., De Winter, H., Bassier, M., Vergauwen, M.: Planimetric rail positioning using UAV photogrammetry: Towards automated and safe railway infrastructure monitoring. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLVIII-1/W2-2023, 589–596 (2023). https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-589-2023

3. Ghassoun, Y., Gerke, M., Khedar, Y., Backhaus, J., Bobbe, M., Meissner, H., Tiwary, P.K., Heyen, R.: Implementation and Validation of a High Accuracy UAV-Photogrammetry Based Rail Track Inspection System. Remote Sensing 13, 384 (2021). https://doi.org/10.3390/rs13030384
4. Yang, H., Li, X., Guo, Y., Jia, L.: RT-GAN: GAN Based Architecture for Precise Segmentation of Railway Tracks. Applied Sciences 12, 12044 (2022). https://doi.org/10.3390/app122312044
5. Li, X., Peng, X.: Rail Detection: An Efficient Row-based Network and A New Benchmark. Proc. 30th ACM International Conference on Multimedia (MM '22) (2022). https://doi.org/10.1145/3503161.3548050
6. Chaumette, F., Hutchinson, S.: Visual servo control. I. Basic approaches. IEEE Robotics & Automation Magazine 13(4), 82–90 (2006). https://doi.org/10.1109/MRA.2006.250573
7. DJI: Mobile SDK v5, Supported Product List e IVirtualStickManager. Documentazione tecnica ufficiale. https://developer.dji.com/doc/mobile-sdk-tutorial/en/ ; https://developer.dji.com/api-reference-v5/android-api/Components/IVirtualStickManager/IVirtualStickManager.html (consultati il 14/10/2026)
8. Ravi, N., et al.: SAM 2: Segment Anything in Images and Videos. arXiv:2408.00714 (2024). https://doi.org/10.48550/arXiv.2408.00714
9. Jocher, G., Qiu, J.: Ultralytics YOLO11, software, versione iniziale 11.0.0 (2024). https://github.com/ultralytics/ultralytics
10. Jocher, G., Qiu, J., Liu, M., Lyu, S., Akyon, F.C., Kalfaoglu, M.E.: Ultralytics YOLO26: Unified Real-Time End-to-End Vision Models. arXiv:2606.03748 (2026). https://doi.org/10.48550/arXiv.2606.03748