GESTURE RECOGNITION PER ROBOTICA COLLABORATIVA: PRIMO APPROCCIO

C. Nuzzi[(1)], S. Pasinetti[(1)], M. Lancini[(1)], F. Docchio[(1)], G. Sansoni[(1)]
[(1)]Dip. di Ingegneria Meccanica e Industriale, Università degli Studi di Brescia
mail autore di riferimento: c.nuzzi@unibs.it

1. INTRODUZIONE

Con il nuovo paradigma di Industria 4.0 si introducono i robot collaborativi, che condividono l'area di lavoro con l'operatore. Risulta necessario non solo elaborare adeguate strategie per assicurare la sicurezza degli operatori, ma anche metodi efficaci per comunicare con i robot collaborativi in modo naturale, tramite comandi vocali o gesti [1].

Come primo approccio al problema della comunicazione umano-robot si è adottato un sistema di riconoscimento gesti basato su un algoritmo di Deep Learning, sviluppato sulla piattaforma **MATLAB 2017b**, in grado di riconoscere quattro diversi tipi di gesto a partire da immagini RGB, come riportato in Fig. 1.

I gesti proposti sono caratterizzati da tre condizioni: devono essere eseguiti usando entrambe le mani con la sinistra chiusa a pugno, il più possibile alla stessa altezza e non troppo distanti tra loro.

Il sistema è stato testato offline su quattro diversi dataset acquisiti sperimentalmente per valutare le performance in diverse condizioni. L'applicazione è stata poi testata in real-time per valutare la velocità del sistema nell'effettuare i riconoscimenti.

Fig. 1. Esempi dei quattro gesti proposti

2. DATASET

Le immagini utilizzate sono state acquisite in laboratorio utilizzando l'acquisizione RGB di una Kinect v2. Vari operatori sono stati ripresi per l'esperimento, ognuno di essi libero di muoversi nella scena eseguendo i vari gesti sia in posizioni frontali alla telecamera sia lateralmente ad essa. Per ogni operatore ripreso sono stati acquisiti tutti e quattro i gesti per un totale tra le 40 e le 80 immagini corrette per operatore (10 – 20 per gesto).

Alcuni esempi di gesto per ogni dataset presentato sono riportati in Fig. 2.

1. Il dataset "**Base**" comprende i gesti eseguiti da 15 diversi operatori senza particolari condizioni;
2. Il dataset "**Colori Chiari**" comprende i gesti eseguiti da 5 diversi operatori indossanti abiti di colore chiaro, in modo tale da confondere le mani con il colore degli abiti o con il colore dello sfondo, anch'esso chiaro;
3. Il dataset "**Guanti**" comprende i gesti eseguiti da 5 diversi operatori indossanti un paio di guanti azzurro brillante, in modo da creare del forte contrasto tra le mani e il resto della scena;
4. Il dataset "**Zoom**" comprende i gesti eseguiti da 7 diversi operatori, con la telecamera posizionata più vicina e ad altezza uomo per inquadrare meglio le mani, con conseguente riduzione della scena inquadrata.

Fig. 2. Esempi presi dai dataset utilizzati.

I gesti sono stati acquisiti nello stesso ordine per ogni operatore, pertanto ogni dataset è stato adeguatamente mescolato prima di essere usato per le prove.

3. IL SISTEMA SVILUPPATO

Per rilevare le mani all'interno delle immagini e classificarle correttamente in uno dei quattro gesti proposti è stato utilizzato l'algoritmo Faster R-CNN [2] già presente in MATLAB [3] e modificato per essere applicato al problema. Per definire il gesto completo come combinazione di entrambe le mani, è stata sviluppata una funzione che utilizza come ingressi i rilevamenti eseguiti dall'algoritmo e, eseguendo una serie di controlli sulla posizione delle RoI e sulla confidenza del rilevamento, filtra i risultati in modo da ottenere in uscita un singolo gesto completo che rispetti le condizioni imposte dal problema.

L'algoritmo è stato allenato su tutti i dataset indipendentemente, ed infine su un dataset generale che comprende tutti i precedenti. Per ogni dataset sono stati usati l'80% dei dati, in modo da assicurare anche nel dataset complessivo di allenare il sistema su un numero adeguato di immagini per ogni dataset. Per la fase di test è stato usato il rimanente 20% di immagini per ogni dataset.

4. RISULTATI SPERIMENTALI

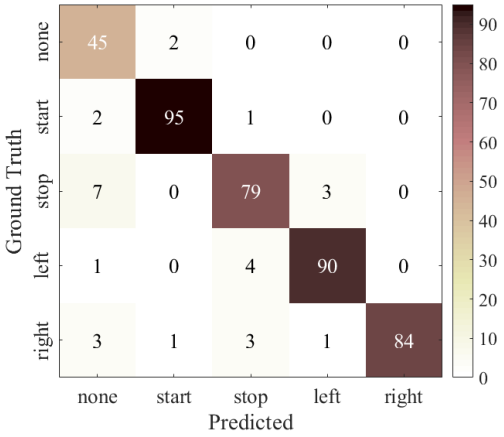

Fig. 3. Matrice di confusione del dataset complessivo di test.

Pur avendo un numero ridotto di immagini su cui addestrare l'algoritmo, il sistema riesce ad ottenere performance molto promettenti, impiegando in fase di allenamento un tempo variabile tra i 40 e i 70 minuti a seconda del dataset addestrato.

Allenando l'algoritmo sul dataset globale è stata raggiunta una accuratezza complessiva del 93% in fase di test, con un numero molto ridotto di predizioni incorrette come evidenziato dalla matrice di confusione in Fig. 3.

L'applicazione real-time impiega in media 0.23 secondi per eseguire la predizione e mostrarla a schermo, tempo adeguato per il tipo di applicazioni obiettivo.

5. CONCLUSIONE

Dai risultati ottenuti risulta evidente che un sistema basato sul Deep Learning come quello testato può essere applicato con successo ad applicazioni di robotica collaborativa. Il lavoro procederà quindi con due diversi obiettivi:

1. Testare il sistema su un robot reale e valutarne l'efficacia nel controllo;
2. Svilupparne una versione completamente embedded abbandonando la piattaforma MATLAB, usando ROS come interfaccia tra applicazione e robot.

RIFERIMENTI BIBLIOGRAFICI

[1] H. Liu and L. Wang, "Gesture recognition for human-robot collaboration: A review", *International Journal of Industrial Ergonomics*, 2017.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "*Faster r-cnn: Towards realtime object detection with region proposal networks*," in Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, ser. NIPS'15. Cambridge, MA, USA: MIT Press, 2015, pp. 91–99.

[3] Mathworks. "Object detection using faster r-cnn deep learning." web: *https://it.mathworks.com/help/vision/examples/objectdetection-using-faster-r-cnn-deep-learning.html*