ADATTAMENTO DI OPENPTRACK v2.0 PER PIATTAFORMA EMBEDDED

C. Nuzzi[(1)], S. Pasinetti[(1)], R. Ferraresi[(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

OpenPTrack è un progetto open source lanciato nel 2013 da UCLA REMAP e Open Perception, in collaborazione con l'Università degli Studi di Padova. Il progetto aveva come obiettivo la creazione di un sistema di tracking real-time per applicazioni di teatro e arte, ed è stato migliorato ed espanso negli anni fino alla più recente versione datata 2017 [1].
Il sistema si basa sull'utilizzo di ROS (Robot Operative System, piattaforma Linux) e di una serie di applicazioni realizzate in C++ per svolgere determinati compiti, come il tracking della persona (*human detection*), il tracking di un determinato oggetto (*object detection*) o il rilevamento dello skeleton della persona nella scena, così da riconoscerne la posa (*pose detection*).
Il team di sviluppo originale ha con successo sviluppato una versione embedded della prima release del progetto su schede NVIDIA Jetson TX1 e TK1 [2], mentre l'ultima versione non è ancora stata adattata per piattaforma embedded.

Fig. 1. La scheda NVIDIA Jetson TX2 montata su board Developer Kit.

L'obiettivo del lavoro qui presentato è quello di compiere questa trasposizione anche per l'ultima e più completa versione su scheda NVIDIA Jetson TX2 (Fig. 1), ed effettuare una caratterizzazione del sistema in configurazione multicamera embedded.

2. SET UP OBIETTIVO

Rispetto al progetto originale, la configurazione da noi provata con successo su PC utilizza fino a tre Kinect v2 su una singola macchina, che si occupa sia di effettuare i rilevamenti sia di gestire il tracking. Il set up obiettivo per la versione embedded è simile, ma sostituendo al PC la scheda NVIDIA Jetson TX2. Questa configurazione non è stata provata nel progetto originale, che invece utilizzava una Jetson TX1/TK1 per ogni Kinect per acquisire i dati dei rilevamenti dalle nuvole di punti, e poi un PC centrale per elaborare questi dati e realizzare il tracking [3].
La scheda NVIDIA Jetson TX2 è un dispositivo di calcolo embedded a basso consumo, sviluppato appositamente per elaborazioni nell'ambito AI. La scheda, in particolare nella versione Developer Kit, dispone di numerose interfacce che permettono al dispositivo di acquisire numerose funzionalità ed essere integrato in sistemi complessi [4].

3. FASE PRELIMINARE: CONVERSIONE LIBRERIE

OpenPTrack si basa su una serie di librerie necessarie per acquisire correttamente i dati dalle videocamere ed elaborarli, librerie che a loro volta si interfacciano con ROS (Fig. 2).
Per questo motivo il lavoro si è concentrato su una parte preliminare di conversione e riadattamento dei file di installazione del progetto originale, in modo tale da poter essere utilizzati con successo sulla Jetson TX2 che, a differenza della TX1, ha una architettura di sistema diversa. Per far ciò è stato necessario modificare le due principali librerie di riferimento del progetto: *libfreenect2* e *iai_kinect2*, entrambe fondamentali per utilizzare e visualizzare i dati delle Kinect con ROS, ed eseguire l'installazione della libreria *OpenCV* funzionante sulla scheda.
Per ottimizzare le prestazioni, è stato riadattato il codice in modo da utilizzare CUDA, il linguaggio utilizzato dalle GPU, al posto di OpenCL (non disponibile sulle schede Jetson).

4. FASI SUCCESSIVE: TEST E CARATTERIZZAZIONE

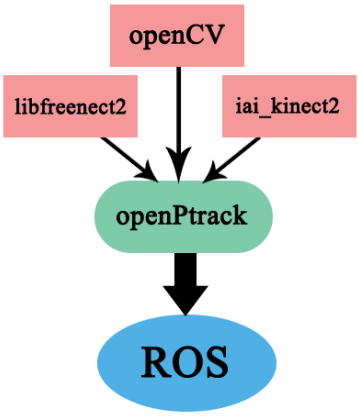

Fig. 2. Diagramma a blocchi del sistema: nella parte superiore sono presenti le principali librerie su cui si basa il software, il quale a sua volta si interfaccia con ROS.

L'obiettivo della fase di **test** è duplice:

- comprendere le difficoltà di elaborazione e calcolo della Jetson TX2 rispetto alla versione funzionante per PC in ogni aspetto del sistema (visualizzazione, calibrazione, rilevamento e tracking real-time in configurazione sia single camera che multi camera);
- ottimizzare il progetto in modo da migliorare ove possibile le prestazioni.

Terminata questa prima fase, il sistema verrà provato e caratterizzato nella stessa configurazione realizzata su PC già caratterizzata, così da valutarne le caratteristiche di misura ed evidenziare le differenze tra i due sistemi.
Rispetto alla versione originale, che utilizzava una Jetson TX1/TX2 o un PC per ogni sensore ed elaborava i dati su una macchina a parte (denominata master), il nostro set up sperimentale anche nelle prove utilizzanti un PC usano una singola macchina. Dopo aver valutato le prestazioni in questa configurazione è nostra intenzione valutarle in una configurazione decentralizzata come suggerito nel progetto originale, e comparare i risultati delle due prove.

5. CONCLUSIONI E LAVORO FUTURO

OpenPTrack è un sistema molto interessante che può essere utilizzato anche in applicazioni più industriali in ottica Industria 4.0. Per questo motivo risulta opportuno valutarne le capacità una volta adattato al funzionamento su schede embedded low cost, come la NVIDIA Jetson TX2.
Lo sviluppo della versione embedded di OpenPTrack v2.0 è ancora in corso. Al termine delle prove e una volta valutate le prestazioni in fase di misura, è nostra intenzione espandere il sistema con nuove applicazioni volte all'utilizzo del progetto anche in ambiti più industriali, come la robotica collaborativa e il riconoscimento gesti per dialogare con il robot.

RIFERIMENTI BIBLIOGRAFICI

[1] OpenPTrack, web: http://openptrack.org/
[2] OpenPTrack Jetson TX2/TK2 installation, web: https://github.com/OpenPTrack/open_ptrack/wiki/Jetson-TX1-Installation
[3] OpenPTrack network configuration, web: https://github.com/OpenPTrack/open_ptrack/wiki/Network-Configuration
[4] NVIDIA Jetson TX2 Developer Kit, web: https://developer.nvidia.com/embedded/buy/jetson-tx2-devkit