

Sperimentazione on-the-road di un'autovettura drive-by-wire a guida autonoma: Esperienze del nodo AI3S dell'Università di Salerno

Vincenzo Carletti, Diego Gragnaniello, Antonio Greco, Francesco Moscato, Leonardo Rundo, Alessia Saggese, Mario Vento, Antonio Vicinanza

Dipartimento DIEM, Università degli Studi di Salerno

{vcarletti, digragnaniello, agreco, fmoscato, lrundo, asaggese, mvento, anvicinanza}@unisa.it

Abstract

Il grande progresso dell'intelligenza artificiale a cui abbiamo assistito nell'ultimo decennio ha reso ormai sempre più concreta la possibilità di realizzare un veicolo a guida autonoma. Questa è la sfida intrapresa dal MIVIA Lab con l'ambizioso progetto della MIVIA Car. Quest'ultima è un veicolo elettrico equipaggiato con sensori GPS, LIDAR, radar e telecamere per percepire l'ambiente circostante e dotato di un motore inferenziale in grado di elaborare i dati in tempo reale e di pianificare il percorso migliore, tra quelli sicuri per l'incolumità delle persone a bordo, dei pedoni e degli altri veicoli, da percorrere da un punto di partenza ad una destinazione desiderata. Sebbene il progetto sia iniziato nel 2021, questo articolo si pone l'obiettivo di descrivere le attività di ricerca in corso e l'architettura hardware e software a supporto.

1 Introduzione

Recenti statistiche dimostrano che la causa principale degli incidenti stradali è la distrazione dell'autista, soprattutto quando quest'ultimo percorre viaggi molto lunghi e deve rimanere concentrato per molte ore [United States Department of Transportation, 2021]. Questo dato allarmante e la dilagante diffusione dell'intelligenza artificiale nella vita di tutti i giorni sono i motivi principali del grande interesse per i veicoli a guida autonoma in ambito industriale ed accademico.

In base alla tassonomia definita in [Yurtsever *et al.*, 2020], esistono due tipologie di approcci per la realizzazione di veicoli a guida autonoma: modulare e end-to-end. Gli approcci modulari identificano specifici task di guida (analisi dei dati, local planning, behavioral planning, global planning), definendo dei componenti dedicati allo svolgimento di tali task. Il principale vantaggio di queste tecniche è il controllo completo dei vari moduli e la possibilità di effettuare diagnosi e manutenzione su un sottoinsieme dell'architettura software. D'altro canto, lo svantaggio è la difficoltà di considerare tutti i possibili scenari di guida reali che il veicolo deve affrontare su strada. Gli approcci E2E, invece, trattano l'intero problema di guida come un singolo task di apprendimento. I metodi più diffusi sono quelli basati su Imitation Learning (IL), che



Figura 1: Diverse viste della MIVIA Car in azione nel campus dell'Università degli Studi di Salerno.

apprendono le policy di guida dalle azioni svolte su scenari di esempio da esperti umani. Pertanto, durante la fase di addestramento, il sistema riceve in input i dati provenienti da sensori eterogenei (telecamere RGB e di profondità, LIDAR, radar, segmentazione semantica, output di moduli di videoanalisi) e i comandi impartiti agli attuatori (accelerazione, frenata, sterzata) dall'autista umano e imparano ad affrontare autonomamente l'intero task di guida.

Il MIVIA Lab sta lavorando alla ricerca in un settore specifico dell'IL, ovvero il Conditional Imitation Learning (CIL), che consente di introdurre dei comandi di alto livello in grado di guidare il veicolo in presenza di incroci, quando più strade sono potenzialmente percorribili. In particolare, l'obiettivo è quello di progettare e realizzare algoritmi di guida autonoma innovativi e robusti e di sperimentarli non solo in ambienti simulati come CARLA [Dosovitskiy *et al.*, 2017], ma anche su strada. Da qui l'esigenza di realizzare la MIVIA Car, un veicolo elettrico equipaggiato con GPS, LIDAR, radar e telecamere, e una piattaforma software in grado di effettuare l'elaborazione in tempo reale dei dati dai sensori e di impartire i comandi di guida al veicolo tramite un sistema drive-by-wire.

In questo articolo descriviamo la MIVIA Car e le attività di ricerca del MIVIA Lab nell'ambito dei veicoli a guida autonoma, con specifico riferimento al paradigma CIL.

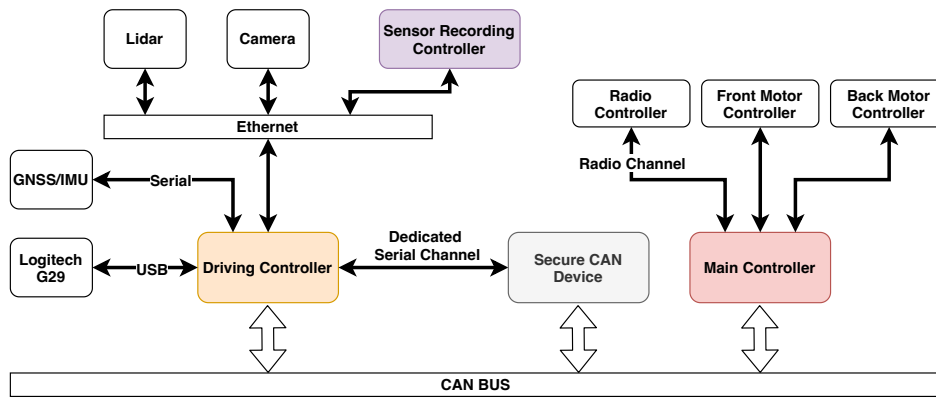


Figura 2: Architettura hardware della MIVIA Car.

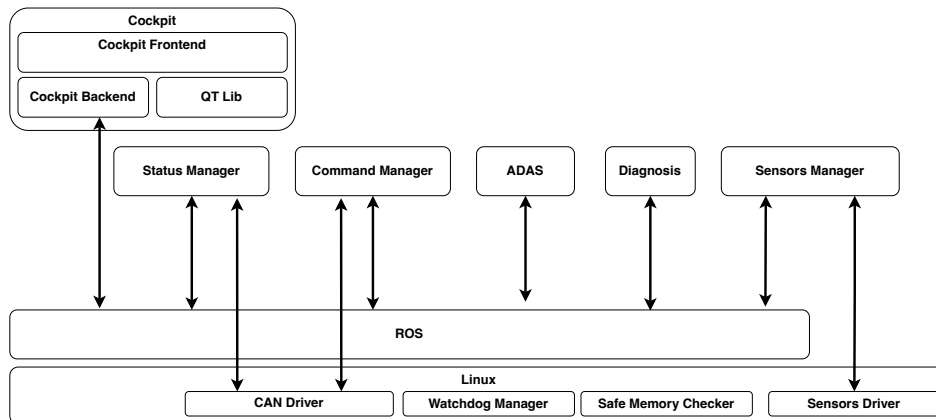


Figura 3: Architettura software della MIVIA Car.

2 MIVIA Car

La ricerca sui veicoli a guida autonoma è spesso portata avanti su ambienti virtuali ipotizzando che questi possano in qualche modo riprodurre sufficientemente la complessità di un contesto reale. Ovviamente, tale ipotesi è valida fino a quando ci si ferma ad un livello di astrazione iniziale. Tuttavia non è pensabile fermarsi in ambienti virtuali per ottenere una valutazione oggettiva e realistica delle metodologie utilizzate per la guida autonoma. A tale proposito, abbiamo selezionato PIXKIT¹, un veicolo elettrico completamente drive-by-wire che consenta un alto livello di personalizzazione ed un controllo completo sul veicolo. In Figura 2 abbiamo riportato una vista di alto livello dell'architettura hardware del veicolo che mette in evidenza i sensori e gli attuatori disponibili e le diverse centraline: (i) la centralina principale (Main Controller) dell'automobile fornita già nel kit, su cui si attestano i motori delle singole ruote, i freni ed i motori per la sterzata con cui comunica il radio comando fornito dal produttore con priorità su tutti gli altri sistemi di controllo in caso di emergenza; (ii) la centralina di guida (Driving Controller), su cui convergono tutti i sensori, su cui sono elaborati i moduli ADAS e sulla quale è installato il virtual cockpit mostrato in Figura 4; (iii) la centralina di monitoraggio del CAN (Secure CAN Device),

che ha lo scopo di effettuare la verifica dei comandi trasmessi sul CAN ed in finer (iv) la centralina di registrazione (Sensor Recording Controller), utilizzata per registrare i dati necessari alla raccolta dei dataset in ambienti reali. Si evidenzia che le ultime tre sono state progettate e realizzate dal MIVIA Lab a supporto delle attività di ricerca. Oltre che nell'architettura hardware il veicolo è stato personalizzato anche da un punto di vista strutturale. Sono stati aggiunti elementi per il fissaggio dei sensori, dei sedili per due passeggeri, uno sterzo con pedaliera utilizzati per la ricerca sul CIL e lo schermo su cui è mostrato il virtual cockpit. In Figura 1 riportiamo alcune foto che raffigurano il veicolo personalizzato dal MIVIA Lab².

Da un punto di vista del software, la centralina di guida su cui vengono eseguiti i diversi algoritmi in fase di test e validazione è basata su ROS (Robot Operating System), un sistema operativo realtime basato su kernel Linux. In Figura 3 riportiamo una vista dei moduli principali; in particolare, è possibile notare come ogni modulo sia disaccoppiato dagli altri, tramite ROS, ed intercambiabile se necessario. Oltre al modulo ADAS, responsabile dell'interfacciamento degli algoritmi di guida con tutti gli altri moduli dell'architettura, si può notare la presenza di: (i) un modulo di diagnostica, responsabile di monitorare lo stato complessivo del veicolo tra-

¹<https://www.pixmoving.com/pixkit>

²<https://mivia.unisa.it>

mite la lettura delle informazioni ottenute dal CAN BUS ed una rappresentazione interna dello stato atteso dell'automobile, (ii) un modulo di interfacciamento con i sensori, che astrae i sensori a tutti gli altri moduli e gestisce le policy di utilizzo; (iii) un command manager, responsabile di trasmettere i comandi sul CAN BUS in maniera sicura, gestendo la priorità tra le diverse sorgenti di comando (es. algoritmi o conducente tramite pedaliera e sterzo aggiuntivi), filtrando eventuali comandi incoerenti con lo stato del veicolo o rispetto a vincoli imposti a tempo di configurazione e verificando l'effettiva trasmissione dei comandi; (iv) un modulo status manager, che monitora lo stato dell'automobile e costruisce una rappresentazione coerente e sintetica da fornire a tutti gli altri moduli; (v) il virtual cockpit, mostrato anche in Figura 4, che fornisce al guidatore una vista complessiva di tutti gli altri moduli, dallo stato del veicolo, ai sensori, ai comandi e alle informazioni di diagnostica.

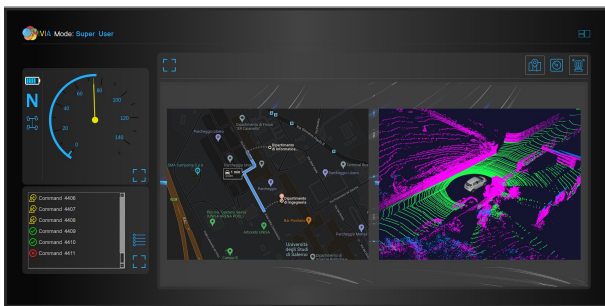


Figura 4: Il Virtual Cockpit dell'automobile, progettato e realizzato dal MIVIA Lab. Sulla sinistra sono presenti le finestre realizzate per consentire il monitoraggio dello stato e del comportamento degli algoritmi ADAS, mentre sulla destra è presente l'area di visualizzazione dei sensori. Il layout dell'interfaccia può essere adattato in modo da dare maggior spazio ad ognuna delle aree.

3 Attività di ricerca

La Figura 5 mostra i moduli software che compongono un sistema di guida autonoma. Dato il percorso di alto livello, stabilito dal Mission Planner, e stimata la posizione del veicolo mediante GPS, IMU ed eventualmente LIDAR, l'algoritmo di guida utilizza i dati dei sensori (telecamere e LIDAR) e/o prodotti dai moduli di perception per analizzare lo stato dell'ambiente circostante (Behavior Planning) e pianificare una traiettoria sicura (Local Planning).

Le attività di ricerca del MIVIA Lab, introdotte nei seguenti paragrafi, ruotano principalmente intorno al CIL, che permette di apprendere tramite esempi tutta la logica di planning; tuttavia, al fine di fornire maggiori informazioni al sistema di guida autonoma basato su CIL, le attività si stanno concentrando anche sulla progettazione di algoritmi di riconoscimento della corsia, riconoscimento segnali e semafori, rilevamento oggetti 3D e tracking di persone e veicoli, robusti in tutte le condizioni meteorologiche e orarie.

3.1 Conditional Imitation Learning

La maggior parte degli approcci di CIL allo stato dell'arte utilizza i dati acquisiti da un unico sensore, tipicamente la te-

lecamera RGB [Codevilla *et al.*, 2019] [Ohn-Bar *et al.*, 2020] [Chen *et al.*, 2020]. Tuttavia, studi recenti hanno dimostrato che la fusione delle immagini con dati provenienti da altri sensori come LIDAR [Eraqi *et al.*, 2020] e telecamera di profondità [Xiao *et al.*, 2020] o con output prodotti da moduli di video-analisi come la segmentazione semantica [Huang *et al.*, 2020], permette di migliorare le performance del sistema di guida basato su CIL. Sulla falsariga di questo trend promettente, abbiamo utilizzato il dataset CARLA100 [Codevilla *et al.*, 2019] per addestrare una ResNet-50 con le seguenti combinazioni di input: immagine RGB e segmentazione semantica (RGBS), immagine RGB e immagine di profondità (RGBD) e tutto insieme (RGBDS). Il meccanismo di combinazione è basato su early-fusion. Le sperimentazioni, in corso sul benchmark NoCrash [Codevilla *et al.*, 2019], iniziano a fornire risultati interessanti, tanto che una prima versione dell'algoritmo RGBD ha guidato la MIVIA Car all'interno del campus dell'Università degli Studi di Salerno ³.

Le attività di ricerca del prossimo futuro sono orientate alla sperimentazione più approfondita di diversi approcci multimodali, con varie backbone CNN e l'aggiunta di input provenienti da altri moduli di video-analisi (descritti nel prossimo paragrafo), con l'obiettivo di fornire informazioni aggiuntive sull'ambiente che possano migliorare l'algoritmo di guida autonoma. Inoltre, l'analisi dei lavori scientifici ha evidenziato l'utilizzo di funzioni di costo che non tengono conto del segno dell'errore e di eventuali sbalzi nell'inseguimento della traiettoria; pertanto, le attività di ricerca saranno indirizzate anche in questa direzione, che riteniamo di fondamentale rilevanza.

3.2 Moduli di video-analisi

Lane detection

Rimanere nella propria corsia è fondamentale per un veicolo ed eventuali traiettorie che invadano la corsia opposta devono essere considerate non percorribili dal local planner basato su CIL. Si ritiene pertanto utile progettare e realizzare un algoritmo efficiente ed efficace non solo su simulatore o su dataset standard, ma anche e soprattutto in test reali su strada [Feniche e Mazri, 2019].

Traffic sign e traffic light recognition

Il riconoscimento della segnaletica stradale, verticale e orizzontale, e dello stato dei semafori incontrati da un veicolo a guida autonoma lungo il suo percorso, sono caratteristiche necessarie per rispettare le regole del codice della strada. Tali informazioni potrebbero quindi essere d'ausilio al sistema basato su CIL per pianificare traiettorie in maniera più consapevole [Ertler *et al.*, 2020]. Pertanto, anche questi algoritmi saranno oggetto di approfonditi studi nel prossimo futuro.

3D object detection e tracking

Un veicolo a guida autonoma deve essere consapevole della posizione attuale dei pedoni e dei veicoli nella scena e deve essere in grado di predire i possibili movimenti di tali entità su strada. E' inoltre fondamentale che la posizione degli oggetti venga stimata in 3D, in modo da poter effettuare le proiezioni degli stessi dal sistema di riferimento dell'immagine

³https://youtu.be/-W_RGL6MKnY

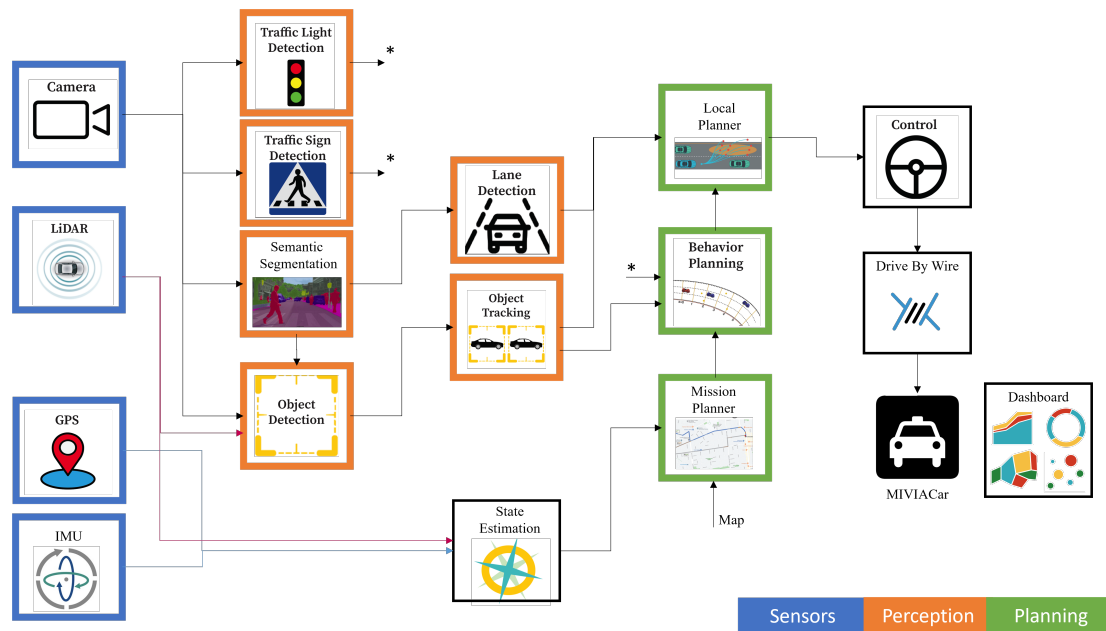


Figura 5: Moduli software per un sistema di guida autonoma. Dato il percorso di alto livello, stabilito dal Mission Planner, e stimata la posizione del veicolo mediante GPS, IMU ed eventualmente LIDAR, l'algoritmo di guida utilizza i dati dei sensori e/o prodotti dai moduli di perception per analizzare lo stato dell'ambiente circostante (Behavior Planning) e pianificare una traiettoria sicura (Local Planning).

al sistema di riferimento del mondo. Pertanto, si ritiene utile lavorare su tale linea di ricerca per fornire al sistema basato su CIL le posizioni attuali e predette di ogni pedone e veicolo nella scena, con un'accuratezza elevata anche in condizioni meteo e di visibilità avverse [Ciaparrone *et al.*, 2020].

Riferimenti bibliografici

- [Chen *et al.*, 2020] Dian Chen, Brady Zhou, Vladlen Koltun, e Philipp Krähenbühl. Learning by cheating. In *Conf. on Robot Learning*, pages 66–75. PMLR, 2020.
- [Ciaparrone *et al.*, 2020] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliarferri, e Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, 2020.
- [Codevilla *et al.*, 2019] Felipe Codevilla, Eder Santana, Antonio M López, e Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *IEEE Int. Conf. on Computer Vision (ICCV)*, pages 9329–9338, 2019.
- [Dosovitskiy *et al.*, 2017] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, e Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on Robot Learning*, pages 1–16. PMLR, 2017.
- [Eraqi *et al.*, 2020] Hesham M Eraqi, Mohamed N Moustafa, e Jens Honer. Efficient occupancy grid mapping and camera-lidar fusion for conditional imitation learning driving. In *IEEE Int. Conf. on Intelligent Transportation Systems (ITSC)*, pages 1–7. IEEE, 2020.
- [Ertler *et al.*, 2020] Christian Ertler, Jerneja Mislej, Tobias Ollmann, Lorenzo Porzi, Gerhard Neuhold, e Yubin Kuang. The mapillary traffic sign dataset for detection and classification on a global scale. In *European Conference on Computer Vision*, pages 68–84. Springer, 2020.
- [Feniche e Mazri, 2019] Mehdi Feniche e Tomader Mazri. Lane detection and tracking for intelligent vehicles: A survey. In *Int. Conf. of Computer Science and Renewable Energies (ICCSRE)*, pages 1–4. IEEE, 2019.
- [Huang *et al.*, 2020] Zhiyu Huang, Chen Lv, Yang Xing, e Jingda Wu. Multi-modal sensor fusion-based deep neural network for end-to-end autonomous driving with scene understanding. *IEEE Sensors Journal*, 21(10):11781–11790, 2020.
- [Ohn-Bar *et al.*, 2020] Eshed Ohn-Bar, Aditya Prakash, Aseem Behl, Kashyap Chitta, e Andreas Geiger. Learning situational driving. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 11296–11305, 2020.
- [United States Department of Transportation, 2021] United States Department of Transportation. Risky Driving. <https://www.nhtsa.gov/>, 2021. Online; accessed 18 November 2021.
- [Xiao *et al.*, 2020] Yi Xiao, Felipe Codevilla, Akhil Gurram, Onay Urfalioglu, e Antonio M López. Multimodal end-to-end autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [Yurtsever *et al.*, 2020] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, e Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8:58443–58469, 2020.