



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Scuola di Scienze Matematiche, Fisiche e Naturali  
Corso di Laurea in Informatica

TITOLO IN ITALIANO

TITLE IN ENGLISH

TERROSI FRANCESCO

BONDAVALLI ANDREA

STRIGINI LORENZO

Anno Accademico 2018-2019



## ABSTRACT

I veicoli autonomi sono sistemi cyber-fisici critici, complessi sotto molteplici aspetti: dalla tecnologia necessaria all'acquisizione di dati esterni come radar, lidar, GPS... all'implementazione del software che si occupa della guida vera e propria. Con l'enorme progresso avuto nel campo del machine learning nell'ultimo ventennio, la prospettiva di macchine capaci di guidare senza alcuna interazione con l'uomo e' sempre piu' vicina.

Il sistema di controllo del veicolo puo' essere visto in maniera semplificata come formato da una rete neurale, che determina l'azione da eseguire (quanto accelerare/decelerare o l'angolo di sterzata) sulla base dei dati ricevuti dai sensori. Dal momento che nelle predizioni effettuate da una rete vi e' insito un errore (i.e. risulta impossibile avere un'accuratezza del 100% sui risultati prodotti) e' di fondamentale importanza avere un *safety monitor*, il cui compito e' quello di controllare e sanificare gli output dell'intelligenza artificiale.

In questo lavoro abbiamo studiato come varia il rapporto fra un safety-monitor relativamente semplice e una rete neurale addestrata per la guida autonoma, andando a definire dei semplici requisiti di safety e osservando come un continuo training della rete neurale vada a impattare (o meno) sull'utilita' del monitor in questione.

Per poter svolgere questo lavoro sono stati utilizzati molti software open-source: grazie al simulatore CARLA e' stato possibile avere una rappresentazione realistica delle leggi della strada e della fisica dei veicoli. Le reti neurali che sono poi state prese in considerazione sono state addestrate con algoritmi di *reinforcement learning* e *imitation learning*, considerati fra i piu' promettenti in questo campo. CARLA inoltre permette di avere una simulazione realistica dei sensori utilizzati sui veicoli autonomi: questo ha permesso di costruire un semplice (ma efficace) safety monitor che effettua dei controlli di sicurezza sulla base dei dati ricevuti dal lidar e sulla velocita' e direzione del veicolo autonomo; sono stati infine condotti gli esperimenti per studiare l'interazione fra questi due componenti.



---

## INDICE

---

1	Introduzione	9
1.1	Section 1.1	9
1.1.1	Subsection 1.1.1	9
2	Stato dell'arte	11
2.1	Self-driving cars	11
2.2	Safety nell'Automotive	13
3	Analisi del Sistema	15
3.1	Strumenti utilizzati	15
3.2	Architettura del software (estrapolazione dati, interazione rete-monitor)	15
3.3	Descrizione della metodologia	16
4	Risultati dell'analisi	17



---

## ELENCO DELLE TABELLE

---





---

## ELENCO DELLE FIGURE

---



---

## INTRODUZIONE

---

Ciao, sono un'introduzione

### 1.1 SECTION 1.1

A section

#### 1.1.1 Subsection 1.1.1

A subsection

##### *Subsubsection 1.1.1.1*

A subsubsection with a list of elements

- element1
- element2
- element3

PARAGRAPH A paragraph



---

## STATO DELL'ARTE

---

### 2.1 SELF-DRIVING CARS

Le macchine a guida autonoma rientrano nella categoria dei sistemi informatici cyber-fisici critici. Anche se già di per sé assimilabili nella categoria dei sistemi di sistemi (dal momento che vi sono più costituenti systems che interagiscono fra loro), l'obiettivo della comunità scientifica è quello di riuscire ad ottenere un sistema di sistemi risultante dall'interconnessione di più veicoli autonomi: ognuno di questi sistemi avrà un obiettivo differente (i.e. diverse destinazioni da raggiungere) ma per ottenerlo è assolutamente necessaria una cooperazione fra essi. Come diretta conseguenza, è indispensabile che il sistema di controllo del veicolo sappia non solo obbedire al codice della strada ma anche essere in grado di riconoscere le situazioni di potenziale pericolo.

Un veicolo autonomo dev'essere in grado di "osservare" l'ambiente circostante, questo è reso possibile grazie ai sensori installati su di essa.

Tipicamente vengono utilizzati:

- Telecamere
  - Necessarie per catturare immagini dell'ambiente
- Radar, Lidar, Sonar
  - Utilizzati per creare una mappa dell'ambiente in cui naviga il veicolo e per percepire gli ostacoli
- GPS, sensori inerziali, odometria
  - Indispensabili per pianificare il percorso da seguire e conoscere la posizione del veicolo nell'ambiente operativo

L'architettura software di un veicolo e' composta da piu' moduli interagenti, dove l'errore di uno di questi potrebbe risultare in una minaccia per la safety del sistema.

Possiamo semplificare il modello architetturale come composto da tre moduli separati:

- Environment Mapping
  - La componente che riceve i dati direttamente dai sensori e si occupa di filtrarli (e.g. per ridurre il rumore e scartare valori poco significativi) ed aggregarli per riconoscere il perimetro dell'area circostante ed eventuali ostacoli
- Motion control
  - Questo modulo riceve in input i dati dopo che questi sono stati elaborati dal sistema di *data processing*. Dopo averli osservati, il *motion planner* decide in che direzione debba proseguire il veicolo; questo comando viene quindi inviato al *controller*, composto da un modulo di controllo longitudinale (accelerazione) e uno di controllo laterale (sterzo del veicolo), il quale va effettivamente a interagire con gli attuatori del sistema
- System Supervisor
  - Un supervisore, o monitor, e' il sistema che si occupa di rilevare guasti o fallimenti sia hardware che software. Dal punto di vista hardware, i controlli effettuati sono principalmente sui guasti a componenti hardware e sugli output del *controller* (ad esempio che appartengano al dominio del sistema). Il monitor software invece si occupa di rilevare inconsistenze fra gli output dei due moduli precedenti. Questo e' di fondamentale importanza in quanto permette di controllare che l'output del *controller* non porti il sistema in una situazione di pericolo, o peggio: ad un fallimento catastrofico che coinvolgerebbe anche vite umane

=====

IMMAGINE ARCHITETTURA SISTEMA DI CONTROLLO [SETTIMANA 1 - LEZIONE 3]

=====

Se classicamente venivano implementati in software modelli statistici noti per mappare l'ambiente (e.g. Kalman filter) e modelli fisici o di teoria del controllo per manovrare il veicolo (e.g. PID controller), grazie al progresso nel campo del *machine learning* avuto negli ultimi anni si e' iniziato a utilizzare reti neurali addestrate alla guida [1] [2].

Le reti neurali hanno dimostrato di sapere reagire meglio a situazioni sconosciute rispetto ai meccanismi classici, tuttavia richiedono tanti piu' dati quanto piu' e' complesso il compito da eseguire.

Questi modelli computazionali inoltre soffrono del cosiddetto *black box problem*: e' molto difficile riuscire perche' la rete abbia associato l'output  $y$  all'input  $x$ . Nonostante siano stati proposti alcuni framework [3] per aiutare a comprendere i meccanismi che regolano le decisioni di un'intelligenza artificiale sotto specifiche assunzioni, non si e' ancora trovata una soluzione universale al problema.





---

## ANALISI DEL SISTEMA

---

In questa sezione spieghiamo il tipo di analisi condotta, elenchiamo i tool utilizzati e le scelte/assunzioni fatte.

- CARLA - cos'è, cosa fa, pregi e difetti
- reti neurali utilizzate
- Quali sono i dati che cerchiamo
- Come vengono condotti gli esperimenti e ricavati i dati

### 3.1 STRUMENTI UTILIZZATI

- CARLA
- Nervana Systems - coach (Intel)
- Reti neurali su git
- Point Cloud Library per filtrare i dati

### 3.2 ARCHITETTURA DEL SOFTWARE (ESTRAPOLAZIONE DATI, INTERAZIONE RETE-MONITOR)

- Interazione rete-monitor
- Safety Monitor - obstacle detection
- Come vengono raccolti i dati
- Come vengono preprocessati

### 3.3 DESCRIZIONE DELLA METODOLOGIA

- Scelta delle metriche (come e perché)
- Come le estraiamo?
- Come vengono effettuati gli esperimenti (durata fissa? ad oltranza? fino ad un fallimento? ...)

---

## RISULTATI DELL'ANALISI

---

In questa sezione elenchiamo i dati che sono stati raccolti e quali sono i risultati che abbiamo ottenuto (errori ricorrenti, grafici, rapporto monitor-rete neurale)



---

## BIBLIOGRAFIA

---

- [1] Jelena Kocic, Nenad Jovicic, Vujo Drndarevi, *An End-To-End Deep Neural Network for Autonomous Driving Designed for Embedded Automotive Platforms* (2019) (Cited on page 13.)
- [2] Qing Rao, Jelena Frtunikj, *Deep learning for self-driving cars: chances and challenges* (2018) (Cited on page 13.)
- [3] Carlos Zednik, Otto-von-Guericke-Universitat Magdeburg *Solving the Black Box Problem: A Normative Framework for Explainable Artificial Intelligence* (Cited on page 13.)