

---

# 01TUJXX IMAGE PROCESSING AND COMPUTER VISION

## ADVANCED IMAGE SEGMENTATION

*Durata prevista dell'esercitazione: 3 ore (2 settimane)*

Lo scopo di questa esercitazione è quello di sperimentare alcune tecniche più avanzate di Image Segmentation, concentrandoci in particolare sull'edge detection (individuazione dei contorni in un'immagine).

### ESERCIZIO: RILEVAMENTO SEGNALETICA ORIZZONTALE

Quanto visto nel laboratorio precedente, unito alla trasformata di Hough, può essere utilizzato per rilevare la segnaletica orizzontale delle strade. Utilizzando prima le immagini (*test\_img.zip*) e poi i video (*test\_videos.zip*) di test disponibili sul portale della didattica, realizzare un'applicazione con OpenCV per il rilevamento della segnaletica orizzontale.

In particolare, dopo le opportune valutazioni sulla scala di colore da utilizzare (RGB/BGR? HVS? ...) e la conseguente sua scelta, il programma dovrà *sogliare* l'immagine/frame per mantenere solo gli oggetti bianchi e gialli<sup>1</sup>, individuare un'appropriata *regione di interesse* per la strada, applicare Canny ed eventualmente gli operatori morfologici per ottenere i *contorni* delle linee, utilizzare la *trasformata di Hough* per rilevare effettivamente le linee, ed infine *evidenziare* con un colore a scelta la segnaletica orizzontale rilevata. Visualizzare tutti i passaggi intermedi realizzati e non preoccuparsi di seguire con precisione le linee curve. Qual è la percentuale di successo del rilevamento? L'applicazione di una sfocatura Gaussiana con un kernel la cui dimensione è nell'ordine di 9, 11, 15, ... per rimuovere il rumore può migliorare il risultato?

### LA TRASFORMATATA DI HOUGH... IN BREVE

La trasformata di Hough è una tecnica di estrazione di feature utilizzata per analizzare immagini nella Computer Vision. L'obiettivo della tecnica è di trovare delle *istanze imperfette* di oggetti all'interno di una certa classe di forme, utilizzando una procedura di votazione. In questa procedura, i candidati oggetti sono ottenuti come un massimo locale in uno spazio detto "accumulatore", esplicitamente costruito dall'algoritmo della trasformata di Hough. La trasformata classica di Hough riguarda l'identificazione di linee nelle immagini, ma esistono anche sue estensioni per l'identificazione di altre forme, tipicamente cerchi ed ellissi. OpenCV implementa sia la trasformata di Hough per le linee sia quella per i cerchi.

Per un'introduzione più approfondita: [https://en.wikipedia.org/wiki/Hough\\_transform](https://en.wikipedia.org/wiki/Hough_transform)

### LINK UTILI

- Canny [https://docs.opencv.org/4.5.5/da/d22/tutorial\\_py\\_canny.html](https://docs.opencv.org/4.5.5/da/d22/tutorial_py_canny.html)
- Operatori morfologici [https://docs.opencv.org/4.5.5/d9/d61/tutorial\\_py\\_morphological\\_ops.html](https://docs.opencv.org/4.5.5/d9/d61/tutorial_py_morphological_ops.html)
- Hough Line Transform [https://docs.opencv.org/4.5.5/d6/d10/tutorial\\_py\\_houghlines.html](https://docs.opencv.org/4.5.5/d6/d10/tutorial_py_houghlines.html)

---

<sup>1</sup> cioè, i due colori delle linee nelle immagini/frame

- Altri metodi utili:
  - `inRange()` per sogliare, [https://docs.opencv.org/4.5.5/d2/de8/group\\_core\\_array.html](https://docs.opencv.org/4.5.5/d2/de8/group_core_array.html)
  - `fillPoly()` per riempire un'area delineata, [https://docs.opencv.org/4.5.5/d6/d6e/group\\_imgproc\\_draw.html](https://docs.opencv.org/4.5.5/d6/d6e/group_imgproc_draw.html)
  - Bitwise operations, [https://docs.opencv.org/4.5.5/d0/d86/tutorial\\_py\\_image\\_arithmetics.html](https://docs.opencv.org/4.5.5/d0/d86/tutorial_py_image_arithmetics.html)

*Disclaimer:* immagini e video sono tratti da un progetto di Udacity disponibile nel repository GitHub: <https://github.com/udacity/CarND-LaneLines-P1>, distribuiti con licenza MIT.