

# Traffic Sign Detection

Giada Pietrocola, Achille Cannavale

Università degli Studi di Cassino e del Lazio Meridionale, Italia

## Sommario

*Il rilevamento dei segnali stradali è una componente fondamentale per molti sistemi di assistenza alla guida e di sicurezza automobilistica. In questo studio, presentiamo un approccio per il rilevamento dei segnali stradali utilizzando la libreria OpenCV, senza l'uso di tecniche di deep learning. La nostra metodologia si basa su tecniche di elaborazione delle immagini per individuare e riconoscere i segnali stradali nelle immagini che costituiscono il nostro dataset. Presentiamo inoltre una valutazione sperimentale del nostro metodo su un dataset di immagini di strade reali. I risultati indicano che il nostro approccio può raggiungere un rilevamento efficace dei segnali stradali senza la necessità di complessi modelli di intelligenza artificiale, offrendo una soluzione leggera e computazionalmente efficiente.*

## Introduzione

Il rilevamento e il riconoscimento dei segnali stradali rivestono un'importanza cruciale per il miglioramento della sicurezza dei veicoli dotati di guida autonoma e dei sistemi di assistenza alla guida. Ogni anno, circa 3000 persone perdono la vita in incidenti automobilistici negli Stati Uniti a causa di disattenzioni dei conducenti, rappresentando tra l'8% e il 9% di tutti gli incidenti stradali nel paese [1].

Questo scenario sottolinea l'urgente necessità per la ricerca globale di sviluppare sistemi di rilevamento dei segnali sempre più efficienti. Tra le tecnologie più promettenti vi sono le Reti Neurali Convoluzionali (CNN), addestrate su vasti dataset di immagini etichettate. Le CNN sono particolarmente efficaci nel riconoscere e classificare i segnali stradali, raggiungendo tassi di accuratezza superiori al 98% [2].

Il nostro studio si propone di sviluppare un sistema avanzato per il rilevamento e il riconoscimento dei segnali di senso vietato, utilizzando tecniche sofisticate di manipolazione ed elaborazione delle immagini [figura 3]. Il processo inizia con l'applicazione di filtri per migliorare il contrasto e ridurre il rumore nelle immagini, seguiti da tecniche di rilevamento dei contorni e segmentazione per identificare i potenziali segnali stradali. Successivamente, impieghiamo degli algoritmi di analisi dei componenti connessi per eliminare i falsi positivi e migliorare ulteriormente l'accuratezza del rilevamento per

poi concludere il nostro sistema con l'utilizzo di una Machine Learning black box, che andrà ad eseguire una Support Vector Machine (SVM) [3].

Questo approccio integrato mira non solo a migliorare la precisione del riconoscimento dei segnali stradali, ma anche a contribuire significativamente alla sicurezza e all'efficienza dei sistemi di guida autonomi e dei sistemi di assistenza alla guida, riducendo così il numero di incidenti dovuti a errori umani sulle strade.

## Metodologie

### Ricerca dei Rettangoli

Il nostro sistema inizia convertendo l'immagine di input in scala di grigi e applicando un filtro GaussianBlur con un kernel 5x5 e parametri  $\sigma_X$  e  $\sigma_Y$  di 0 e 0.6 rispettivamente. Questo passaggio è fondamentale per migliorare la qualità dell'immagine e ridurre il rumore.

Successivamente, viene eseguita un'operazione morfologica di tophat utilizzando uno Structuring Element di dimensioni (11, 61). Questo processo è progettato per enfatizzare i rettangoli presenti nell'immagine, rendendoli più evidenti e facilitando il loro rilevamento successivo.

Dopo di ciò, l'immagine viene binarizzata utilizzando un Adaptive Thresholding. Questa tecnica di binarizzazione si adatta dinamicamente alle variazioni locali di luminosità, garantendo una segmentazione più accurata degli og-



**Figura 1:** Esempio di un cartello prima e dopo il *tophat enhancement*

getti nell'immagine. Successivamente, vengono estratti i contorni degli oggetti rilevati.

**Filtraggio dei Rettangoli** In questa fase, viene applicato un filtro per eliminare i contorni che non soddisfano specifici requisiti. Questo processo è cruciale per eliminare falsi positivi e migliorare l'accuratezza complessiva del sistema. I criteri utilizzati per il filtraggio includono l'area dei rettangoli, il loro rapporto di aspetto e altri attributi geometrici che consentono di distinguere correttamente i segnali stradali dagli altri oggetti nell'immagine.

**Tabella 1:** Criteri di Filtraggio dei Rettangoli

Area	Rect	Aspect Ratio	Orient
(800-12000)	0.75	0.16	(45-135)

## Creazione delle ROI

Arrivati a questo punto, il nostro sistema genera delle Regioni di Interesse (ROI) rettangolari attorno ad ogni rettangolo rimasto dopo la fase di filtraggio. Le ROI sono scalate di un fattore 2.5 rispetto alla larghezza del rettangolo, in modo tale da garantire che racchiudano anche la parte circolare del segnale, sebbene limitate dai bordi esterni delle immagini.

## Region Growing

In questa fase si itera sull'insieme di regioni di interesse (ROI), generando per ognuna di esse una sottoimmagine, successivamente convertita in scala di grigi e sfocata con un filtro gaussiano per ridurre il rumore. Per ogni ROI viene creata una corrispondente matrice di semi, collocati in punti specifici intorno al rettangolo corrente, rispettando i limiti dell'immagine. Questi punti vengono selezionati dell'intorno immediato

del rettangolo, posizionandoli con sufficiente sicurezza nella zona rossa del cartello.

Successivamente, viene chiamata la funzione `RegionGrowingHSV` per effettuare la segmentazione. Questa funzione converte l'immagine d'ingresso nello spazio colore HSV e separa i canali H, L e S. Il canale H viene segmentato utilizzando intervalli di colore predefiniti, creando una maschera binaria, mentre una soglia adattiva viene applicata al canale S per ottenere una seconda maschera binaria. La combinazione delle due maschere porta alla selezione dei punti dell'immagine con valori di Hue e saturazione compresi tra quelli plausibili per il cartello stradale cercato. I semi vengono iterativamente dilatati e combinati con la maschera ottenuta, fin quando non ci sono più cambiamenti significativi o fino al raggiungere un massimo di 60 iterazioni (numero che si è dimostrato empiricamente sufficiente per la ricostruzione completa dei segnali). Se la porzione di area segmentata supera una certa soglia, la ROI corrente viene considerata un candidato valido e aggiunta a una lista di regioni candidate.



**Figura 2:** Da sinistra verso destra: *seeds*, segmentazione per saturazione, segmentazione per valore di hue, risultato finale del *region growing*

## Rilevamento dei Cerchi

Ogni ROI viene analizzata singolarmente per individuare i pattern circolari. La rappresentazione in scala di grigi della ROI viene sfocata utilizzando un filtro Gaussiano per ridurre il rumore. La Trasformata di Hough per cerchi è una tecnica robusta per il rilevamento di forme circolari in un'immagine. Viene utilizzata per identificare i cerchi all'interno di ciascuna ROI, basandosi su parametri specifici che guidano l'algoritmo nella ricerca. Questi parametri includono: Distanza minima tra i cerchi, valore minimo e massimo per il raggio del cerchio e il valore di soglia più alto per l'algoritmo di segmentazione Canny.

**Tabella 2:** Criteri per il Rilevamento dei Cerchi

minDist	Canny	Accumulatore	Raggio
120	100	18	(20-150)

I cerchi rilevati vengono validati per garantire che contengano all'interno il rettangolo precedentemente rilevato. Se viene rilevato almeno un cerchio con queste proprietà viene creata una nuova ROI quadrata di dimensioni pari a 1.5 per la larghezza del rettangolo.

## Estrazione delle Feature

Gli Histogram of Oriented Gradients (HOG) descriptors sono ampiamente utilizzati per catturare informazioni sui gradienti di intensità di un'immagine, descrivendo la struttura e le forme degli oggetti presenti.

Ogni ROI viene ridimensionata a una dimensione standard di 80x80 pixel per facilitare un'analisi uniforme e ridurre la variabilità. I descrittori HOG vengono quindi calcolati sull'immagine ridimensionata con parametri specifici: dimensione della finestra di rilevamento, dimensione dei blocchi, passo dei blocchi, dimensione della cella e numero di bin per l'istogramma delle direzioni dei gradienti. Viene così prodotto un vettore di features che rappresentano i gradienti di intensità e le loro orientazioni nelle celle dell'immagine. Per garantire la comparabilità tra immagini diverse, i descrittori HOG vengono normalizzati nell'intervallo  $[0, 1]$ .

## Machine Learning

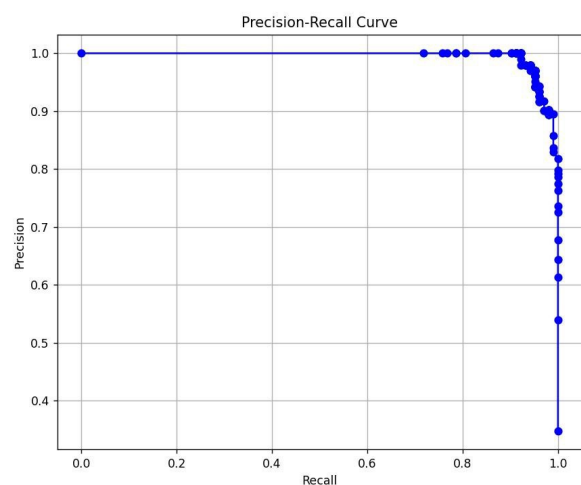
Le HOG (Histogram of Oriented Gradients) features descritte nel paragrafo precedente sono state utilizzate come input per un programma a riga di comando che implementa una Support Vector Machine (SVM). La SVM è uno dei metodi più performanti di Machine Learning per la classificazione binaria. In particolare, abbiamo utilizzato la modalità di cross validation, in cui in dataset viene suddiviso in due set. Un set viene utilizzato per il training, e il modello risultante viene validato sul secondo. Il risultato finale è un confidence score per ogni immagine classificata come True Positive e False Positive.

Questo punteggio di confidenza ci permette di valutare con maggiore precisione l'affidabilità delle classificazioni effettuate dal modello, consentendo ulteriori analisi e ottimizzazioni del sistema di rilevamento.

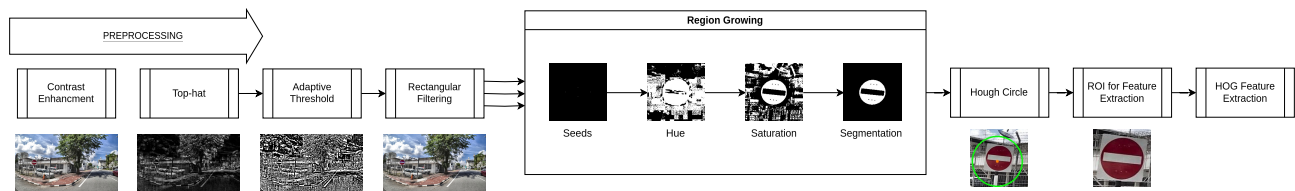
I risultati finali sono stati utilizzati per realizzare un riscontro visivo delle performance del nostro sistema. Abbiamo circondato ogni cartello da una bounding box verde o rossa (true positive o false positive) affiancata dallo score ottenuto.

## Risultati

Con i confidence score ottenuti abbiamo costruito una precision-recall curve [figura 4]. Una Precision-Recall Curve è uno strumento grafico utilizzato per valutare le prestazioni di un modello di classificazione binaria, in particolare in contesti con classi sbilanciate. La precisione (precision) misura la proporzione di veri positivi rispetto al totale delle previsioni positive, mentre il richiamo (recall) misura la proporzione di True Positive rispetto al totale dei Real Positive. La curva viene tracciata rappresentando la precisione sull'asse y e il richiamo sull'asse x, variando la soglia di decisione del modello. Una Precision-Recall Curve aiuta a identificare il trade-off ottimale tra precisione e richiamo e a valutare l'efficacia del modello soprattutto quando l'obiettivo è massimizzare i veri positivi minimizzando i falsi positivi.



**Figura 4:** Precision-Recall Curve con area sotto la curva di 0.994



**Figura 3:** Pipeline completa del nostro sistema. Costituito da una prima fase di Preprocessing, una successiva fase di ROI Creation che verranno date in input ad un sottosistema di Region Growing e, dopo l'ausilio di Hough Circle, le ROI risultanti serviranno da base per la HOG Feature Extraction.

## Cassino dataset

Come ulteriore sfida all'efficienza e all'affidabilità del nostro sistema, abbiamo costruito personalmente un piccolo dataset composto da 25 immagini raffiguranti segnali di senso unico presenti nella nostra città, Cassino. Questo dataset è stato accuratamente progettato per includere una varietà di condizioni reali, come diverse illuminazioni, angolazioni, e possibili ostruzioni, al fine di testare la capacità del sistema di adattarsi a nuove e potenzialmente difficili situazioni.

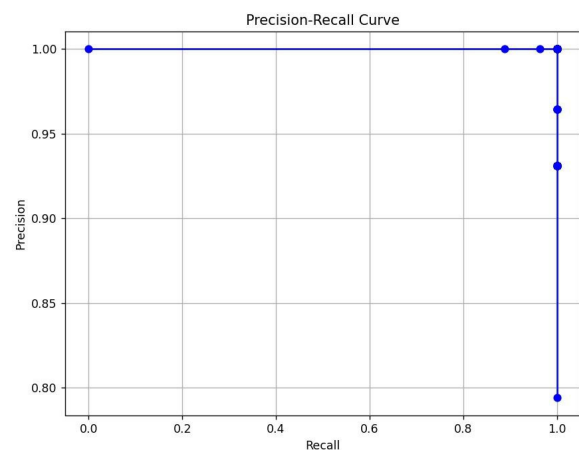
Quando abbiamo sottoposto il nostro sistema a queste nuove immagini, esso ha dimostrato un livello notevole di funzionamento e robustezza, riuscendo a rilevare correttamente tutti i segnali di senso unico presenti nel dataset. Questo risultato positivo sottolinea la capacità del nostro sistema di non solo operare con dati predefiniti, ma anche di generalizzare efficacemente su nuovi dati che non erano stati precedentemente visti.

La precisione del sistema nel rilevare i segnali stradali in diverse condizioni reali di Cassino conferma il potenziale del nostro approccio, dimostrando di essere efficace nel distinguere i segnali di senso vietato dal background, nonostante le variazioni in condizioni di illuminazione e la presenza di altri elementi visivi nel contesto urbano.

Inoltre, il successo del sistema su questo dataset costruito ad hoc suggerisce che esso potrebbe essere ulteriormente esteso e adattato per riconoscere altri tipi di segnali stradali e simboli, rendendolo uno strumento versatile e utile per diversi tipi di applicazioni usate in scenari del mondo reale dove l'accuratezza e l'affidabilità sono cruciali.

## Cassino Results

Anche per quanto riguarda questo dataset, abbiamo utilizzato il programma di machine learning, questa volta in modalità test, ottenendo così la seguente Precision-Recall Curve [figura 5]. Durante la fase di test, il modello addestrato è stato applicato a un insieme di dati non precedentemente visto, permettendoci di valutare le sue prestazioni in condizioni realistiche. La Precision-Recall Curve ottenuta mostra la relazione tra la precisione e il richiamo del modello su questo specifico dataset di test.



**Figura 5:** Precision-Recall Curve sul dataset di Cassino, con area unitaria sotto la curva

## Riferimenti bibliografici

- [1] Forbes. *Distracted Driving Statistics and Facts In 2024*. <https://www.forbes.com/advisor/legal/auto-accident/distracted-driving-statistics/>.

- [2] Dan Ciregan, Ueli Meier e Jürgen Schmidhuber. "Multi-column deep neural networks for image classification". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 3642–3649. DOI: 10.1109/CVPR.2012.6248110.
- [3] Wikipedia contributors. *Support vector machine* — *Wikipedia, The Free Encyclopedia*. [Online; accessed 13-July-2024]. 2024. URL: [https://en.wikipedia.org/w/index.php?title=Support\\_vector\\_machine&oldid=1230931205](https://en.wikipedia.org/w/index.php?title=Support_vector_machine&oldid=1230931205).