Individuazione dei Bordi Strada o della Corsia di Marcia mediante Computazione Massivamente Parallela*

Giovanni Adorni, Massimo Bertozzi, Alberto Broggi

Dipartimento di Ingegneria dell'Informazione
Università di Parma, Viale delle Scienze
43100, Parma

Sommario

Questo lavoro descrive un sistema per l'individuazione dei bordi di una strada (oppure di una corsia di marcia) in grado di operare in tempo reale su sequenze di immagini acquisite da telecamera posta a bordo di un autoveicolo. Caratteristica principale dell'approccio utilizzato è quella di eseguire elaborazioni "locali" sulle immagini al fine di trarre il massimo vantaggio dall'architettura hardware a parallelismo massivo, PAPRICA, utilizzata nella realizzazione di un prototipo dimostrativo istallato a bordo del laboratorio mobile MOB-LAB.

Massively Parallel Road/Lane Detection*

Giovanni Adorni, Massimo Bertozzi, Alberto Broggi

Dipartimento di Ingegneria dell'Informazione
Università di Parma, Viale delle Scienze
43100, Parma

Abstract

This paper describes an approach to real-time road/lane detection from image sequences acquired from a vehicle running on extra urban roads. The distinguishing characteristic of the approach is that the algorithm is suited for implementation on PAPRICA massively parallel architecture. The real time road/lane detecion system has been tested on the MOB-LAB experimental vehicle.

*Questo lavoro è stato finanziato dal "Progetto Finalizzato Trasporti II" del CNR, nell'ambito dell'obbiettivo PROMETHEUS, e dal progetto MURST 40% "Sistemi Intelligenti".

1 Introduzione

In questi ultimi anni sono sorte diverse iniziative di ricerca a livello internazionale con l'obbiettivo di migliorare il traffico stradale e incrementare la sicurezza nella guida di veicoli. Tali iniziative sono state rivolte verso l'utilizzo delle "tecnologie dell'informazione" sia per il miglioramento della sicurezza delle infrastrutture (si veda,ad esempio il progetto AMTICS [Oka 88] in Giappone, il programma Mobility 2000 [WGA 90] negli USA e il progetto europeo DRIVE [CEC 91]), che per il miglioramento della sicurezza dei veicoli. In questo ultimo ambito la Comunità Europea ha avviato a partire dalla metà degli anni '80 il programma di ricerca PROMETHEUS per migliorare la sicurezza dei veicoli. Il presente lavoro descrive alcuni risultati ottenuti durante lo svolgimento del progetto PROMETHEUS, con particolare riferimento agli obbiettivi PRO-ART e PRO-CHIP riguardanti un sistema per l'ausilio alla guida che permette di determinare la posizione di un veicolo rispetto alla carreggiata (oppure rispetto alla corsia di marcia) elaborando sequenze di immagini acquisite da una telecamera installata a bordo dello stesso veicolo.

In letteratura sono riportati diversi sistemi per l'ausilio alla guida di questo genere sviluppati per operare in ambienti non strutturati [Joc 93] (strade senza segnaletica orizzontale), o per navigazione al di fuori di ambienti stradali [Sch 93, Met 93]. La maggioranza delle ricerche sono però state rivolte all'individuazione della corsia di marcia (o dei bordi strada) tramite il riconoscimento della segnaletica orizzontale come, ad esempio, le linee di mezzeria. In entrambi i casi l'introduzione di conoscenza sull'ambiente esterno può agevolare l'algoritmo di riconoscimento e/o accelerare l'intero processo. Tale conoscenza può riguardare, ad esempio, le posizioni in cui è maggiore la probabilità di trovare le linee di segnaletica [Dic 92], la configurazione della strada [Pom 95, Rui 95, Brg 95] oppure la particolare forma geometrica (clotoidi) della strada [Dic 92, Kol 95, Web 95].

Nella letteratura del settore è ormai classica la distinzione tra approcci guidati dai dati ed approcci guidati dalle aspettative [Bal 82]. La differenza tra le due metodologie è nel tipo di strategia di controllo utilizzata dal processo di visione: nel primo caso, il tipo di elaborazione da effettuare viene deciso in base alle caratteristiche di ogni singola immagine; nel secondo caso, invece, è il tipo di risultato desiderato (le aspettative) a guidare il processo di elaborazione. In entrambi i casi si rende comunque necessaria una fase di elaborazione di *basso livello*. Tali elaborazioni sono caratterizzate dal fatto che ogni elemento dell'immagine (*pixel*) viene modificato secondo una regola fissa, in modo sincrono con tutti gli altri pixel dell'immagine, e comunque dipendente solo da un proprio intorno. La struttura hardware (topologia) delle architetture a parallelismo massivo con connessione a griglia bidimensionale (MC2 - "two dimensional mesh connected architectures") è particolarmente adatta per ottimizzare tali operazioni.

Il sistema presentato in questo lavoro è basato su di un approccio guidato dai dati e controllato dalla conoscenza sul mondo esterno, e trae vantaggio da una particolare architettura MC2: PAPRICA [Bro 94]. L'originalità del sistema consiste nell'utilizzo di un particolare algoritmo basato quasi esclusivamente su elaborazioni di basso livello dell'immagine. Tale algoritmo è particolarmente adatto ad essere implementato su architetture MC2 e quindi anche sul sistema PAPRICA.

L'intero sistema è stato sperimentato a bordo del laboratorio mobile MOB-LAB, sviluppato nell'ambito del progetto PROMETHEUS [Ado 88, Ado 96]. Tale laboratorio mobile è dedicato allo sviluppo e verifica di sistemi per la visione artificiale, operanti in tempo reale, orientati al controllo della traiettoria di un veicolo viaggiante su strade extraurbane.

Nel paragrafo successivo viene fornita una descrizione dell'architettura dell'intero sistema. Nel paragrafo 3 viene descritto l'algoritmo e la teoria su cui esso si basa, mentre nel paragrafo 4 vengono discussi i risultati ottenuti durante le prove conclusive del progetto PROMETHEUS.

2 Il Sistema

L'architettura di riferimento del sistema descritto in questo lavoro, PAPRICA (PArallel PRocessor for Image Checking and Analysis) [Bro 94, Bro 96], è un'architettura massivamente parallela di tipo SIMD progettata per essere collegata come co-processore dedicato alle applicazioni di morfologia matematica ad una stazione di lavoro di tipo "general purpose". PAPRICA è integrata su una singola scheda VME connessa ad una workstation di tipo Sparc, ed è composta da 5 moduli funzionali principali:

- la Memoria di Programma (contenente fino a 256 kistruzioni);
- la Memoria di Immagine (contenente fino a 8 Mbytes di dati);
- la Matrice di Processori (contenente 16x16 Elementi di Elaborazione);
- l'Unità di Controllo;
- un Dispositivo di Acquisizione di immagini interfacciato direttamente con la memoria immagine, che consente anche di visualizzare i risultati su monitor.

La Matrice di Processori è composta da 16×16 Elementi di Elaborazione (EE) ad 1 bit disposti sui nodi di una griglia bidimensionale, ognuno dei quali è direttamente connesso con i propri 8 EE primi vicini; il tempo di ciclo è 100 ns. Ogni EE ha una memoria interna di 64 bit usati per la memorizzazione dei risultati intermedi. In generale il numero di EE è risulta essere minore del numero dei pixel delle immagini da elaborare, e per questo motivo il meccanismo di funzionamento di PAPRICA è basato sulla *virtualizzazione* della matrice dei processori [Brg 94]. L'immagine da elaborare è divisa in finestre che vengono caricate dalla memoria immagine nella memoria degli EE, elaborate e i risultati riscritti nella memoria immagine. Durante il trasferimento di ogni singola finestra dalla memoria immagine alla matrice degli EE si può scegliere il modo di indirizzamento in memoria. Più precisamente se la finestra è formata da pixel adiacenti viene eseguita una elaborazione di tipo *tradizionale*, viceversa se il modo di indirizzamento in memoria immagine seleziona pixel non adiacenti (per esempio 1 su 2, 1 su 3, ...) l'elaborazione avviene su di un'immagine sottocampionata. Mutualmente è possibile eseguire sovracampionamenti, permettendo quindi un funzionamento di tipo piramidale [Can 93, Tan 80].

PAPRICA è utilizzata come il supporto per le elaborazioni di basso livello di un sistema in grado di individuare i bordi di una strada (o di una corsia di marcia) da sequenze di immagini acquisite da un autoveicolo in movimento.

La filosofia dell'approccio realizzato è la seguente: due processi di elaborazione, uno a basso livello ed uno ad alto livello, che cooperano per la soluzione di un compito prefissato. Il processo a basso livello, guidato essenzialmente dai dati, viene controllato dalla conoscenza sulle scene all'interno delle quali ci stiamo muovendo (processo ad alto livello e guidato dalle aspettative). Per esempio, se sappiamo che il tempo è "piovoso", e quindi l'asfalto è bagnato, possiamo utilizzare questa informazione per regolare opportunamente i parametri di alcuni filtraggi a basso livello. Se poi sappiamo che stiamo percorrendo una strada a doppio senso di marcia a due corsie, e vogliamo individuare i bordi della corsia di marcia, possiamo guidare il processo di elaborazione fornendo al sistema la forma “presunta” della corsia, e cioè segnalando il loro numero. Questo particolare tipo di conoscenza viene codificata nel sistema in termini di immagini sintetiche bidimensionali come quelle di figura 1.

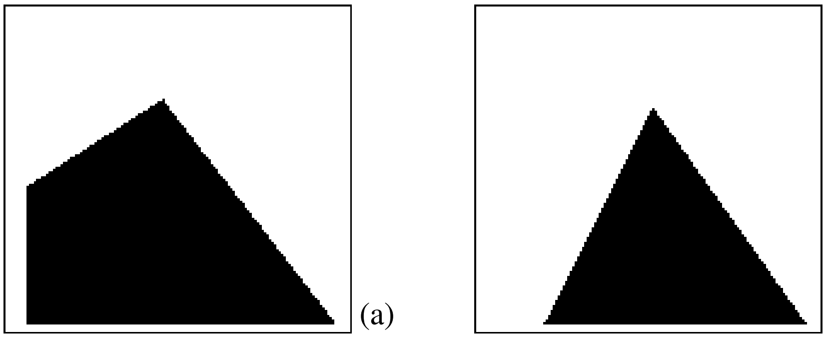

Figura 1: Esempi di immagini sintetiche bidimensionali che codificano a) il modello di una strada a doppio senso di marcia, e b) il modello di una corsia per una strada a più corsie

Le immagini sintetiche vengono confrontate con le immagini naturali seguendo lo schema di figura 2, dove i numeri sulle frecce rappresentano la profondità delle immagini in bit/pixel. Inizialmente viene scelta un’immagine sintetica dalla base di dati delle immagini sintetiche in funzione della conoscenza esterna. L’algoritmo di individuazione bordi (descritto nella sezione successiva) permette di *adattare* un’immagine sintetica iniziale che rappresenta una ipotesi di forma della strada/corsia di marcia, alla strada/corsia di marcia contenuta nell’immagine naturale. L’immagine sintetica adattata, risultante dall’analisi di una singola immagine di una sequenza, viene utilizzata, sia per il confronto con le immagini successive della sequenza stessa, che per il calcolo dela posizione del veicolo rispetto al centro della strada ovvero della corsia di marcia. Quest’ultima informazione viene fornita al guidatore tramite un pannello di controllo composto da led colorati.

Il confronto tra immagine sintetica e immagine naturale non può però essere direttamente eseguito tramite operazioni locali, in quanto le immagini naturali contengono molti più dettagli di quelle sintetiche. Inoltre, i bordi di una strada, così come le delimitazioni di una corsia di marcia, rappresentano caratteristiche con correlazione a lunga distanza. In questi casi il riconoscimento di tali caratteristiche risulta in generale facilitato dall’utilizzo di immagini a bassa risoluzione spaziale rispetto ad immagini ad alta risoluzione spaziale in cui possono apparire dettagli fuorvianti. Ciononostante, la verifica e la descrizione completa delle caratteristiche individuate necessitano di dettagli presenti solo in immagini ad alta risoluzione.

Queste considerazioni suggeriscono la scelta di una struttura dati piramidale, in cui l’immagine possa essere codificata a differenti livelli di risoluzione. In tal caso, l’individuazione di caratteristiche viene effettuata a basse risoluzioni, mentre il riconoscimento è raffinato durante il passaggio a risoluzioni maggiori [Bro 95].

3 L’Algoritmo Piramidale

La struttura dell’algoritmo è illustrata in figura 3, dove i numeri in ingresso ed in uscita ad ogni blocco rappresentano la profondità dell’immagine in bits/pixel. L’immagine naturale e l’imma-

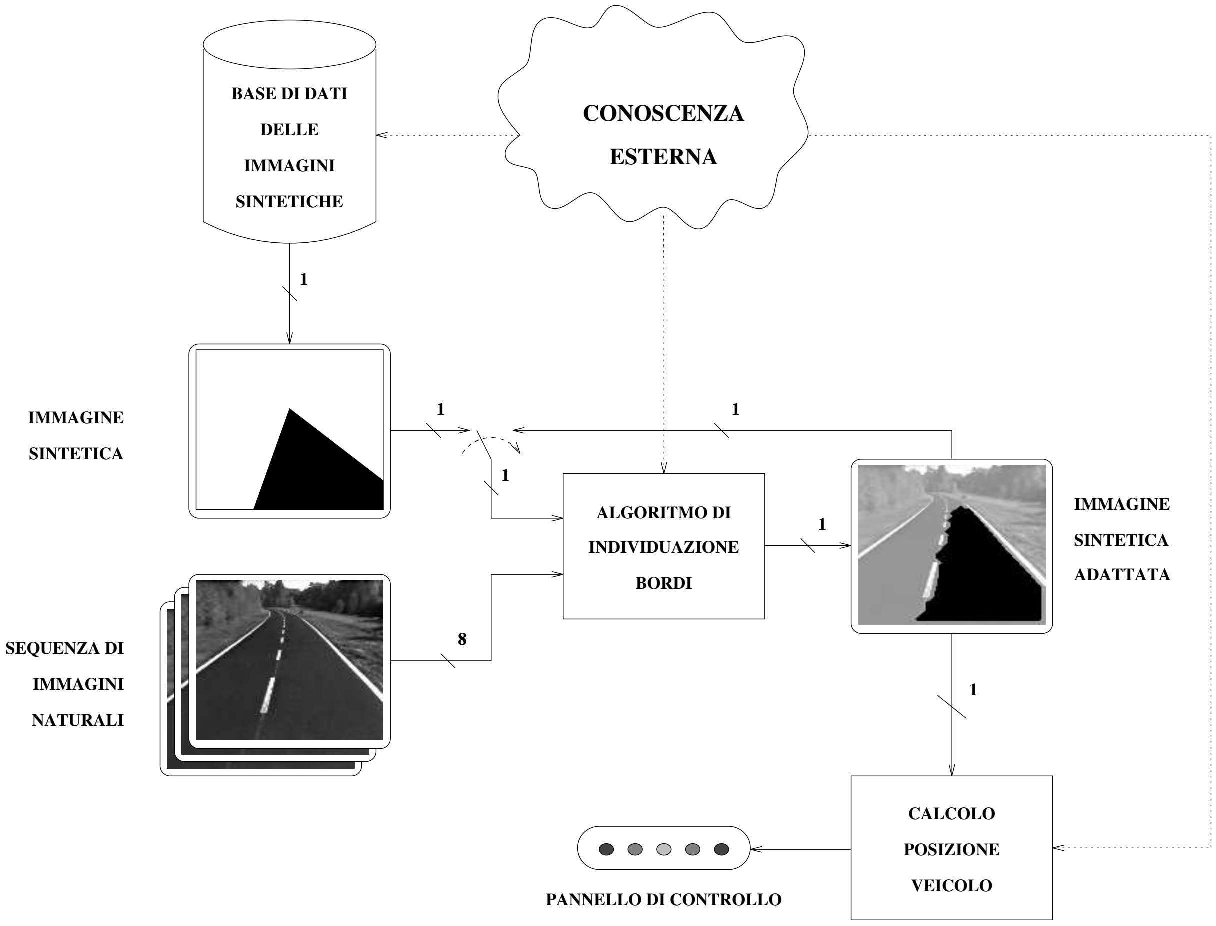

Figura 2: Diagramma di flusso del processo di individuazione dei bordi strada o della corsia di marcia

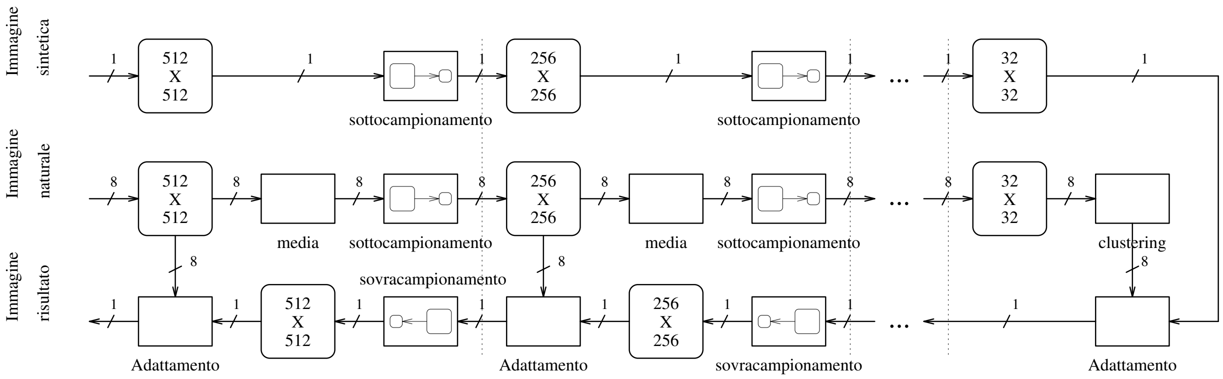

Figura 3: Diagramma di flusso dell'algoritmo piramidale

gine sintetica seguono inizialmente due flussi di elaborazione distinti fino al raggiungimento del livello di risoluzione più basso prescelto, attraverso processi di sottocampionamento [Ado 95].

Prima di ogni sottocampionamento, l'immagine naturale viene filtrata attraverso un filtro passa-basso (filtro di media su di un vicinato 2×2). In questo modo è possibile ridurre l'influenza del rumore, delle informazioni ridondanti, e delle distorsioni dovute a fenomeni di *aliasing* introdotti dal processo di sottocampionamento stesso. Una volta raggiunto il livello di risoluzione prescelto (32×32 nel caso di figura 3) i "bordi" dell'immagine naturale vengono evidenziati mediante applicazione iterativa di un algoritmo di *clustering* [Bro 93]. Successivamente, il risultato dell'analisi viene confrontato con il flusso proveniente dall'immagine sintetica attraverso un "algoritmo di adattamento". Il risultato dell'analisi viene quindi sovracampionato e successivamente migliorato utilizzando lo stesso algoritmo di adattamento. Questi passi dell'analisi vengono iterati fino a ritornare alla risoluzione dell'immagine di partenza (512×512 nel caso di figura 3).

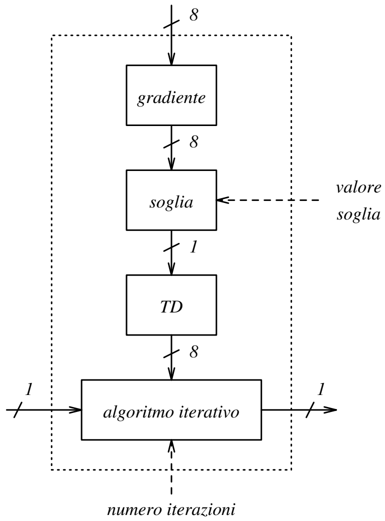

Figura 4: Diagramma di flusso dell'algoritmo di adattamento

L'algoritmo di adattamento (vedi figura 4) ha il compito di "deformare" l'immagine sintetica in modo da adattarla ad una immagine naturale e di fornire come risultato dell'analisi una nuova versione dell'immagine sintetica che verrà utilizzata per l'analisi delle immagini successive della sequenza. Per capire a fondo il comportamento di tale algoritmo, occorre ricordare che l'immagine originale è codificata in termini di livelli di grigio, e che in tale contesto si suppone che il contorno di un qualunque oggetto sia rappresentato da discontinuità di luminanza nell'immagine che lo rappresenta. Viene quindi calcolata una immagine gradiente, a cui è applicata una soglia per evidenziare solamente i bordi più significativi. Per poter adattare l'immagine sintetica all'immagine sogliata (binaria) viene utilizzato un algoritmo che iterativamente sposta ogni pixel del bordo dell'immagine sintetica verso la posizione del pixel più vicino nell'immagine sogliata . Per tale motivo, usando la metrica *chessboard*, si calcola la "trasformata distanza" (TD) dell'immagine gradiente binarizzata che fornisce punto per punto la distanza dai pixel rimasti dopo l'operazione di soglia [Bor 86],

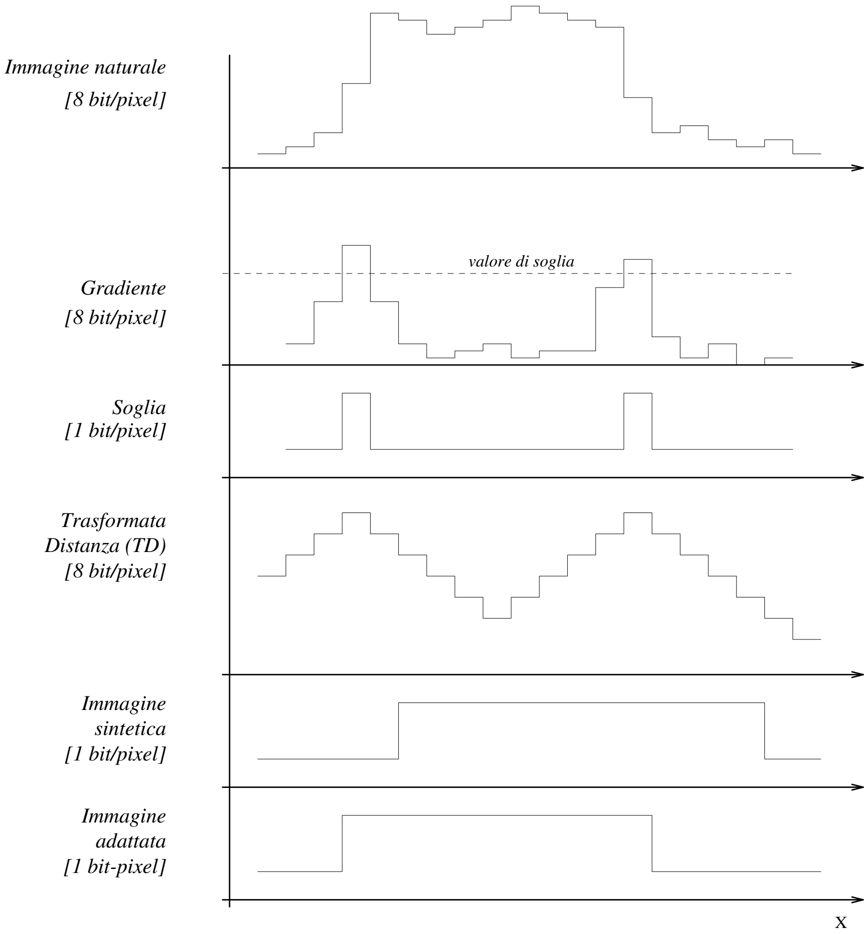

Figura 5: Esempio dell'algoritmo di adattamento nel caso di immagini monodimensionali con 4 iterazioni

Per descrivere l'algoritmo di adattamento richiamiamo brevemente alcuni concetti derivati dalla Morfologia Matematica [Ser 82]. L'algoritmo di adattamento è basato sull'applicazione successiva dei due operatori morfologici binari di espansione ($\oplus$) e di erosione ($\ominus$), utilizzando il seguente elemento strutturante N:

$$N = \Big\{(-1,0),(1,0),(0,0),(0,-1),(0,1)\Big\} = \begin{array}{|c|c|c|}\hline & \bullet & \\ \hline \bullet & \bullet & \bullet \\ \hline & \bullet & \\ \hline\end{array} \qquad (1)$$

Una immagine binaria bidimensionale S viene rappresentata come un sottoinsieme dello spazio euclideo a due dimensioni, i cui elementi corrispondono ai pixel dell'immagine, dove il vettore (x, y) rappresenta le coordinate del generico elemento s_i dell'immagine S. L'insieme degli elementi generati dalla espansione di S a seguito dell'applicazione dell'elemento strutturante N, viene chiamato *bordo esterno* di S:

$$\mathcal{B}_e(S) = (S \oplus N) \cap \overline{S} \tag{2}$$

In modo analogo, l'insieme degli elementi eliminati dall'insieme S per erosione, a seguito dell'applicazione dell'elemento strutturante N, viene chiamato *bordo interno* di S:

$$\mathcal{B}_i(S) = \overline{(S \ominus N)} \cap S \tag{3}$$

L'unione logica tra (3) e (4) rappresenta il *bordo* S:

$$\mathcal{B}(S) = \mathcal{B}_e(S) \cup \mathcal{B}_i(S) \tag{4}$$

Gli elementi di $\mathcal{B}(s)$ sono i soli che possono cambiare il proprio stato (da immagine a sfondo e viceversa) mediante l'applicazione di una singola iterazione dell'algoritmo di adattamento. Ai due tipi di bordi vengono applicate due regole specifiche. La prima regola, applicata al bordo esterno, determina quali elementi devono essere inclusi nell'insieme S, mentre la seconda regola, applicata al bordo interno, determina gli elementi che devono essere eliminati dall'insieme S. Tali regole vengono riportate nel seguito.

Regola per il bordo esterno:

- calcola per ogni pixel del bordo esterno di S, mediante TD, il valore minimo della distanza rispetto a i 4 primi vicini che appartengono all'insieme S;
- inserisci nell'insieme S tutti i pixel: (i) il cui valore di distanza calcolato mediante TD sia più grande del valore precedentemente calcolato, e (ii) che non appartengono all'immagine sogliata ed il cui valore di TD sia uguale al valore precedentemente calcolato.

Regola per il bordo interno:

- calcola per ogni pixel del bordo interno di S, mediante TD, il valore massimo della distanza rispetto ai 4 primi vicini che non appartengono all'insieme S;
- elimina dall'insieme S tutti i pixel il cui valore di distanza calcolato mediante TD sia più grande del valore precedentemente calcolato.

Applicando questo algoritmo all'immagine sintetica proveniente dalla base di dati, o a quella precedentemente elaborata, si può notare come il numero delle iterazioni necessarie per un adattamento non dipenda dalla risoluzione. Inizialmente, infatti, l'immagine è a bassa risoluzione; è sufficiente quindi un numero ridotto di iterazioni per adattare l'immagine sintetica all'immagine naturale. Muovendoci lungo la piramide (incrementando cioè la risoluzione) vengono sfruttati i risultati dell'adattamento già ottenuto alle risoluzioni più basse e, quindi, è nuovamente sufficiente un numero ridotto di iterazioni per raffinare il risultato precedente. Nella figura 5 è riportato un esempio di adattamento relativamente ad un caso monodimensionale.

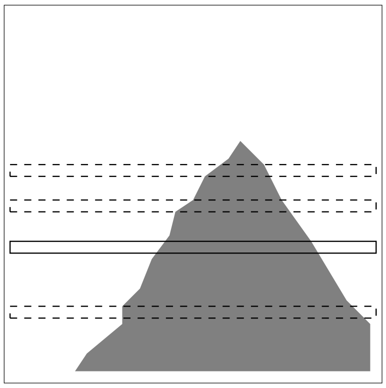

Figura 6: Porzioni dell'immagine adattata che vengono utilizzate per la determinazione della posizione del veicolo relativamente alla carreggiata o alla corsia di marcia

L'immagine risultante dalla fase di adattamento viene utilizzata per il calcolo della posizione del veicolo. Inizialmente il sistema di visione viene calibrato mediante l'acquisizione di immagini in condizioni note (strada piana e rettilinea); questo permette di ricavare la relazione spaziale che lega i pixel dell'immagine alla strada, e quindi anche la distanza di ogni porzione di strada rappresentata nell'immagine dalla telecamera [Mal 91]. Più precisamente, dall'analisi di alcune porzioni dell'immagine risultante dalla fase di adattamento si possono ottenere alcuni parametri, quali l'andamento della carreggiata o corsia di marcia rispetto all'orientamento del veicolo. Le porzioni di immagine analizzate sono quelle individuate da finestre orizzontali, come mostrato in figura 6. Ad ogni finestra vengono associate 2 coordinate, che corrispondono all'inizio e al termine della parte deformata che corrisponde alla carreggiata o alla corsia di marcia. Attualmente si utilizza una sola finestra (quella rappresentata in linea continua), ma l'algoritmo è stato concepito per generare un'immagine deformata completa, che quindi offre la possibilità di analizzare un maggior numero di finestre. Da un'analisi teorica si è notato che l'utilizzo di 4 finestre (ovvero con 8 parametri) è possibile determinare con precisione sia la posizione del veicolo rispetto alla carreggiata, sia la curvatura della stessa.

4 Le Prestazioni

Il sistema descritto nei paragrafi precedenti è stato sperimentato per il calcolo dei bordi strada o della corsia di marcia elaborando sequenze di immagini acquisite da veicolo in movimento su strade extraurbane. Più precisamente PAPRICA è stata installata a bordo del laboratorio mobile MOB-LAB in un cestello VME insieme al front-end (una stazione Sparc) ed ad una interfaccia A/D per convertire i risultati dell'analisi in segnalazioni su di un pannello di controllo posto sul cruscotto di guida. La stazione Sparc ha il compito di gestire il controllo di flusso e di ospitare i processi di elaborazione ad alto livello che decidono in funzione di dati provenienti dall'esterno quali informazioni fornire ai processi di basso livello operanti su PAPRICA. La sperimentazione del sistema a bordo del veicolo MOB-LAB (figura 8.a) è stata fatta durante le prove conclusive del progetto EUREKA PROMETHEUS sul circuito extraurbano di Mortefontaine (Francia) nell'ottobre 1994. Tale veicolo è basato su di un FIAT DUCATO equipaggiato con telecamere B/N e colore poste sia nella parte anteriore che nella parte posteriore del veicolo per l'analisi dell'ambiente stradale esterno, un radar a onde millimetriche per il controllo

Operazione		Dimensioni immagine	Tempo [ms]
Riduzione di Risoluzione		$256^2 \rightarrow 16^2$	155.7
TD, Adattamento & Sovracampionamento		$16^2 \rightarrow 32^2$	3,55
TD, Adattamento & Sovracampionamento		$32^2 \rightarrow 64^2$	18,55
TD, Adattamento & Sovracampionamento		$64^2 \rightarrow 128^2$	69,13
TD, Adattamento & Sovracampionamento		$128^2 \rightarrow 256^2$	296,87
Configurazione			
A	visualizzazione nel pannello di controllo	$256^2 \rightarrow 16^2 \rightarrow 64^2$	177,80
B	visualizzazione nel pannello di controllo più ricostruzione immagine da visualizzare	$256^2 \rightarrow 16^2 \rightarrow 256^2$	543,80

Tabella 1: Prestazioni dell'algoritmo sul sistema PAPRICA: i valori si riferiscono a 5 iterazioni del processo TD e altrettante iterazioni del filtro di adattamento per ogni livello della piramide

del bordo stradale destro, quattro accelerometri per la misura dell'accelerazione longitudinale e trasversale, due sistemi di elaborazione di immagini basati su architetture sia tradizionali che massivamente parallele, due personal computers per l'analisi dei dati provenienti dal radar e dagli accelerometri e un pannello di controllo (vedi figura 8.b). Mediante tale equipaggiamento il veicolo è in grado di svolgere funzioni di:

- Riconoscimento di ostacoli [Den 94] (*obstacle and road detection*);
- Controllo della distanza di sicurezza [Bra 95] (*cruise control*);
- Controllo sulla deriva laterale [Cug 93] (*lateral control*);
- Controllo sul sorpasso [Tis 95] (*overtaking control*);
- Controllo dell'assetto del veicolo [Cre 95] (*lane tracking control*).

Tali funzioni fanno riferimento al pannello di figura 8.b.

Durante le prove, in cui sono stati percorsi circa 1000 km, l'operatore forniva indicazioni al sistema sulla situazione meteorologica (sole/pioggia), sul numero di sensi di marcia della strada (senso unico/doppio senso), sul numero di corsie per senso di marcia (1/2/3), in modo tale da permettere al sistema di scegliere (e/o cambiare durante l'esecuzione) i parametri dei filtri e la forma dell'immagine sintetica da adattare. In questo modo veniva simulata l'acquisizione della conoscenza esterna. I risultati delle computazioni venivano forniti al guidatore tramite un pannello di controllo in cui erano visualizzate le informazioni relative alle funzionalità di cui sopra in forma discretizzata attraverso dei led. In particolare i risultati dell'elaborazione relativa all'individuazione dei bordi strada o corsia di marcia venivano forniti al guidatore tramite 5 led (rosso-giallo-verde-giallo-rosso) indicanti la posizione del MOB-LAB rispetto alla mezzeria della corsia di marcia. Verde: posizione più o meno coincidente con la mezzeria della corsia di marcia; Giallo: leggera deviazione a sinistra/destra rispetto alla mezzeria; Rosso: pericolo, deviazione eccessiva a sinistra/destra rispetto alla mezzeria. Tali informazioni erano determinate dal modulo "calcolo posizione veicolo" di figura 2.

In figura 7 sono mostrate alcune immagini ottenute durante le prove effettuate, la prima colonna rappresenta le immagini acquisite, mentre la seconda rappresenta le immagini sintetiche adattate sovrapposte alle immagini acquisite. Nella terza colonna è stata sovrapposta la schematizzazione delle indicazioni fornite dal pannello dei led all'immagine acquisita. Le prove effettuate hanno dimostrato che: nel 90 % dei casi si riesce ad adattare con successo l'immagine sintetica con 5 iterazioni. Le prove sono state condotte in condizioni di illuminazione costante, con ombre, con e senza ostacoli, in rettilineo e in curva. Si è notato che l'algoritmo ha probabilità di fallimento maggiori in presenza di ombre o strada eccessivamente curva (bordi della carreggiata non presenti nel campo visivo della telecamera).

In tabella 1 vengono riportati i tempi di esecuzione dell'algoritmo a vari livelli di risoluzione delle immagini in due configurazioni. Nella prima configurazione (A) al guidatore viene fornito unicamente lo scostamento del veicolo rispetto la mezzeria della corsia di marcia attraverso i cinque led presenti sul pannello di controllo. In questo caso, vista la discretizzazione imposta dal numero dei led sul pannello di controllo, non è necessario il totale recupero della risoluzione iniziale. Nella seconda configurazione (B) viene fornito al guidatore anche l'immagine elaborata sovrapposta all'immagine originale. In questo caso si rende necessario recuperare la risoluzione originaria che, nell'esempio riportato in tabella 1, è 256×256. Le prove nel circuito di Mortefontaine sono state effettuate ad una velocità di circa 60 km/h. Questo significa che nell'ipotesi di configurazione A veniva acquisita un'immagine ogni circa 3 m, mentre nell'ipotesi di configurazione B veniva acquisita un'immagine ogni circa 9 m. Secondo la *American Associaton of State Highway Officials* lo spazio di arresto alla velocità di 60 km/h, considerando anche il tempo di reazione del guidatore (circa 2,5 s), risulta essere di circa 90 m su strada piana e bagnata, e circa 80 m su strada piana e asciutta [Pig 73]. Il sistema permetteva quindi di fornire all'utente informazioni in tempo utile per farlo intervenire in condizioni di ragionevole sicurezza.

È in fase di test la nuova versione del sistema PAPRICA [Bro 96] in cui sono state ottimizzate le operazioni di accesso alla Memoria Immagine ed è stata aumentata la frequenza di clock dei processori migliorando le prestazioni dell'intero sistema di un fattore stimato 2,5.

5 Ringraziamenti

Gli autori ringraziano tutti i ricercatori delle unità PROMETHEUS PRO-ART e PRO-CHIP senza il cui contributo non sarebbe stato possibile la dimostrazione delle funzionalità del MOB-LAB. Un ringraziamento particolare va a Giorgio Quaglia e ai suoi collaboratori dell'Istituto Elettrotecnico Nazionale "Galileo Ferraris" di Torino per l'allestimento del MOB-LAB. Gli autori sono infine in debito con Francesco Gregoretti, Claudio Sansoé, Leonardo Reyneri del Politecnico di Torino e con Gianni Conte dell'Università degli Studi di Parma per lo sviluppo del sistema PAPRICA.

Riferimenti bibliografici

[Ado 88] G. Adorni. Artificial Intelligence for a Safer and More Efficient Car Driving. Atti del SPIE - Mobile Robots III, Cambridge, Regno Unito, Novembre 1988.

[Ado 95] G. Adorni, M. Bertozzi, e A. Broggi. Riconoscimento della Corsia di Marcia mediante Tecniche Massivamente Parallele. *Atti del Secondo Convegno Nazionale Progetto Finalizzato Trasporti 2*, CNR, Roma, Italia, Maggio 29-31 1995.

[Ado 96] G. Adorni, A. Broggi, G. Conte, e V. D'Andrea. Real-Time Image Processing for Automotive Applications. In P. A. Laplante e A. D. Stoyenko, editori, *Real-Time Imaging: Theory, Techniques, and Applications*, pagine 161–194.

[Bal 82] D. H. Ballard e C. M. Brown. *Computer Vision*. Prentice Hall, 1982.

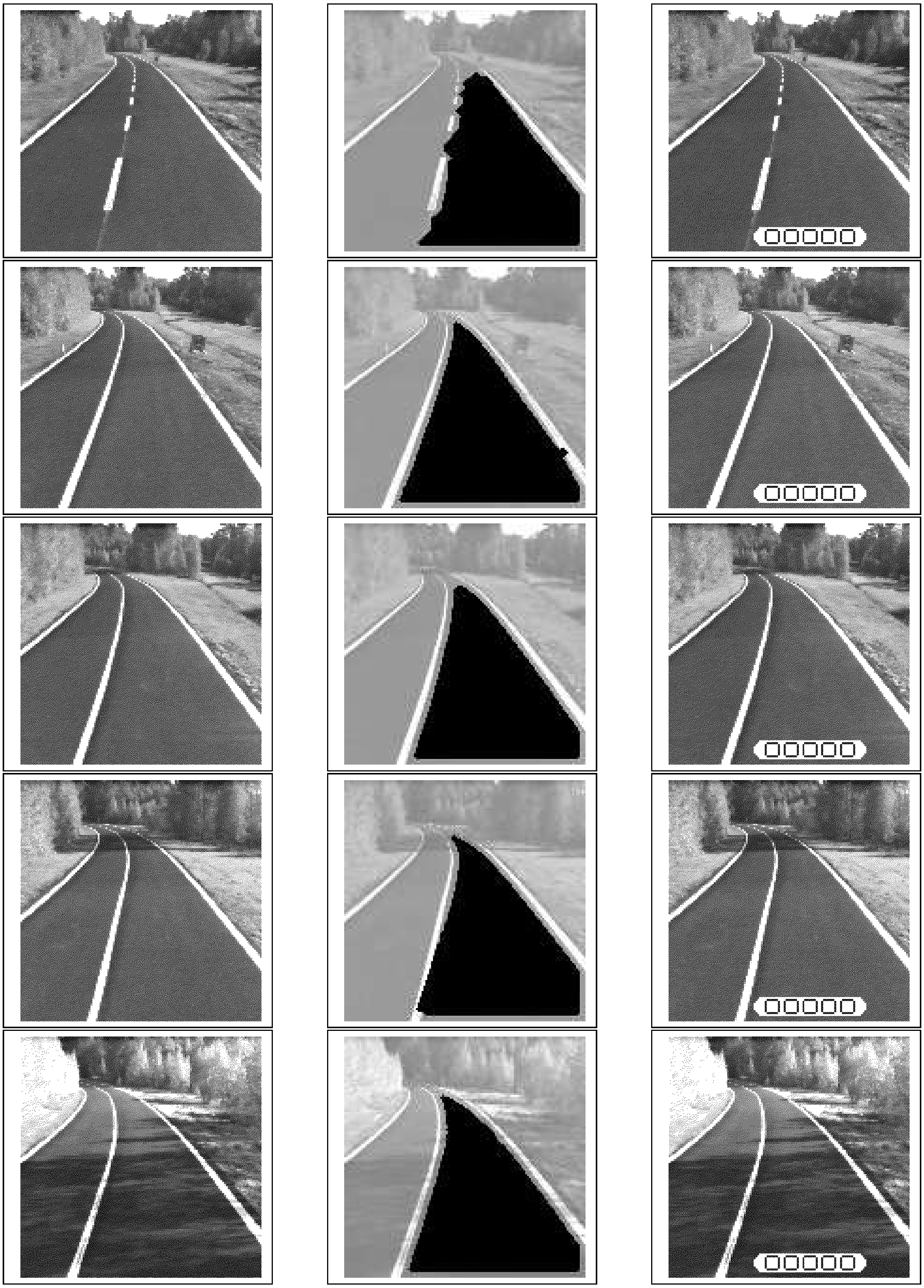

Figura 7: Risultato dell'elaborazione di una sequenza di immagini acquisite a bordo del MOB-LAB

Figura 8: (a) Il veicolo sperimentale MOB-LAB e (b) il pannello di controllo

[Bro 95] A. Broggi e S. Bertè. Vision-Based Road Detection in Automotive Systems: a Real-Time Expectation-Driven Approach. *Journal of Artificial Intelligence Research*, 3:325–348, Dicembre 1995.

[Bro 94] A. Broggi, G. Conte, F. Gregoretti, C. Sansoè, e L. M. Reyneri. The PAPRICA Massively Parallel Processor. *Atti del MPCS*, pagine 16–30, Ischia, Italia, Maggio 2-6 1994.

[Bro 96] A. Broggi, G. Conte, F. Gregoretti, C. Sansoè, e Leonardo M. Reyneri. The Evolution of the PAPRICA System. *Integrated Computer-Aided Engineering Journal - Special Issue on Massively Parallel Computing*, 3(4), 1996.

[Bro 93] A. Broggi e A. Gandini. Parallel Image Clustering: A Real-Time Application for a Special-Purpose Architecture. In K. A. Hogda, B. Braathen, e K. Heia, editori, *Atti del* 8^{o} *SCIA*, volume 1, pagine 297–304, Tromsoe, Norvegia, Maggio 25-28 1993.

[Bra 95] C. Braccini, A. Grattarola, e L. Crovato. Cruise control. *Atti del 2nd. National Conference of PFT2*, pagine 255–2555, Genova, Italia, Maggio 1995.

[Bor 86] G. Borgefors. Distance Transformations in Digital Images. In *Computer Vision, Graphics and Image Processing*, volume 34, pagine 344–371, 1986.

[Brg 94] A. Broggi. Performance Optimization on Low-Cost Cellular Array Processors. *Atti del MPCS*, pagine 334–338, Ischia, Italia, Maggio 2-6 1994.

[Brg 95] A. Broggi. Robust Real-Time Lane and Road Detection in Critical Shadow Conditions. *Atti del International Symposium on Computer Vision*, Coral Gables, Florida, Novembre 19-21 1995.

[Cre 95] D. Crescini, G. De Cicco, D. Febbrari, D. Marioli, A. Masoero, B. Morten, M. Prudenziati, E. Sardini, e A. Taroni. Sistema Multisensoriale per il Rivelamento delle Componenti dell'Accelerazione del Veicolo. *Atti del 2o. National Conference of PFT2*, pagine 2511–2522, Genova, Italia, Maggio 1995.

[Can 93] V. Cantoni e M. Ferretti. *Pyramidal Architectures for Computer Vision*. Plenum Press, London, 1993.

[Cug 93] C. Cugiani, L. Giubbolini, e R. Tascone. Millimeter-wave guard rail tracking system for the vehicle lateral control. *Atti del IEEE 4rt.Int. Conf. on Vehicle Navigation and Information System*, pagine 521–524, Ottawa, Canada, Ottobre 1993.

[Den 94] S. Denasi, C. Lanzone, P. Martinese, G. Pettiti, G. Quaglia, e L. Viglione. Real-time system for road following and obstacle detection. *Atti del SPIE on Machine Vision Applications, Architectures, and Systems Integration III*, pagine 70–79, Boston, MA, Ottobre 1994.

[Dic 92] E. D. Dickmans e B. D. Mysliwetz. Recursive 3-D Road and Relative Ego-State Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:199–213, Maggio 1992.

[Joc 93] T. M. Jochem, D. A. Pomerleau, e C. E. Thorpe. MANIAC: A Next Generation Neurally Based Autonomous Road Follower. *Atti del International Conference on Intelligent Autonomous Systems: IAS -3*, Pittsburgh, Pennsylvania, USA, Febbraio 1993.

[Kol 95] D. Koller, J. Malik, Q. T. Luong, e J. Weber. An integrated stereo-based approach to automatic vehicle guidance. *Atti del Fifth ICCV*, pagine 12–20, Boston, 1995.

[Mal 91] H. A. Mallot, H. H. Bülthoff, J. J. Little, e S. Bohrer. Inverse perspective mapping simplifies optical flow computation and obstacle detection. *Biological Cybernetics*, 64:177–185, 1991.

[Met 93] L. E. Mettala e O. Firschein. Reconnaissance, surveillance and target acquisition research for the unmanned ground vehicle program. *Atti del International Conference on Intelligent Autonomous Systems*, pagine 257–279, 1993.

[WGA 90] Working Group on ATMS. Advanced Traffic Management System - Mobility 2000. In *Final Report, The National Intelligent Vehicle Higway System Workshop*, Dallas, Texas, 1990.

[Oka 88] H. Okamoto. Advanced mobile traffic information and communications systems (AMTICS). *Atti del Annual Meeting of Transportation Research Board*, Washington, D.C., 1988.

[CEC 91] Commission of the European Communities. R+D in advanced road transport telematics in Europe - DRIVE'91. *Report DRI202*, DRIVE Central Office, Brussells, 1991.

[Pig 73] L. J. Pignataro. *Traffic Engineering*. Prentice-Hall Inc., Englewood Cliffs, N.J., 1973.

[Pom 95] D. A. Pomerleau. RALPH: Rapidly Adapting Lateral Position Handler. I. Masaky, editore, *Proceedings Intelligent Vehicles'95*, pagine 506–511, Detroit, September 25-26 1995.

[Rui 95] Y. Ruichek e J. G. Postaire. Real-Time Neural Vision for Obstacle Detection Using Linear Cameras. I. Masaky, editore, *Proceedings Intelligent Vehicles'95*, pagine 524–529, Detroit, September 25-26 1995. Society.

[Sch 93] H. Schneidermann e J. S. Albus. Progress and prospects for vision-based automatic driving. *Atti del 1993 annual meeting of IVHS AMERICA*, pagine 569–575, 1993.

[Ser 82] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, London, 1982.

[Tis 95] M. Tistarelli. Vision-based overtaking control to aid car driving. *Atti del 2nd. National Conference of PFT2*, pagine 2539–2549, Genova, Italia, Maggio 1995.

[Tan 80] S. L. Tanimoto e K. Kilger. *Structured Computer Vision: Machine Perception trough Hierarchical Compuation Structures*. Academic Press, NY, 1980.

[Web 95] J. Weber, D. Koller, Q. T. Luong, e J. Malik. New results in stereo-based automatic vehicle guidance. In Ichiro Masaky, editore, *Atti del IEEE Intelligent Vehicles'95*, pagine 530–535, Detroit, 1995.