UNIVERSITÀ DEL SALENTO

FACOLTÀ DI INGEGNERIA

Corso di Laurea Magistrale in Ingegneria Meccanica

TESI DI LAUREA

in

MECCANICA DEL VEICOLO

TECNICHE DI STEREOVISIONE PER IL RILEVAMENTO DI OSTACOLI IN CAMPO AGRICOLO

Relatore:

Ing. Giulio REINA

Laureando:

Antonio NOTARISTEFANO

ANNO ACCADEMICO 2011-2012

Ai miei genitori,

che hanno sempre creduto in me e mi hanno sostenuto durante questo lungo percorso di studi.

A mia sorella Claudia,

semplicemente con affetto.

Alla mia ragazza Daisy,

che mi è sempre stata accanto e che ha reso questo importante giorno ancora più belle.

All'Ing. Giulio Reina,

per avermi offerto la possibilità di affrontare un argomento di grande interesse e attualità e per la disponibilità dimostrata in ogni occasione.

SOMMARIO

INTRODUZIONE

In natura la maggior parte degli esseri viventi più evoluti è in grado di percepire la realtà mediante l'uso della vista.

La capacità di ottenere informazioni sull'ambiente circostante dagli stimoli luminosi e dal loro cambiamento è così importante in natura che sin dagli albori dell'intelligenza artificiale si è presa in considerazione l'idea di dotare di funzionalità visive anche i sistemi robotici ed informatici. Purtroppo in questi sistemi la capacità visiva è spesso accompagnata da un requisito di elaborazione in tempo reale che fino a poco tempo fa difficilmente poteva essere garantito dalla tecnologia a nostra disposizione. Ancora oggi i sistemi visivi di molti esseri viventi hanno prestazioni irraggiungibili da qualsiasi macchina; basti pensare ad esempio alla potenza di calcolo necessaria per gestire un sistema visivo complesso come quello dell'uomo: non è un caso che ben il 30 % circa dei neuroni (circa 60 miliardi) che compongono un cervello umano è dedicato all'elaborazione delle informazioni visive.

Dai numerosi tentativi compiuti dall'uomo per comprendere i principi alla base della visione umana sono emersi negli anni altrettanti progressi per tentare di imitarla a livello computazionale.

Da ciò è nata l'esigenza della Machine Vision, che consiste nello sviluppo di tecniche di visione artificiale di ausilio alla vita dell'uomo.

L'idea alla base della visione computazionale consiste nell'avere una o più telecamere connesse ad un computer, il quale deve automaticamente interpretare le immagini di una scena reale, ottenendo informazioni per svolgere determinate azioni.

Una delle facoltà più importanti del nostro cervello è sicuramente la capacità di fondere le immagini retiniche che provengono dagli occhi e di percepirle come un'unica immagine.

È proprio quest'ultimo processo, definito *Stereopsi*, che ci consente la localizzazione relativa degli oggetti visivi in profondità, donandoci la percezione della

tridimensionalità dello spazio. Ciò costituisce lo scopo della *stereoscopia* che significa "*visione spaziale*", dalle parole greche "stereo" che significa "spazio" e "skopein" che significa "vedere".

Fino a qualche decennio fa l'approccio stereoscopico era spesso trascurato dalla scienza a causa delle ingenti problematiche connesse alla realizzazione pratica del sistema di stereovisione, che imponeva l'impiego di due telecamere perfettamente uguali e ben allineate sullo stesso asse, l'uso di due sistemi di acquisizione, in grado di lavorare parallelamente, e la conoscenza esatta dei parametri di calibrazione dei dispositivi.

Infatti, sebbene le leggi prospettiche, che legano le coordinate 3D dei punti dello spazio a quelle delle rispettive proiezioni sui piani immagine siano note sin dai tempi di Leonardo da Vinci, la possibilità di utilizzare quelle relazioni geometriche era comunque subordinata alla conoscenza esatta dei parametri interni delle telecamere utilizzate e alla conoscenza quasi perfetta della disposizione geometrica dei sensori (parametri esterni).

Grazie ai progressi della tecnologia, le difficoltà relative all'hardware sono state superate, mentre lo studio di algoritmi di ottimizzazione e risoluzione di funzioni non lineari complesse ha consentito di misurare con un grado di accuratezza notevole i parametri intrinseci ed estrinseci delle telecamere correggendo via software le non idealità del sistema.

Grazie a tutto ciò i sistemi di visione stereoscopica hanno trovato applicazione nei più disparati ambiti di utilizzo: dalla robotica ai sistemi di video sorveglianza, dai veicoli a guida automatica all'esplorazione robotica su Marte.

I vantaggi della visione stereoscopica risiedono pertanto nel recupero della profondità, che, non solo consente di ricostruire la tridimensionalità degli oggetti presenti sulla scena, ma anche di percepire la vicinanza di ostacoli o dedurre l'avvicinamento o allontanamento di corpi in moto ed, infine, di risalire all'intera struttura 3D di una scena, ottenendo informazioni utilissime per la navigazione automatica, la manipolazione e il riconoscimento di oggetti da parte di robot autonomi.

Infatti, la ricostruzione tridimensionale di una scena consente ad un veicolo autonomo di navigare liberamente in un ambiente, percependo la presenza di eventuali ostacoli

da evitare, e di capire in quale punto dello spazio si trovi (autolocalizzazione) grazie alle elevate prestazioni stereometriche della visione stereoscopica.

Progetto *"QUAD-AV"* e obiettivo della tesi

Negli ultimi anni, si è assistito a un crescente interesse verso l'impiego di veicoli autonomi in campo agricolo al fine di migliorare la produttività e l'efficienza degli stessi. Perché un veicolo possa operare in maniera autonoma e sicura, è necessario e fondamentale che esso sia dotato di capacità avanzate di percezione e interpretazione dell'ambiente esterno.

L'Università del Salento, insieme a partner europei come il Centro di Ricerca francese Cemagref, il Danish Technology Institute (DTI) di Odense, il Fraunhofer IAIS di Bonn e il produttore di macchinari agricoli tedesco CLAAS (secondo produttore mondiale di trattori), è coinvolta in un progetto europeo denominato "*QUAD-AV*" (*Ambient Awareness for Autonomous Agricultural Vehicles*) per la realizzazione di un trattore a guida autonoma.

Tale progetto ha come obiettivo lo sviluppo e l'integrazione di diverse modalità sensoriali verificando la loro idoneità al riconoscimento di diverse tipologie di ostacoli nella scena e consentendo la costruzione di un database degli ostacoli, utile per il controllo del veicolo.

Gli ostacoli che si possono incontrare in campo agricolo possono essere classificati in quattro categorie: ostacoli positivi, ostacoli negativi, ostacoli in movimento (persone, animali, ecc.) e ostacoli legati alle differenze di struttura del terreno (suoli irregolari o non attraversabili).

Ostacoli positivi

Ostacoli negativi

Ostacoli in movimento (persone, animali, ecc.)

Differenze di struttura del terreno

Tali ostacoli, inoltre, possono essere molto variabili in quanto dipendenti dal tipo di terreno, dalla crescita di frutta e vegetazione, ecc.

A causa della varietà di situazioni che si possono incontrare, non esiste un unico sensore in grado di garantire risultati attendibili in ogni caso. Pertanto, il riconoscimento degli ostacoli viene effettuato valutando la potenzialità di quattro diverse modalità sensoriali (visione stereo, radar, ladar e termografia), il cui obiettivo è quello di incrementare il livello complessivo di sicurezza di un veicolo autonomo agricolo in relazione a se stesso, agli animali e alle persone presenti nel suo ambiente, nonché in relazione alle proprietà (www.quad-av.eu).

L'obiettivo della tesi consiste nell'utilizzo di un sistema di visione stereoscopica che consenta di individuare la presenza di ostacoli nella scena.

CAPITOLO 1

LA VISIONE STEREOSCOPICA

1.1 Introduzione

La percezione tridimensionale che l'uomo ha dell'ambiente che lo circonda deriva dalla capacità del nostro cervello di fondere le immagini retiniche che provengono dagli occhi e di percepirle come un'unica immagine. Infatti, poiché gli occhi sono posti ad una certa distanza uno dall'altro (circa 65 mm) essi osservano lo stesso oggetto sotto due angolazioni leggermente diverse fornendo al cervello due immagini retiniche che, entro certi limiti, si formano su punti della retina leggermente disparati. Il cervello poi fonde queste immagini e sfrutta proprio le differenze tra di esse per ricavare le informazioni relative alla conformazione tridimensionale della scena. È proprio quest'ultimo processo, definito *Stereopsi*, che ci consente di assegnare un senso di maggiore o minore profondità agli oggetti dello spazio visivo, donandoci la percezione tridimensionale dello spazio. Viene così realizzata la *Visione Stereoscopica* che ci consente di percepire l'oggetto come "unico" e "solido".
La *stereopsi computazionale* è invece il processo che consente di ottenere l'informazione di profondità da una coppia di immagini provenienti da due telecamere che inquadrano una scena da differenti posizioni.
Essa costituisce una branca articolata e complessa della visione computazionale all'interno della quale si possono individuare in generale due sottoproblemi:

1. Il calcolo delle corrispondenze;
2. La ricostruzione della struttura 3D.

Il primo consiste nel riconoscere quali punti di ciascuna immagine sono la proiezione di uno stesso punto della scena. I punti così individuati sono detti *coniugati* oppure *omologhi*. L'individuazione di tali punti risulta notevolmente agevolata se le immagini in esame differiscono in maniera lieve tra loro, in modo che uno stesso particolare della scena risulti simile in entrambe le immagini. A tale scopo assume quindi una

grande importanza il posizionamento e l'orientazione delle telecamere e sarà necessario introdurre alcuni vincoli per ridurre al massimo i falsi accoppiamenti,
come ad esempio il *vincolo epipolare*, secondo cui il corrispondente di un punto in un'immagine, può trovarsi solo su una retta (*retta epipolare*) nell'altra immagine. Questo vincolo rende il problema del calcolo delle corrispondenze unidimensionale piuttosto che bidimensionale, comportando ovviamente notevoli vantaggi in termini di semplicità e rapidità d'esecuzione.
Una volta note le coppie di punti coniugati delle due immagini e noti i parametri delle telecamere (posizionamento, orientazione e caratteristiche interne del sensore), è possibile ricostruire la posizione nella scena dei punti proiettati sulle due immagini (ricostruzione della struttura 3D).
In realtà, tale problema non risulta di difficile soluzione se i dati di partenza sono corretti, e per questo motivo assumerà una considerevole importanza il problema della calibrazione delle telecamere ed il calcolo dei relativi parametri intrinseci.

1.2 Sistema di acquisizione di immagini digitali

Un generico sistema di visione ha lo scopo di fornire in uscita un'immagine della realtà che sta osservando. Nel caso in cui essa debba essere processata da un computer si deve trattare di un'immagine di tipo digitale. Un sistema di acquisizione di immagini digitali consiste di tre componenti fondamentali: una *telecamera*, un *frame grabber* ed un calcolatore *host*.
All'interno di un sistema di visione stereoscopica si utilizzano due telecamere che catturano una coppia di immagini stereo analogiche. La conversione in immagini digitali viene realizzata dal frame grabber. Una rappresentazione schematica di questo sistema è riportata in Figura 1.1.

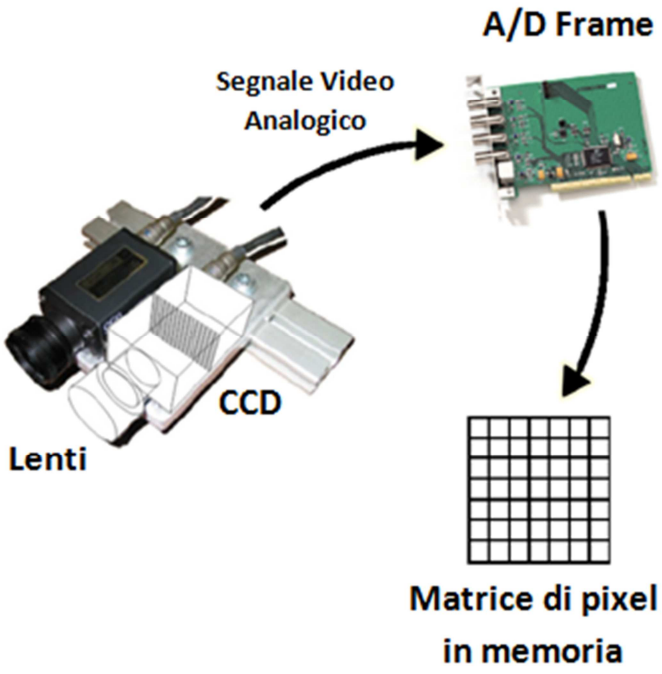


Figura 1.1: Struttura di un sistema di visione stereo.

Il processo di formazione dell'immagine ha inizio con i raggi luminosi che entrano nella telecamera attraverso un'apertura detta pupilla. La formazione dell'immagine è resa possibile per il fatto che questi raggi luminosi colpiscono uno schermo, o *piano immagine*, su cui risiede il dispositivo fotosensibile che registra le intensità dei raggi luminosi, il CCD (Charge Coupled Device).

Molti di questi raggi sono il risultato della riflessione, da parte degli oggetti nella scena, dei raggi provenienti da una sorgente luminosa. All'interno della telecamera, per meglio raccogliere i raggi luminosi, sono presenti delle lenti.

Il CCD è il sensore che riproduce la retina dell'occhio umano. Esso consiste in un circuito integrato formato da una griglia di $n \times m$ elementi semiconduttori sensibili alla luce, cioè in grado di accumulare una carica elettrica proporzionale all'intensità della radiazione luminosa che li colpisce. Tali elementi sono accoppiati in modo che ognuno di essi, sollecitato da un impulso elettrico, possa trasferire la propria carica ad un altro elemento adiacente. Inviando al dispositivo una sequenza temporizzata d'impulsi, si ottiene in uscita un segnale elettrico grazie al quale è possibile ricostruire la matrice dei pixel che compongono l'immagine proiettata sulla superficie del CCD

stesso. L'uscita della telecamera a CCD è un segnale elettrico analogico, ottenuto leggendo il potenziale degli elementi della matrice CCD per righe. Il segnale video (analogico) viene così inviato al *frame grabber*. Quest'ultimo è una scheda di acquisizione che viene montata sul PC e a cui vengono collegati i segnali in uscita da entrambe le telecamere. Esso ha la funzione di digitalizzare il segnale video analogico in ingresso, convertendolo in una matrice $N \times M$ (tipicamente 512×512) di valori interi, memorizzati in un'opportuna area di memoria chiamata *frame buffer*. Gli elementi della matrice prendono il nome di *pixels* o *picture elements*.

I pixels sono i più piccoli elementi autonomi che compongono la rappresentazione di una immagine nella memoria di un computer. Solitamente i punti sono cosi piccoli e numerosi da non essere distinguibili ad occhio nudo, apparendo fusi in un'unica immagine quando vengono stampati su carta o visualizzati su un monitor.

Il numero di pixel presente nel sensore esprime la risoluzione della telecamera, per cui maggiore è il loro numero, migliore è la qualità dell'immagine che si ottiene. Ciascun pixel e caratterizzato dalla propria posizione e da valori come colore e intensità, variabili in funzione del sistema di rappresentazione adottato.

Se l'immagine è monocromatica ogni pixel assumerà un valore in scala di grigi su 8 bit che forniscono 256 diverse gradazioni di luminosità dell'immagine che vanno da 0 (nero) a 255 (bianco). Nel caso di immagine a colori le componenti su 8 bit associate a ogni pixel sono tre, secondo la codifica RGB.

Indicando con $I(u, v)$ il valore dell'immagine (ovvero il valore della luminosità) nel pixel individuato dalla riga v e dalla colonna u (sistema di coordinate (u, v) avente l'origine nell'angolo in alto a sinistra), e con $n \times m$ le dimensioni del piano immagine (quindi le dimensioni del CCD), la matrice degli elementi fotosensibili e quella dei pixels sono legate tra loro mediante le seguenti relazioni:

$$u_{pixel} = \frac{N}{n} u_{CCD}$$

$$v_{pixel} = \frac{M}{m} v_{CCD}$$

da cui appare chiaro che la posizione di un punto sul piano immagine risulta diversa se misurata in pixel (u_{pixel}, v_{pixel}) o elementi del piano immagine (u_{CCD}, v_{CCD}). Tuttavia, risulta comodo assumere che vi sia una relazione uno ad uno tra pixel ed elementi del piano immagine, pensando che ad ogni pixel corrisponda un'area rettangolare sul piano immagine, le cui dimensioni sono dette *dimensioni efficaci del pixel*.

1.3 Formazione dell'immagine

Per estrarre delle informazioni dalle immagini è necessario conoscere le fasi relative alla formazione delle stesse. Studiare l'intero reale processo di formazione dell'immagine all'interno della telecamera porterebbe a dover analizzare operazioni complesse e specifiche. Pertanto, ai fini dello studio della formazione dell'immagine, il funzionamento della telecamera viene ricondotto a quello di un modello che ne riproduce le caratteristiche fondamentali.
Naturalmente esso rappresenta solamente parte del processo di acquisizione dell'immagine, giungendo a un compromesso tra accuratezza descrittiva del funzionamento reale e complessità del modello.

1.3.1 Modello pinhole della telecamera

Esistono diversi modelli di telecamera, ognuno con proprie caratteristiche e limiti di applicazione. Essi differiscono principalmente per il grado di approssimazione e per la quantità di elementi considerati nel modello (lenti, CCD, ecc.). Il più utilizzato è il *modello pinhole* che è il più semplice e ideale ma, allo stesso tempo, fornisce un'approssimazione accettabile del processo di formazione dell'immagine con convenienza anche dal punto di vista matematico e computazionale.
Secondo il modello pinhole la telecamera viene modellata come una scatola costituita da un foro infinitesimo, detto *punto di fuoco*. Attraverso tale foro i raggi luminosi provenienti dal mondo esterno penetrano nella scatola andando a formare

un'immagine rovesciata del mondo esterno sulla parete opposta, che costituisce il *piano dell'immagine*. Il campo visivo, ovvero l'insieme dei punti che può essere proiettato nell'immagine, costituisce una piramide infinita di vertice il punto di fuoco.

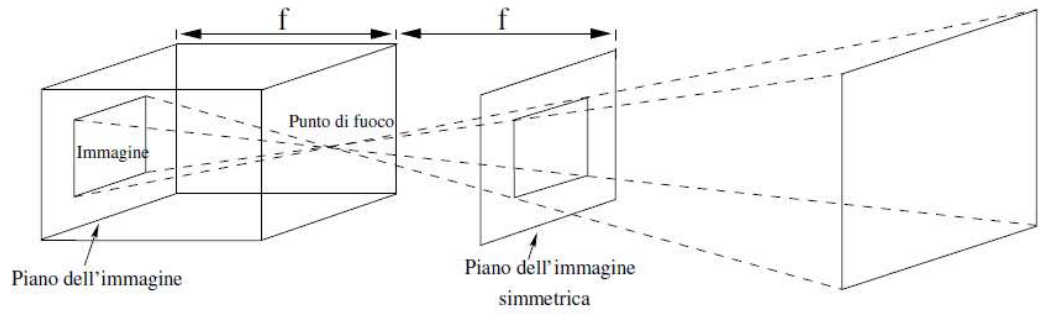


Figura 1.2: Modello pinhole box.

Il modello pinhole possiede tuttavia dei limiti dovuti al fatto che in realtà il fuoco non è puntiforme ma ha una dimensione non trascurabile. Pertanto, il raggio che unisce un punto 3D, il pinhole e un punto 2D non è unico, ma ogni punto sul piano immagine raccoglie un cono di raggi luminosi provenienti dall'esterno.

Gli occhi dei vertebrati, le macchine fotografiche e le telecamere utilizzano lenti al fine di garantire che l'immagine sia a fuoco e, allo stesso tempo, luminosa a sufficienza, risultato non ottenibile con la sola struttura a pinhole di una semplice telecamera. Infatti per ottenere immagini nitide è necessario ridurre la dimensione del pinhole, ma in questo modo l'immagine sarebbe poco luminosa. Viceversa, per aumentare la luminosità sarebbe necessario ingrandire il pinhole, ma in tal caso a ogni punto dell'immagine non corrisponderebbe più un solo raggio luminoso ma un cono di raggi luminosi convergenti che darebbero come risultato immagini offuscate. Il compromesso tra le due situazioni descritte si ottiene pertanto aggiungendo un obiettivo, cioè un sistema di lenti.

1.3.2 Modello a lenti sottili

Il sistema ottico più semplice, più utilizzato e che raccoglie i principi di base di un obiettivo è quello delle *lenti sottili*.

Una lente sottile è definita come una lente il cui spessore è considerato piccolo rispetto alle distanze generalmente associate alle sue proprietà ottiche (raggi di curvatura delle calotte, distanze focali, distanze degli oggetti e delle immagini).

Essa è caratterizzata da un asse ottico, passante per il centro della lente C e perpendicolare al piano della lente, e due fuochi F e F', cioè due punti esterni alla lente che giacciono sull'asse ottico ad una certa distanza dal centro C della lente e che si trovano sui due lati opposti della stessa.

F prende il nome di *fuoco primario* e rappresenta un punto dell'asse ottico per il quale ogni raggio proveniente da esso o diretto in esso si propaga parallelamente all'asse a seguito della rifrazione.

F' prende il nome di *fuoco secondario* e rappresenta un punto dell'asse ottico per il quale ogni raggio che si propaga parallelamente all'asse è diretto in esso o appare provenire da esso a seguito della rifrazione.

Sebbene in generale i fuochi siano asimmetrici rispetto al centro C della lente, si assume per semplicità che assi abbiano la stessa distanza da C. Tale distanza è detta *lunghezza focale* e viene indicata con f.

In definitiva, il funzionamento delle lenti sottili è riassunto dalle due seguenti affermazioni:

- I raggi paralleli all'asse ottico e incidenti sulla lente vengono rifratti in modo da passare per il fuoco secondario F';
- I raggi che passano per il centro C della lente restano inalterati;
- I raggi che passano per il fuoco primario F vengono trasmessi dalla lente tutti paralleli alla direzione dell'asse ottico.

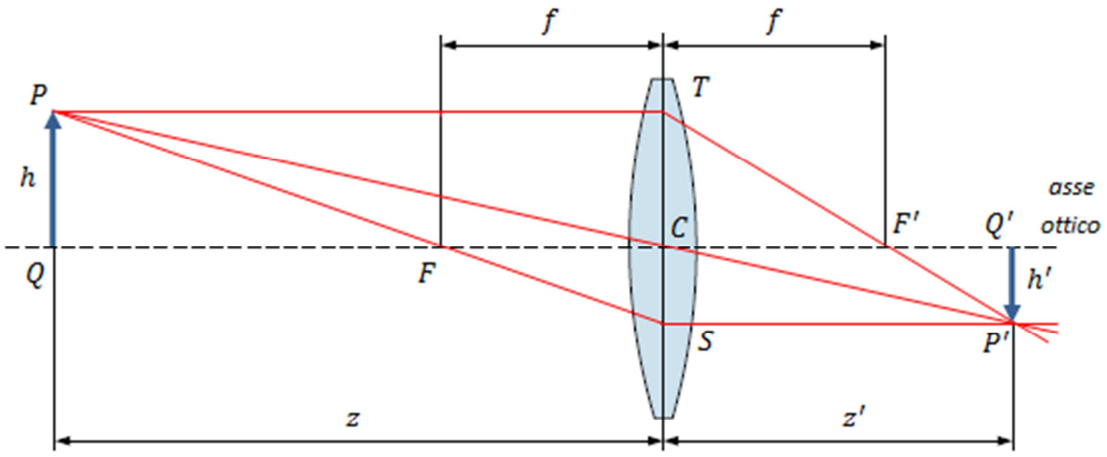


Figura 1.3: Modello a lenti sottili.

Ora, dato un punto P della scena è possibile costruirne graficamente l'immagine P' considerando due raggi particolari che partono da P: il raggio parallelo all'asse ottico, che dopo la rifrazione passa per il fuoco F ed il raggio che passa inalterato per il centro della lente C.

Dalla similitudine tra i triangoli TSP' e TCF' e i triangoli PST ed FSC si ottengono le seguenti due relazioni:

$$\begin{cases} \dfrac{h}{h+h'} = \dfrac{f}{z'} \\ \dfrac{h'}{h+h'} = \dfrac{f}{z} \end{cases} \tag{1.1}$$

Sommando le espressioni membro a membro si ottiene la *legge di Fresnel*, che esprime la relazione fra la distanza z dell'oggetto e la distanza z' dell'immagine dal centro C della lente:

$$\frac{1}{z} + \frac{1}{z'} = \frac{1}{f}$$

Nel caso in cui $|z| \gg f$, dalla legge di Fresnel risulta che:

$$|z'| \cong f$$

ottenendo così un modello di telecamera molto più semplice.

In definitiva, il modello della telecamera consiste di un *piano retina* (o *piano immagine*) $\mathcal{R}$ e di un punto C, *centro ottico* (o *centro di proiezione*) distante f (*lunghezza focale*) dal piano. La retta passante per C e ortogonale a $\mathcal{R}$ è l'*asse ottico* (asse z nella Figura 1.4) e la sua intersezione con $\mathcal{R}$ prende il nome di *punto principale*. Il piano $\mathcal{F}$ parallelo ad $\mathcal{R}$ e contenente il centro ottico prende il nome di *piano focale*.

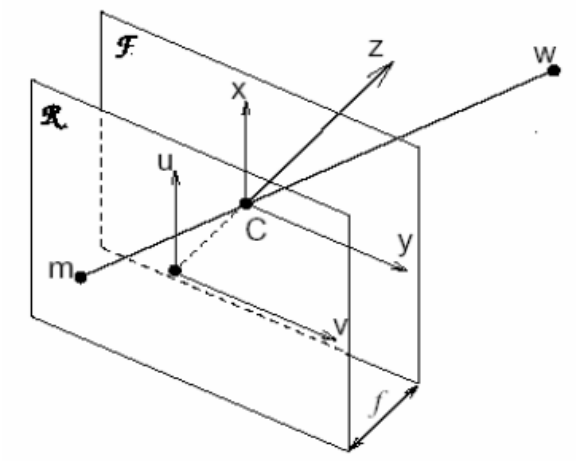


Figura 1.4: Modello pinhole della telecamera.

Si supponga di fissare un sistema di riferimento cartesiano $Oxyz$, avente origine coincidente con il centro ottico C e asse z coincidente con l'asse ottico della telecamera. Sia $w = (x, y, z)^T$ un punto dello spazio e si indichi con $m = (u, v)^T$ la sua proiezione su $\mathcal{R}$ attraverso C.

Mediante semplici considerazioni sulla similitudine dei triangoli è possibile scrivere le seguenti relazioni di proiezione prospettica:

$$\begin{cases} u = \frac{f}{z} \cdot x \\ v = \frac{f}{z} \cdot y \end{cases} \tag{1.2}$$

Poiché la proiezione dallo spazio 3D a quello ottico 2D è non lineare (a causa della presenza della variabile z a denominatore) risulta opportuno esprimere i punti w ed m in coordinate omogenee come segue:

$$\widetilde{w} = (x, y, z, 1)^T \qquad \widetilde{m} = (u, v, 1)^T$$

dove l'apice $\sim$ individua il nuovo sistema di coordinate. Mediante tale trasformazione è possibile scrivere le equazioni di proiezioni nella seguente forma matriciale:

$$\widetilde{m} = \begin{bmatrix} ku \\ kv \\ k \end{bmatrix} = \begin{bmatrix} f \cdot x \\ f \cdot y \\ z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \tilde{P} \cdot \widetilde{w}$$

La matrice $\tilde{P}$ è denominata *Matrice di Proiezione Prospettica MPP*.
Si noti che k coincide con la terza coordinata di w, ovvero con la distanza dal piano xy.
I punti per cui k è nullo sono punti all'infinito e coincidono col piano focale $\mathcal{F}$.
Un modello realistico di una telecamera che descriva la trasformazione da coordinate 3D a coordinate pixel, oltre che della trasformazione prospettica, deve tenere conto dei seguenti due processi:

1. La *pixelizzazione* o *discretizzazione* dovuta al sensore CCD (visto come matrice bidimensionale di pixel) e sua posizione rispetto all'asse ottico;
2. La trasformazione isometrica tra il sistema di riferimento Mondo e quello della telecamera: *rototraslazione*.

1.3.3 La pixelizzazione

La pixelizzazione deve tener conto del fatto che:

- Il centro ottico della telecamera non coincide con il centro fisico del CCD ma ha coordinate (u_0, v_0);
- Le coordinate di un punto nel sistema di riferimento standard della telecamera sono misurate in pixel : si introduce pertanto un fattore di scala;
- La forma dei pixel non è quadrata: occorre, pertanto, considerare due fattori di scala diversi lungo gli assi x ed y e che vengono indicati rispettivamente con $k_u = 1/s_u$ e $k_v = 1/s_v$ espressi in termini di pixel/mm lungo le direzioni

orizzontale e verticale, essendo s_u ed s_v le dimensioni orizzontale e verticale dell'areola del sensore della telecamera;

- A causa di sfasamenti nella scansione di righe successive dello schermo, gli assi di riferimento immagine u, v non sono ortogonali ma inclinati di ϑ.

I primi tre punti vengono presi in considerazione mediante l'introduzione nella (1.2) della traslazione del centro ottico e della riscalatura indipendente degli assi u e v:

$$\begin{cases} u = k_u \frac{f}{z} \cdot x + u_0 \\ v = k_v \frac{f}{z} \cdot y + v_0 \end{cases} \tag{1.3}$$

essendo (u_0, v_0) le coordinate del centro immagine C_c ed i parametri k_u e k_v le unità del sistema di riferimento immagine $O_I UV$.

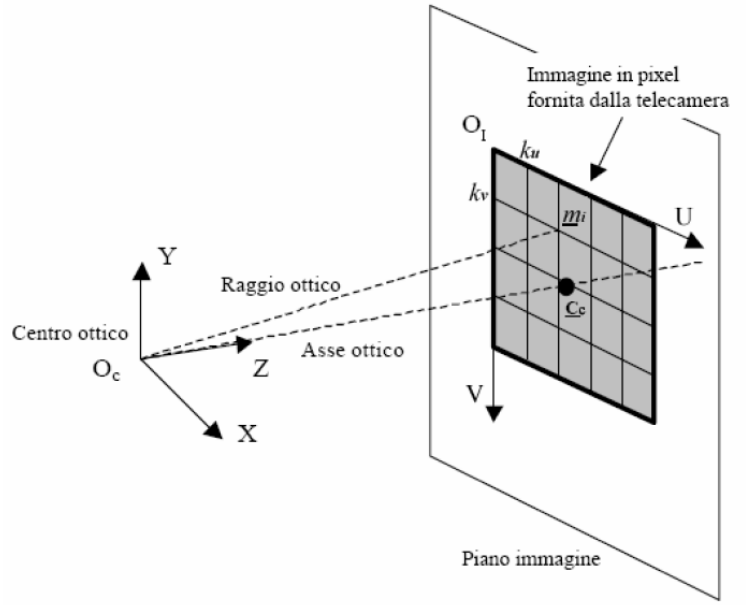


Figura 1.5: Proiezione di un punto sul piano immagine.

Si può notare che se le coordinate di un generico punto m_i sull'immagine vengono espresse in pixel e quelle del punto che lo ha generato sono espresse in metri così come anche la distanza focale f, allora $1/k_u$ e $1/k_v$ rappresentano le dimensioni, in

metri, di un singolo pixel; mentre fk_u e fk_v possono essere interpretate come la dimensione della distanza focale in termini di pixel orizzontali e verticali.
Pertanto la MPP può essere riscritta nel modo seguente:

$$\tilde{P} = \begin{bmatrix} fk_u & 0 & u_0 & 0 \\ 0 & fk_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = A \cdot [I \quad 0] \qquad \text{con} \quad A = \begin{bmatrix} fk_u & 0 & u_0 \\ 0 & fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

Il modello più generale in realtà deve prevedere anche la possibilità che gli assi u, v non siano ortogonali ma inclinati di un angolo ϑ. La matrice A più generale può essere pertanto riscritta nel modo seguente:

$$A = \begin{bmatrix} fk_u & -fk_v \cdot \cot\vartheta & u_0 \\ 0 & fk_v / \sin\vartheta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

Tuttavia, nei moderni sistemi di acquisizione si è cercato di attenuare la disomogeneità lungo u e v della matrice di CCD, per cui si può considerare $\vartheta = \pi/2$.
Le quantità f, k_u, k_v, u_0 e v_0 non dipendono né dall'orientazione né tanto meno dalla posizione della telecamera ed è per questo motivo che vengono chiamati *parametri intrinseci* o *interni* della telecamera.

1.3.4 La trasformazione rigida tra la telecamera e la scena

Per tenere conto del fatto che, in generale, il sistema di riferimento mondo non coincide con il sistema di riferimento standard della telecamera, è necessario introdurre una trasformazione rigida che lega i due sistemi di riferimento.

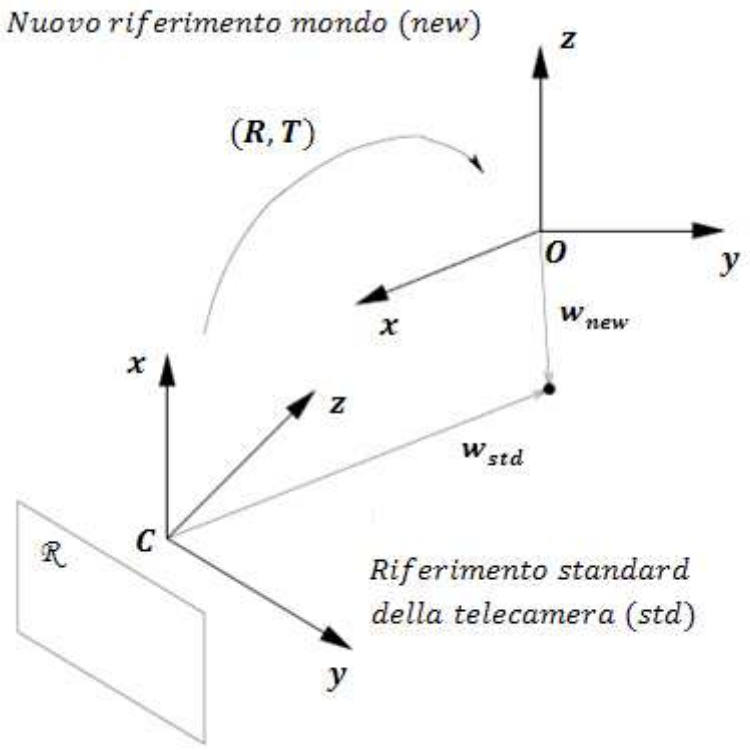


Figura 1.6: Trasformazione rigida tra la telecamera e la scena.

Si introduce, pertanto, un cambio di coordinate costituito da una rotazione R seguita da una traslazione T che esprimono l'orientazione e la posizione della telecamera rispetto ad una terna solidale con il mondo esterno.

Pertanto, nell'ipotesi di considerare un punto che nel sistema di riferimento standard della telecamera abbia coordinate $(x_{std}, y_{std}, z_{std})$ ed in quello solidale con il mondo esterno abbia coordinate $(x_{world}, y_{world}, z_{world})$, è possibile legare le coordinate di uno stesso punto nei due sistemi di riferimento mediante la seguente relazione:

$$w_{std} = R \cdot w_{world} + T$$

e che in coordinate omogenee si riscrive come:

$$\widetilde{w}_{std} = G \cdot \widetilde{w}_{world}$$

essendo $G = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}$ la matrice che codifica i *parametri estrinseci* o *esterni* (R e T) della telecamera.

Ricordando che $\widetilde{m} = \tilde{P} \cdot \widetilde{w}_{std}$, è possibile mettere in relazione la posizione di $\widetilde{w}_{world}$ nello spazio 3D nel sistema di riferimento mondo con la sua proiezione sul piano immagine $\widetilde{m}$.

$$\widetilde{m} = \tilde{P} \cdot \widetilde{w}_{std} = \tilde{P} \cdot G \cdot \widetilde{w}_{world} = \tilde{P}_{new} \cdot \widetilde{w}_{world}$$

essendo $\tilde{P}_{new}$ è la nuova MPP in coordinate omogenee, espressa come segue:

$$\tilde{P}_{new} = A \cdot [I \quad 0] \cdot G = A \cdot [R \quad T]$$

Nell'ipotesi spesso verificata in cui $\vartheta = \pi/2$ e ponendo:

$$\alpha_u = f \cdot k_u$$

$$\alpha_v = f \cdot k_v$$

$$T = [t_1 \quad t_2 \quad t_3]^T$$

$$R = [r_1{}^T \quad r_2{}^T \quad r_3{}^T]^T$$

la matrice $\tilde{P}$ può essere riscritta come segue:

$$\tilde{P} = \begin{bmatrix} \alpha_u \cdot r_1{}^T + u_0 \cdot r_3{}^T & \alpha_u \cdot t_1 + u_0 \cdot t_3 \\ \alpha_v \cdot r_2{}^T + v_0 \cdot r_3{}^T & \alpha_v \cdot t_2 + v_0 \cdot t_3 \\ r_3{}^T & t_3 \end{bmatrix}$$

Tale matrice è di fatto costituita da 10 parametri indipendenti la cui conoscenza è ignota in maniera esatta anche al costruttore il quale, per poterli stimare, deve ricorrere ad un accurato processo di calibrazione [1].

1.3.5 Modello della distorsione radiale e tangenziale

Il modello pinhole della telecamera a 10 parametri descritto sinora non è ancora sufficiente a descrivere in maniera completa la trasformazione geometrica che subiscono le immagini acquisite.

Infatti, le considerazioni fatte riguardo al modello pinhole della telecamera sono state desunte dal modello delle lenti sottili.

Tuttavia, le lenti sottili costituiscono un'astrazione matematica e, per quanto i progressi compiuti dall'ottica abbiano portato alla costruzione di lenti sempre più prossime a queste ipotesi di idealità, in esse è sempre presente una certa componente dominante di *distorsione radiale* (dovuta alle dimensioni reali della lente) ed una lieve *distorsione tangenziale* (causata dal decentramento dell'asse della lente causata da un cattivo allineamento della componentistica della lente).

Ragionando in un sistema di coordinate polari con origine nel centro dell'immagine, la distorsione radiale agisce sui pixel dell'immagine in funzione della sola distanza dal centro ρ, mentre quella tangenziale dipende anche dall'angolo ϑ.

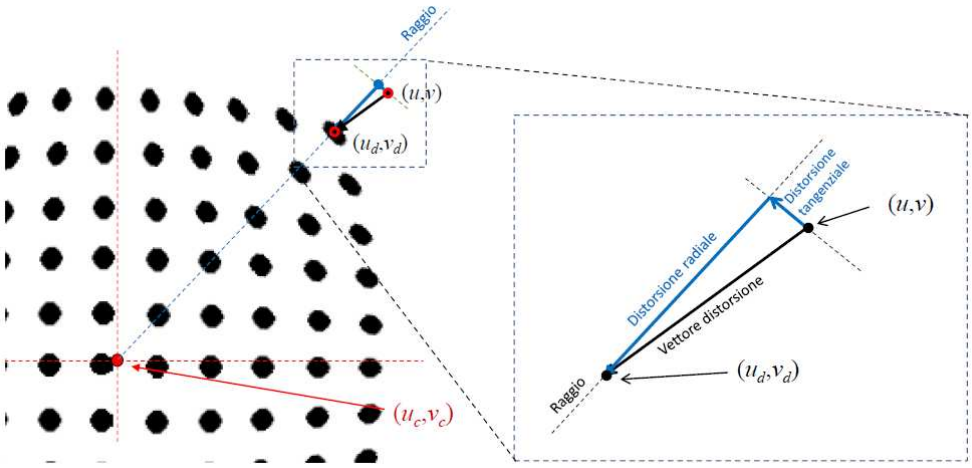
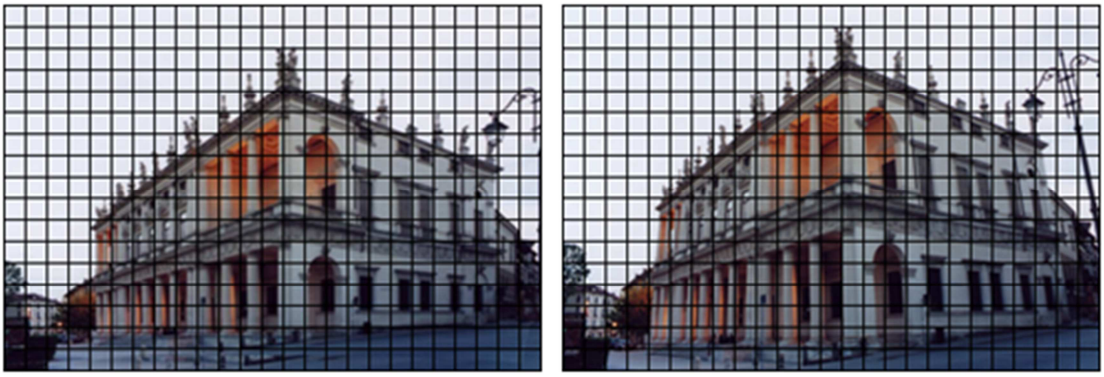


Figura 1.7: Distorsione radiale e tangenziale della lente.

Nella pratica si procede individuando il punto dell'immagine di coordinate (u_c, v_c) che presenta distorsione nulla, cioè coordinate distorte e non distorte coincidenti. Ci si aspetta che il punto (u_c, v_c), detto *centro di distorsione*, sia vicino al punto centrale dell'immagine di coordinate (u_0, v_0). Quindi si scompone il segmento che congiunge il

punto di coordinate distorte (u_d, v_d) con quello di coordinate non distorte (u, v) in due componenti: una orientata secondo la retta che congiunge (u_d, v_d) e (u_c, v_c) e detta distorsione radiale, ed una ortogonale a tale retta, detta distorsione tangenziale. Tale scomposizione è mostrata in Figura 1.7, nella quale si è ipotizzata l'acquisizione di una immagine di una griglia di cerchi organizzata secondo le direzioni orizzontale e verticale [2].

Di seguito vengono messe a confronto due immagini di uno stesso edificio in cui è possibile notare, mediante una griglia opportuna, come l'immagine a destra sia affetta da una notevole distorsione.



(a) (b)

Figura 1.8: a) Immagine non distorta; b) Immagine affetta da una notevole distorsione.

Sono stati sviluppati diversi modelli matematici di distorsione al fine di correggerla e tornare così alle ipotesi di lente sottile: tale processo di correzione prende il nome di *compensazione della distorsione*.

La distorsione può essere pensata come una funzione non lineare che si applica ai singoli pixel dell'immagine originaria non distorta di coordinate (u, v) che dipende, oltre che dalle stesse coordinate, anche dalla distanza dal centro $\rho = u^2 + v^2$ e dall'angolo $\vartheta = \tan^{-1}(v/u)$.

Indicate con (u_d, v_d) le coordinate distorte, la relazione che esprime la trasformazione da quelle non distorte è la seguente:

$$\begin{bmatrix} u_d \\ v_d \end{bmatrix} = f(u, v, \rho, \vartheta)$$

La maggior parte dei modelli matematici è basata sulla stima dei coefficienti dello sviluppo in serie di Taylor della funzione f arrestato al secondo ordine e non tengono quasi mai conto della distorsione tangenziale: pertanto essi forniscono soltanto due parametri di distorsione, che risultano essere sufficienti a correggere fenomeni di aberrazione molto lievi, ma inefficaci per alterazioni di notevole entità.

Ora, la distorsione può essere modellata come somma di due contributi: uno radiale ed uno tangenziale (equazione di correzione di Brown). Pertanto il legame tra le coordinate distorte e quelle non distorte può essere scritto nel modo seguente:

$$\begin{bmatrix} u_d \\ v_d \end{bmatrix} = \underbrace{(1 + k_1\rho^2 + k_2\rho^4 + k_3\rho^6) \cdot \begin{bmatrix} u \\ v \end{bmatrix}}_{\textbf{\textit{Distorsione radiale}}} + \underbrace{\begin{bmatrix} 2\tau_1 \cdot u \cdot v + \tau_2(\rho^2 + 2u^2) \\ 2\tau_2 \cdot u \cdot v + \tau_1(\rho^2 + 2v^2) \end{bmatrix}}_{\textbf{\textit{Distorsione tangenziale}}}$$

dove k_1, k_2, k_3 sono delle costanti chiamate *coefficienti di distorsione radiale*, mentre τ_1, τ_2 sono delle costanti chiamate *coefficienti di distorsione tangenziale*. Questi cinque parametri dipendono dal tipo di lente e, pertanto, fanno parte dei parametri intrinseci della telecamera.

Il parametro più importante è k_1 in quanto è quello che influenza maggiormente l'entità della distorsione: se $k_1 < 0$ si ha una *distorsione a barile*, se $k_1 > 0$ si ottiene una *distorsione a cuscino* entrambe illustrate in Figura 1.9:

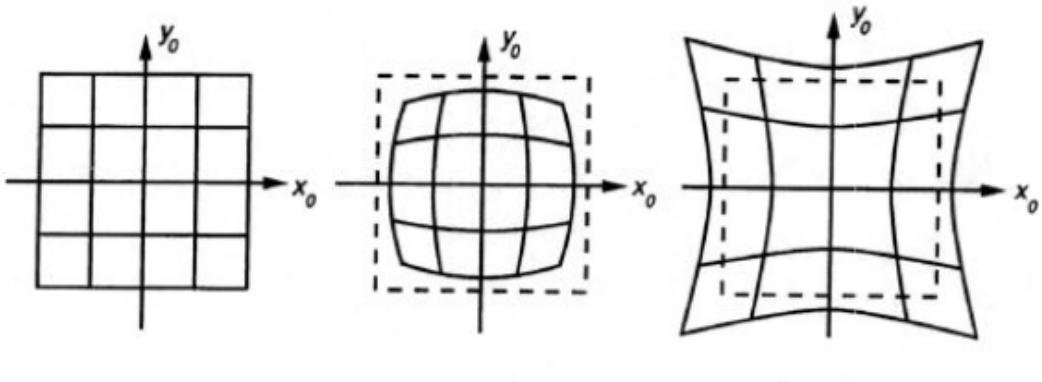


Figura 1.9: Distorsione a barile e a cuscino.

La prima è causata da un effetto di ingrandimento dell'immagine nella parte centrale che diminuisce man mano che ci si allontana dall'asse ottico. L'effetto ottico è di un immagine che è stata "avvolta" attorno ad una sfera (o barile).
La seconda è esattamente l'opposto della prima: l'ingrandimento dell'immagine cresce all'aumentare della distanza dall'asse ottico. L'effetto ottico causa la tendenza delle linee non passanti per il centro ottico ad essere inclinate verso lo stesso, creando, appunto, una forma simile ad un cuscino.

1.4 Calibrazione delle telecamere

1.4.1 Parametri di calibrazione

Per conoscere la matrice di proiezione prospettica MPP è necessario valutare i parametri intrinseci che costituiscono la matrice A e quelli estrinseci relativi alla matrice G. I parametri intrinseci rappresentano la geometria interna della telecamera e le caratteristiche ottiche, mentre quelli estrinseci fanno riferimento alla posizione e all'orientazione del sistema di riferimento telecamera rispetto al sistema di riferimento mondo.

Il processo di misurazione di tali parametri è detto *calibrazione* e si basa sul presupposto che si conoscano le proiezioni di alcuni punti 3D, detti *punti di calibrazione*, le cui coordinate sono note.

La necessità di calcolare queste due categorie di parametri, interni ed esterni, porta ad una naturale suddivisione del processo di calibrazione in:

- Calibrazione interna: consente di determinare la lunghezza focale f; le coordinate (u_0, v_0) del centro immagine espresse in pixel nel sistema di riferimento dell'immagine; le dimensioni k_u e k_v dei pixel che costituiscono dei fattori di scala che tengono conto della forma rettangolare, e non quadrata, degli stessi; i parametri k_1, k_2, k_3, τ_1 e τ_2 che caratterizzano la distorsione radiale e tangenziale delle lenti;
- Calibrazione esterna: consente di determinare la matrice di rotazione R ed il vettore di traslazione T che definiscono le trasformazioni necessarie al passaggio dal sistema di riferimento della telecamera a quello del mondo e viceversa.

La fase di *calibrazione interna* è richiesta una sola volta, mentre la fase di *calibrazione esterna* deve essere ripetuta ogni qualvolta le telecamere vengono spostate e/o ruotate. Infatti, per quanto detto in precedenza, i parametri interni ottenuti dalla fase di calibrazione omonima sono legati all'hardware della telecamera. Quindi, fintanto che la telecamera resta la stessa anche tali parametri rimangono invariati. Nel caso

della calibrazione esterna, invece, poiché i parametri che la interessano (R, T) dipendono dall'orientazione e dalla posizione della telecamera, è sufficiente che questa venga mossa, anche accidentalmente, affinché si renda necessaria una nuova fase di calibrazione esterna.

Esistono diversi metodi di calibrazione la cui idea di base consiste nel ricavare i parametri della telecamera risolvendo un sistema lineare di equazioni che mette in relazioni un insieme di punti 3D di coordinate note alle loro proiezioni sull'immagine. Ciò consente, pertanto, di calcolare la matrice di proiezione prospettica e la sua decomposizione in parametri intrinseci ed estrinseci.

Le coordinate dei punti 3D vengono solitamente fissate su un oggetto (pattern di calibrazione) i cui punti interni (punti di calibrazione) hanno coordinate note. Tali punti sono costituiti dai vertici di N elementi che nella immagine devono essere riconoscibili senza ambiguità ed avere coordinate note con accuratezza. Gli N elementi in genere sono quadrati disposti a scacchiera, solitamente di colore nero su sfondo bianco.

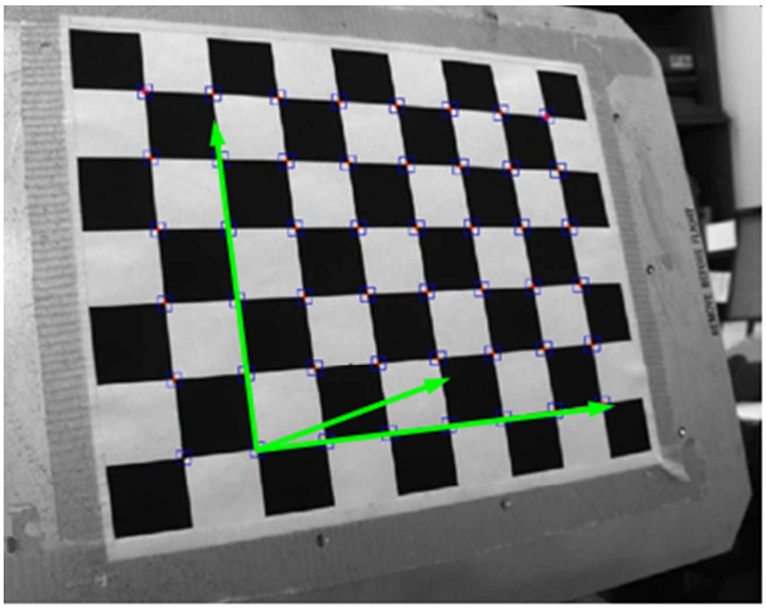


Figura 1.10: Scacchiera di calibrazione della telecamera.

L'acquisizione di tali pattern in diverse posizioni e orientazioni, utilizzando procedure ripetute fino a convergenza, consente di stimare i parametri intrinseci ed estrinseci che caratterizzano il sistema.

1.5 Il calcolo delle corrispondenze

Il calcolo delle corrispondenze consiste nell'individuare quali punti o quali porzioni delle immagini destra e sinistra sono la proiezione dello stesso elemento nella scena.

Da questa ricerca dipende gran parte dell'elaborazione successiva volta a ottenere il posizionamento tridimensionale e la struttura degli oggetti osservati.

La precisione che si ha nella ricostruzione tridimensionale della scena dipende in maniera diretta dalla precisione nella determinazione dei cosiddetti *punti coniugati* o *omologhi*, tanto che spesso il problema della ricerca delle corrispondenze è considerato il principale problema della visione stereo. La determinazione dei punti omologhi può essere computazionalmente onerosa se ci si basa solo su vincoli di similarità. Infatti, per ciascun punto di un'immagine sarebbe necessario verificare la similarità su tutte le righe e tutte le colonne dell'altra immagine generando inoltre numerosi falsi accoppiamenti [3].

Per semplificare il problema e ridurre lo spazio di ricerca dei punti coniugati è stato introdotto il *vincolo epipolare*, ipotizzando che due punti coniugati debbano trovarsi necessariamente su una linea retta, detta appunto *retta epipolare*.

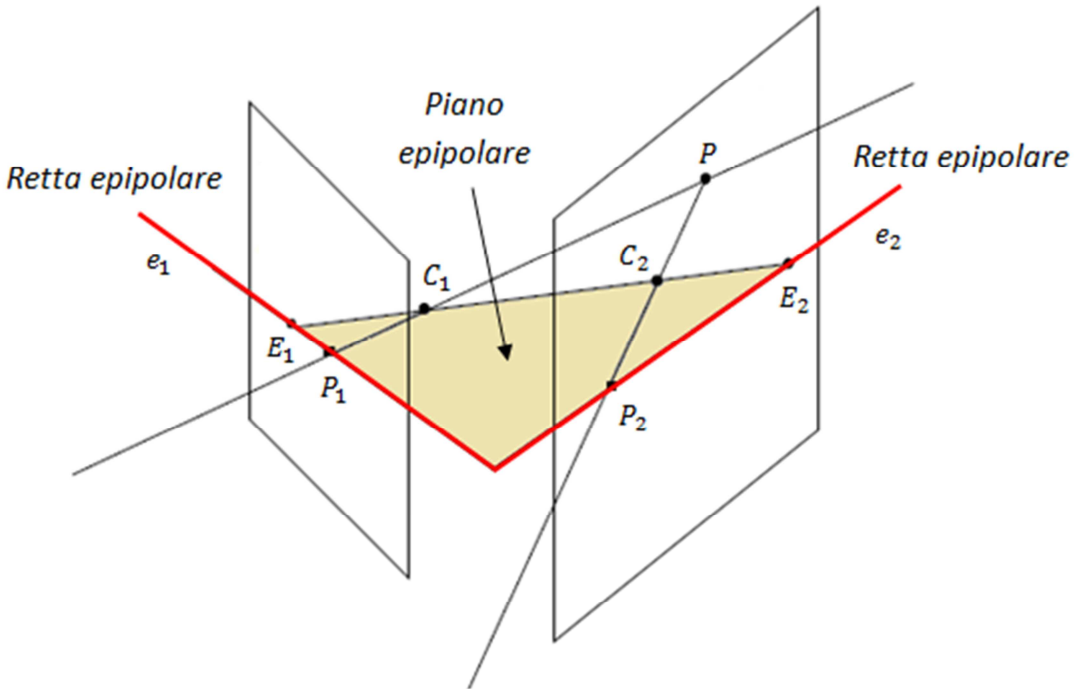
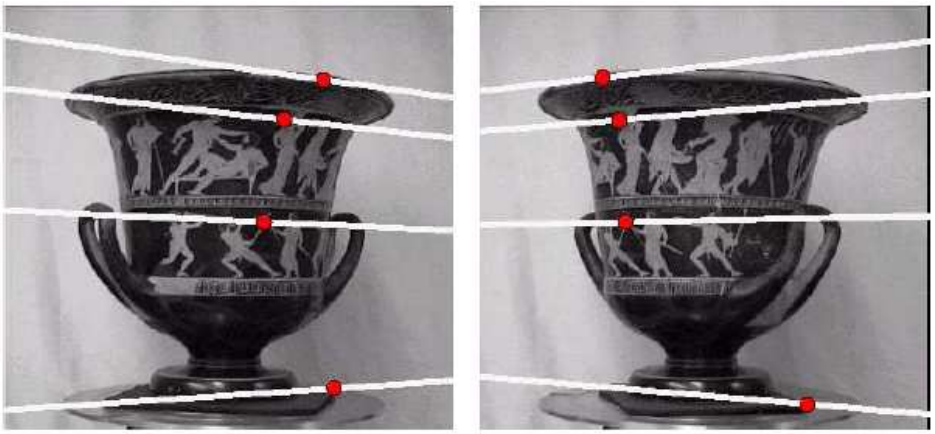


Figura 1.11: Costruzione della retta epipolare.

Si consideri ad esempio il caso in Figura 1.11 in cui il punto P nello spazio tridimensionale ha come proiezioni il punto P_1 nell'immagine I_1 e il punto P_2 nell'immagine I_2. Il piano passante per P, P_1 e P_2 è detto *piano epipolare*, e l'intersezione di quest'ultimo con i due piani immagine determina le rette epipolari e_1 ed e_2. Si osserva inoltre che tutte le linee epipolari di un'immagine passano per uno stesso punto, chiamato *epipolo* (E_1 per l'immagine sinistra e E_2 per quella destra), e che i piani epipolari costituiscono un fascio di piani che hanno in comune la retta passante per i centri ottici C_1 e C_2. Dato quindi il punto P_1 nell'immagine I_1, il suo corrispondente nell'immagine I_2 sarà vincolato a trovarsi sulla retta epipolare e_2, e viceversa. Il segmento che unisce i centri ottici delle due telecamere prende il nome di *baseline*.



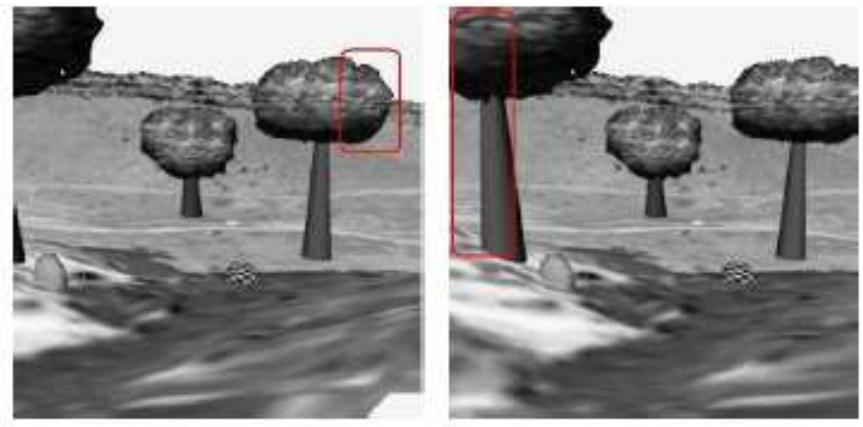
(a) (b)

Figura 1.12: Esempio di rette epipolari: il corrispondente di un punto nell'immagine sinistra (a) si trova su una linea retta in quella destra (b), e viceversa. Tale linea è detta retta epipolare.

I principali problemi che si riscontrano nel calcolo delle corrispondenze e che possono portare a falsi accoppiamenti sono:

- Occlusioni: poiché le telecamere si trovano in posizioni differenti, è inevitabile che alcune parti della scena compaiano solo in una delle due immagini. In altri casi, invece, il problema si manifesta a causa della prospettiva: può capitare, infatti, che in un'immagine un particolare dello sfondo venga nascosto da un oggetto in primo piano, mentre nell'altra sia perfettamente visibile. Questi

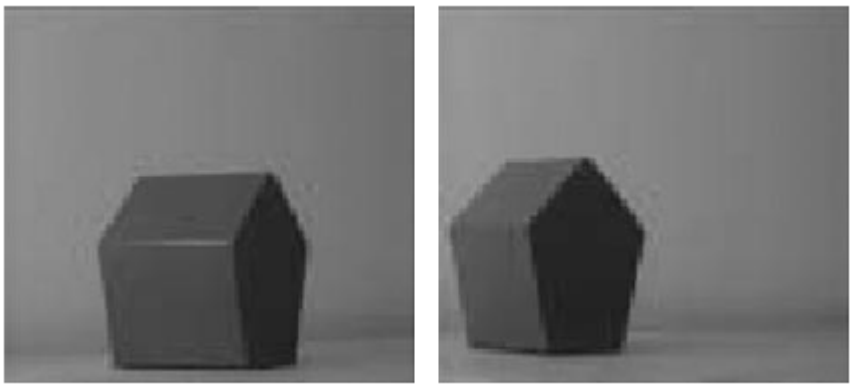
punti quindi non hanno una corrispondenza e, in casi particolari, possono essere fonte di errori e causa di falsi o mancati accoppiamenti;



(a) (b)

Figura 1.13: Problema delle occlusioni: nell'ultima immagine sinistra (a) sono visibili particolari della scena che non sono presenti in quella destra (b), e viceversa.

- Distorsione proiettiva**:** a causa della proiezione prospettica, lo stesso oggetto della scena può essere proiettato in modo diverso sulle due immagini, rendendo molto difficile il riconoscimento dei punti coniugati. Tale effetto è tanto maggiore quanto più distanti tra loro sono i punti di ripresa della scena.



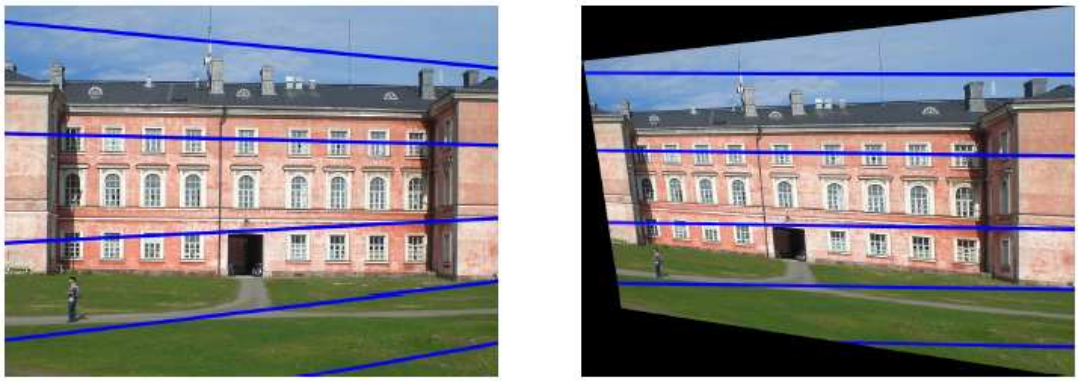
(a) (b)

Figura 1.14: Distorsione proiettiva: nell'immagine sinistra (a) si nota come l'oggetto venga proiettato in maniera differente rispetto all'immagine destra (b).

1.6 Rettificazione epipolare

Per semplificare ulteriormente la ricerca di punti corrispondenti, è possibile effettuare un'operazione di rettifica delle proiezioni, detta *rettificazione epipolare*, che consente di trasformare una qualunque coppia di immagini in modo che le coppie coniugate di rette epipolari risultino parallele, orizzontali e collineari in ciascuna immagine, dove per collinearità si intende che una retta epipolare dell'immagine destra sia la prosecuzione della sua corrispondente nell'immagine sinistra.

Infatti, nell'ipotesi in cui gli assi ottici delle telecamere siano convergenti, le rette epipolari di ogni telecamera convergono negli epipoli. Per cui, considerato un punto in una delle due immagini, la ricerca del suo corrispondente nell'altra immagine sarà effettuata su una retta in due dimensioni. Rendendo invece tutte le rette epipolari parallele, perfettamente orizzontali e collineari, la ricerca della corrispondenza si ridurrebbe ad un problema unidimensionale (lungo una linea orizzontale).



(a) (b)

Figura 1.15: Visuale della camera sinistra prima (a) e dopo (b) la rettifica, con alcune rette epipolari d'esempio.

Le immagini così rettificate possono essere pensate come acquisite da un nuovo sistema di stereovisione ottenuto per rototraslazione delle telecamere originarie.

Il vantaggio più importante della rettificazione epipolare risiede nel fatto che il calcolo delle corrispondenze fra punti coniugati risulta enormemente semplificato in quanto, nell'ipotesi che i punti coniugati giacciano sulla stessa riga, la ricerca del corrispondente di un punto nell'altra immagine potrà limitarsi ai soli punti

appartenenti alla stessa riga di pixel. Ciò comporta un notevole risparmio computazionale in quanto l'algoritmo di ricerca passa da una complessità quadratica ($\propto n^2$) ad una lineare ($\propto n$), essendo n una delle dimensioni trasversali (base o altezza) dell'immagine.

La rettificazione va effettuata a valle della calibrazione, ovvero una volta calcolati e resi disponibili i parametri intrinseci ed estrinseci dei sensori che determinano le Matrici di Proiezione Prospettica $\tilde{P}_1$ e $\tilde{P}_2$ delle due telecamere e che legano le coordinate omogenee 3D di un punto $\widetilde{w}$ nello spazio alle coordinate omogenee delle sue proiezioni $\widetilde{m}_1$ ed $\widetilde{m}_2$ sui piani immagine dei due sensori.

È ormai noto che:

$$\begin{cases} \widetilde{m}_1 = \tilde{P}_1 \cdot \widetilde{w} \\ \widetilde{m}_2 = \tilde{P}_2 \cdot \widetilde{w} \end{cases}$$

dove la generica matrice $\tilde{P}$ è espressa come segue:

$$\tilde{P} = A \cdot [R \quad T] = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}$$

L'idea alla base della rettificazione epipolare è quella di definire due nuove matrici $\tilde{P}_{n1}$ e $\tilde{P}_{n2}$ ottenute per rotazione di quelle originarie attorno ai loro centri ottici C_1 e C_2 fintanto che i piani focali non diventino complanari e contengano la baseline. Ciò assicura che gli epipoli siano punti all'infinito e che, pertanto, le rette epipolari risultano parallele.

Al fine di avere rette epipolari orizzontali, invece, la baseline deve essere parallela all'asse x del nuovo sistema di riferimento standard delle due telecamere. In aggiunta, per soddisfare alla richiesta di collinearità tra punti coniugati, le coppie di rette epipolari devono avere la stessa coordinata verticale. Questo viene ottenuto richiedendo che le nuove matrici $\tilde{A}_{n1}$ e $\tilde{A}_{n2}$ siano caratterizzate dagli stessi parametri intrinseci, ovvero dagli stessi valori delle coordinate del centro ottico (u_0, v_0) e dagli

stessi valori delle lunghezze focali (meglio detti coefficienti di scalatura orizzontale e verticale α_u, α_v).

Imporre l'uguaglianza delle lunghezze focali fa sì che anche i piani immagine diventino complanari, come mostrato in Figura 1.16.

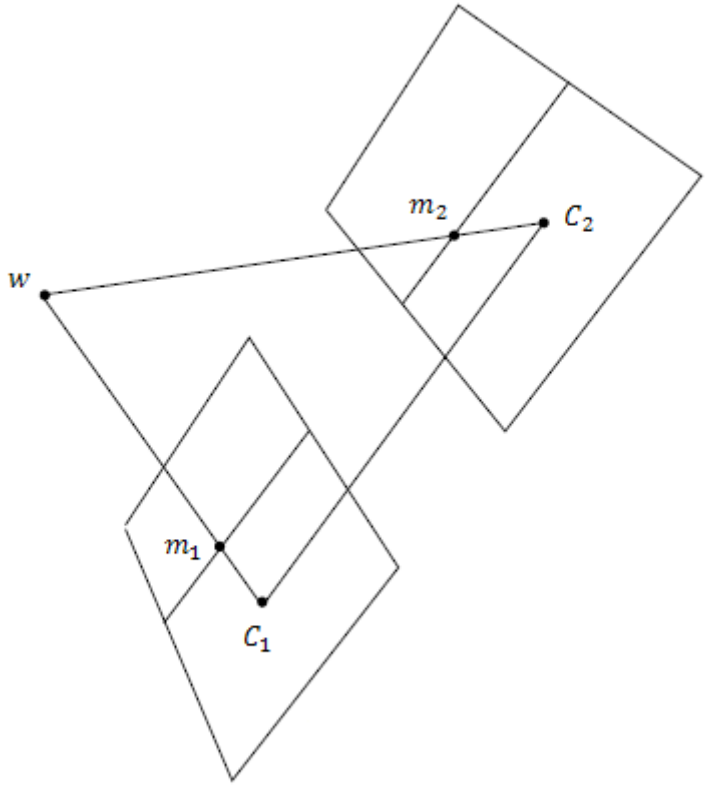


Figura 1.16: Telecamere rettificate. I piani retina sono complanari e paralleli alla baseline.

Per far sì, inoltre, che oggetti quadrati appiano ancora quadrati piuttosto che rettangolari, è necessario imporre l'uguaglianza tra α_u e α_v. Nella pratica si sceglie $\alpha_u = \alpha_v = \alpha = \min(\alpha_u, \alpha_v)$. In definitiva, la posizioni dei centri ottici C_1 e C_2 risultano essere le stesse di quelle originarie mentre gli orientamenti delle terne di riferimento di ciascuna telecamera differiscono dai precedenti solo per rotazione di angoli opportuni. I parametri intrinseci delle telecamere vengono imposti uguali e con gli stessi valori dei coefficienti di scalatura orizzontale e verticale.

Pertanto le matrici $\tilde{P}_{n1}$ e $\tilde{P}_{n2}$ differiranno solo per le posizioni dei loro centri ottici e le due telecamere possono essere pensate come una singola telecamera traslata lungo l'asse x del suo sistema di riferimento.

1.7 Ricostruzione 3D

1.7.1 Stereo Matching

Si è visto che utilizzando due immagini che rappresentano la stessa scena da due prospettive differenti, è possibile eseguire la ricostruzione dei punti delle immagini nella scena 3D. Per far ciò, è necessario considerare i punti appartenenti ad un'immagine ed identificarli nell'altra. Il vincolo epipolare e la rettificazione delle immagini consentono solo di semplificare tale problema restringendo la zona di ricerca dei punti coniugati, ma non consentono di conoscere con esattezza la loro collocazione. Per risolvere tale problema è possibile ricorrere alle tecniche di *Stereo Matching*.
Per il calcolo delle corrispondenze sono stati sviluppati svariati algoritmi basati principalmente su due metodi diversi:

- *Area-based*: questi algoritmi considerano una piccola porzione (finestra) di un'immagine, e cercano nell'altra la regione che più vi somiglia. Solitamente quest'operazione avviene tramite una misura di correlazione tra le due finestre;
- *Feature-based*: in questo caso gli algoritmi ricercano nelle immagini delle caratteristiche di particolare interesse (*edge*, spigoli, linee curve, ecc...) e le mettono in corrispondenza tra loro.

Per questi ultimi algoritmi è possibile un'ulteriore classificazione in base al metodo utilizzato per realizzare l'accoppiamento tra le feature ricavate dalle due immagini. In particolare è possibile distinguere tra:

- *Correlation-based*: la corrispondenza viene stabilita tramite una misura di correlazione tra porzioni delle immagini;
- *Relaxation-based*: partendo da un insieme di corrispondenze (tra tutti i punti o solo tra alcuni di essi), l'accoppiamento viene stabilito riorganizzando tali corrispondenze secondo certi vincoli (relazioni) stabiliti a priori;

- *Dynamic programming*: il problema del calcolo delle corrispondenze può anche essere interpretato come la minimizzazione di una funzione costo, risolvibile con il metodo della Programmazione Dinamica.

I metodi Feature-based sono generalmente più veloci di quelli Area-based in quanto permettono di rivolgere l'attenzione su determinati particolari delle immagini, ma sono poco affidabili nel rilevamento di oggetti di forma libera, i quali in genere non presentano caratteristiche da poter essere utilizzate come vincoli di similarità.

1.7.2 Disparità

Una volta determinata con esattezza la corrispondenza tra punti delle due immagini è possibile ricavare le coordinate tridimensionali dell'oggetto di cui tali punti sono la proiezione.

In particolare, individuate le corrispondenze di un punto della scena in entrambe le immagini (sinistra e destra) provenienti da una coppia di telecamere stereo, è possibile ricavare la *disparità* d tra questi due punti. Essa rappresenta la base da cui ogni sistema di stereovisione parte per calcolare le coordinate 3D di ogni punto della scena ed esprime la distanza tra il punto considerato nell'immagine sinistra ed il suo coniugato nell'immagine destra.

Infatti, considerato un punto P appartenente alla scena, esso verrà rappresentato sulle due immagini destra e sinistra in due posizioni diverse P_l e P_r. Chiamate x_l e x_r le distanze tra queste posizioni e gli assi ottici delle telecamere, la differenza $d = x_l - x_r$ rappresenta la disparità.

Riuscendo a mettere in corrispondenza ogni pixel dell'immagine di sinistra con uno dell'immagine di destra è possibile costruire una mappa che racchiuda l'informazione sulla disparità di ogni punto della scena. Tale mappa prende il nome di *mappa di disparità* o, usando la terminologia inglese, *disparity map*. Essa è una rappresentazione bidimensionale della scena ripresa dalle telecamere che codificano con gradazioni di grigio l'informazione sulla distanza e la profondità dagli oggetti presenti. Le due immagini destra e sinistra si fondono in un'unica immagine in scala di grigi, che cambia

intensità a seconda della distanza dei punti dall'osservatore: colori più scuri indicano oggetti più distanti, mentre colori chiari oggetti più vicini.

(a) (b) (c)

Figura 1.17: Immagine telecamera sinistra (a), immagine telecamera destra (b), mappa di disparità (c) in cui le tonalità più scure indicano disparità minori e quindi identificano punti della scena che presentano distanze maggiori rispetto alla coppia di telecamere.

Nella figura seguente è possibile notare come gli oggetti presentino disparità sempre minori man mano che questi si trovino ad una maggiore distanza dal punto di osservazione.

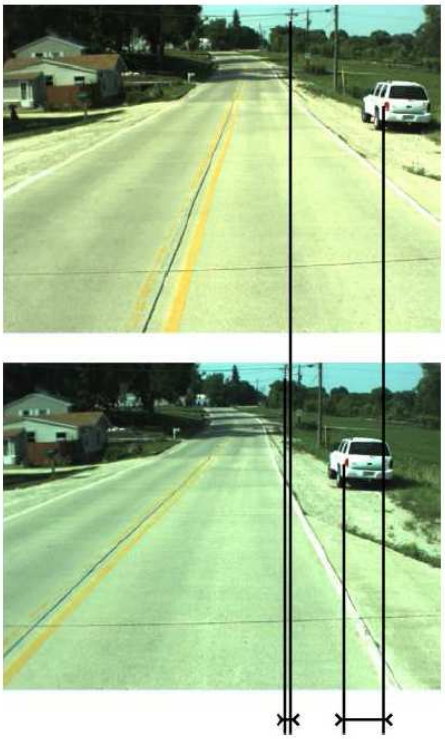


Figura 1.18*: Disparità di alcuni ostacoli. Gli oggetti più vicini al punto di osservazione presentano una maggiore disparità.*

La disparità cresce al diminuire della distanza di un oggetto dall'osservatore; diminuisce per oggetti via via più lontani.
Infine, dalla disparità di punti corrispondenti delle due immagini e dalla conoscenza dei parametri delle telecamere (che governano la proiezione prospettica di punti 3D sui piani ottici dei sensori) è possibile calcolare la posizione 3D di oggetti fisici nello spazio e ricostruire l'intera struttura tridimensionale della scena visibile.

1.7.3 Sistema stereo semplificato

Si supponga di considerare un sistema stereo semplificato, costituito da due telecamere disposte con assi ottici paralleli tra loro e posti ad una distanza b (baseline) l'uno dall'altro. Si supponga, inoltre, che i fuochi si trovino alla stessa altezza di modo che un qualsiasi punto nell'immagine sinistra abbia la stessa ordinata nella corrispondente immagine destra, e che non vi sia alcuna distorsione radiale. Si indichi con f la distanza focale, ovvero la distanza tra il centro della lente e il piano immagine e con C_l e C_r i centri delle due lenti.
Avendo fissato il sistema di riferimento $Oxyz$ con origine coincidente con il punto medio della baseline e avendo fissato i sistemi di riferimento delle due telecamere con origine nel centro ottico di ciascuna di esse, sia P un punto nella scena di coordinate (x, y, z) e siano P_l (x_l, y_l) e P_r (x_r, y_r) le proiezioni prospettiche di tale punto sui piani immagine sinistro e destro ottenute dalla ricerca delle corrispondenze come illustrato nella Sezione 1.5.

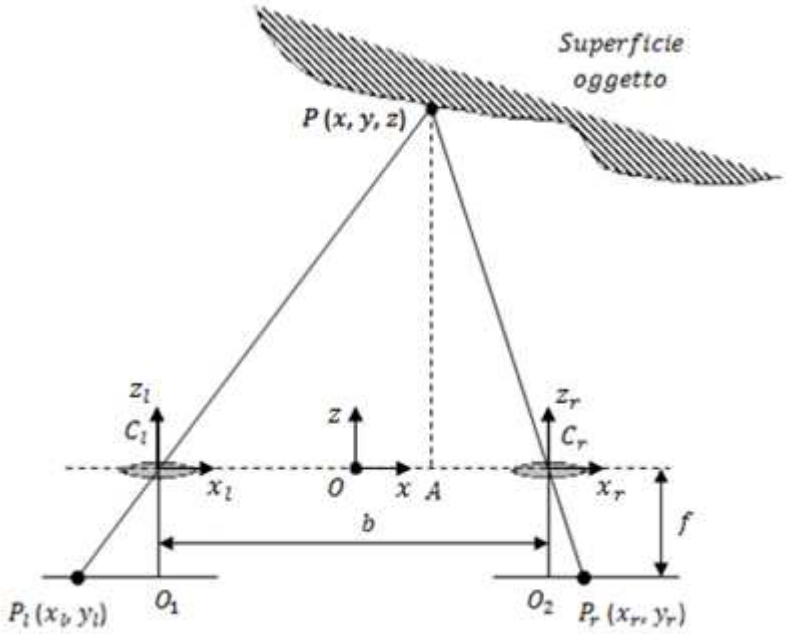


Figura 1.19: Un semplice sistema di visione stereo.

Dalla similitudine tra i triangoli $C_l P_l O_1$ e $P C_l A$ e i triangoli $C_r O_2 P_r$ e $P A C_r$ è possibile scrivere che:

$$\frac{x_l}{f} = \frac{x + b/2}{z}$$

$$\frac{x_r}{f} = \frac{x - b/2}{z}$$

$$\frac{y_l}{f} = \frac{y_r}{f} = \frac{y}{z}$$

Risolvendo ora il sistema nelle incognite (x, y, z) si ottengono i seguenti risultati:

$$x = \frac{b(x_l + x_r)}{2(x_l - x_r)}$$

$$y = \frac{b(y_l + y_r)}{2(x_l - x_r)}$$

$$z = \frac{bf}{x_l - x_r} = \frac{bf}{d}$$

La quantità $(x_l - x_r)$ che compare in ciascuna equazione rappresenta la disparità d. Nota dunque la disparità tra i punti omologhi, la distanza tra le telecamere b e la loro lunghezza focale f, è possibile ricostruire l'esatta posizione nella scena del punto P.
Tuttavia permangono alcuni problemi di natura pratica:

- Il calcolo delle coordinate tridimensionali può risultare affidabile per oggetti vicini, ma molto impreciso, o addirittura impossibile, per oggetti lontani. Generalmente b ed f sono valori costanti, per cui dall'ultima espressione si deduce che la distanza z risulta essere inversamente proporzionale alla disparità d tra i punti coniugati. Ma nella misura della disparità non è possibile ottenere una precisione superiore al pixel e si commettono errori specialmente in caso d'oggetti a grande distanza;
- La disparità è proporzionale alla distanza b che separa le telecamere.
  Questo implica che, se l'errore che si commette nel calcolo della disparità è costante, la stima della coordinata z del punto P è più precisa all'aumentare della baseline b.

Si arriva dunque ad una contraddizione: infatti, precedentemente, era emerso che, per ridurre i problemi relativi alla distorsione proiettiva e alle occlusioni fosse preferibile disporre le telecamere piuttosto vicine; per quanto riguarda invece il problema della ricostruzione vale esattamente il contrario. Occorre quindi giungere ad un compromesso per ottenere la situazione ottimale. Tipicamente però si tende a mantenere le telecamere vicine in quanto spesso un errore nella stima della distanza è trascurabile, mentre è molto più importante poter stabilire con esattezza le corrispondenze. In alcuni casi poi, la scelta è obbligata, come nel caso di sistemi di visione per l'assistenza alla guida di autoveicoli, in cui le telecamere si troveranno inevitabilmente molto ravvicinate. Generalmente in queste situazioni si tiene conto del fatto che per oggetti piuttosto lontani la distanza ricavata potrebbe risultare errata e si tende a restringere il range di ricerca.

1.7.4 La risoluzione

Nota la geometria di base di un sistema di stereovisione, è lecito domandarsi quale sia la massima risoluzione che un tale sistema è in grado di raggiungere nella ricostruzione delle coordinate tridimensionali.

Per *risoluzione* si intende la distanza minima che un sistema stereo riesce a distinguere e rappresenta l'ampiezza dell'area in cui possono collocarsi differenti punti pur assumendo lo stesso valore di disparità. Essendo l'algoritmo stereo una procedura di triangolazione, tale risoluzione peggiora con la distanza dalle telecamere. Infatti, si è visto precedentemente che disparità e profondità sono legate mediante la seguente relazione:

$$z = \frac{bf}{d}$$

essendo b la baseline ed f la distanza focale.

Derivando tale espressione rispetto alla disparità d e tenendo conto della relazione scritta sopra si ha che:

$$\Delta z = \frac{z^2}{bf}\Delta d$$

essendo Δz la più piccola variazione di distanza che un sistema di stereovisione è in grado di rilevare in presenza di una variazione di disparità Δd.

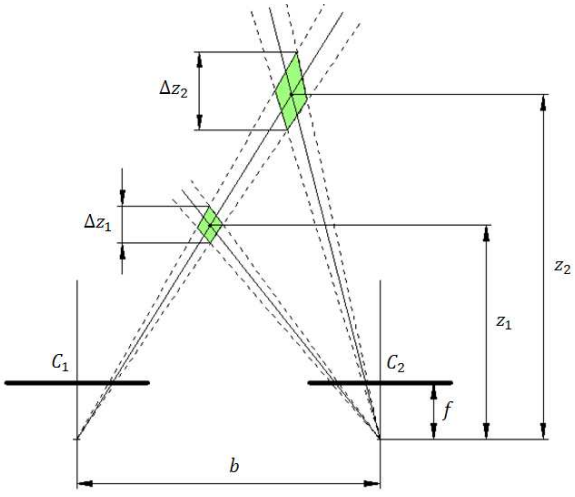


Figura 1.20: Andamento dell'incertezza all'aumentare di z.

Si può notare dall'immagine che gli oggetti vicini godono di una maggiore accuratezza rispetto a quelli distanti.

Una risoluzione migliore consente di distinguere meglio le differenze di posizione quando gli oggetti sono lontani dalla telecamera. Inoltre dalla formula appena scritta, si ricava un risultato importante: all'aumentare della distanza a cui si trovano gli oggetti, aumenta il beneficio di utilizzo di baseline più ampie.

Infatti, poiché baseline e distanza focale influiscono inversamente in tale relazione, ciò implica che una baseline ampia e/o una distanza focale maggiore consentono di aumentare il numero di dettagli che un sistema di stereovisione è in grado di percepire, garantendo una maggiore accuratezza nella definizione della profondità. Tuttavia, all'aumentare della baseline si riduce il numero di punti reali visibili da entrambe le telecamere dal momento che la zona di sovrapposizione dei coni di vista di entrambe le telecamere risulta essere limitata rispetto a quella ottenibile con una baseline minore. Inoltre, diventa più difficile l'individuazione di coppie coniugate, poiché la differente prospettiva rende molto diverse le features corrispondenti.

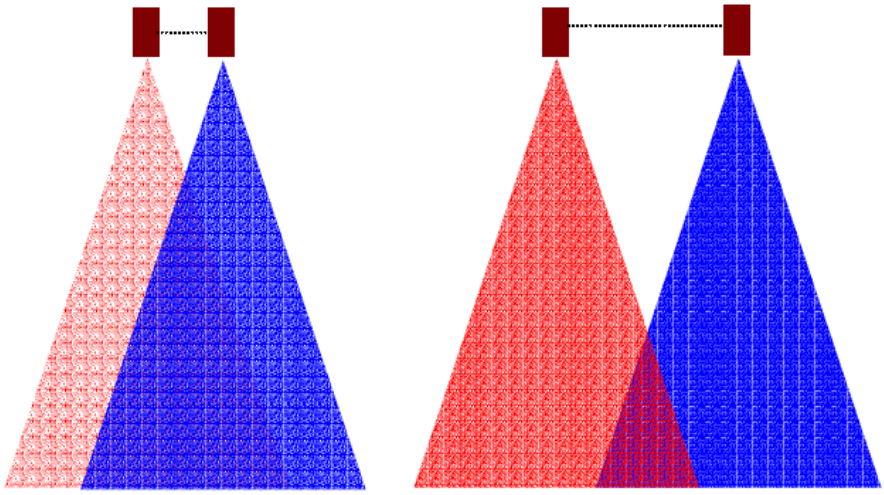


Figura 1.21: Variazione del campo di vista con la linea di base.

CAPITOLO 2

STATO DELL'ARTE

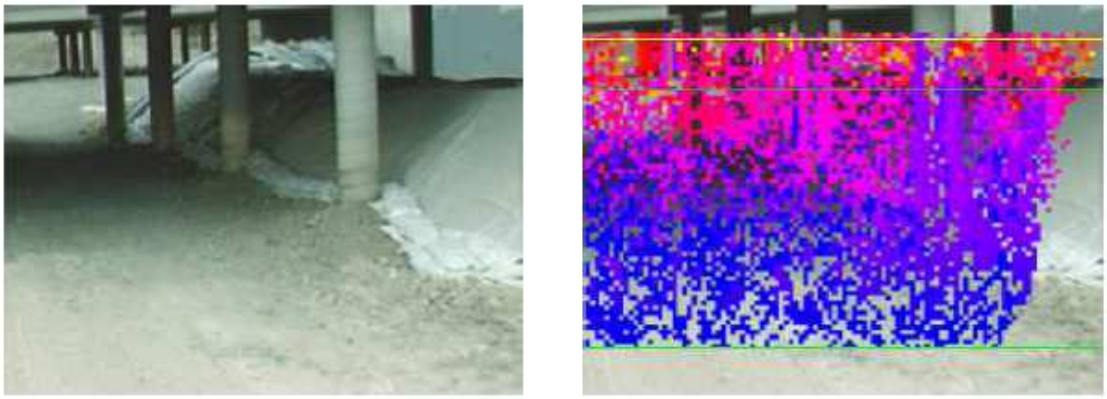
2.1 Tecniche di rilevamento di ostacoli in letteratura

Come detto, l'obiettivo della tesi è quello di ideare un algoritmo che consenta di riconoscere la presenza di ostacoli sfruttando le informazioni provenienti da un sistema trinoculare montato su un trattore agricolo a guida autonoma.

L'identificazione degli ostacoli rappresenta ancora oggi un problema aperto e diversi sono i metodi proposti in letteratura volti allo sviluppo di algoritmi che forniscano soluzioni che siano computazionalmente semplici e garantiscano allo stesso tempo una accurata percezione del mondo esterno, incrementando il livello di sicurezza del veicolo [4].

L'approccio utilizzato nel Terramax consiste nella ricerca per ogni punto dell'immagine che si è scelta di prendere come riferimento, del corrispondente punto nell'altra immagine.

Questo consente di creare una *Disparity Space Image* (DSI), o immagine di disparità, in cui ad ogni regione dell'immagine presa come riferimento è assegnato il valore di disparità che individua la regione dell'altra immagine che le assomiglia maggiormente. Inoltre, per facilitare la comprensione, ciascun valore di disparità viene contrassegnato con un colore differente corrispondente alla distanza del punto della scena dalla telecamera. In Figura 2.1 si osserva che all'aumentare della distanza la colorazione dei pixel tende dal blu al rosso.



(a) (b)

Figura 2.1: Immagine originale destra (a) e immagine di disparità (b).

Ottenuta la DSI, la ricerca degli ostacoli viene effettuata aggregando le regioni a disparità costante. In particolare, essendo lo studio focalizzato sull'individuazione di oggetti paliformi, l'immagine di disparità viene suddivisa in colonne, contrassegnate con il valore predominante di disparità e scartando, per ciascuna colonna dell'immagine di disparità, tutti i valori che si discostano di più di una unità (parametro scelto arbitrariamente) dalla disparità predominante sulla colonna stessa.

I dati raccolti vengono così aggregati dando più valore a insiemi di colonne vicine contrassegnate dalla stessa disparità. L'insieme delle colonne considerate "vicine" è inversamente proporzionale alla distanza corrispondente al valore di disparità. Le colonne caratterizzate da una disparità isolata vengono eliminate.

Al termine dell'aggregazione, attraverso una soglia assegnata, vengono stabiliti gli insiemi che hanno raggiunto un valore tale da far ritenere più probabile la presenza di un ostacolo. La soglia viene quindi moltiplicata per un valore tanto più alto quanto più l'ostacolo è vicino, in quanto oggetti prossimi alle telecamere occupano una regione più grande dell'immagine [5], [6].

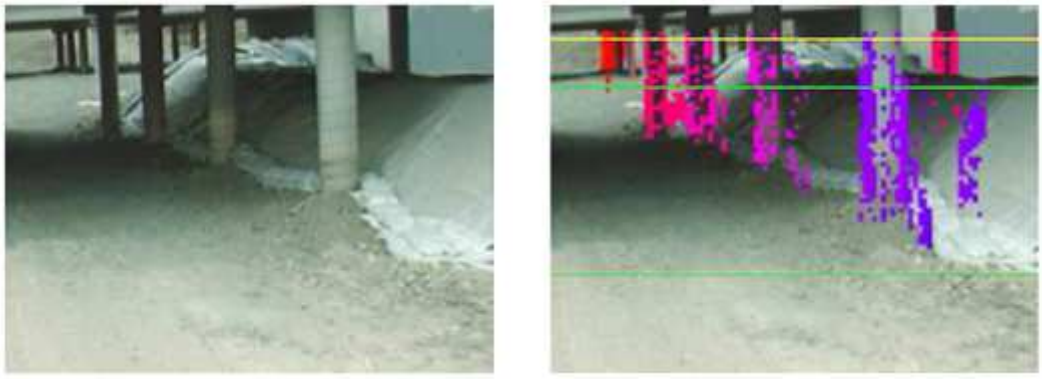


Figura 2.2: Rilevazione di colonne.

Tale algoritmo ha reso possibile l'individuazione di diversi tipi di ostacoli.

Figura 2.3: Rilevazione di paletti piantati nel terreno.

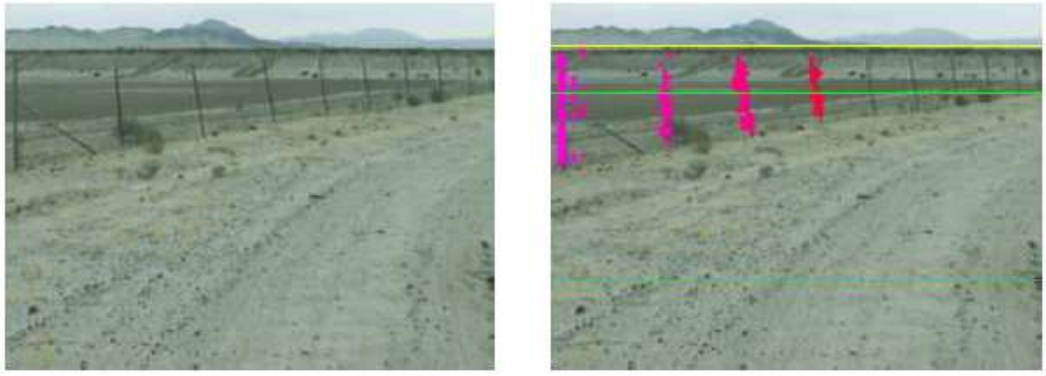


Figura 2.4: Rilevazione di paletti di una rete.

L'algoritmo è in grado di abbinare altrettanto facilmente gli elementi delle persone, in quanto sono caratterizzate da una forte componente verticale e da bordi marcati.

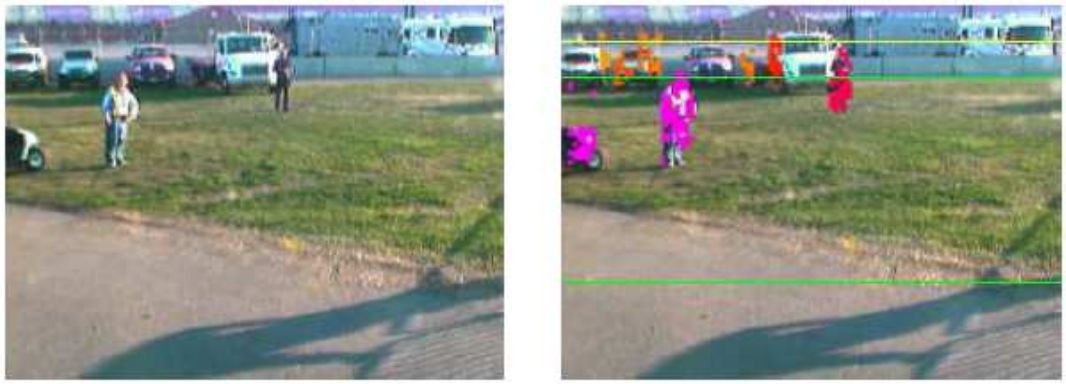


Figura 2.5: Rilevazione di persone.

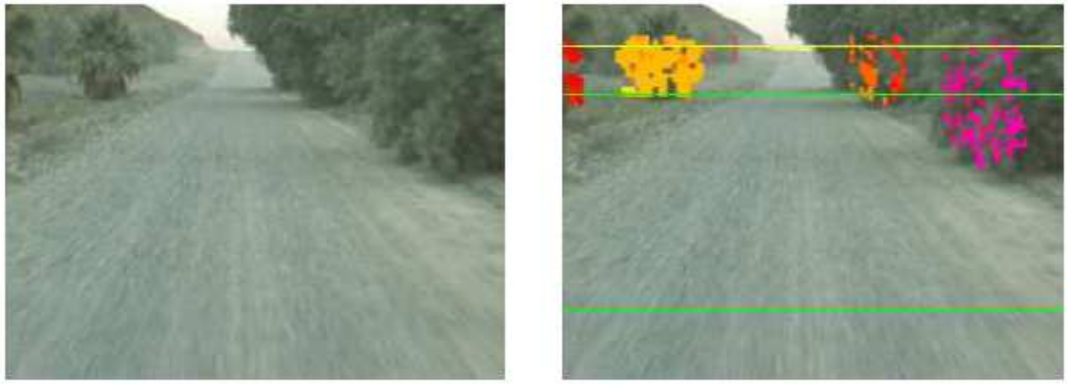


Figura 2.6: Rilevazione di una fila di alberi.

L'Università degli studi di Parma ha sviluppato un software denominato *GOLD (Generic Obstacle Lane Detection)* basato su una coppia di telecamere stereo, utilizzabile su veicoli autonomi per incrementare la sicurezza stradale. Tra le sue principali funzioni vi sono l'individuazione della posizione delle linee di corsia sulla carreggiata (Lane Detection) e il rilevamento di generici ostacoli sul percorso (Obstacle Detection).

Tali risultati sono ottenuti tramite l'elaborazione d'immagini acquisite da due telecamere (sistema di visione stereoscopico) e mediante una trasformazione geometrica, *l'Inverse Perspective Mapping (IPM)* che elimina l'effetto prospettico dalle immagini in ingresso.

Infatti, un problema nell'analisi delle immagini reali consiste nella distribuzione non uniforme delle informazioni nell'immagine stessa. A causa dell'effetto prospettico un pixel rappresenta un'area variabile dello spazio reale a seconda della distanza dal punto di ripresa: pertanto, lo stesso oggetto in una zona vicina alla telecamera risulta composto da un numero di pixel maggiore di quanto non sia in una zona più distante.

Per questo motivo le immagini sono state elaborate eseguendo un'operazione di Inverse Perspective Mapping che elimina l'effetto della prospettiva dall'immagine acquisita rimappandola in un nuovo dominio 2D in cui ciascun pixel rappresenta la stessa area spaziale.

Basandosi sull'ipotesi di strada piana si compie un'elaborazione che determina un'immagine ottenuta come differenza tra le immagini stereo trasformate tramite IPM; ogni valore non nullo della differenza e al di sopra di una certa soglia, rivela la presenza di possibili ostacoli. La diversa angolazione da cui la scena viene ripresa dalle telecamere fa in modo che un ostacolo a profilo quadrato fornisca degli insiemi di pixel a forma triangolare nell'immagine differenza, in corrispondenza dei suoi lati verticali. Il processo di determinazione degli ostacoli si basa proprio sulla localizzazione di coppie di questi triangoli.

In realtà, a causa delle forme irregolari e della luminosità non uniforme, la loro individuazione è molto complessa; tuttavia aree di pixel di forma quasi triangolare sono comunque presenti nell'immagine differenza e, grazie ad algoritmi che realizzano ed analizzano diagrammi polari su di essa, è comunque possibile rintracciarli ed ottenere la loro posizione nello spazio tridimensionale operando la trasformazione inversa dell'IPM.

I diagrammi polari sono caratterizzati dalla presenza di un picco in corrispondenza di ciascun triangolo e, poiché la presenza di un ostacolo da luogo nell'immagine differenza ad una coppia di triangoli disgiunti, la rilevazione di un ostacolo si riduce alla ricerca di coppie di picchi adiacenti. La forma, l'ampiezza e la vicinanza dei picchi consentono di raggruppare quelli appartenenti ad uno stesso ostacolo, ottenendo l'angolo di vista per ciascun ostacolo.

Infine, dall'immagine differenza e dalla posizione dei picchi nel diagramma polare è possibile stimare la distanza dell'ostacolo.

Per ciascun picco viene realizzato un istogramma radiale ottenuto esaminando una specifica regione dell'immagine differenza la cui larghezza dipende dalla larghezza del picco e calcolando, all'interno di quella regione, il numero di pixel che superano il valore di soglia stabilito al fine di individuare la posizione dell'angolo del triangolo che rappresenta il punto di contatto tra l'ostacolo e il piano stradale. Ciò consente di ottenere, così, l'informazione sulla distanza dell'ostacolo [7]-[9].

L'intero processo di individuazione dell'ostacolo è mostrato in Figura 2.7.

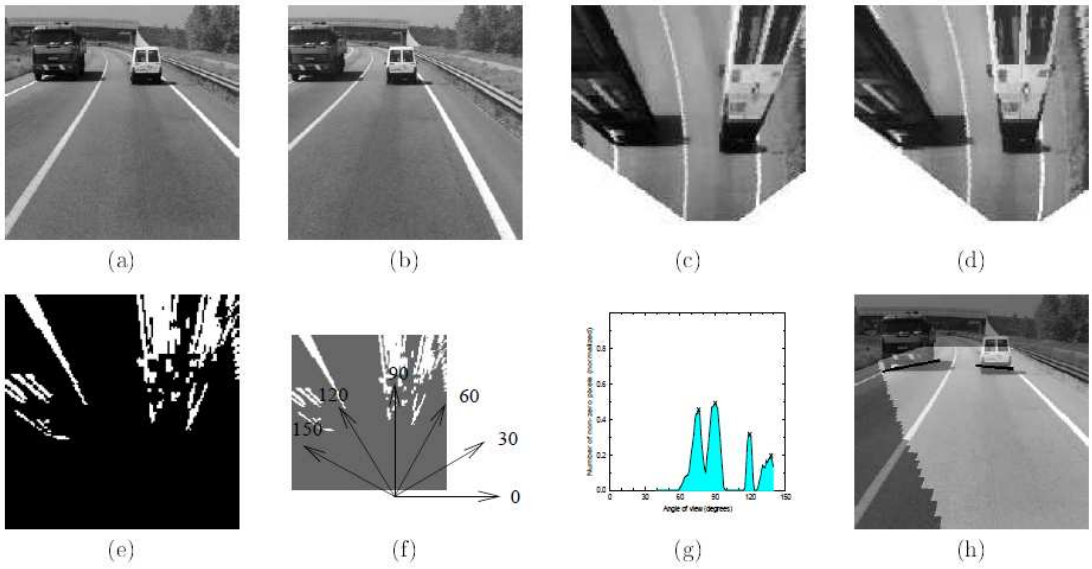


Figura 2.7: Rilevazione di ostacoli: (a) immagine stereo sinistra, (b) immagine stereo destra, (c) e (d) immagini rimappate, (e) immagine differenza, (f) angoli di vista sovrapposti all'immagine differenza, (g) istogramma polare e (h) risultato della rilevazione dell'ostacolo attraverso un segno nero sovrapposto all'immagine sinistra; l'area dell'immagine illuminata di grigio rappresenta la regione visibile da entrambe le telecamere.

In Figura 2.8 seguente sono mostrati i risultati ottenuti in diverse situazioni (con uno o due ostacoli di differente forma e colore).

Figura 2.8: Risultati della rilevazione di ostacoli in differenti condizioni stradali.

All'interno del software GOLD è, inoltre, implementato anche l'algoritmo di *Vehicle Detection* che si basa sul fatto che una vettura ha caratteristiche generalmente simmetriche e può essere individuata in una specifica regione dell'immagine che costituisce l'area d'interesse all'interno della quale eseguire la ricerca.

Dopo aver individuato il veicolo e averlo localizzato con un bounding box (un'area rettangolare, di cui sono note la posizione e le dimensioni, che racchiude al suo interno una regione dell'immagine di particolare interesse) si procede col determinare la sua distanza. Per fare questo si utilizza l'altra immagine per ricercarvi lo stesso contenuto del bounding box, determinando in quale posizione è massima la correlazione con l'immagine utilizzata in partenza: questo parametro riflette la distanza del veicolo e, grazie alla conoscenza dei parametri di calibrazione delle telecamere, è possibile calcolarla. L'inseguimento si realizza, quindi, massimizzando la correlazione tra il bounding box in una frame ed in quella successiva. Infatti, visualizzando una sequenza di frame la percezione visiva dell'occhio viene ingannata e, non riuscendo a distinguere i cambiamenti, crede di percepire una scena animata.

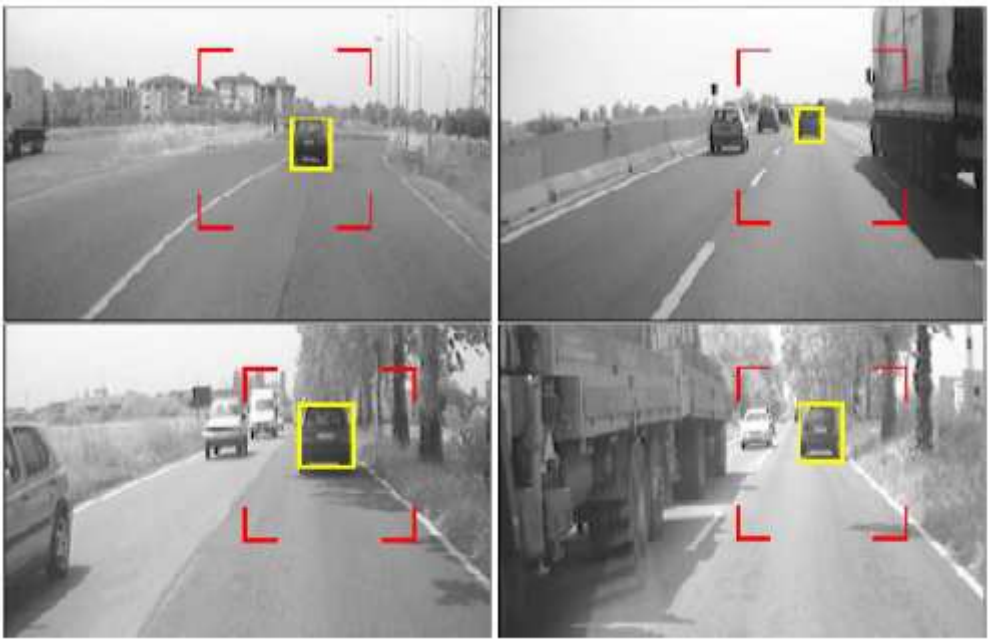


Figura 2.9: Esempi di rilevazione di veicoli.

CAPITOLO 3

AMBIENT AWARENESS FOR AUTONOMOUS AGRICULTURAL VEHICLES

3.1 Obiettivi del progetto

L'Università del Salento, insieme a partner europei come il Centro di Ricerca francese Cemagref, il Danish Technology Institute (DTI) di Odense, il Fraunhofer IAIS di Bonn e il produttore di macchinari agricoli tedesco CLAAS (secondo produttore mondiale di trattori), è coinvolta in un progetto europeo denominato "*QUAD-AV*" (*Ambient Awareness for Autonomous Agricultural Vehicles*) per la realizzazione di un trattore a guida autonoma.

Obiettivo del progetto è quello di sviluppare sistemi e metodi di percezione finalizzati al miglioramento del livello di sicurezza e di autonomia di veicoli agricoli intelligenti. In particolare, la ricerca proposta è focalizzata sul problema della rilevazione e del riconoscimento degli ostacoli in campo agricolo. Si investigheranno varie tipologie sensoriali e approcci multi modali. L'idea è quella di usare un sistema composto da più sensori che, utilizzati in modo integrato e complementare, possano compensare a vicenda i propri limiti applicativi e fornire misure accurate e robuste in tutte le condizioni ambientali.

Obiettivi specifici del progetto sono:

- Selezionare, modificare e sviluppare un set di sensori con elevato potenziale ai fini dello sviluppo di sistemi di percezione avanzata per veicoli agricoli autonomi;
- Preparare l'infrastruttura (protocolli e formati di dati) per una soluzione multi-sensoriale;
- Implementare il sistema su un veicolo all-terrain destinato ad applicazioni agricole;

- Identificare un set di scenari agricoli rappresentativi per la verifica sperimentale del sistema;
- Condurre una campagna prove sul campo.

3.2 Descrizione del progetto

Questa sezione illustra l'attività di ricerca svolta presso l'Università del Salento nell'ambito del progetto QUAD-AV "Ambient Awareness for Autonomous Agricultural Vehicles". Il documento descrive le attività svolte e i risultati ottenuti riguardo allo sviluppo di un sistema multisensoriale finalizzato al riconoscimento automatico di ostacoli in campo agricolo. In particolare, l'Università del Salento si è occupata dello studio di un sistema tipo multi – stereoscopico a baseline fissa, adatta alla particolare applicazione che è stato implementato e testato in laboratorio e, in seguito, verificato sperimentalmente sul campo. Quest'ultima fase è stata realizzata in una campagna prove condotta, insieme agli altri partner del progetto, presso una fattoria a Helsingør, Danimarca, dal 27 al 29 Settembre 2011.

Nel seguito di questa sezione, è fornita una descrizione del contesto generale in cui il progetto si colloca anche in relazione allo stato dell'arte, mentre le attività svolte e i risultati sono presentati in dettaglio nella sezione successiva.

Negli ultimi anni, l'automazione del settore agricolo è diventata un'esigenza di importanza prioritaria. La maggior parte degli sforzi della ricerca scientifica in tale ambito si è concentrata sui compiti di raccolta di frutta e verdura, che sono, generalmente, dispendiosi in termini di tempo, forza lavoro e costi. In molte coltivazioni, il lavoro di raccolta copre dalla metà ai due terzi del costo di lavoro totale. Inoltre, l'automazione delle operazioni di raccolta si rende sempre più necessaria in conseguenza della riduzione della popolazione agricola [10]. Quello dei veicoli autonomi è un particolare tipo di automazione dei sistemi di coltivazione, avente l'obiettivo di migliorare la produttività e l'efficienza. Numerosi gruppi di ricerca internazionali hanno sviluppato veicoli autonomi per il settore agricolo. Essi utilizzano tecnologie come sistemi di visione, GPS e sistemi inerziali per navigare attraverso

campi e frutteti, svolgere operazioni di semina, coltivazione e altri compiti specifici che richiedono intelligenza. Ad esempio, i ricercatori del National Robotics Engineering Consortium hanno sviluppato un trattore senza guidatore che utilizza un controllo elettronico di tipo differential drive e tecniche di machine vision per tagliare alfalfa [11]. Hague e Tillett [12] hanno incorporato tecniche di visione e sensori inerziali su un veicolo per orticoltura che utilizza i filari della coltivazione come supporto alla navigazione. Un sistema di raccolta robotizzato che impiega una telecamera posizionata sull'end-effector per la raccolta di piante di radicchio è presentato in [13]. Un esempio di sistema di guida autonoma basato su GPS è il John Deere's AutoTrac system [14]. Altri esempi notevoli di robot agricoli sono presentati in [15]-[17].

Benché questi sistemi si siano dimostrati efficaci sul campo, essi sono ben lontani dal poter essere definiti veicoli autonomi. La maggior parte dei lavori si è focalizzata sullo sviluppo di sistemi di controllo di traiettoria al fine di migliorare la precisione di guida di veicoli e manipolatori e si basano su uno stile industriale di coltivazioni agricole in cui si assume che tutto sia noto a priori e che le macchine lavorino in modo predefinito come in una tipica linea di produzione. La sfida è quella di sviluppare veicoli intelligenti, completamente autonomi, in grado di operare in ambienti popolati che si modificano nel tempo. Per raggiungere tale obiettivo, sono necessari ulteriori sforzi da parte della ricerca scientifica volti a sviluppare sistemi sensoriali che forniscano ai veicoli una chiara percezione dell'ambiente circostante.

Un problema chiave per la reale utilizzazione dei veicoli autonomi in campo agricolo è quello della sicurezza sia del veicolo stesso sia delle persone, degli animali e delle cose che si trovano nell'ambiente operativo del veicolo. A tal fine è fondamentale che il veicolo sia in grado di rilevare e riconoscere gli ostacoli. Questo compito è particolarmente difficile per un veicolo agricolo, in quanto esso può incontrare sul campo numerosi ostacoli visibili o nascosti, statici o dinamici.

L'idea del progetto QUAD-AV è quella di utilizzare diverse modalità sensoriali e metodi multi-algoritmici per rilevare varie tipologie di ostacoli e costruire un database che possa essere impiegato per il controllo di un veicolo agricolo autonomo. Il progetto si propone di investigare quattro tecnologie: stereo visione, radar, ladar e termografia. In

particolare, l'Università del Salento si occupa dell'acquisizione ed elaborazione di immagini prodotte da un sistema stereo trinoculare.

La visione è uno degli input sensoriali più utili per la rilevazione ed il riconoscimento degli ostacoli. I metodi basati su visione proposti in letteratura possono essere suddivisi in due categorie: metodi monoculari e metodi stereovisivi. Gli approcci basati su telecamere monoculari sono fondati sull'assunzione che gli ostacoli sono oggetti che differiscono nel loro aspetto (appearance) dal terreno [18]. Comprendono, generalmente, una fase di segmentazione basata su colore o tessitura, seguita da stima delle proprietà 3D degli oggetti sfruttando la conoscenza a priori della geometria degli oggetti o del terreno (ad esempio assunzione di terreno piano). La stereo visione consente, invece, di stimare direttamente le coordinate 3D delle features nell'immagine fornendo al tempo stesso importanti informazioni circa l'appearance degli oggetti come il colore e la tessitura che possono essere poi usate per la classificazione. Le tecniche di computer vision 3D sono ideali per individuare e categorizzare oggetti visibili sul campo a varie distanze a seconda della baseline delle telecamere. Diversi lavori in letteratura hanno dimostrato l'efficacia e la robustezza dei sistemi di stereo visione per la rilevazione di ostacoli su veicoli off-road [19]-[22]. In campo agricolo, la stereo visione può essere utilizzata per individuare ostacoli, apprendendo ed utilizzando modelli di colore e tessitura dell'ambiente cosi come caratteristiche 3D di terreno ed oggetti. I dati range registrati simultaneamente alle informazioni di apparenza rendono un veicolo agricolo potenzialmente in grado di rilevare ostacoli distinguendoli dal terreno e di determinarne l'appartenenza a classi di interesse come persone, alberi, foglie, animali ecc.

3.3 Attività svolte

L'attività svolta presso l'Università del Salento nell'ambito del progeto QUAD-AV ha riguardato lo sviluppo di un sistema di visione trinoculare. In questa fase sono stati definiti i requisiti del sistema stereo in relazione agli obiettivi specifici da raggiungere ed alle caratteristiche generali del contesto applicativo. A conclusione delle analisi effettuate è stata operata la scelta di un sistema trinoculare pre-calibrato da utilizzare nella fase di test iniziale dei sistemi, per la prima campagna prove sul campo e per lo studio preliminare degli algoritmi di segmentazione e classificazione. Particolare attenzione è stata rivolta alla scelta della baseline. La baseline ottimale di un sistema stereo dipende da due fattori contrapposti, ma ugualmente importanti: campo di vista e massimo range. Una baseline più corta aumenta il campo di vista comune alle due telecamere, ma consente un massimo range ridotto. Viceversa, una baseline più lunga riduce il campo di vista comune, ma consente un massimo range più elevato e maggiore precisione a distanze maggiori. Al fine di combinare i vantaggi di due diverse baseline, si è scelto di utilizzare un sistema multi-baseline trinoculare anziché un sistema binoculare standard. Dopo una analisi dei sistemi disponibili in commercio, si è optato per l'acquisto della telecamera Bumblebee XB3 della PointGrey (*http://www.ptgrey.com/*) (Figura 3.1). Si tratta di un sistema multi-baseline a tre sensori a colori da 1.3 mega-pixel IEEE-1394b (800 Mb/s) con due diverse baseline e ottiche a 3.8 mm. La baseline maggiore, pari a 24 cm, consente maggiore precisione a distanze più elevate mentre la baseline più corta migliora il matching a distanze ravvicinate. Le specifiche sono riportate in Tabella 3.1.

Figura 3.1: Bumblebee XB3, PointGrey.

Sensor	Three Sony 1/3" progressive scan CCDs, Color
Baseline	12 cm and 24 cm
Resolution and FPS	1280 x 960 at 15 FPS
Focal Lengths	3.8 mm with 66° HFOV
Aperture	f/2.0
A/D Converter	12-bit analog-to-digital converter
White Balance	Manual
Video Data Output	8 and 16-bit digital data
Interfaces	2 x 9-pin IEEE-1394b for camera control and video data transmit; 4 general-purpose digital input/output (GPIO) pins
Voltage Requirements	8-32 V via IEEE-1394 interface or GPIO connector
Power Consumption	4W at 12V
Gain	Automatic/Manual
Shutter	Automatic/Manual, 0.01 ms to 66.63 ms at 15 FPS
Trigger Modes	DCAM v1.31 Trigger Modes 0, 1, 3, and 14
Signal To Noise Ratio	54 dB
Dimensions	277 x 37 x 41.8 mm
Mass	505 grams
Camera Specification	IIDC 1394-based Digital Camera Specification v1.31
Lens mount	3 x M12 microlens mount
Emissions Compliance	Complies with CE rules and Part 15 Class A of FCC Rules
Operating Temp.	Commercial grade electronics rated from 0° to 45°C

Tabella 3.1: Specifiche tecniche del sistema trinoculare Bumblebee XB3.

Sono stati sviluppati i codici di acquisizione e calibrazione della telecamera Bumblebee XB3. Per il software di acquisizione sono state utilizzate le librerie FlyCapture e Triclops della PointGrey e OpenCV. Per la calibrazione è stato utilizzato il Matlab Calibration Toolbox (*http://www.vision.caltech.edu/bouguetj/calib_doc/*).

Software di acquisizione

Il software è stato sviluppato in Visual Studio 2008 e comprende le seguenti applicazioni basate su libreria Microsoft Foundation Class (MFC) con finestre di dialogo:

1) XB3_Acquisition_Interface: questa applicazione consente di gestire l'acquisizione delle immagini;
2) ConvertRawAndShow: questa applicazione consente di convertire le immagini da formato raw a formato bmp e visualizzare le immagini generando anche un video della sequenza.

Entrambi i progetti implementano funzionalità di interfaccia utente standard di Windows, come pulsanti di ingrandimento, di riduzione a icona e chiusura finestra, e pulsanti o campi per l'immissione di testo e valori numerici finalizzati alla gestione delle funzionalità della telecamera (es. start/stop acquisizione, set up dei parametri di acquisizione, etc.) e delle immagini acquisite (es. formato e nome dei file).

Calibrazione

La telecamera Bumblebee XB3 è una telecamera pre-calibrata. Pertanto i parametri di calibrazione intrinseci ed estrinseci per la ricostruzione stereo sono noti. Tuttavia, si è resa necessaria un'ulteriore fase di calibrazione al fine di stimare la matrice di trasformazione tra il sistema di riferimento solidale alla telecamera (camera frame) e il sistema di riferimento solidale al veicolo (vehicle frame) (Figura 3.2).
A tal fine è stato utilizzato il Camera Calibration Toolbox for Matlab. Questo toolbox utilizza un pattern di calibrazione planare a scacchiera e produce in output il file Calib_Results.m contenente i parametri di calibrazione stimati. La calibrazione è stata realizzata con riferimento alla telecamera centrale usando una griglia planare di 1.5 m x 1 m e quadrati di 10 cm, come mostrato in Figura 3.3. Assumendo i tre sensori identici e nota la posizione relativa tra essi è poi possibile riferirsi ad una qualsiasi delle tre telecamere una volta nota la matrice di trasformazione per la telecamera centrale.

Sono stati stimati i parametri estrinseci ovvero i parametri di rotazione e traslazione che determinano la trasformazione tra camera frame e vehicle frame. A tal fine, è stata dapprima stimata la trasformazione tra sistema di riferimento solidale alla griglia di calibrazione (grid frame) e camera frame mediante il Toolbox di Matlab. Successivamente, la posizione del grid frame rispetto al body frame è stata misurata manualmente posizionando la griglia parallelamente alla strada. Il risultato della calibrazione è riportato in Tabella 3.2. La tabella riporta gli offset di traslazione (X_C, Y_C, Z_C) e rotazione (φ, θ, ψ), che descrivono come muovere il vehicle frame per allinearlo con il camera frame, espressi nel vehicle frame. Le distanze sono espresse in metri e gli angoli in gradi. La matrice di trasformazione da telecamera a veicolo è la seguente:

$$T = \begin{bmatrix} c\psi \cdot c\theta & c\psi \cdot s\theta \cdot s\varphi - s\psi \cdot c\varphi & c\psi \cdot s\theta \cdot c\varphi + s\psi \cdot s\varphi & X_C \\ s\psi \cdot c\theta & s\psi \cdot s\theta \cdot s\varphi + c\psi \cdot c\varphi & s\psi \cdot s\theta \cdot c\varphi - c\psi \cdot s\varphi & Y_C \\ -s\theta & c\theta \cdot s\varphi & c\theta \cdot c\varphi & Z_C \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

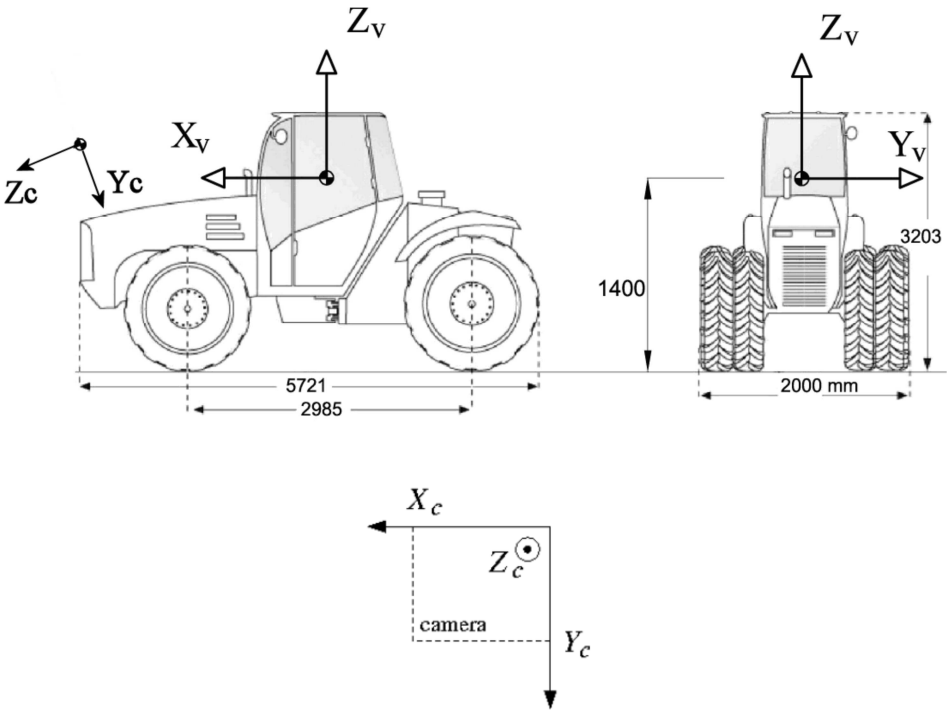


Figura 3.2: Sistemi di riferimento veicolo e telecamera.

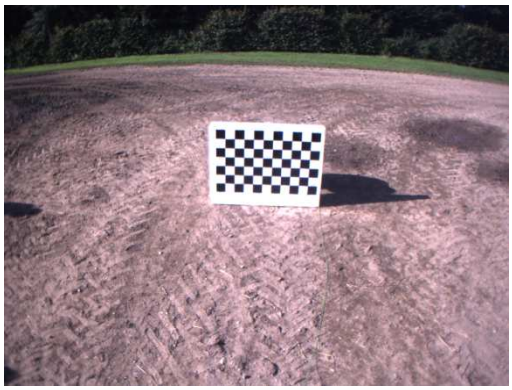


Figura 3.3: Griglia di calibrazione.

$X_C\ [m]$	$Y_C\ [m]$	$Z_C\ [m]$	$\varphi\ [deg]$	$\theta\ [deg]$	$\psi\ [deg]$
3	0	0.57	-117.14	0.59	-86.14

Tabella 3.2: Parametri di trasformazione da camera frame a vehicle frame espressi nel sistema di riferimento veicolo.

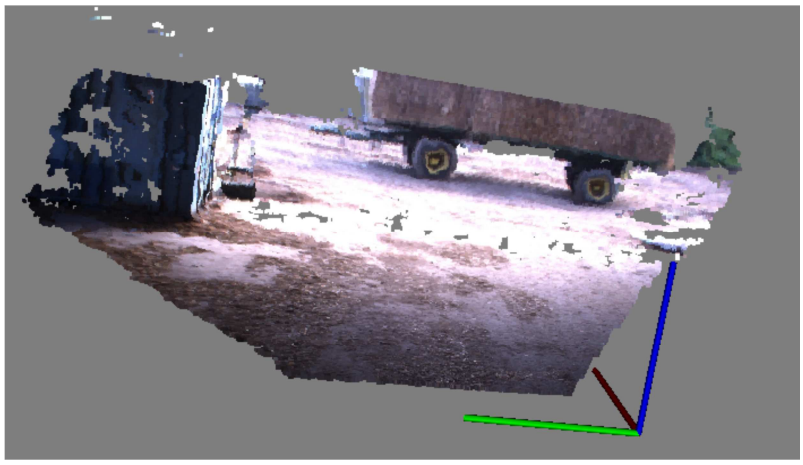
3.4 Validazione sperimentale del sistema trinoculare

Il sistema stereoscopico trinoculare è stato validato durante la prima campagna prove del progetto, realizzata presso una fattoria vicino Helsingør, Danimarca, in cui i diversi sensori sono stati montati a bordo del trattore CLAAS AXION 840 4WD (Figura 3.4). Alcune immagini ottenute dal sistema trinoculare con una risoluzione di 640 × 480 pixels sono mostrate in Figura 3.5. Le prove sono state realizzate guidando il trattore con un operatore umano a diverse velocità comprese tra 2 e 15 km/h mentre i sensori acquisivano informazioni dall'ambiente circostante. Differenti scenari agricoli sono stati analizzati con ostacoli di tipo positivo (muri, altri veicoli, tralicci della luce), di tipo negativo (depressioni del terreno, dirupi), ostacoli umani e animali e in presenza di terreni particolarmente difficili (fango, terreni cedevoli). Le prove sono state ripetute a diverse condizioni di luminosità (mattina, pomeriggio, tramonto).

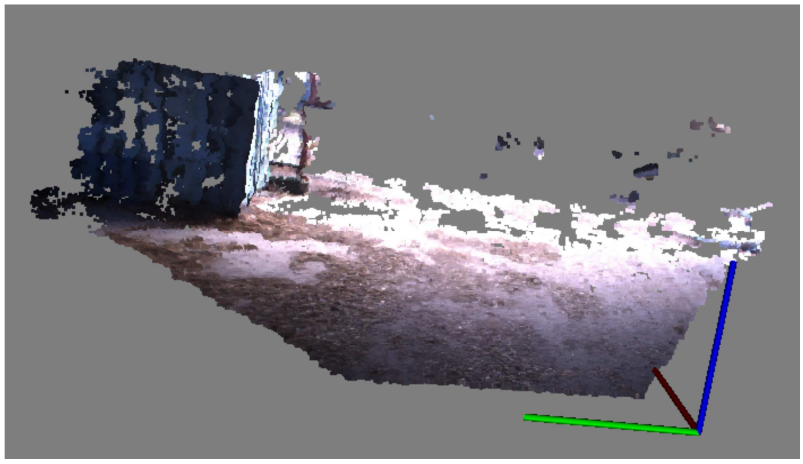
Figura 3.4: CLAAS AXION 840 con il set di sensori di bordo.

Figura 3.5: Immagini acquisite durante la campagna prove di Helsingør, Danimnarca.

(a)



(b)



(c)

Figura 3.6: (a) Immagini acquisite dalle tre telecamere; Stereo ricostruzione della scena 3D: (b) wide baseline (left+right cameras), e (c) narrow baseline (right+central camera).

I punti ricostruiti dalle telecamere stereo sia con baseline larga che stretta sono fusi in un'unica nuvola di punti utilizzando un unico sistema di riferimento centrato nella telecamera destra. Quando un punto della scena viene ricostruito sia da una coppia di telecamere con baseline lunga che da una con baseline corta, solo l'informazione proveniente dalla coppia con baseline lunga viene conservata, in quanto una baseline più lunga assicura una migliore accuratezza ad ogni distanza. Ciò consente di combinare il minimo range della baseline corta con la migliore accuratezza e il massimo range della baseline lunga.

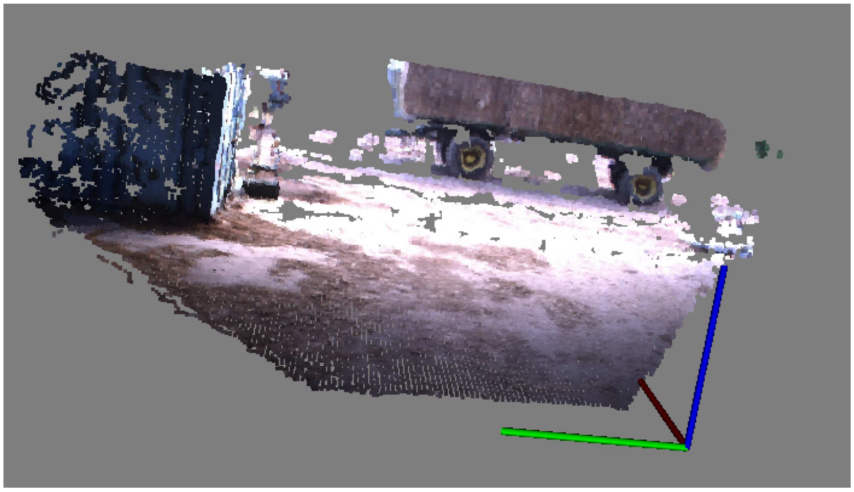


Figura 3.7: Ricostruzione 3D della scena mediante multiplexing della baseline maggiore e minore.

CAPITOLO 4

ALGORITMO

4.1 Descrizione dell'algoritmo

La navigazione autonoma di un robot mobile all'interno di una scena 3D richiede una caratterizzazione attenta ed efficiente dell'intera struttura della scena percepita, come ad esempio una mappa di attraversabilità che possa essere esplorata da algoritmi di pianificazione di un percorso.

La maggior parte degli algoritmi presenti in letteratura adottano l'ipotesi di mondo piatto ed il riconoscimento degli ostacoli consiste nell'identificare gli oggetti che "sporgono" dal terreno (considerati come ostacoli positivi). Tuttavia, in ambienti non strutturati come terreni erbosi, tali assunzioni non risultano in genere valide.

L'approccio utilizzato nella tesi per il riconoscimento degli ostacoli è applicabile sia ad ambienti strutturati (come la strada), sia ad ambienti non strutturati (come terreni erbosi).

In contrasto con i lavori presenti in letteratura che mirano alla creazione di algoritmi in grado di identificare in maniera esplicita gli ostacoli presenti nella scena, quello realizzato si propone di riconoscere in maniera esplicita solo quelle regioni della scena che sono "attraversabili" e, quindi, sicure per il movimento del trattore dalla posizione attuale. Ciò consente al trattore di evitare, durante il suo movimento, gli ostacoli sia positivi che negativi, i quali, di fatto, non vengono riconosciuti esplicitamente.

Il concetto di attraversabilità può essere facilmente compreso definendo una griglia 2D di celle. Considerata la nuvola di punti 3D acquisita da ciascuna delle telecamere montate sul trattore, questi vengono assegnati a ciascuna cella e per ciascuna di esse vengono realizzati degli istogrammi di elevazione dei punti appartenenti. L'informazione relativa all'altezza di ciascun punto viene, quindi, utilizzata per realizzare un algoritmo in grado di riconoscere e classificare le celle come "attraversabili" ed identificare indirettamente gli ostacoli nella scena.

La mappa degli ostacoli presenti nella scena è stata realizzata mediante l'ausilio del software *Matlab®*, definendo una griglia di riferimento parallela al piano orizzontale nel sistema di coordinate della telecamera con origine nel punto di coordinate $(4.6, 0, -1.97)\ m$. La griglia, che è simmetrica rispetto all'asse x, è caratterizzata da $4.6\ m \leq x \leq 22\ m$ e da $-9\ m \leq y \leq 9\ m$.

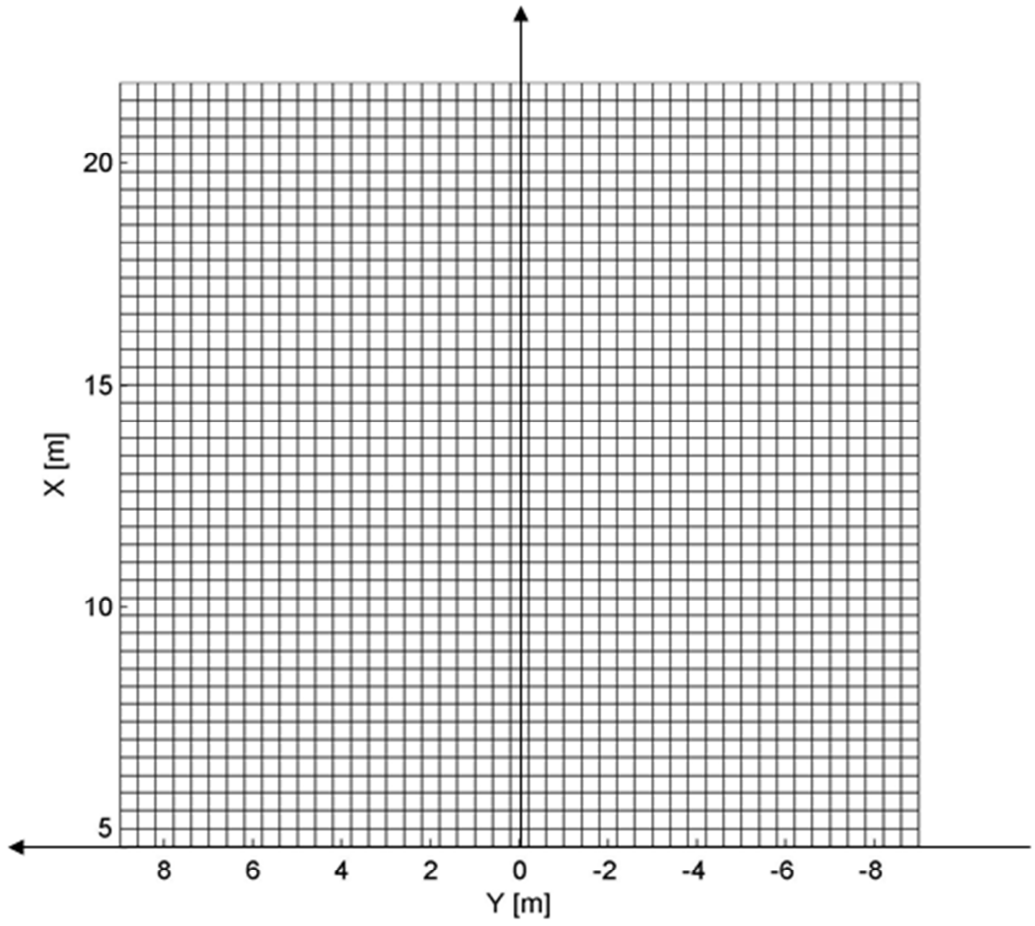


Figura 4.1: Griglia 2D di celle.

Tale griglia è stata poi suddivisa in celle quadrate uguali e di lato $r = 0.4\ m$ con l'obiettivo di classificare ciascuna di esse come "attraversabile", "ostacolo" o "non definita".

In seguito, sono stati caricati i dati relativi alla nuvola di punti 3D acquisita dalla telecamera trinoculare e memorizzati al termine dell'acquisizione all'interno di un file con estensione *.pcd*. I punti della nuvola 3D sono stati successivamente plottati relativamente alla regione di interesse definita dalla griglia come in Figura 4.2.

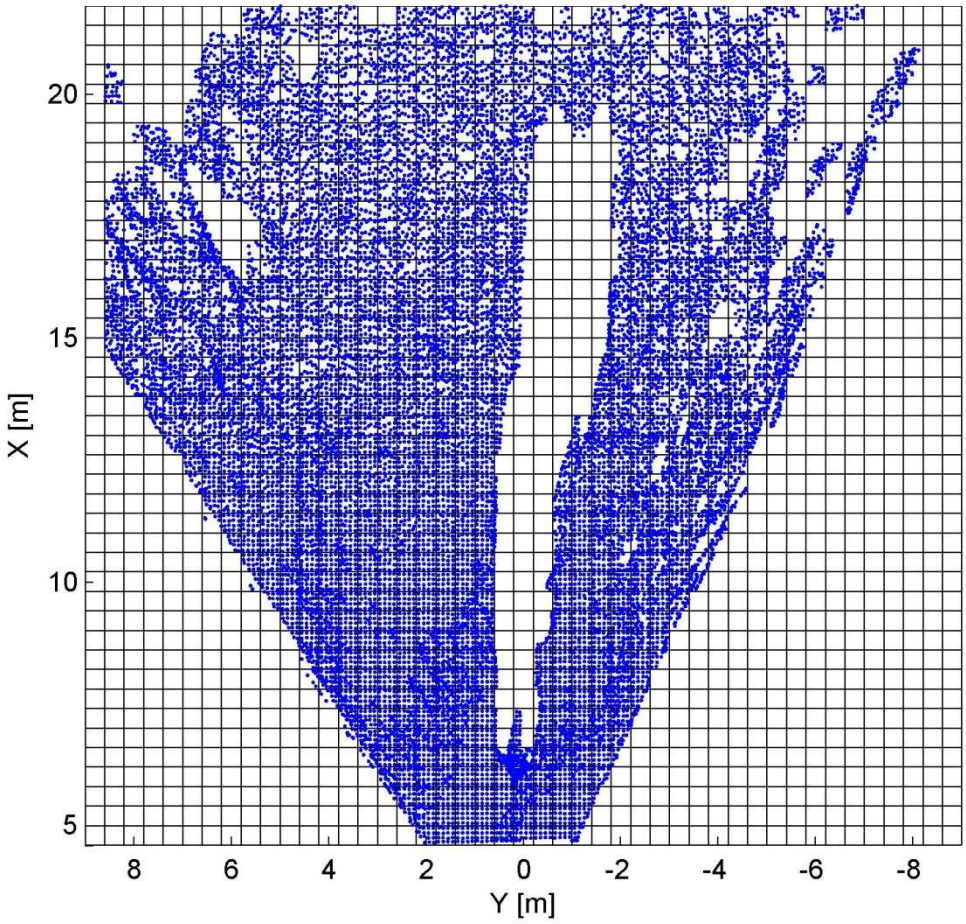


Figura 4.2: Vista nel piano XY della nuvola di punti 3D acquisita dal sistema trinoculare.

L'algoritmo consiste in cinque fasi principali di seguito elencate ed analizzate successivamente in maniera dettagliata:

1. Assegnazione dei punti 3D della scena a ciascuna cella della griglia di riferimento;
2. Determinazione degli istogrammi di elevazione dei punti appartenenti a ciascuna cella;
3. Riconoscimento ed eliminazione delle strutture sporgenti sopraelevate (ad es. rami di alberi, sottopassaggi, ecc.);
4. Classificazione delle celle;
5. Riproiezione dei punti sull'immagine.

4.1.1 Assegnazione dei punti 3D della scena a ciascuna cella della griglia

Il primo passo è quello di individuare sulla griglia 2D la cella corrispondente a ciascuno dei punti 3D acquisiti da ciascuna telecamera montata sul trattore.

In primo luogo, si è creato un vettore GP (Grid Points) contenente le informazioni sulle coordinate dei nodi della griglia, i quali sono stati numerati come segue:

$$GP = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ \vdots & \\ x_i & y_i \\ \vdots & \\ x_n & y_n \end{bmatrix} \begin{matrix} \rightarrow 1 \\ \rightarrow 2 \\ \rightarrow 3 \\ \\ \rightarrow i \\ \\ \rightarrow n \end{matrix}$$

Di seguito è rappresentato il criterio di numerazione adottato per i nodi della griglia.

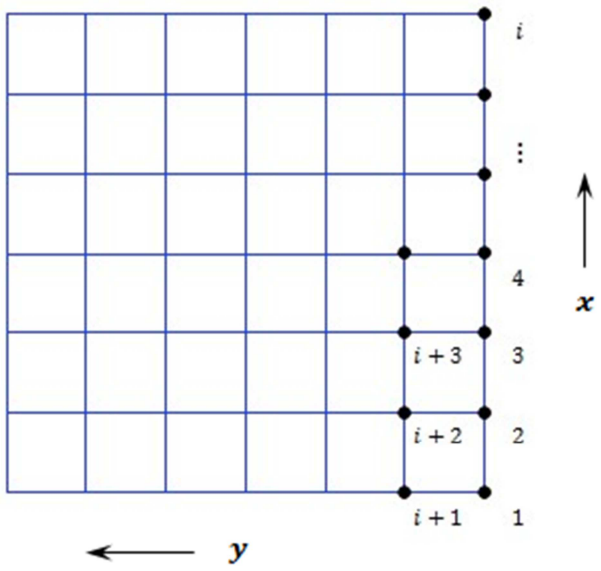


Figura 4.3: Criterio di numerazione dei nodi della griglia.

Pertanto, considerata una generica cella i ed indicati con (x_i, y_i), (x_{i+1}, y_i), (x_i, y_{i+i}) e (x_{i+1}, y_{i+i}) i vertici della stessa, si sono ricercati tutti i punti $P(x, y, z)$ della nuvola 3D di modo che le coordinate x ed y fossero tali da soddisfare le due seguenti relazioni:

$$\begin{cases} x_{i+1} \leq x < x_i \\ y_i \leq y < y_{i+1} \end{cases}$$

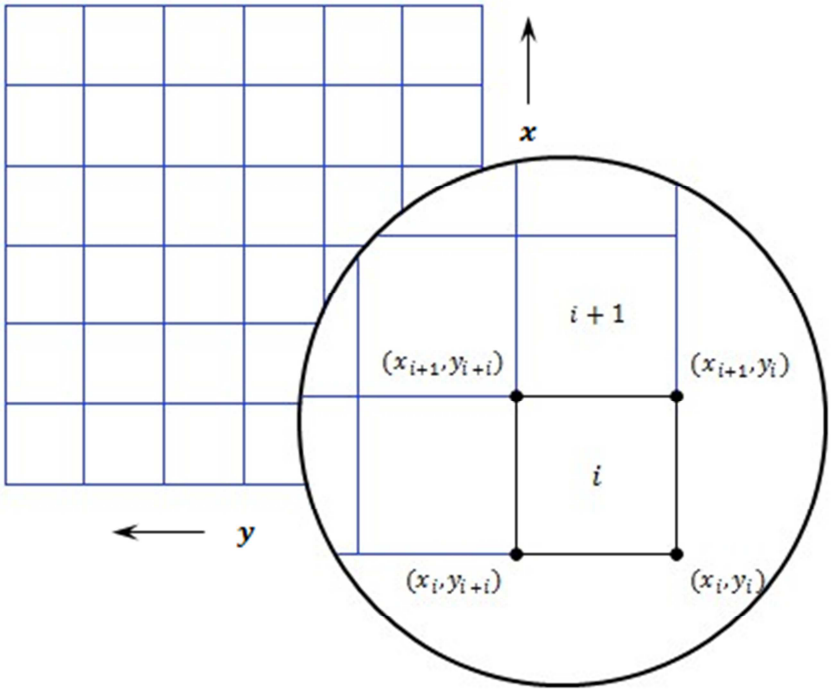


Figura 4.4: Caratterizzazione di una generica cella i della griglia.

In definitiva si è creata una struttura di dati caratterizzata da un numero di matrici pari al numero di celle contenenti tutti i punti che cadono all'interno di ciascuna di esse.
Ciascuna matrice è caratterizzata da sei colonne: le prime tre relative alle coordinate x, y e z dei punti appartenenti alla cella e le altre tre alle componenti R, G e B relative al colore. Ciascuna delle matrici è stata indicata con il nome di $Patch\ \{i\}$, essendo i la i-esima cella presa in esame.
In Figura 4.5 sono rappresentati tutti i punti della nuvola 3D appartenenti alla cella evidenziata sulla griglia.

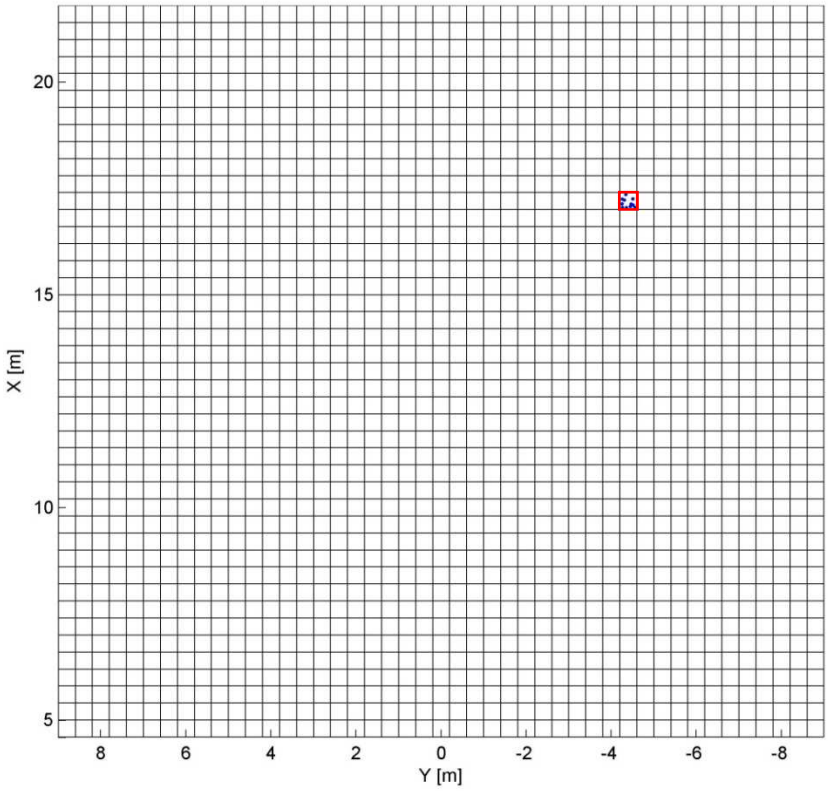


Figura 4.5: Visualizzazione dei soli punti appartenenti ad una generica cella della griglia.

Ripetendo lo stesso procedimento per tutte le celle, è stato possibile conoscere i punti 3D appartenenti a ciascuna cella. Le celle sono state infine numerate come in Figura 4.6.

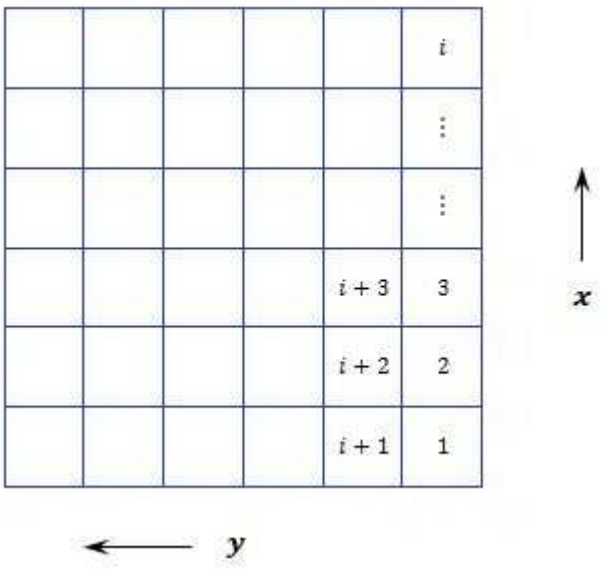


Figura 4.6: Criterio di numerazione delle celle della griglia.

4.1.2 Determinazione degli istogrammi di elevazione per ciascuna cella

Il secondo passo dell'algoritmo è stato quello di determinare per ciascuna cella della griglia gli istogrammi di elevazione (coordinata z) di tutti i punti 3D appartenenti a quella cella. Tutti gli istogrammi sono caratterizzati da un certo numero di bin, ciascuno dei quali è caratterizzato da un'altezza $b = 0.1\ m$.

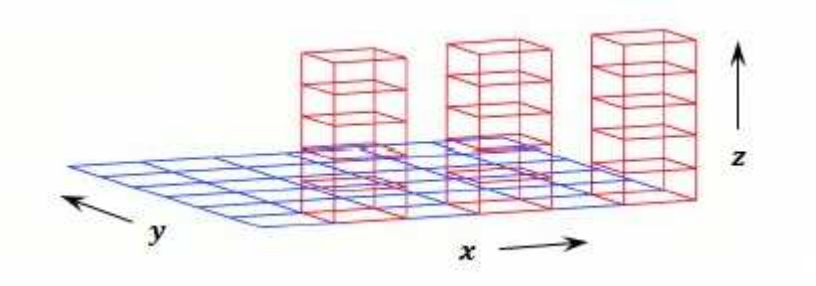


Figura 4.7: Visualizzazione dei bin degli istogrammi di elevazione per alcune celle della griglia.

Ciascun punto della nuvola 3D apparterrà, pertanto, ad un particolare bin dell'istogramma della cella corrispondente. In tal modo è stato possibile realizzare un istogramma di elevazione per ciascuna cella.

In pratica, per ciascuna cella sono stati realizzati degli istogrammi i cui i dati vengono visualizzati mediante colonne. La lettura di tali istogrammi è piuttosto semplice. In ascissa è riportata l'altezza da terra di tutti i punti che cadono in ciascuna cella ed in ordinata il numero di punti.

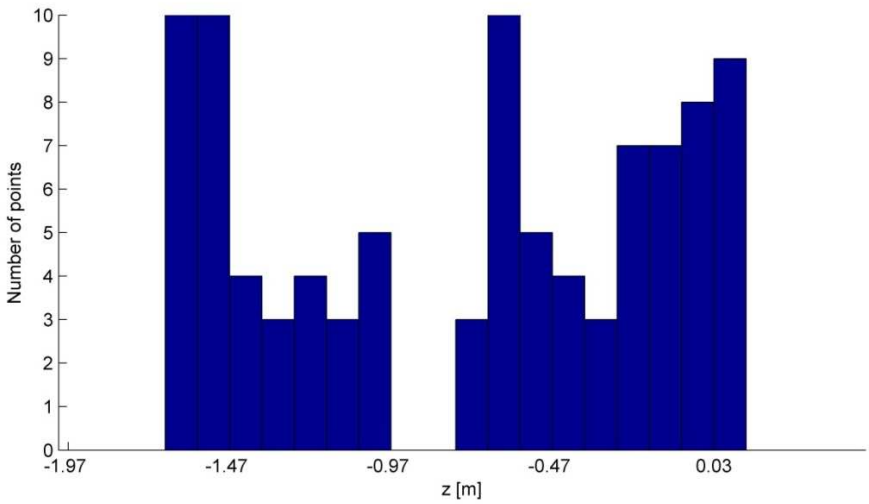


Figura 4.8: Esempio di istogramma di elevazione dei punti di una cella.

A partire da terra ($z = -1.97\ m$), per ogni 0.1 m di altezza, si è espresso in forma di istogramma il numero di punti aventi un valore della coordinata z contenuto nel range considerato. Pertanto, ciascuna colonna è rappresentativa di un bin dell'istogramma di elevazione per una determinata cella. L'assenza di colonne nell'istogramma è indicativo della presenza di bin vuoti.

Inoltre, al fine di rendere l'algoritmo più robusto ed eliminare l'effetto di outlier isolati nella nuvola di punti 3D, nell'ipotesi in cui un bin di un istogramma sia caratterizzato da un numero di punti inferiore ad un valore α_l assegnato, esso viene invalidato e settato pari a zero. Nel caso in esame il valore di α_l è stato settato pari ad 1, in quanto si è ritenuto che la presenza di un solo punto all'interno di un bin non fosse significativo ai fini del riconoscimento di un ostacolo.

4.1.3 Riconoscimento ed eliminazione delle strutture sporgenti sopraelevate

Per capire l'utilità di questo terzo passo dell'algoritmo si supponga di considerare uno scenario in cui si abbiano celle aventi alcuni punti 3D in prossimità o sul terreno ed altri piuttosto distanti dallo stesso e costituiti, per esempio, da un ramo sporgente di un albero. Se il ramo si trova ad un'altezza ben maggiore di quella del trattore, quest'ultimo può transitare sotto il ramo in sicurezza ed in tal caso i punti 3D costituenti il ramo vengono ignorati al fine del riconoscimento dell'ostacolo.

Tali strutture sopraelevate possono essere facilmente riconosciute esaminando tutti i bin di ciascun istogramma di elevazione partendo dal bin non nullo più basso nella cella e procedendo verso l'alto. Se durante la scansione si osserva la presenza di una serie di bin vuoti (Figura 4.9) la cui altezza complessiva risulta essere maggiore dell'altezza del trattore, tutti i bin non vuoti dell'istogramma al sopra di questi vengono settati pari a zero. I rimanenti bin non vuoti dell'istogramma consentono di determinare l'altezza del punto più alto nella cella, informazione utile per il riconoscimento dell'ostacolo.

Nel caso in esame, essendo l'altezza del veicolo pari 3.203 m e l'origine della telecamera situata ad un'altezza da terra pari a 1.97 m, si sono eliminati per ciascuna

cella tutti i punti aventi una coordinata z superiore a 1.433 m da terra, avendo considerato un fattore di sicurezza pari a 0.2 m.

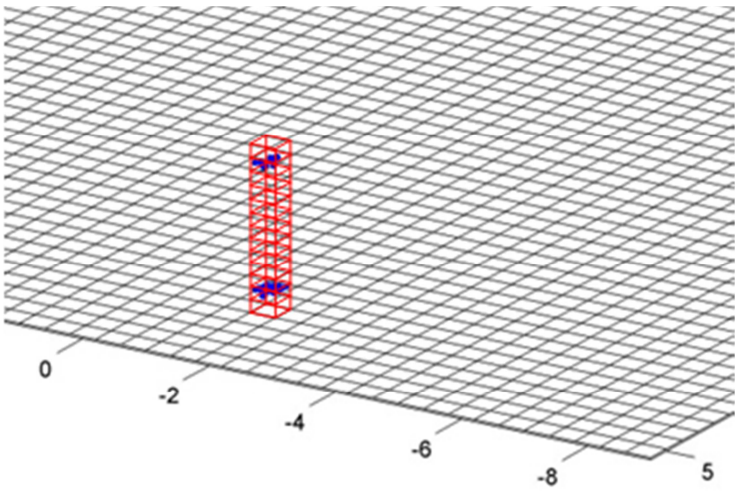


Figura 4.9: Esempio di struttura sporgente sopraelevata.

4.1.4 Classificazione delle celle

Come detto, l'algoritmo utilizzato si basa sul concetto di attraversabilità di una cella. L'obiettivo è quello di individuare le celle, ovvero le regioni della scena che il trattore può attraversare in maniera sicura senza correre il rischio di incontrare ostacoli durante il suo movimento.

L'algoritmo utilizzato è il *Breadth-First-Search (BFS)* che è un algoritmo di ricerca in ampiezza per grafi, il quale consente di esplorare tutti i nodi raggiungibili a partire da un nodo sorgente s.

Esso si avvale di una coda in cui memorizza man mano i nodi visitati, i quali vengono inoltre marcati. L'idea di base è quella di partire con un solo vertice qualsiasi nella coda ed eseguire ripetutamente l'operazione di "visitare" il vertice in cima alla coda, marcarlo ed aggiungere in fondo alla coda tutti i suoi vertici adiacenti non ancora marcati.

Per poter applicare tale algoritmo al caso in esame, dovendo esplorare celle e non nodi, ciascuna cella è stata identificata nel suo piano con il proprio baricentro, al quale

è stata associata come coordinata z quella del punto più alto della nuvola 3D che cade all'interno di ciascuna cella.

Pertanto, partendo da una cella sorgente, l'algoritmo ha come obiettivo quello di ispezionare tutte le altre celle della griglia classificando ciascuna di esse come "attraversabile", "ostacolo" o "non definita" ed assegnando ad esse una differente colorazione: verde per le celle attraversabili, rosse per quelle classificate come ostacolo e ciano per quelle non definite.

Per comprendere il criterio alla base dell'algoritmo, è necessario definire il concetto di attraversabilità tra due celle adiacenti.

Due celle adiacenti di baricentri (u, v) e (p, q) si dicono attraversabili se la coordinata z di tali punti è simile.

A tal fine si è definita una *funzione di compatibilità* che esprime la pendenza tra due celle adiacenti come segue:

$$C_l(u, v, p, q) = \frac{|z_{\max}(u, v) - z_{\max}(p, q)|}{r\|(u, v) - (p, q)\|}$$

La quantità C_l rappresenta un'approssimazione della pendenza tra le due celle, in cui il numeratore descrive la variazione di altezza e il denominatore la distanza tra le due celle. Definita tale funzione, due celle (u, v) e (p, q) sono definite attraversabili se:

$$C_l(u, v, p, q) < \varepsilon_l$$

essendo ε_l una costante espressa in gradi che definisce la massima pendenza che due celle adiacenti possono avere per essere definite attraversabili e che, nel caso in esame, è stata impostata pari a 20°.

In particolare, come cella sorgente si è considerata quella in prossimità dell'origine della telecamera e caratterizzata da un baricentro avente coordinata y nulla.

La scelta della cella sorgente non è stata casuale. Essa infatti, trovandosi subito davanti alla telecamera montata sul trattore, non può rappresentare un ostacolo ma una cella verde, e quindi, attraversabile in maniera sicura dal veicolo.

In Figura 4.10 è evidenziata la cella sorgente da cui ha inizio l'algoritmo.

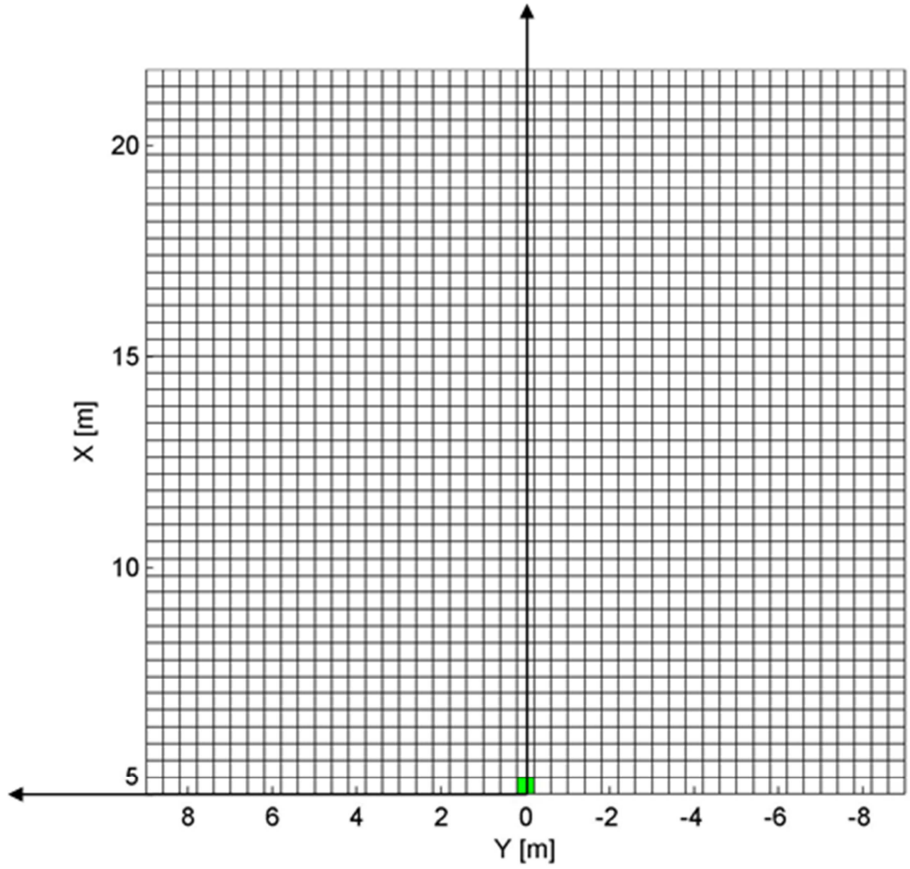


Figura 4.10: Cella sorgente della griglia 2D.

Prima di descrivere il funzionamento vero e proprio dell'algoritmo, si precisa che, d'ora in poi, parlare di baricentro di una cella o di cella sarà del tutto equivalente.

All'inizio dell'algoritmo tutte le celle della griglia sono bianche ad eccezione della sorgente (verde) da cui parte l'esecuzione dell'algoritmo.

Il passo successivo è quello di visitare le celle adiacenti a quella sorgente. Visitare una cella adiacente significa valutare la pendenza C_l rispetto a quella sorgente e, a seconda del valore assunto, colorare la cella visitata di verde o di rosso, classificandola "attraversabile" oppure "ostacolo".

Man mano che le celle vengono visitate, esse vengono inserite all'interno di un vettore etichettato come "visitato", contenente sia celle verdi che rosse. Tali celle, a loro volta, vengono inserite all'interno di due vettori etichettati come "Ground Visitors (GV)" e "Non-Ground Visitors (NGV)" contenenti rispettivamente le celle verdi e quelle rosse visitate e utili per eseguire la riproiezione finale dei punti di ciascuna cella sull'immagine.

Una volta visitate (e quindi colorate) tutte le celle adiacenti a quella sorgente, solo quelle verdi diverranno nuove sorgenti, e per ciascuna di esse verranno visitate solo le celle adiacenti non ancora visitate e cosi via, fino al completamento delle celle della griglia. Le celle rosse, invece, rappresentano un punto di arrivo e per esse non saranno visitate le celle adiacenti.

Pertanto, al termine dell'algoritmo potranno esistere celle che non sono state visitate o perché separate dal resto delle celle esaminate o perché occluse da celle rosse. Tali celle vengono classificate "non definite" e colorate di ciano. Queste ultime, non essendo state definite, non sono state considerate per la riproiezione finale dei punti sull'immagine.

Di seguito si riporta un diagramma di flusso utile per una comprensione più agevole ed immediata del funzionamento dell'algoritmo.

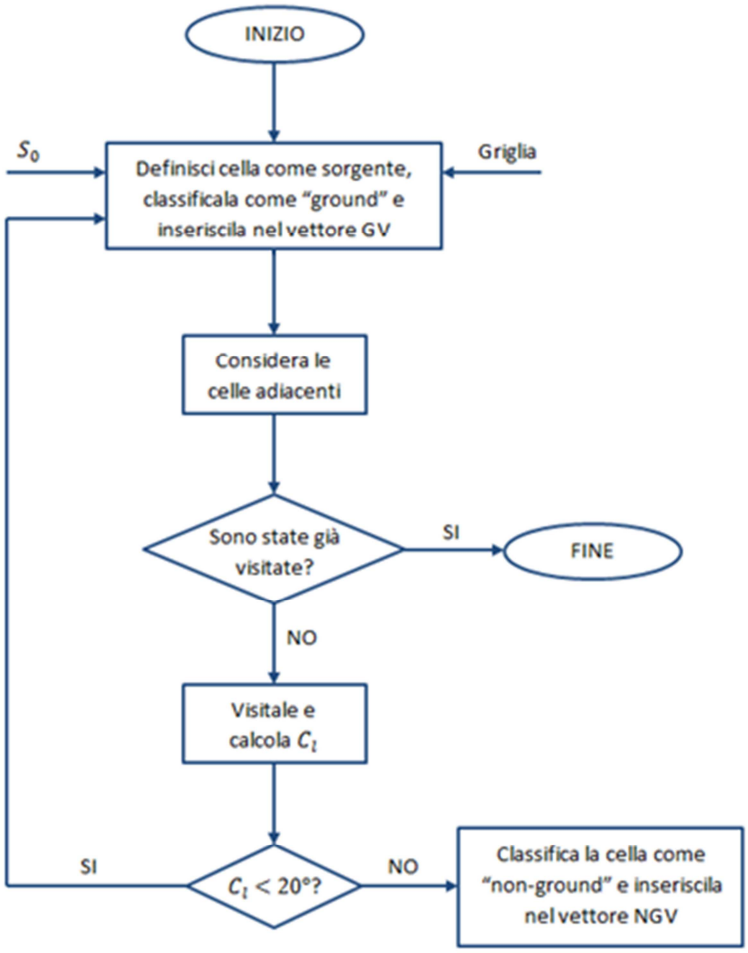


Figura 4.11: Diagramma di flusso dell'algoritmo; Input al sistema sono la cella sorgente di partenza S_0 e la griglia; Output del sistema è la classificazione delle singola cella in attraversabile (ground) o ostacolo (non-ground).

A titolo di esempio si riporta il risultato dell'algoritmo relativo all'immagine numero 240. Come si può facilmente osservare, le celle verdi rappresentano le zone che il trattore può attraversare in sicurezza, quelle rosse identificano il perimetro degli ostacoli presenti nella scena e quelle in ciano le celle non definite.
Il codice dell'intero algoritmo è riportato in Appendice A.

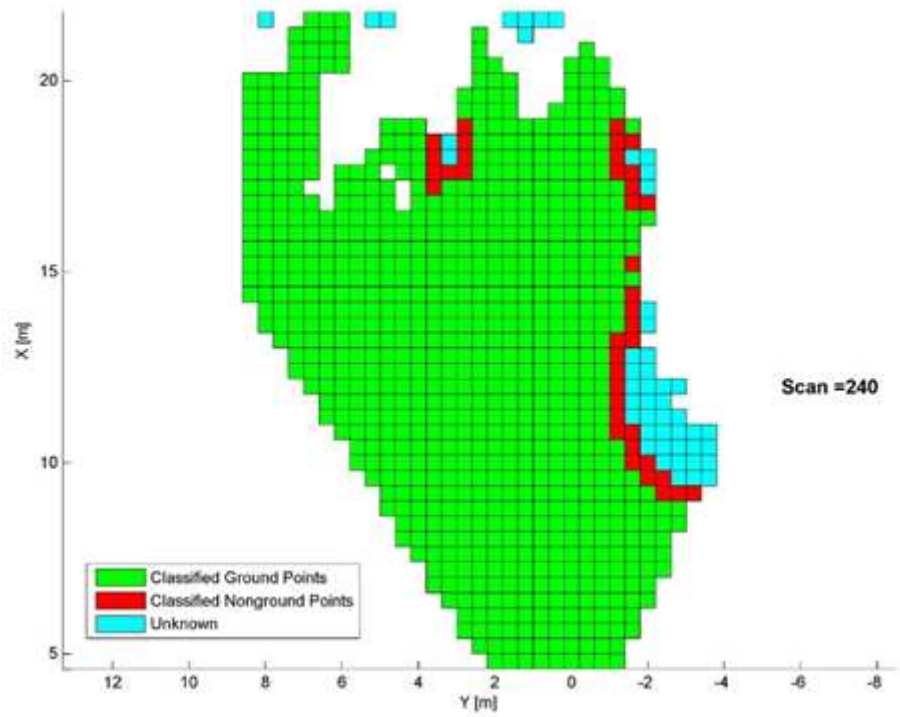


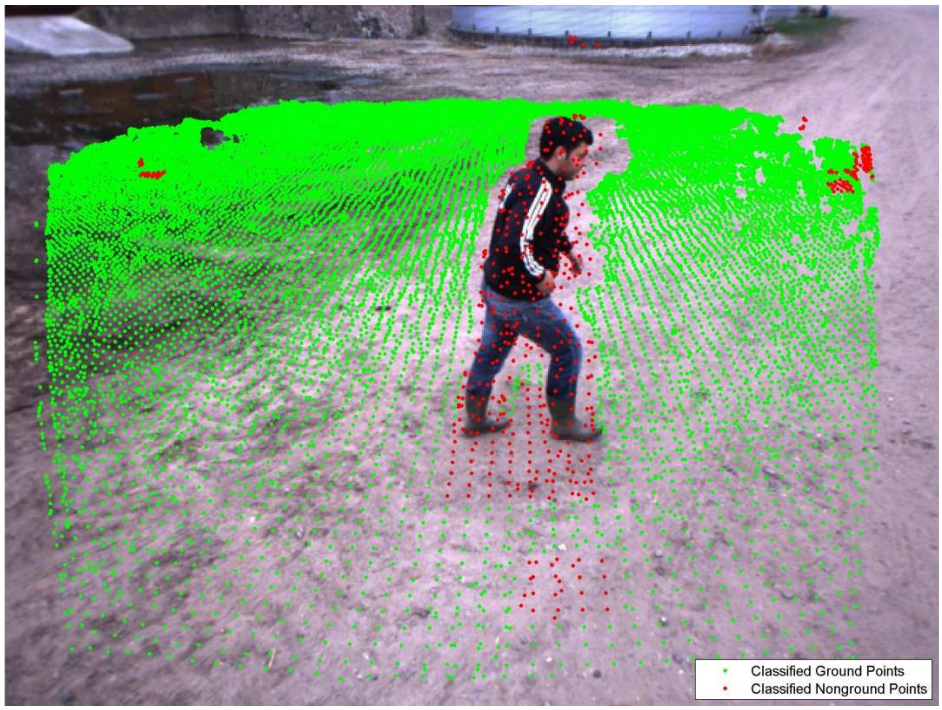
Figura 4.12: Risultato dell'algoritmo per l'immagine n° 240.

4.1.5 Riproiezione dei punti sull'immagine

Una volta classificate le celle della griglia, i punti 3D appartenenti a ciascuna cella verde e rossa sono stati classificati come "ground" (terreno) e "non-ground" (ostacolo), inserendoli rispettivamente in un vettore CGP (Classified Ground Points) ed un vettore CNGP (Classified Non-Ground Points). Fatto ciò, è stata caricata l'immagine originale della scena acquisita dalla telecamera e memorizzata al termine dell'acquisizione in un file di estensione *.pgm* e, su di essa, sono stati quindi proiettati i punti della nuvola 3D, ciascuna col proprio colore.
Di seguito si riporta l'applicazione dell'algoritmo ad alcune delle immagini acquisite dalla telecamera nelle quali sono evidenziate in verde le zone che il trattore può

attraversare in sicurezza, mentre in rosso evidenziati gli ostacoli rilevati (persone, auto, paletti, ecc.).

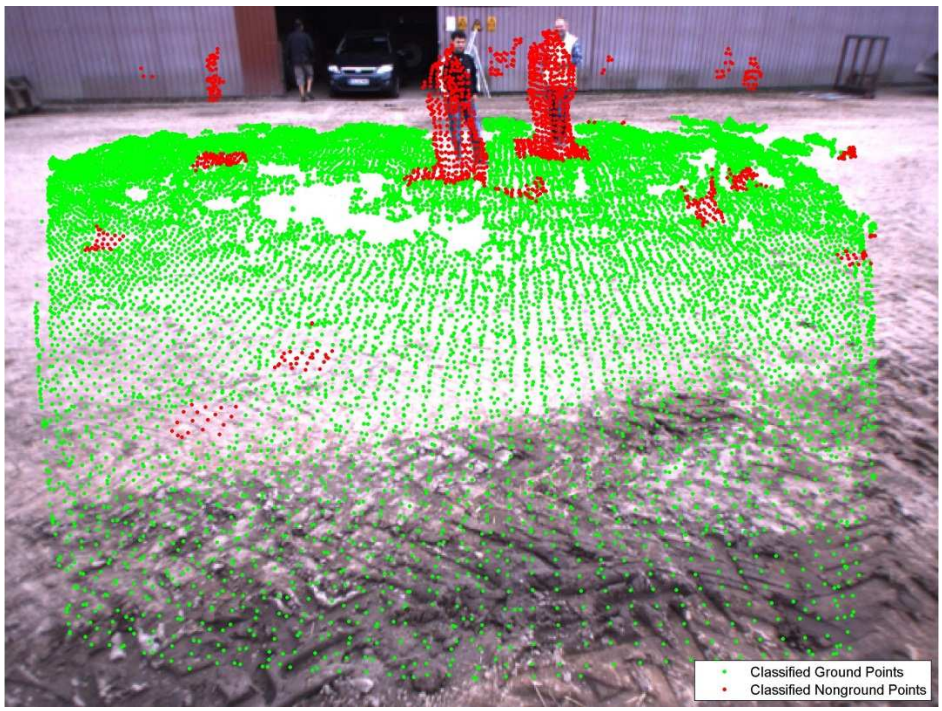
(a)



(b)

Figura 4.13: Immagine originale n° 89 (a) e risultato della riproiezione dei punti sull'immagine (b).

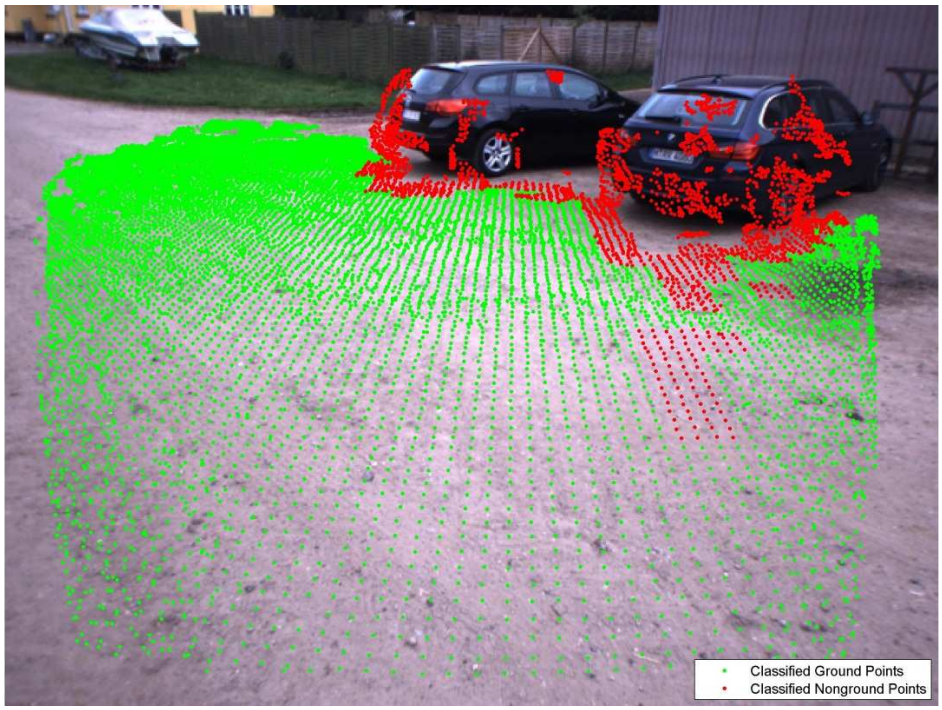
(a)



(b)

Figura 4.14: Immagine originale n° 191 (a) e risultato della riproiezione dei punti sull'immagine (b).

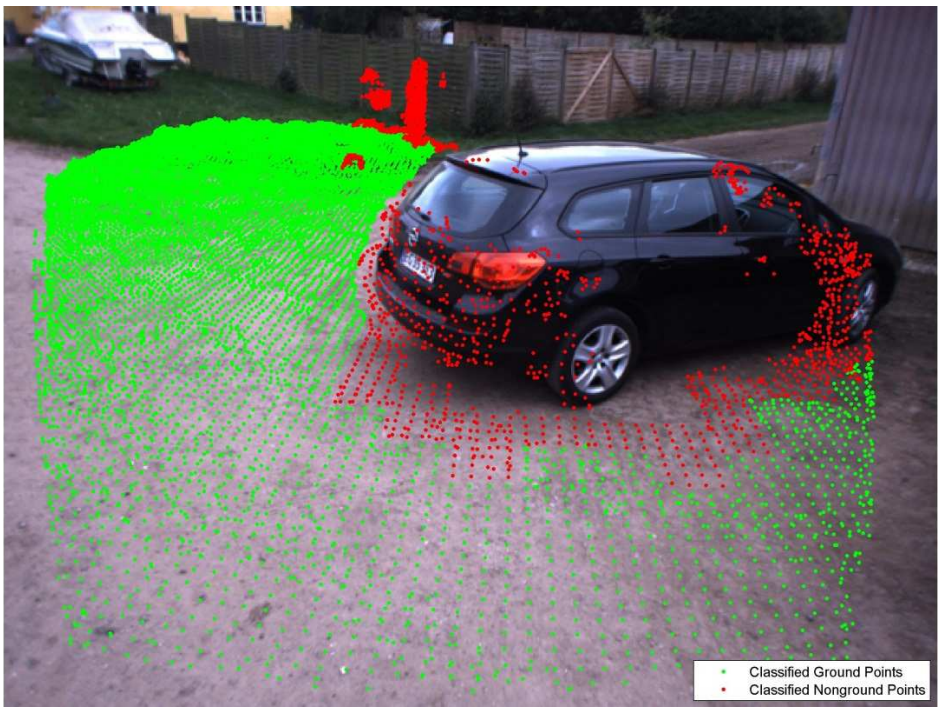
(a)



(b)

Figura 4.15: Immagine originale n° 196 (a) e risultato della riproiezione dei punti sull'immagine (b).

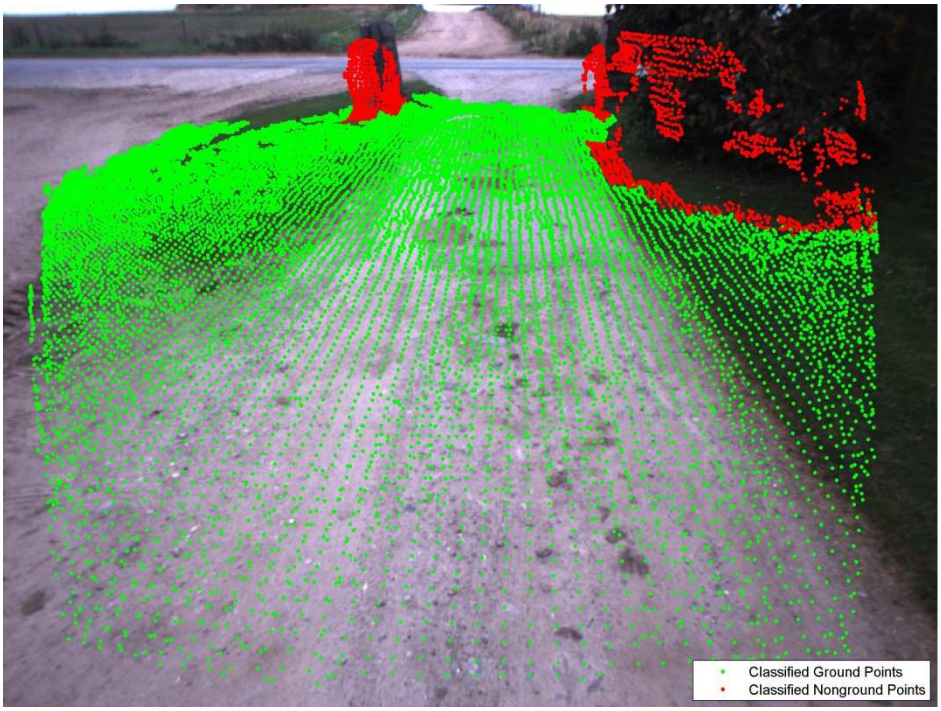
(a)



(b)

Figura 4.16: Immagine originale n° 202 (a) e risultato della riproiezione dei punti sull'immagine (b).

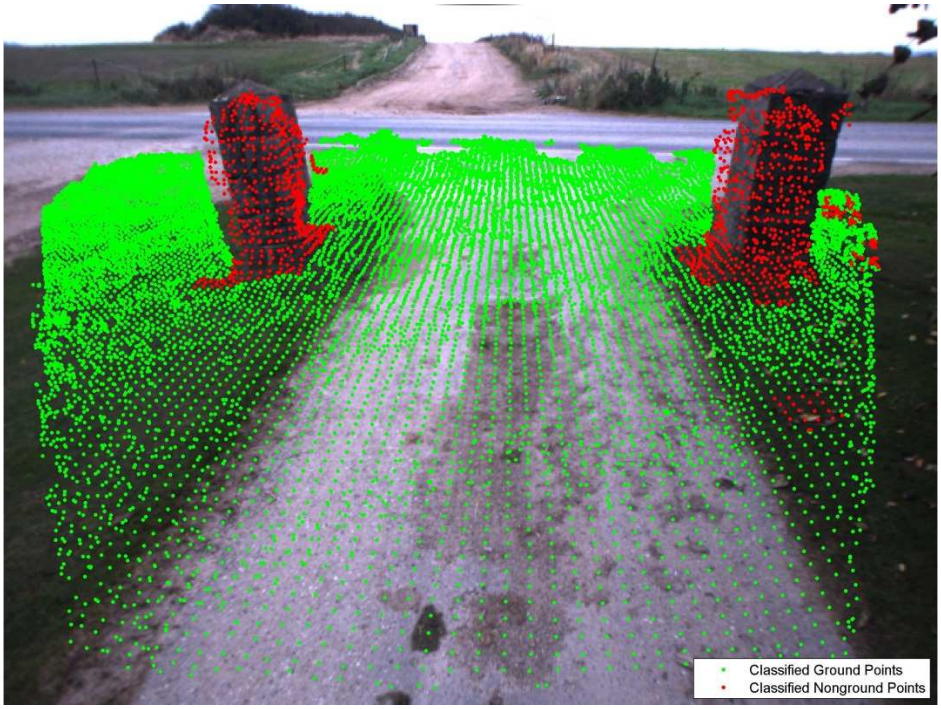
(a)



(b)

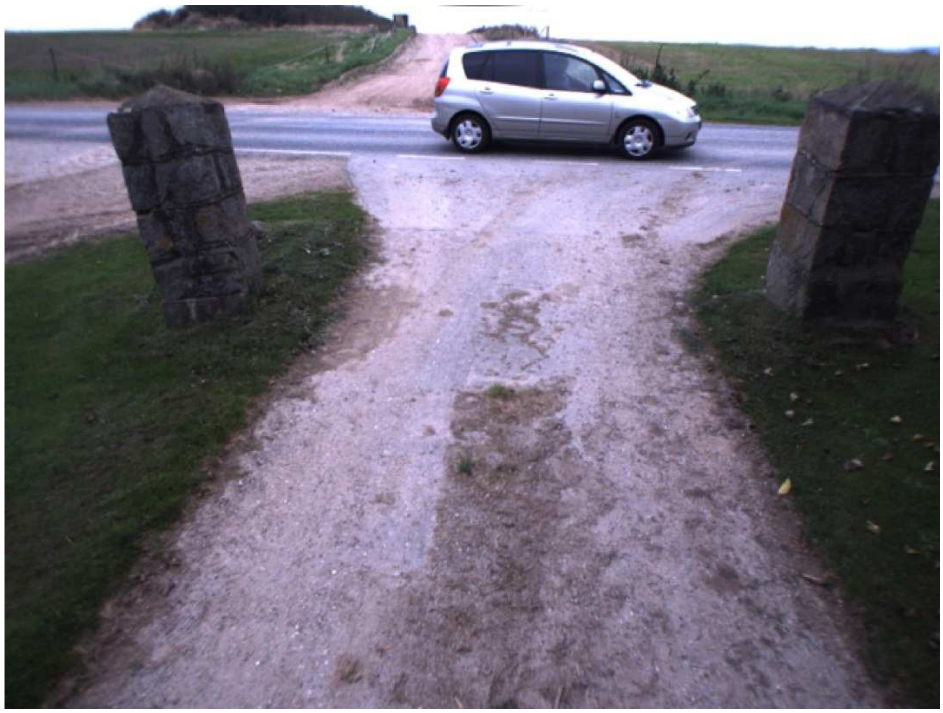
Figura 4.17: Immagine originale n° 239 (a) e risultato della riproiezione dei punti sull'immagine (b).

(a)

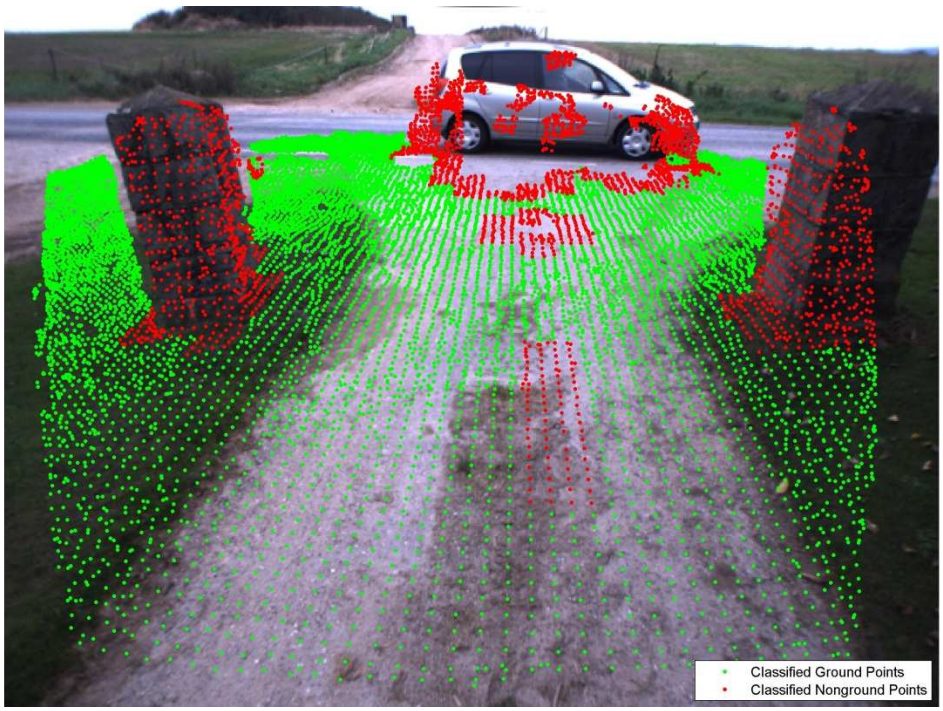


(b)

Figura 4.18: Immagine originale n° 250 (a) e risultato della riproiezione dei punti sull'immagine (b).



(a)



(b)

Figura 4.19: Immagine originale n° 257 (a) e risultato della riproiezione dei punti sull'immagine (b).

In definitiva l'applicazione dell'algoritmo ha fornito risultati più che soddisfacenti dal momento che gli ostacoli presenti nelle diverse immagini risultano rilevati in maniera abbastanza precisa.

Tuttavia le immagini presentano alcune zone in cui non si ha una precisa corrispondenza tra la colorazione delle celle (e quindi dei punti appartenenti a ciascuna di esse) e la realtà.

Al fine di valutare quantitativamente la performance dell'algoritmo, per ispezione visiva, si è quantificato l'errore commesso prendendo in considerazione la classe del "ground" (terreno) e quella del "non-ground" (ostacoli) e definendo per ciascuna di esse i *Falsi Positivi* (in inglese False Positive - FP) e i *Falsi Negativi* (in inglese False Negative - FN).

Riferendosi alla classe del "ground", i falsi positivi sono celle classificate come attraversabili ma che in realtà sono degli ostacoli, mentre i falsi negativi sono celle classificate dal sistema come ostacoli ma che, in realtà, sono attraversabili. Simile discorso può essere ripetuto per la classe degli ostacoli.

Facendo riferimento alla classe del "ground", indicato con GP_{tot} (Ground Points) il numero totale di punti classificati come "ground", è stato possibile definire l'errore percentuale legato alla presenza di falsi positivi e di falsi negativi come segue:

$$Err_{FP} = \frac{FP}{GP_{tot} - FP + FN} \cdot 100$$

$$Err_{FN} = \frac{FN}{GP_{tot} - FP + FN} \cdot 100$$

Per il set di immagini preso in esame, l'errore relativo alla presenza di falsi positivi risulta nullo, mentre quello relativo alla presenza di falsi negativi risulta essere dell'ordine del 4 % circa.

4.2 Analisi statistica della distribuzione dei punti

Nonostante le prestazioni del classificatore abbiano fornito risultati abbastanza soddisfacenti, si è evidenziato un numero di Falsi Negativi isolati per i quali si è ritenuto necessario condurre uno studio più approfondito, volto ad un'analisi statistica più accurata della distribuzione dei punti all'interno di ciascuna cella della griglia.
In Figura 4.20 sono evidenziate alcune delle celle isolate.

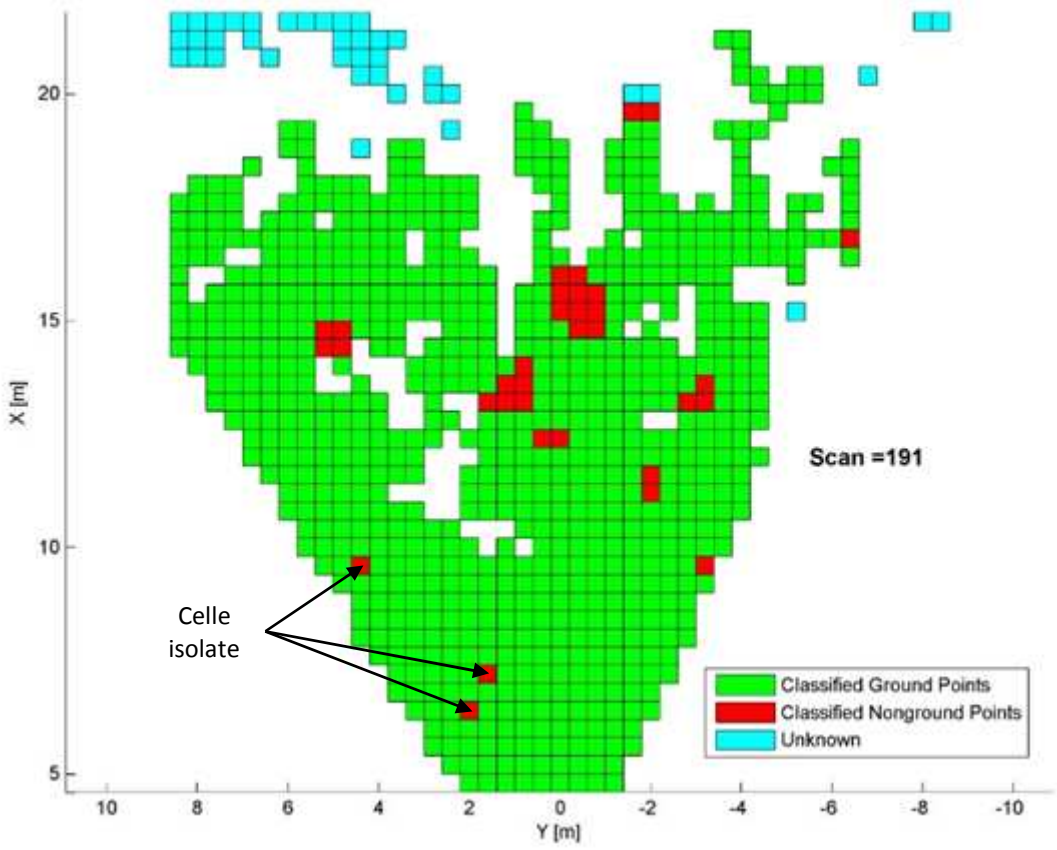


Figura 4.20: Risultato dell'algoritmo per l'immagine n° 191 e individuazione di celle rosse isolate.

In particolare, analizzando la distribuzione dei punti per le celle suddette, si è riscontrata la presenza di outlier.
Con il termine *outlier*, tradotto in italiano con i termini "dato anomalo" o "valore fuori limite", si intende una osservazione che appare differente dalle altre dello stesso gruppo. Il concetto spesso è limitato a un solo dato, ma può essere esteso a più valori contemporaneamente, rispetto al gruppo più ampio di osservazioni raccolte nelle

stesse condizioni. In termini più tecnici, un dato si definisce outlier quando non appare consistente con gli altri, cioè quando altera uno o più parametri contemporaneamente tra media, varianza e simmetria.

A titolo di esempio, si riporta l'istogramma relativo alla distribuzione dei punti per la cella 473 dell'immagine 191, in cui è possibile osservare chiaramente la presenza di outlier, ovvero di punti che sono molto distanti dalla zona di maggiore concentrazione degli stessi.

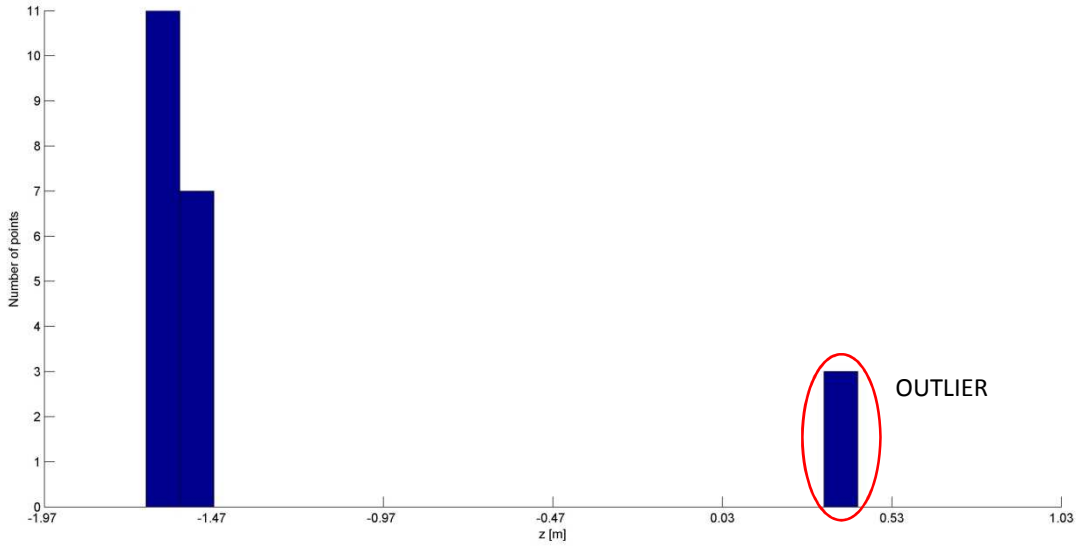


Figura 4.21: Istogramma relativo alla distribuzione dei punti appartenenti alla cella 473 dell'immagine 191 e individuazione degli outlier.

4.2.1 Misure di tendenza centrale

In genere, nella maggior parte dei casi, i dati mostrano una tendenza a raggrupparsi attorno a un valore centrale. Pertanto, risulta in genere possibile selezionare un valore tipico per poter descrivere un insieme di dati.

Tale valore descrittivo rappresenta una *misura di posizione* o di *tendenza centrale*.

Esistono tre differenti tipologie di misure di tendenza centrale: la *media aritmetica*, la *moda* e la *mediana*. La prima è una media di calcolo mentre le altre due sono medie di posizione.

La media aritmetica è la misura di posizione più comune ed è calcolata dividendo la somma dei valori di ciascun dato per il numero totale di dati. In pratica, la media rappresenta un "punto di equilibrio" tale che i dati più piccoli bilanciano quelli più grandi.

Tuttavia, poiché il calcolo della media si basa sull'intero insieme di dati $(x_1, x_2, x_3, \dots, x_n)$, tale valore di tendenza centrale risulta essere poco significativa per il caso in esame in quanto, essendo notevolmente influenzata dai valori estremi della distribuzione.

La moda è, invece, definita come il dato o la classe di dati che ha la massima frequenza. Tale valore riveste grande importanza in quanto rappresenta un'osservazione concreta sul fenomeno che non deriva da calcoli aritmetici e, a differenza della media, non è influenzata da outlier. Nell'istogramma della distribuzione, la classe modale corrisponde alla base del rettangolo di altezza massima ed è, quindi, facilmente individuabile.

Tuttavia, la moda presenta dei limiti. Infatti, un campione di dati può avere più di una moda. Inoltre, essa risulta essere molto sensibile alla larghezza e al numero degli intervalli di classe, al variare dei quali la moda può cambiare in maniera considerevole.

Il valore di tendenza centrale più idoneo a descrivere la distribuzione dei punti all'interno di ciascuna cella è rappresentato dalla mediana.

La *mediana* è una media di posizione che, in una successione di dati ordinati in ordine crescente, occupa la posizione centrale tale che il 50 % dei dati si trovi al di sotto e il 50 % al di sopra della stessa. Pertanto, assegnato un insieme di n valori ordinati in senso crescente, se n è dispari la mediana rappresenta il valore centrale, se n è pari la

semisomma dei due valori centrali. Inoltre, la mediana divide l'istogramma della distribuzione in due aree uguali e, nell'ambito delle frequenze cumulate essa corrisponde all'ascissa del punto la cui ordinata è 1/2, ovvero il 50 %.

Come la moda, essa non è influenzata da valori estremi ed identifica, di fatto, la zona in cui i punti si addensano maggiormente. Di seguito viene riportata sull'istogramma la retta che identifica la posizione della mediana per la distribuzione di punti in esame.

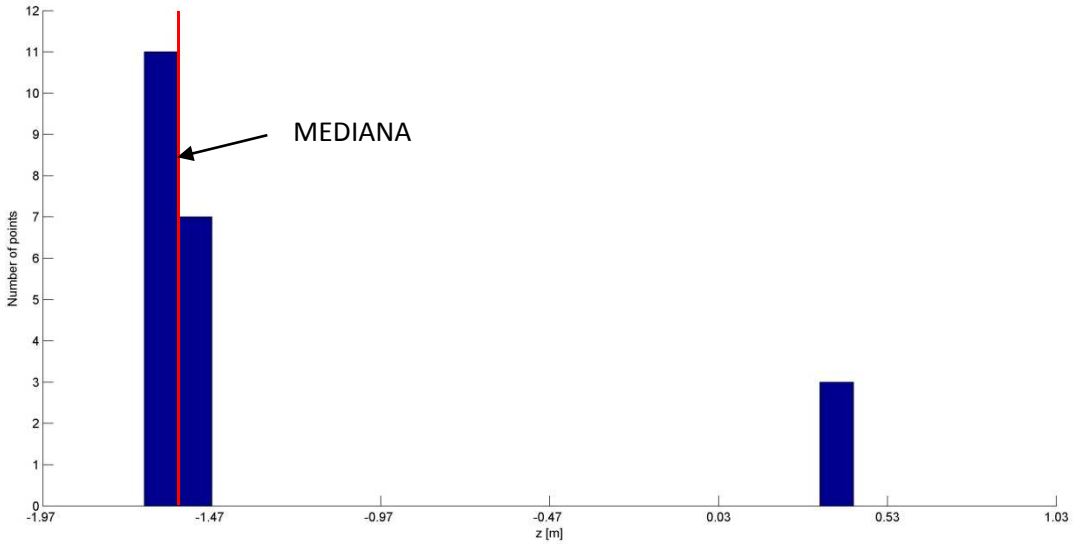


Figura 4.22: Individuazione della mediana per la distribuzione dei punti nella cella 473 dell'immagine 191.

4.2.2 Indici di dispersione

Una seconda caratteristica importante di un insieme di dati è la *variabilità* che misura la dispersione dei dati, ovvero la tendenza dei singoli dati di una distribuzione ad allontanarsi dalla tendenza centrale. Due insiemi di dati possono differire sia nella posizione che nella variabilità; oppure possono essere caratterizzati dalla stessa variabilità, ma da diversa misura di posizione; o ancora, possono essere dotati della stessa misura di posizione, ma differire notevolmente in termini di variabilità.

Esistono diversi indici di dispersione in grado di esprimere la variabilità di un insieme di dati. Il più semplice è il *range* che rappresenta la differenza tra il massimo e il minimo valore dei dati in una distribuzione. Tuttavia, esso non rappresenta un buon indice di variabilità in quanto risente in maniera sensibile della presenza di outlier; infatti se i

dati contengono degli outlier, il massimo o il minimo dei dati sarà proprio un outlier, e il valore del range rispecchierà l'ordine di grandezza degli outlier.

Per ovviare a ciò, si preferiscono pertanto degli indici di dispersione più robusti.

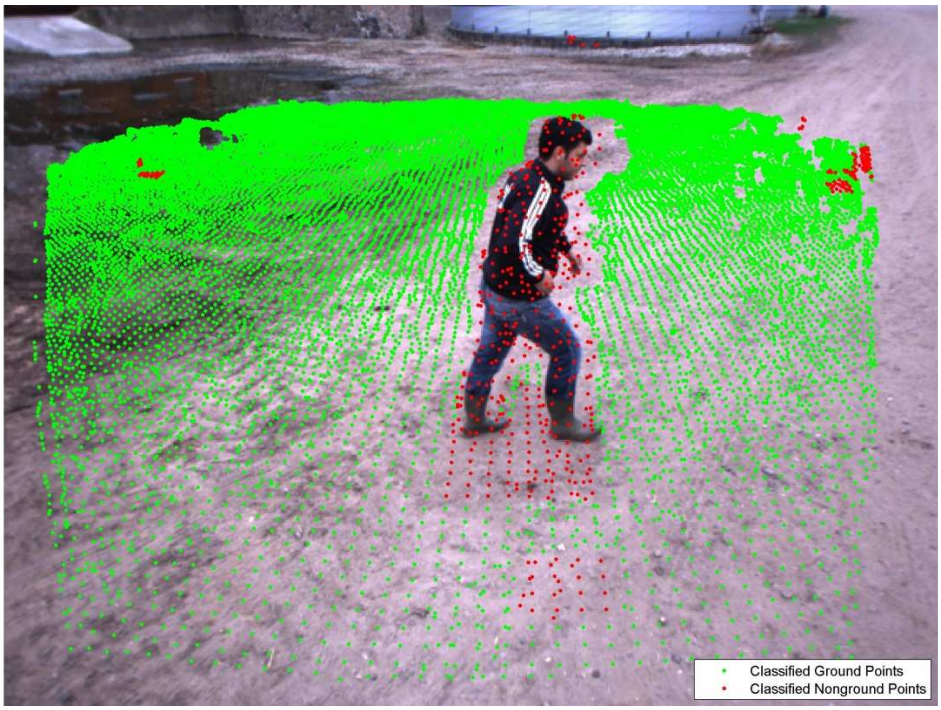
L'indice di dispersione che ha consentito di ottenere i risultati migliori è stato quello relativo alla *deviazione assoluta mediana (MAD – Median Absolute Deviation)* che rappresenta la mediana delle differenze assolute tra i dati e la loro mediana, in grado di fornire una dispersione robusta della variabile in quanto poco influenzata dalla presenza di outlier [23], [24]. La scelta di utilizzare tale misura di variabilità in luogo della più popolare deviazione standard si spiega con il fatto che quest'ultima risulta essere ottimale quando i dati sono distribuiti normalmente, ma poco attendibile in presenza di distribuzioni asimmetriche come nel caso in esame. La deviazione assoluta mediana viene calcolata come segue:

$$MAD(x_1, \dots, x_n) = median(\sum_{i=1}^{n} |x_i - median(x_1, \dots, x_n)|)$$

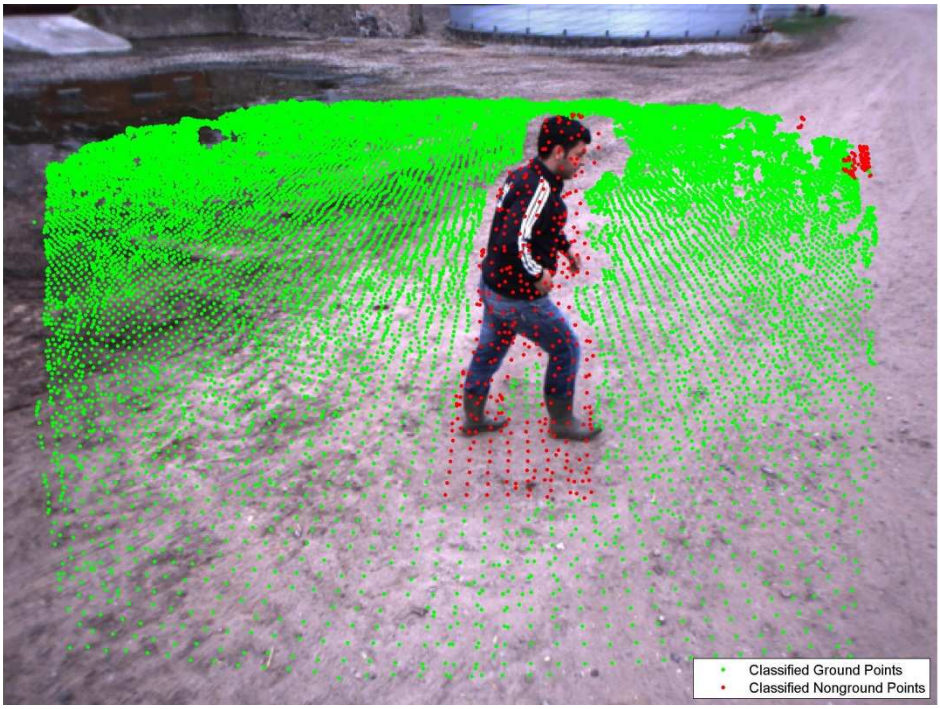
Ai fini dell'algoritmo in esame, si è considerato, rispetto alla mediana della distribuzione, un intervallo di semi-ampiezza pari a $2.9 \cdot MAD(x_1, \dots, x_n)$ che, nell'ipotesi in cui i punti fossero distribuiti normalmente all'interno di ciascuna cella, corrisponderebbe a considerare un intervallo di confidenza pari al 95 %.

4.2.3 Risultati

Di seguito vengono riportati i risultati dell'algoritmo mettendo a confronto le immagini prima e dopo l'analisi statistica condotta.

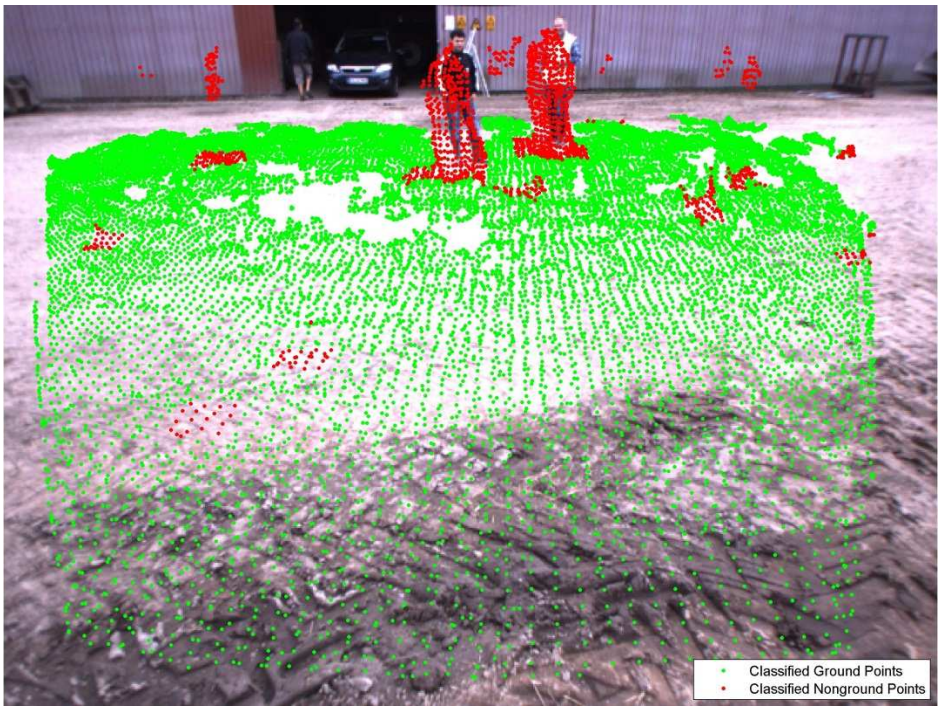


(a)

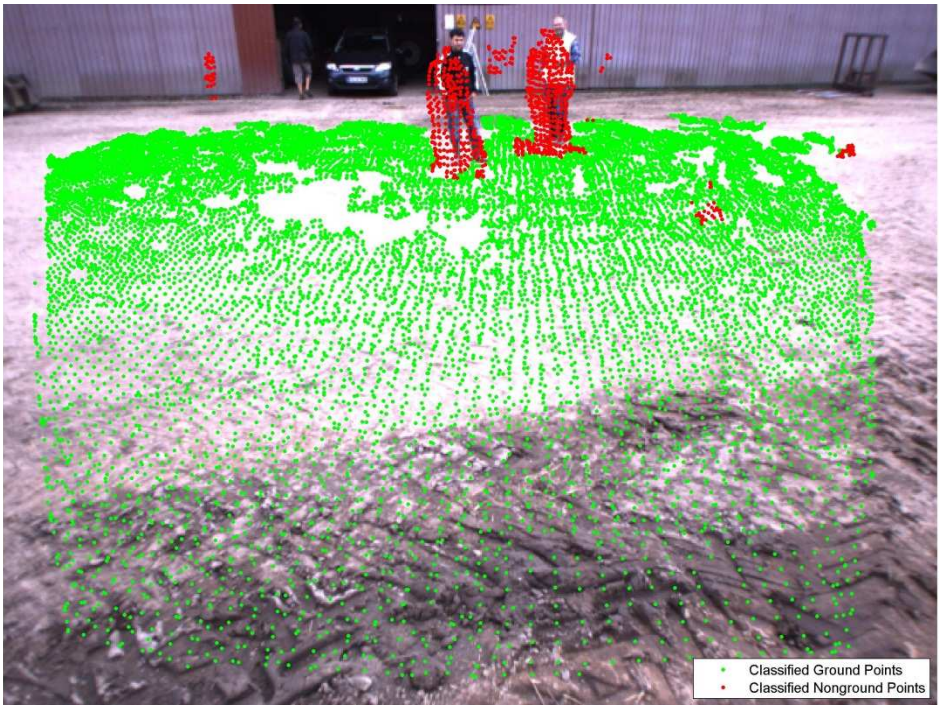


(b)

Figura 4.23: Confronto dei risultati per l'immagine n° 89 prima (a) e dopo (b) l'analisi statistica.

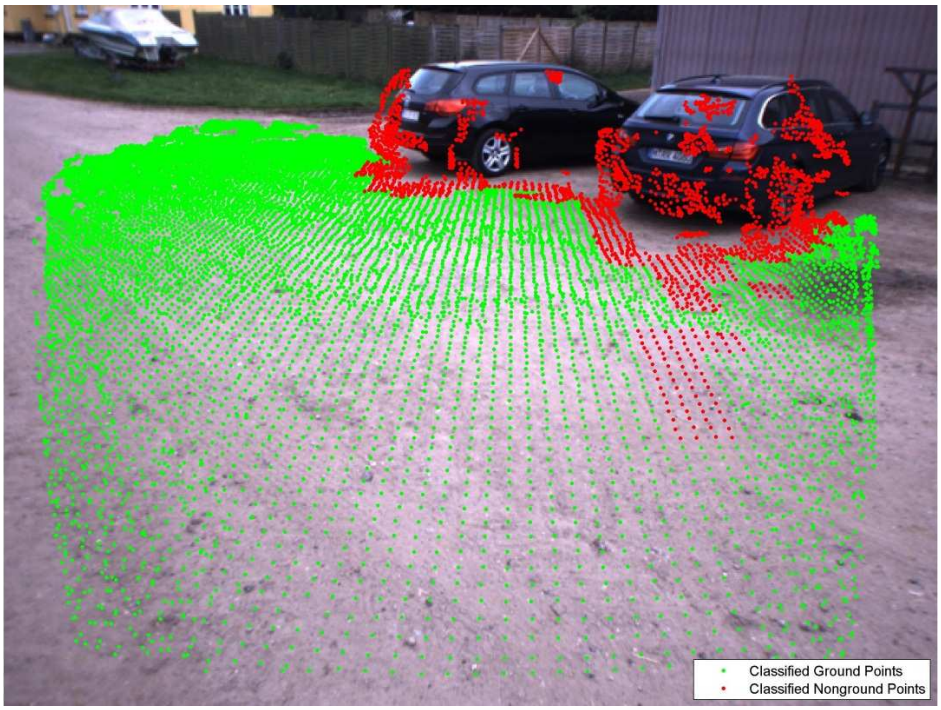


(a)

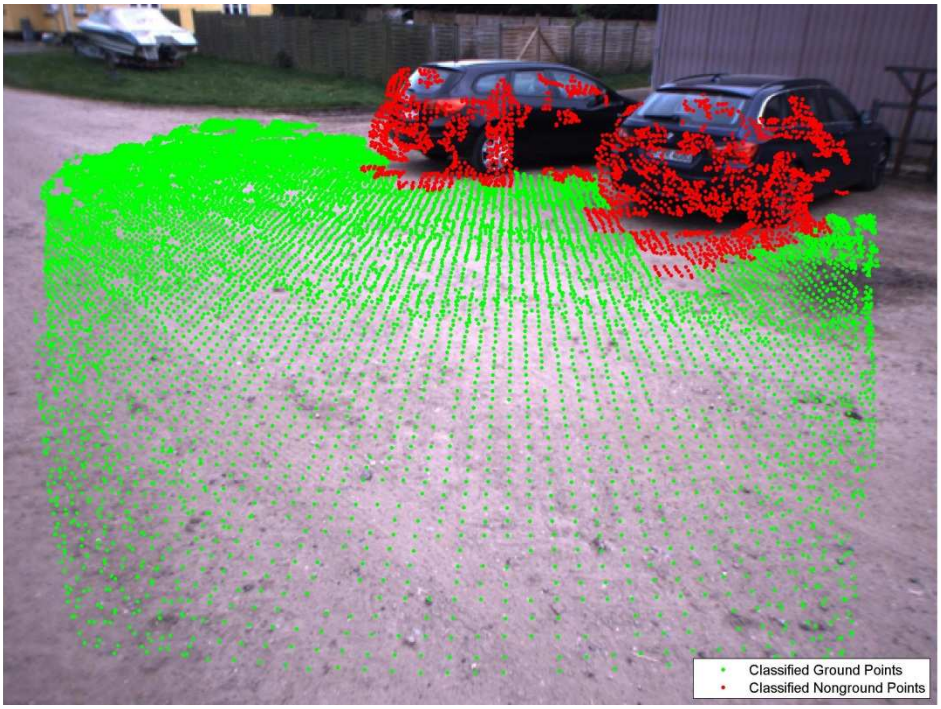


(b)

Figura 4.24: Confronto dei risultati per l'immagine n° 191 prima (a) e dopo (b) l'analisi statistica.

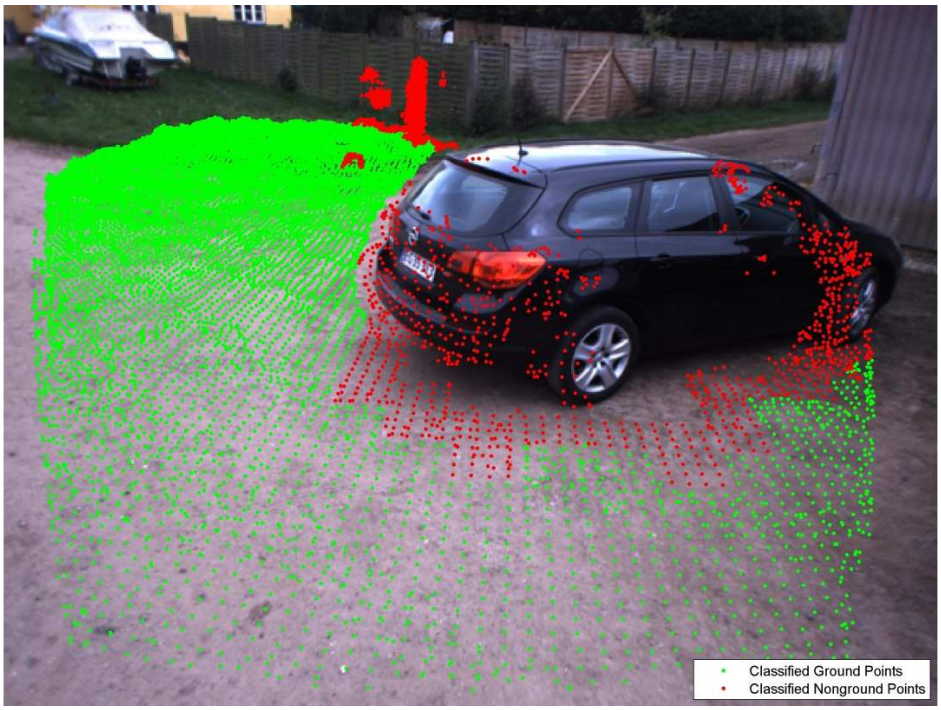


(a)

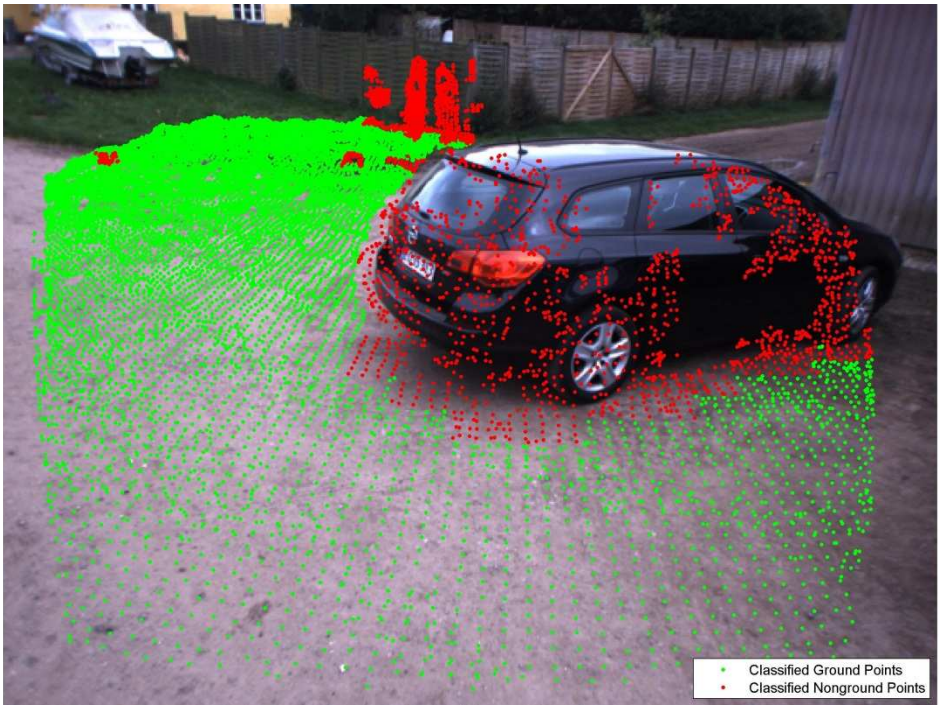


(b)

Figura 4.25: Confronto dei risultati per l'immagine n° 196 prima (a) e dopo (b) l'analisi statistica.

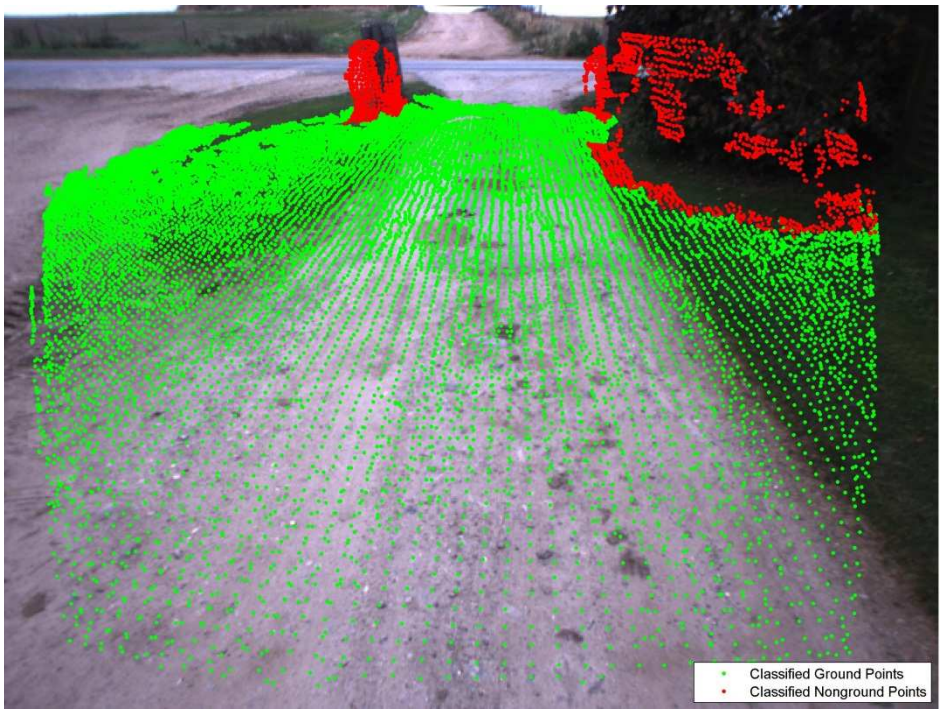


(a)

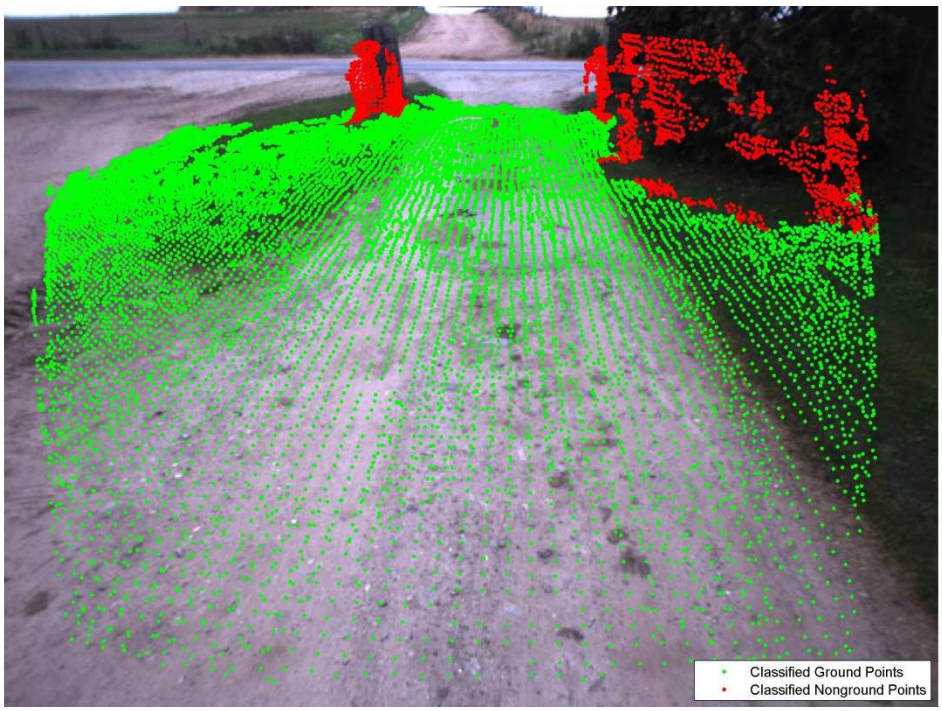


(b)

Figura 4.26: Confronto dei risultati per l'immagine n° 202 prima (a) e dopo (b) l'analisi statistica.

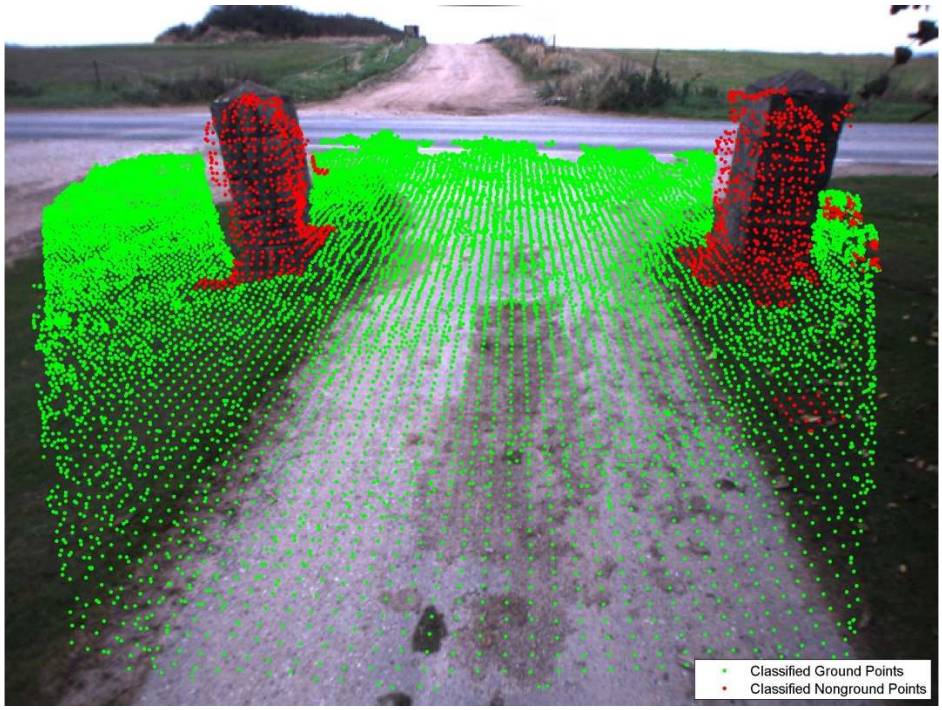


(a)

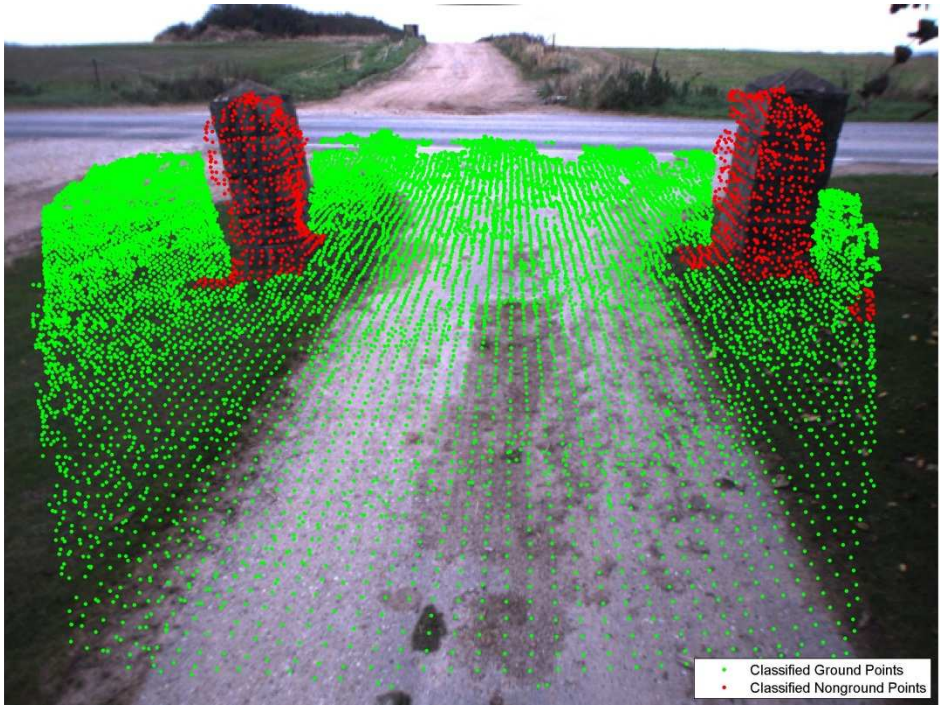


(b)

Figura 4.27: Confronto dei risultati per l'immagine n° 239 prima (a) e dopo (b) l'analisi statistica.

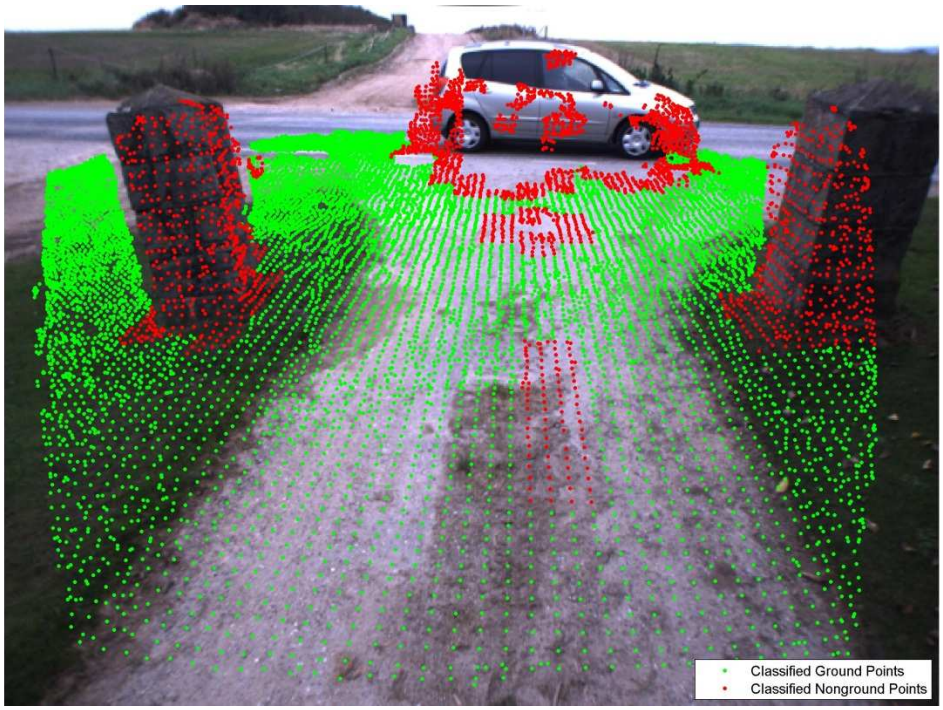


(a)

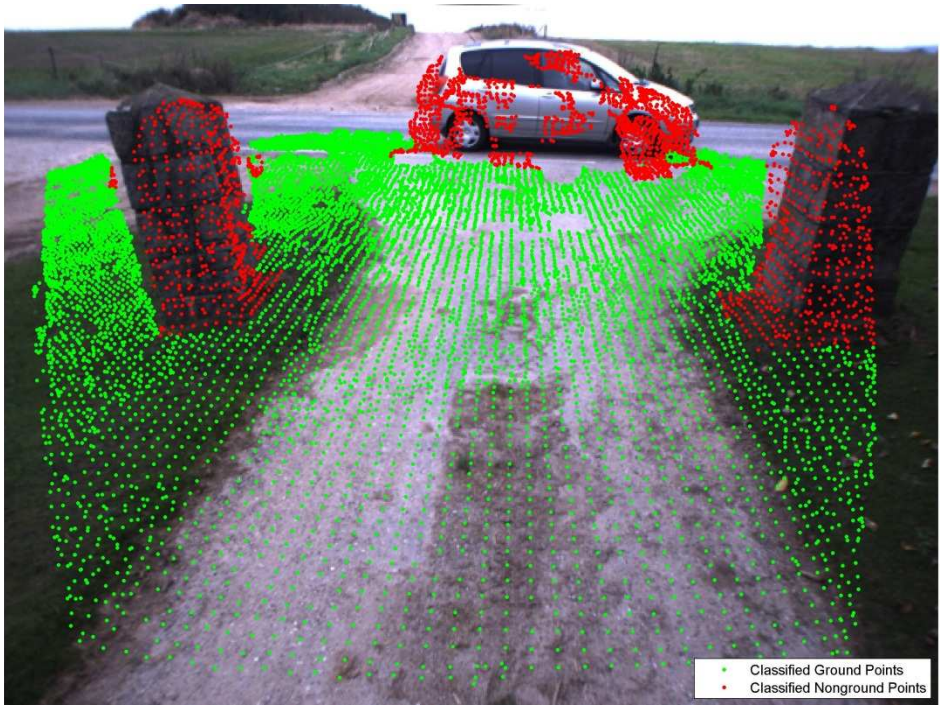


(b)

Figura 4.28: Confronto dei risultati per l'immagine n° 250 prima (a) e dopo (b) l'analisi statistica.



(a)



(b)

Figura 4.29: Confronto dei risultati per l'immagine n° 257 prima (a) e dopo (b) l'analisi statistica.

Come si evince dal confronto delle immagini prima (Sezione 4.1) e dopo l'analisti statistica condotta, l'identificazione delle zone attraversabili in sicurezza dal trattore e degli ostacoli presenti nella scena risulta quasi ottimale, tanto che l'errore relativo alla presenza di falsi negativi è stato ridotta a circa l'1.7 %, mentre l'errore relativo alla presenza di Falsi Positivi è rimasto nullo.

CONCLUSIONI

La tesi, inserendosi nell'ambito del progetto europeo denominato *QUAD-AV (Ambient Awareness for Autonomous Agricultural Vehicles)*, ha avuto come obiettivo quello di sviluppare un algoritmo finalizzato al riconoscimento e alla rilevazione automatica di ostacoli in campo agricolo per un trattore a guida autonoma, mediante l'utilizzo di un sistema multi-stereoscopico a baseline fissa.

Partendo dalla nuvola di punti 3D acquisita dal sistema trinoculare montato sul trattore, l'algoritmo si è proposto di definire in maniera esplicita le regioni della scena considerate "attraversabili" e, quindi, sicure per il movimento del veicolo, riconoscendo indirettamente la presenza di ostacoli nella scena.

L'idea di base dell'algoritmo è stata quella di suddividere la nuvola di punti 3D acquisiti dal sistema di telecamere mediante celle di una griglia 2D e, sfruttando l'informazione relativa all'altezza dei punti all'interno di ciascuna di esse, realizzare un algoritmo in grado di classificare ciascuna cella come "attraversabile", "ostacolo" o "non definita", colorandole rispettivamente di verde, rosso e ciano.

L'algoritmo, partendo da una cella sorgente, consiste nel visitare le celle ad essa adiacenti classificandole come "attraversabili" o "ostacoli" attraverso la definizione di una funzione di compatibilità che esprime la pendenza massima tra due celle adiacenti. Le celle classificate come attraversabili, e quindi colorate di verde, diventano a loro volta nuove sorgenti per ciascuna delle quali vengono visitate quelle adiacenti e così via, fino a quando tutte le celle della griglia non saranno state visitate. Quelle rosse, invece, costituiscono per l'algoritmo un punto d'arrivo e le celle ad esse adiacenti non vengono pertanto visitate. Tutte le celle non visitate vengono, infine, classificate "non definite" e colorate di ciano.

I punti appartenenti alle celle verdi e rosse sono stati colorati come le medesime celle di appartenenza e, infine, riproiettati sull'immagine della scena, ottenendo dei risultati più che soddisfacenti con un errore pari a circa il 4 %.

Tuttavia, l'algoritmo è stato successivamente migliorato conducendo un'analisi statistica più approfondita sulla distribuzione dei punti nelle celle al fine di correggere la presenza di celle rosse particolarmente isolate.

Tale analisi ha prodotto risultati quasi ottimali avendo ridotto notevolmente l'errore che è risultato essere inferiore al 2 %.

SVILUPPI FUTURI

Per quanto riguarda gli sviluppi futuri, in primo luogo, si può pensare di aumentare la discretizzazione della griglia riducendo la dimensione delle celle e la dispersione dei punti all'interno di ciascuna di esse, nel piano della griglia.

Un'altra possibilità potrebbe essere quella di considerare una griglia non più rettangolare ma che vada diradandosi man mano che ci si allontana dalle telecamere in modo da avere via via celle più grandi. Infatti, poiché la profondità dei punti acquisiti dal sistema trinoculare (e in generale da un sistema stereoscopico) è inversamente proporzionale alla loro disparità, la densità dei punti non risulta essere uniforme, in quanto, man mano che ci si allontana dalle telecamere, essi presentano una maggiore dispersione nel piano della griglia.

Pertanto, l'utilizzo di una griglia diradata consentirebbe alle celle più lontane dalle telecamere di accogliere, per via della loro maggiore dimensione, un numero di punti più significativo e di caratterizzare meglio la profondità della scena.

Infine, facendo riferimento sempre a una griglia di celle 2D, si potrebbe provare a cambiare il criterio di classificazione delle celle, definendo in luogo della pendenza tra celle adiacenti un parametro che faccia riferimento alla densità volumetrica dei punti all'interno di ciascuna cella. In tal caso, ci si potrebbe aspettare che i punti presentino una maggiore densità volumetrica proprio in corrispondenza degli ostacoli presenti nella scena.

APPENDICE A

Di seguito si riporta il codice elaborato mediante il software *Matlab®* relativamente all'algoritmo di rilevamento degli ostacoli.

```
% Script to divide a 3D point cloud into a regulary-sampled grid,
define
% the memebrship to a patch and define geoemetric and color properties
of
% the patch
% Author: G.Reina, 4/12


clear all, close all, clc
addpath '.\Functions'
%addpath(genpath('.\Functions'))
%addpath 'C:\Users\Giulio
Reina\Documents\2_Research\8_Sydney\Research\1_RadarGroundSegmentation
\Matlab\Functions'
fnampcd=dir('C:\Tesi\Tesi\28_9_11\*.pcd');
fnam=dir('C:\Tesi\Tesi\28_9_11\*.pgm');
dir1='C:\Tesi\Tesi\28_9_11';
txt=load('C:\Tesi\Tesi\28_9_11\test3.txt');
CameraTimer=txt(:,4); Time=CameraTimer-CameraTimer(1,1);


for i=300 %nscan0+n0:58
    n=i;
%% Load data
% XYZRGB = readPcd1('C:\Tesi\Tesi_new\28_9_11\Im2576.750126.pcd');
% XYZRGB = readPcd1('samples\Im2568.747312.pcd');
% XYZRGB = readPcd1('C:\Tesi\Tesi_new\28_9_11\Im2697.743938.pcd');
% XYZRGB = readPcd1('C:\Tesi\Tesi_new\28_9_11\Im2690.755126.pcd');
imNamepcd=fnampcd(i).name;
f = fullfile(dir1, imNamepcd); % Image name i
XYZRGB = readPcd1(f);


% fi='C:\Tesi\Tesi_new\28_9_11\Im2542.742066.pgm';
% load 2542.742066.mat%% Location of data
imName=fnam(i).name;
fi = fullfile(dir1, imName); % Image name i

%XYZRGB = readPcd1('C:\Tesi\Tesi_new\28_9_11\Im2542.742066.pcd');


%%  Define short range grid (SRG)

xlim=22; % Maximum value along x-axis
ylim=9; % Maximum value along y-axis
indexr=find(XYZRGB(:,1)<=xlim &  abs(XYZRGB(:,2))<=ylim);
XYZRGB=XYZRGB(indexr,:); %Region of interest for the SRG
%clearvars indexr
tic
```

```
r=0.4; % size of the square grid Patch
tx = 4.6:r:xlim;
ty = -ylim:r:ylim;
[xi,yi] = meshgrid(tx,ty);
zi=ones(size(xi))*(-1.97);
% zi = griddata(XYZRGB(:,1),XYZRGB(:,2),XYZRGB(:,3),xi,yi);

GP=[];
for i=1:size(xi,1)
    for j=1:size(yi,2)
        P=[xi(i,j) yi(i,j) zi(i,j)];
        GP=[GP;P];
    end
end
GP=GP(~isnan(GP(:,3)),:); % Remove Nan


%% Mapping 3D points to cells

Grid={}; % Center points of each patch
Patch={};
for i=1:size(GP,1)-1
    if GP(i,2)<ylim-r
       index=find(XYZRGB(:,1)<GP(i+1,1) &  XYZRGB(:,1)>=GP(i,1) &
XYZRGB(:,2)>=GP(i,2) & XYZRGB(:,2)<GP(i,2)+r &
isempty(GP(GP(i+1:end,1)==GP(i,1)))~=1);
       if (isempty(index)~=1 & length(index)>4)
         Patchi=XYZRGB(index,:); % Points falling into Patch i
         Patch{i}=Patchi;
         Grid{i}=[0.5*(GP(i+1,1)+GP(i,1)) GP(i,2)+r*0.5];
       end
    else
        break
    end
end
Patch=Patch(~cellfun('isempty', Patch));
Grid=Grid(~cellfun('isempty', Grid));


%% Computing elevation histograms for cells and eliminating
overhanging structures

H_vehicle=3.203;
z_camera=1.97;
for i=1:size(Patch,2)
    zi_Patch=Patch{i}(:,3);
    x=-2.12:0.1:1.28;
    z{i}=zi_Patch;
    [counts,bin]=hist(z{i},x);
    indexc{i}=find(counts==1);
    indexb{i}=bin(:,indexc{i});
    le{i}=indexb{i}-0.05;
    re{i}=indexb{i}+0.05;
    indexz=[];
    for j=1:size(le{i},2)
        if ~isempty(indexc{i})
        indexzi=find(z{i}(:,1)<re{i}(:,j) & z{i}(:,1)>le{i}(:,j));
        indexz=[indexz;indexzi];
```

```
        end
    end
    z{i}(indexz,:)=[];
    Patch{i}(indexz,:)=[];
    cs=0.2; % safety constant;
    m=find(z{i}(:,1)>=(H_vehicle-z_camera)+cs);
    z{i}(m,:)=[];
    Patch{i}(m,:)=[];
    zmax{i}=max(Patch{i}(:,3));
end


%% Robust statistics for outlier rejection

for i=1:size(Patch,2);
    Me{i}=median(z{i}(:,1)); % median
    MAD{i}=mad(z{i}(:,1),1); % median absolute devition
    w=find(abs(z{i}(:,1)-Me{i})<2.9*MAD{i});
    z{i}=z{i}(w,:);
    Patch{i}=Patch{i}(w,:);
    zmax{i}=max(Patch{i}(:,3));
end

figure(1)
% mesh(yi,xi,zi,'EdgeColor',[0 0 0]), hold on, alpha(0.04)
% hold on
% plot3(GP(:,1), GP(:,2), GP(:,3),'.g')
axis equal
xlabel('Y [m]','FontSize',10); ylabel('X [m]','FontSize',10);
zlabel('Z [m]','FontSize',10);
% legend('Classified Ground Points','Classified Nonground
Points','Unknown',4,'Location','EastOutside')
grid off; view(2);
text(-5.5,12,-1.4,['Scan
=',num2str(n,'%.f'),''],'Color','k','FontWeight','bold','FontSize',12)
% for i=1:size(Grid,2)
%     plot3(Grid{i}(:,2),Grid{i}(:,1),ones(1:length(GP),1)*1.4,'.b')
% end
set(1,'units','normalized','position',[0.39 0.292 0.4 0.6]);
set(gca,'XDir','reverse')
hold on
% for jj=674
%     scatter3(Patch{jj}(:,2), Patch{jj}(:,1),
Patch{jj}(:,3),10,'filled','b')
% end

XGRID=[];
YGRID=[];
for i=1:size(Grid,2)
    x_grid=Grid{i}(:,1);
    y_grid=Grid{i}(:,2);
    XGRID=[XGRID,x_grid];
    YGRID=[YGRID,y_grid];
end

GRID=[XGRID',YGRID'];

for i=1:size(Grid,2)
```

```
    aa=find(GRID(:,1)<GRID(i,1)+1.5*r & GRID(:,1)>GRID(i,1)-1.5*r &
GRID(:,2)<GRID(i,2)+1.5*r & GRID(:,2)>GRID(i,2)-1.5*r);
    hh=find(aa(:,1)==i);
    aa(hh,:)=[];
    A{i}=aa;

end


%% Breadth-First-Search (BFS) algorithm and computing the
traversable/obstacle map

s=find(GRID(:,1)<4.6+r & GRID(:,1)>4.6 & GRID(:,2)<r/2 & GRID(:,2)>-
r/2); % Starting Patch
GV=[s]; % Ground Patch Visited
NGV=[]; % NonGround Patch Visited
Visitors=[s]; % Vector patches visited
Al=A{s}; % Patches around s
CGP=[Patch{s}]; % Classified Ground Points
CNGP=[]; % Classified NonGround Points
%GGG=[];
slope=20;

for j=1:size(Al,1)
    k=Al(j,:);
   if (atand((zmax{k}-zmax{s})/sqrt((GRID(k,1)-
GRID(s,1))^2+(GRID(k,2)-GRID(s,2))^2))<slope &
isempty(find(Visitors(:)==k)))
      GV=[GV;k];
      CGP=[CGP;Patch{k}];
      %GGG=[GGG; GRP{k}];
    elseif (atand((zmax{k}-zmax{s})/sqrt((GRID(k,1)-
GRID(s,1))^2+(GRID(k,2)-GRID(s,2))^2))>slope &
isempty(find(Visitors(:)==k)))
           NGV=[NGV;k];
           CNGP=[CNGP;Patch{k}];
   end
   Visitors=[Visitors;k];
end

if ~isempty(GV)
  for n=1:50
    for i=1:size(GV,1)
        Al=A{GV(i,:)};
     for j=1:size(Al,1)
         k=Al(j,:);
       if (atand((zmax{k}-zmax{GV(i,:)})/sqrt((GRID(k,1)-
GRID(GV(i,:),1))^2+(GRID(k,2)-GRID(GV(i,:),2))^2))<slope &
isempty(find(Visitors(:)==k)))
          GV=[GV;k];
          Visitors=[Visitors;k];
          CGP=[CGP;Patch{k}];
          %GGG=[GGG; GRP{k}];
        elseif (atand((zmax{k}-zmax{GV(i,:)})/sqrt((GRID(k,1)-
GRID(GV(i,:),1))^2+(GRID(k,2)-GRID(GV(i,:),2))^2))>slope &
isempty(find(Visitors(:)==k)))
               NGV=[NGV;k];
               Visitors=[Visitors;k];
               CNGP=[CNGP;Patch{k}];
```

```
       end
       n=n+1;
     end
    end
  end
 else break
end

for i=1:size(GV,1)
    XX=[Grid{GV(i,:)}(:,1)+r/2 Grid{GV(i,:)}(:,1)-r/2
Grid{GV(i,:)}(:,1)-r/2 Grid{GV(i,:)}(:,1)+r/2];
    YY=[Grid{GV(i,:)}(:,2)+r/2 Grid{GV(i,:)}(:,2)+r/2
Grid{GV(i,:)}(:,2)-r/2 Grid{GV(i,:)}(:,2)-r/2];
    ZZ=[-1.97 -1.97 -1.97 -1.97];
    h=patch(YY',XX',ZZ','g');
end

for i=1:size(NGV,1)
    XX=[Grid{NGV(i,:)}(:,1)+r/2 Grid{NGV(i,:)}(:,1)-r/2
Grid{NGV(i,:)}(:,1)-r/2 Grid{NGV(i,:)}(:,1)+r/2];
    YY=[Grid{NGV(i,:)}(:,2)+r/2 Grid{NGV(i,:)}(:,2)+r/2
Grid{NGV(i,:)}(:,2)-r/2 Grid{NGV(i,:)}(:,2)-r/2];
    ZZ=[-1.97 -1.97 -1.97 -1.97];
    h2=patch(YY',XX',ZZ','r');
end

Nan=[]; % Unknown Patches

for i=1:size(GRID,1)
    if isempty(find(Visitors(:)==i));
        Nan=[Nan;i];

    end
end

for i=1:size(Nan,1)
    XX=[Grid{Nan(i,:)}(:,1)+r/2 Grid{Nan(i,:)}(:,1)-r/2
Grid{Nan(i,:)}(:,1)-r/2 Grid{Nan(i,:)}(:,1)+r/2];
    YY=[Grid{Nan(i,:)}(:,2)+r/2 Grid{Nan(i,:)}(:,2)+r/2
Grid{Nan(i,:)}(:,2)-r/2 Grid{Nan(i,:)}(:,2)-r/2];
    ZZ=[-1.97 -1.97 -1.97 -1.97];
    h1=patch(YY',XX',ZZ','c');
end
legend('Classified Ground Points','Classified Nonground
Points','Unknown',4)

toc

figure(2)
    set(gcf,'doublebuffer','on'); cla
%   load 2542.742066.mat
    text(-5.5,12,-1.4,['Scan
=',num2str(i,'%.f'),''],'Color','k','FontWeight','bold','FontSize',12)

    I=imread(fi);
    imshow(fi,'InitialMagnification', 70,'Border','tight');
    set(2,'units','normalized','position',[0.0078 0.535 0.369 0.357]);
    %[imName,timeInd,im] = loadImage(i,Time, fnam, dir1);
imshow(im,'InitialMagnification', 50,'Border','tight');
```

```
    tes=1;
    if tes==1
        hold on
       if exist('GV','var')~=0
%         ZZ=GV(:,3); % remove node height higehr than -1.2 m
%         zz=(ZZ>-1.2).*(-1.2)+(ZZ<-1.2).*ZZ;
%         GV(:,3)=zz;
        [Cpts, px,py, validindices]=image_projection(CGP(:,1:3)*1000,
I); %

plot(px,py,'og','LineWidth',1,'MarkerFaceColor','g','MarkerSize',2);
%         ZZ=CNGP(:,3);
%         zz=(ZZ>-1.4).*(-1.7)+(ZZ<-1.4).*ZZ;
%         CNGP(:,3)=zz;
        [NCpts,
Npx,Npy,validindices]=image_projection(CNGP(:,1:3)*1000, I); %

plot(Npx,Npy,'or','LineWidth',1,'MarkerFaceColor','r','MarkerSize',2);
       end
    end
%     legend('Classified Ground Points','Classified Nonground
Points',4)
    waitforbuttonpress
end
```

BIBLIOGRAFIA

[1] Scaramuzza, D., Tesi di Laurea: *Progetto e realizzazione di un sistema di visione stereoscopica per la robotica, con applicazione all'inseguimento di corpi in moto e all'autolocalizzazione*, Facoltà di Ingegneria Elettronica, Università degli Studi di Perugia, A.A. 2003-2004.

[2] Paolillo, A., Appunti: *Appunti di misure basate sulla visione*, Facoltà di Ingegneria Elettronica, Università degli Studi di Salerno, 7 Aprile 2011.

[3] Lasagni, A., Tesi di Laurea: *Elaborazione di immagini stereoscopiche all'infrarosso termico per la localizzazione di pedoni*, Facoltà di Ingegneria Informatica, Università degli Studi di Parma, A.A. 2003-2004.

[4] Kuthirummal, S., Das, A., Samarasekera, S., (2011): *A graph traversal based algorithm for obstacle detection using lidar*, IEEE/RSJ International Conference on Intelligent Robots and Systems, September 25-30, 2011. San Francisco, CA, USA.

[5] Zani, P., Tesi di Laurea: *Algoritmi ottimizzati per la localizzazione di ostacoli in ambienti non strutturati mediante visione stereo*, Facoltà di Ingegneria Informatica, Università degli Studi di Parma, A.A. 2004-2005.

[6] Caraffi, C., Tesi di Laurea: *Stabilizzazione di immagini ed individuazione di ostacoli mediante visione artificiale stereo in ambienti non strutturati*, Facoltà di Ingegneria Informatica, Università degli Studi di Parma, A.A. 2003-2004.

[7] Bertozzi, M., *Student Member*, *IEEE*, Broggi, A., *Associate Member*, *IEEE*: *GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection*, IEEE Transactions on image processing, VOL. 7, NO. 1, January 1998.

[8] Bertozzi, M., Broggi, A., Conte, G., Fascioli, A.: *Obstacle and lane detection on the ARGO autonomous vehicle*, Proceedings of IEEE Intelligent Transportation Systems conference, 1997. Boston, MA, USA.

[9] Bertozzi, M., Broggi, A., Fascioli, A.: *Real time obstacle detection using stereo vision*, Proceedings EUSIPCO-96 - VIII European Signal Processing Conference,1996, Trieste, Italy.

[10] Burks, T., Villegas, F., Hannan, M., Flood, S., Sivarabam, B., Subramanian V., Sikes, J., (2005): *Engineering and horticultural aspects of robotic fruit harvesting: opportunities and constraints*, HortTecnhology, Vol. 15 No. 1, pp. 79-87.

[11] Ollis, M., Stentz, A. (1996): *First results in vision–based crop line tracking,* Proceedings IEEE International Conference on Robotics and Automation, Minneapolis, MN , USA, Vol. 1, pp. 951–956.

[12] Hague, T. and Tillett, N. D., (1995): *Navigation and control of an autonomous horticulture robot*, Mechatronics, Vol. 6 No. 2, pp. 165–180.

[13] Milella, A., Reina, G., Foglia, M., (2006): *Computer vision technology for agricultural robotics,* Sensor Review, Vol. 26 No. 4, pp. 290–300.

[14] Kise, M., Bonefas, Z. T., Moorehead, S. J., Reid, J. F. (2010): *Performance Evaluation on Perception Sensors for Agricultural Vehicle Automation*, Proc. of MCG 2010.

[15] Arima, S., Kondo, N. Monta, M., (2004): *Strawberry harvesting robot on table-top culture*, ASAE Paper No. 04-3089, ASAE, St Joseph, MI.

[16] Kanemitsu, M., Yamamoto, K., Shibano, Y., Goto, Y. and Suzuki, M. (1993): *Development of a Chinese cabbage harvester (Part 1)*, JSAM, Vol. 55 No. 5, pp. 133-140.

[17] Edan, Y. and Rogozin, V., (1992): *Robotic melon harvesting: prototype and field tests*, ASAE Paper No. 94-3073, ASAE, St Joseph, MI.

[18] Ulrich, I. and Nourbakhsh, I., (2000): *Appearance-Based Obstacle Detection with Monocular Color Vision*, Proceedings of the AAAI National Conference on Artificial Intelligence, Austin, TX.

[19] Rankin, A., Huertas, A., Matthies, L., (2005): *Evaluation of Stereo Vision Obstacle Detection Algorithms for Off-Road Autonomous Navigation*, Proc. of the 32nd AUVSI Symposium on Unmanned Systems.

[20] Broggi, A., Caraffi, C., Fedriga, R.I., Grisleri, P., (2005): *Obstacle Detection with Stereo Vision for Off-Road Vehicle Navigation*, Proc. of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05).

[21] Maimone, M., Cheng, Y., Matthies, L., (2007): *Two years of visual odometry on the Mars Exploration Rovers*, Journal of Field Robotics, Special Issue on Space Robotics.

[22] Matthies, L., Bergh, C., Castano, A., Macedo, J., Manduchi, R., (2003): *Obstacle Detection in Foliage with Ladar and Radar*, International Symposium on Robotic Research.

[23] Mosteller, F. and J. Tukey: *Data Analysis and Regression*, Addison-Wesley, 1977.

[24] Sachs, L., Applied Statistics: *A Handbook of Techniques*, Springer-Verlag, 1984, p. 253.