

**ALMA MATER STUDIORUM – UNIVERSITA' DI BOLOGNA  
SEDE DI CESENA  
SECONDA FACOLTA' DI INGEGNERIA CON SEDE A CESENA  
CORSO DI LAUREA IN INGEGNERIA BIOMEDICA**

**SVILUPPO E CARATTERIZZAZIONE DI ALGORITMI DI  
SEGMENTAZIONE PER L'ANALISI DEL CAMMINO IN ACQUA  
CON TECNICA MARKERLESS**

Elaborato in  
Biomeccanica

Relatore  
Prof.ssa Silvia Fantozzi

Presentata da  
Valeria Mondini

Correlatore  
Ing. Luca Tersì

Sessione Seconda  
Anno Accademico 2011-2012



# Abstract

Recenti studi hanno evidenziato come il cammino in ambiente acquatico possa portare a notevoli benefici nell'ambito di un processo riabilitativo: il cammino in acqua è infatti oggi considerato una delle principali terapie per pazienti con disturbi nella deambulazione, oltre ad essere impiegato per migliorare il recupero a seguito di interventi ed infortuni. Una caratterizzazione biomeccanica del cammino umano in acqua permetterebbe tuttavia di giungere a una conoscenza più approfondita degli effetti di quest'attività sul processo riabilitativo, e dunque a una sua prescrizione più mirata come parte delle terapie. Nonostante il crescente interesse, uno dei motivi per cui ancora pochi studi sono stati condotti in questo senso risiede nell'inadeguatezza di molti dei tradizionali sistemi di Motion Capture rispetto all'impiego subacqueo. La nuova branca della Markerless Motion Capture potrebbe invece in questo senso rappresentare una soluzione. In particolare, ci si occuperà in questo lavoro di tesi della tecnica markerless basata sulla ricostruzione del visual hull per retroproiezione delle silhouette. Il processo iniziale che permette di ottenere le silhouette dai video delle acquisizioni è detto *segmentazione*, la quale è anche una fase particolarmente importante per ottenere una buona accuratezza finale nella ricostruzione della cinematica articolare. Si sono pertanto sviluppati e caratterizzati in questo lavoro di tesi sette algoritmi di segmentazione, nati specificamente nell'ottica dell'analisi del cammino in acqua con tecnica markerless. Si mostrerà inoltre come determinate caratteristiche degli algoritmi influenzino la qualità finale della segmentazione, e sarà infine presentato un ulteriore algoritmo di post-processing per il miglioramento della qualità delle immagini segmentate.



# Indice

<b>Abstract.....</b>	<b>3</b>
<b>Introduzione .....</b>	<b>8</b>
<b>1. Acquisire il movimento.....</b>	<b>12</b>
<b>1.1 Motion Capture: definizione e ambiti applicativi .....</b>	<b>12</b>
<b>1.2 L'acquisizione del movimento allo stato dell'arte .....</b>	<b>13</b>
1.2.1 I metodi non ottici .....	16
1.2.2 La stereofotogrammetria .....	19
<b>1.3 La tecnica markerless.....</b>	<b>23</b>
1.3.1 Descrizione della tecnica .....	25
1.3.2 Vantaggi e problemi .....	32
<b>1.4 L'analisi del cammino in acqua con tecnica markerless .....</b>	<b>33</b>
1.4.1 Vantaggi offerti dall'ambiente acquatico sul processo riabilitativo .....	33
<b>1.5 Obiettivo della tesi .....</b>	<b>35</b>
<b>2. Materiali e Metodi .....</b>	<b>38</b>
<b>2.1 Materiale analizzato .....</b>	<b>38</b>
<b>2.2 Algoritmi sviluppati per la segmentazione del soggetto .....</b>	<b>39</b>
2.2.0 Creazione di un modello per il background.....	40
2.2.1 Sottrazione/divisione del background e Sogliatura .....	41
2.2.2 Naive Bayes Classifier.....	43
2.2.3 Naive Bayes Classifier, applicato a immagini già divise per il background.....	47
2.2.4 Naive Bayes Classifier, con aggiornamento guidato dei parametri del modello assumendo andamenti lineari dei parametri..	48
2.2.5 Naive Bayes Classifier, con aggiornamento guidato dei parametri del modello assumendo andamenti lineari dei parametri, applicato a immagini già divise per il background .....	50
2.2.6 Naive Bayes Classifier, con update automatico del modello ad ogni frame analizzato .....	52

2.2.7 Naive Bayes Classifier, con update automatico del modello ad ogni frame analizzato, applicato a immagini già divise per il background.....	53
<b>2.3 Algoritmi per il miglioramento della segmentazione .....</b>	<b>53</b>
2.3.1 L'algoritmo "regionGrowing" .....	53
<b>2.4 Metodi di valutazione degli algoritmi .....</b>	<b>55</b>
2.4.1 Gold-Standard e parametri per la valutazione .....	55
<b>3. Risultati.....</b>	<b>66</b>
<b>3.1 Tabelle riassuntive e istogrammi .....</b>	<b>73</b>
3.1.1 Algoritmo $SD_B$ .....	74
3.1.2 Algoritmo NBC .....	75
3.1.3 Algoritmo $NBC_B$ .....	76
3.1.4 Algoritmo $NBC_{AG}$ .....	77
3.1.5 Algoritmo $NBC_{AG,B}$ .....	78
3.1.6 Algoritmo $NBC_{AA}$ .....	79
3.1.7 Algoritmo $NBC_{AA,B}$ .....	80
<b>3.2 Distanza di Hausdorff.....</b>	<b>81</b>
<b>4. Discussione dei risultati.....</b>	<b>83</b>
<b>4.1 L'algoritmo Sottrazione/Divisione e Sogliatura .....</b>	<b>83</b>
<b>4.2 Algoritmi con aggiornamento automatico del classificatore ...</b>	<b>84</b>
<b>4.3 Efficacia dell'algoritmo regionGrowing.....</b>	<b>86</b>
<b>4.4 Il risultato migliore .....</b>	<b>88</b>
<b>4.5 Contestualizzazione del presente lavoro di tesi .....</b>	<b>92</b>
<b>Conclusioni .....</b>	<b>99</b>
<b>Bibliografia .....</b>	<b>103</b>



# Introduzione

Un sistema per la Motion Capture è un sistema che ha lo scopo di fornire una rappresentazione matematica quantitativa del movimento reale, tale da rendere il movimento stesso fruibile come input per successive elaborazioni. Anche se le origini sono piuttosto antiche, soprattutto negli ultimi decenni le tecniche di Motion Capture hanno conosciuto un grande sviluppo. In particolare, l'analisi del cammino in ambito clinico ha ricoperto un ruolo sempre più importante sia per la diagnosi di malattie muscolo-scheletriche che per la quantificazione dei benefici ottenuti dalle terapie riabilitative. Recentemente, diversi studi hanno inoltre evidenziato come il cammino in ambiente acquatico possa risultare particolarmente utile nell'ambito di un processo riabilitativo. Sarebbe dunque importante essere in grado di fornire una caratterizzazione biomeccanica del cammino umano anche in acqua, per ottenere una più mirata prescrizione di quest'attività come parte di una terapia riabilitativa. La maggior parte delle tecniche Motion Capture prevede tuttavia ad oggi strumentazioni che difficilmente possono essere impiegate anche in ambiente acquatico. Una possibile soluzione a questo problema può tuttavia scaturire dalla nuova branca della Markerless Motion Capture. I punti di forza di queste nuove tecniche riguardano la necessità di una strumentazione più ridotta e meno specifica e, soprattutto, la totale assenza di dispositivi applicati al corpo del soggetto, rendendo in questo modo possibile lo studio di un movimento naturale e libero da costrizioni. Ad oggi, diversi metodi sono stati proposti per eseguire l'analisi del movimento con tecnica markerless. Uno dei più diffusi è un metodo model-based che sfrutta il matching sui dati sperimentali di un modello del soggetto, dove i dati sperimentali assumono la forma tridimensionale di visual hull (approssimazione del volume di spazio occupato dal soggetto in un istante). Il primo passo delle elaborazioni in questo tipo di tecnica

consiste nel riconoscimento automatico del soggetto rispetto al background, in tutti i video provenienti dalle telecamere sincronizzate che riprendono il movimento. Questo tipo di riconoscimento del soggetto è detto *segmentazione*, e produce una binarizzazione delle immagini in ingresso allo scopo di ottenere le sagome (*silhouette*) che, retroproiettate nello spazio a partire dai piani delle telecamere, permettono la creazione del visual hull. La segmentazione delle immagini è un passo di fondamentale importanza per garantire una buona ricostruzione del visual hull e quindi, in ultima analisi, un'accurata ricostruzione della cinematica del soggetto. Anche se vi è un crescente interesse, ancora pochi studi sono stati ad oggi effettuati sull'acquisizione del cammino in acqua con tecnica markerless e, conseguentemente, relativamente pochi strumenti specifici sono stati messi a punto per questo particolare ambito applicativo. In questo lavoro di tesi sono dunque stati sviluppati e caratterizzati sette algoritmi di segmentazione, nati specificamente nell'ottica dell'analisi del cammino in acqua. In particolare, uno degli obiettivi sarà quello di mostrare come varia la qualità della segmentazione aggiungendo/rimuovendo particolari gradi di complessità all'algoritmo di segmentazione. Un ulteriore algoritmo per il post-processing delle immagini binarizzate, il *regionGrowing*, sarà inoltre impiegato, valutando anche in questo caso gli effetti indotti sulla qualità della segmentazione.

Nel Capitolo 1, *Acquisire il movimento*, sarà data una definizione più precisa della Motion Capture, discutendo brevemente anche l'evoluzione delle tecniche dalle origini storiche fino allo stato dell'arte. Una panoramica delle principali tecniche di Motion Capture allo stato dell'arte sarà dunque proposta, fino a descrivere più in dettaglio la nuova Markerless Motion Capture. Si concluderà infine il capitolo ricordando quali siano i principali vantaggi offerti dall'ambiente acquatico nell'ambito del processo riabilitativo.

Il Capitolo 2, *Materiali e Metodi*, può essere idealmente diviso in tre parti. Nella prima parte del capitolo si fornirà una descrizione dettagliata del materiale impiegato nel presente lavoro di tesi. Nella

seconda parte del capitolo si descriveranno in dettaglio i sette algoritmi di segmentazione implementati e l'algoritmo di miglioramento della segmentazione post-processing *regionGrowing*. Nell'ultima parte del capitolo si definiranno infine i criteri impiegati per valutare la qualità della segmentazione ottenuta con i diversi algoritmi, definendo da un lato quali siano i gold-standard assunti e dall'altro quali parametri siano stati considerati per le valutazioni. In particolare, saranno principalmente valutate le percentuali di pixel del background e del foreground riconosciute correttamente nelle segmentazioni.

Il Capitolo 3, *Risultati*, riporta i risultati ottenuti, mostrando per ogni algoritmo i valori dei parametri considerati per la valutazione della qualità della segmentazione, sia in forma sintetica che per via grafica, con l'uso di istogrammi e box-plot. Saranno inoltre riportati in questa sezione gli esiti dei test statistici ANOVA eseguiti sui diversi risultati, e definiti alla fine del Capitolo 2.

Nel Capitolo 4, *Discussione dei risultati*, si discuteranno alcuni dei risultati ottenuti e mostrati al Capitolo 3, cercando di motivarli. Si discuterà inoltre su quale tra gli algoritmi implementati fornisca i risultati complessivamente migliori, cercando infine di rapportare questo lavoro di tesi alla letteratura sull'argomento.



# 1. Acquisire il movimento

## 1.1 Motion Capture: definizione e ambiti applicativi

Se parliamo di “motion capture” o, come spesso si abbrevia, di “MoCap”, una definizione piuttosto comune e largamente accettata è quella fornita da Menache nel suo libro [1]: “Motion Capture is the process of recording a live motion event and translating it into usable mathematical terms by tracking a number of key points in space over time and combining them obtaining a single 3D representation of performance”.

In accordo con la definizione appena data, scopo della motion capture è dunque quello di “digitalizzare” il movimento di un soggetto, fornendo del movimento reale una rappresentazione matematica quantitativa che renda il movimento stesso facilmente fruibile come input per successivi studi e successive elaborazioni. Tale caratteristica rende le tecniche di motion capture utili e interessanti per un grande ventaglio di applicazioni, e non solo per studi di biomeccanica [2], [3], ma anche ad esempio per scopi di sorveglianza, nella realtà virtuale o nel cinema.

Tra i vari usi delle tecnologie per la motion capture, ricordiamo innanzitutto quelli in ambito clinico e in ambito sportivo. Per quanto riguarda gli studi clinici, le tecniche di motion capture sono necessarie per pervenire ad una conoscenza accurata della cinematica del soggetto, per poter diagnosticare i problemi di locomozione dei pazienti. Per quanto riguarda l’ambito sportivo, le tecniche MoCap vengono comunemente utilizzate per registrare il movimento al fine di individuare eventuali raffinamenti da attuare per il miglioramento della performance sportiva nel suo complesso [2].

Un altro ambito di utilizzo delle tecnologie di motion capture è quello della sicurezza: le tecniche MoCap possono infatti essere impiegate nella progettazione di sistemi di sorveglianza intelligenti. A titolo di

esempio, sistemi di MoCap sono stati utilizzati per il riconoscimento dell'andatura nel cammino di un soggetto come segnale di riconoscimento biometrico. È stato infatti proposto [4] che l'andatura sia una caratteristica sufficientemente distintiva da permettere la discriminazione tra soggetti diversi [2].

Citiamo infine, a conclusione di questa breve carrellata di esempi, un altro importante campo di applicazione per queste tecnologie: la realtà virtuale. Grazie alle tecnologie MoCap, infatti, essendo in grado di tradurre il movimento umano in modelli digitali comprensibili, eventualmente, anche dalle macchine, è possibile infine per utenti umani interagire con contenuti digitali in tempo reale: questo aspetto può risultare molto utile ad esempio nella costruzione di simulazioni di allenamento, o per eseguire test di percettibilità visiva, nelle terapie di biofeedback oppure ancora per donare in generale un movimento naturale a personaggi creati interamente al computer (caratteristica, questa, molto apprezzata e utilizzata migliorare i rendering degli effetti speciali in TV, nei film e nei video games [3] [5]).

Una volta introdotto cosa sia la motion capture ed elencati alcuni tra i molti possibili ambiti applicativi, andiamo a descrivere nel prossimo paragrafo quali siano, allo stato dell'arte, le principali tecniche di acquisizione del movimento.

## **1.2 L'acquisizione del movimento allo stato dell'arte**

L'interesse per l'analisi del movimento in generale e, in particolare, per l'analisi del cammino, è in realtà un interesse molto antico, che si rifà ai tempi dell'antica Grecia, e anche se la sua evoluzione passa attraverso i nomi dei più grandi scienziati, filosofi e teorici di sempre, da Aristotele (384-322 a.C.) a Thomas Young (1773-1829), fino ai giorni nostri [6], è stata in particolare l'invenzione della fotografia nell'800 a contribuire all'evoluzione di ciò che oggi è noto come motion capture [3]. In particolare, è piuttosto comune l'identificazione nel lavoro di Etienne Jules Marey (1830-1904) e Eadweard Muybridge (1830-1904) un passo fondamentale per lo sviluppo della motion

capture, fino alla sua evoluzione a come la conosciamo oggi [2], [3]: fu infatti grazie all'invenzione del cronofotografo ad opera di Marey (Figura 1.1) che fu possibile per Muybridge, negli anni successivi, continuare l'analisi sul modo in cui sia gli animali che gli umani si muovevano, scattando fotografie a umani e animali nell'atto di compiere svariate attività [3]. Questi esperimenti di Muybridge furono collezionati, e sono diventati famosi con i nomi di *Animal Locomotion* [7] e di *The Human Figure in Motion*. [8] (Figura 1.2).

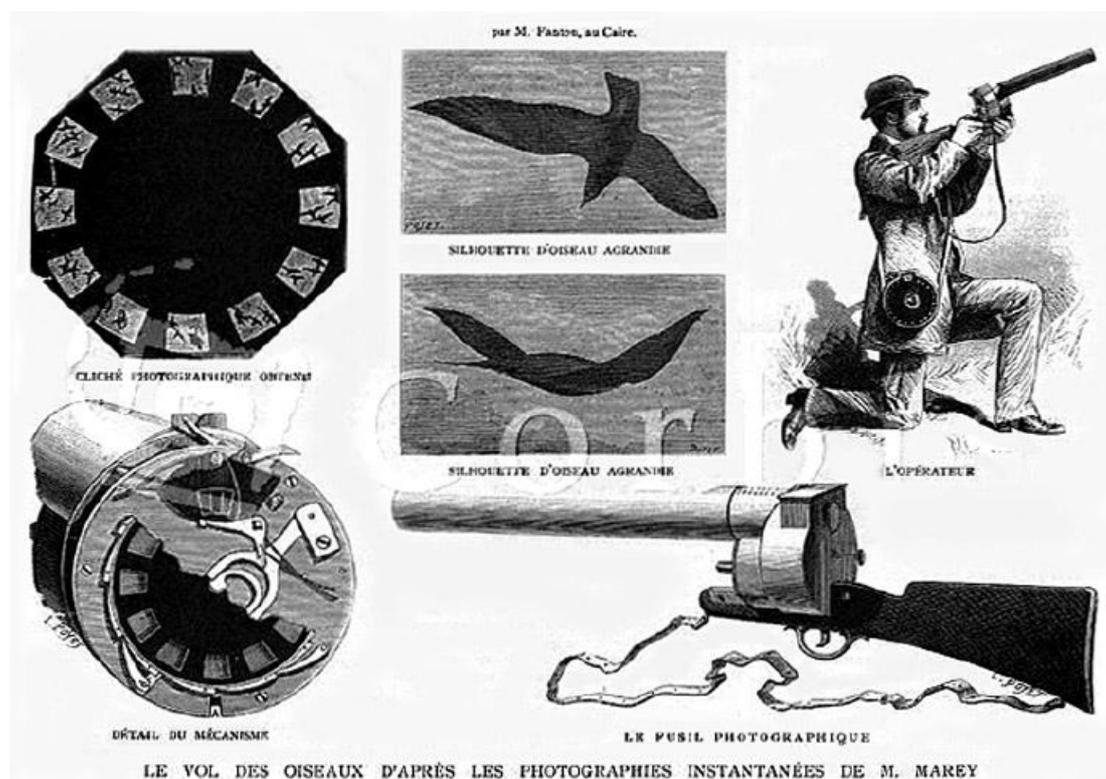


Figura 1.1 Il fucile fotografico di Marey, 1882

Dopo il lavoro di Marey e Muybridge, nel citare quali altri contributi abbiano permesso l'evoluzione delle tecniche di motion capture da allora fino ad oggi, è d'obbligo menzionare anche il lavoro di G. Johansson che, nel 1973, nei suoi Moving Light Display Experiments [9] utilizzò dei piccoli marker riflettenti applicati alle giunture di soggetti umani per registrare il loro movimento. L'obiettivo del lavoro di Johansson era quello di essere in grado di identificare movimenti noti solo a partire dall'analisi delle traiettorie dei marker [2], [3], il che ci avvicina sempre più a quella che è la

moderna analisi del movimento, ad esempio, tramite stereofotogrammetria.

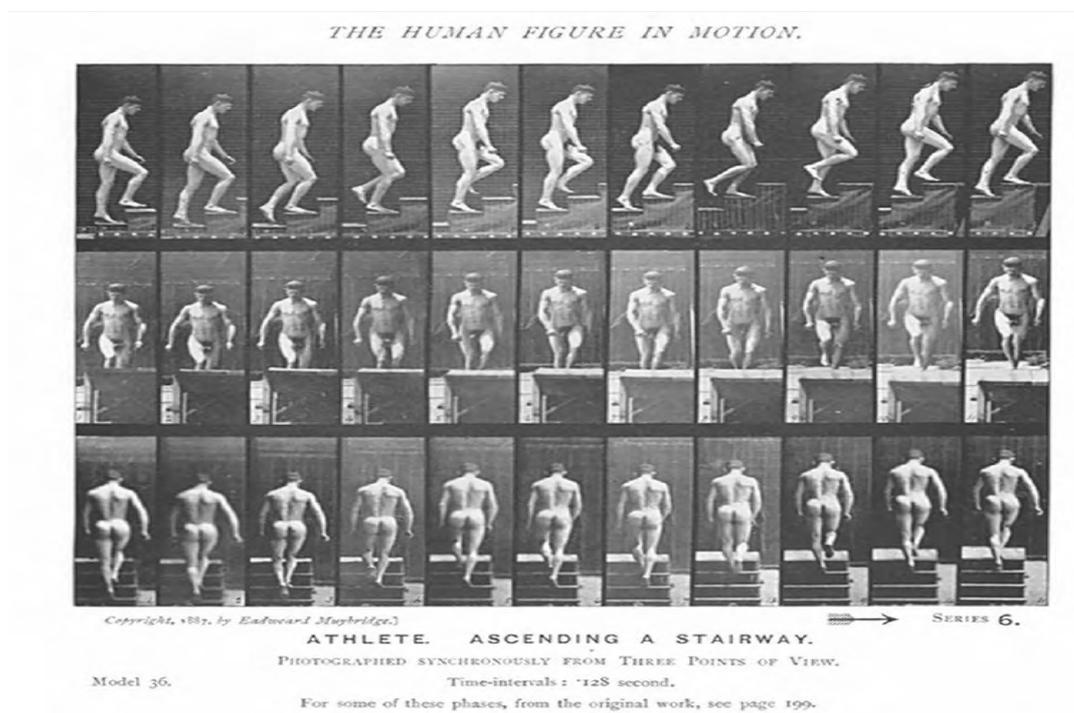


Figura 1.2 E. Muybridge. Athlete. Ascending a Stairway [8]

Ad oggi, generalmente parlando, possiamo dividere le tecniche per l'analisi del movimento in tre grandi branche, a seconda delle finalità dall'analisi stessa: analisi del movimento facciale e riconoscimento dei volti, analisi e riconoscimento della gestualità, analisi del movimento del corpo. La prima branca riguarda il riconoscimento dei volti in un'immagine (solitamente in congiunzione con sistemi di sicurezza), oppure può essere finalizzata alla creazione di modelli di volti che vengano utilizzati successivamente nelle animazioni. L'analisi della gestualità può essere intesa come un riconoscimento della gestualità o di entrambe le mani con le braccia o anche delle sole mani, al fine di poter utilizzare questo tipo di linguaggio del corpo come input per una comunicazione uomo-macchina. Infine, l'analisi del movimento del corpo nella sua globalità, con un livello di dettaglio che tendenzialmente esclude i movimenti più fini (come ad esempio le espressioni del volto), si riduce fondamentalmente alla stima della posa del soggetto istante per istante [2], assunto un modello che descriva il soggetto stesso.

In biomeccanica e nelle applicazioni cliniche e sportive, è principalmente l'ultima branca descritta quella di maggior interesse, poiché è grazie a questo tipo di analisi che sarà poi possibile studiare la cinematica del soggetto. Maggiori dettagli verranno dunque di seguito forniti sulle principali tecniche MoCap relative a questo tipo di analisi.

Allo stato dell'arte, le tecniche MoCap esistenti per l'analisi della cinematica di un soggetto si possono concettualmente dividere in due grandi categorie: le tecniche che si basano su approcci non ottici, e quelle che sfruttano principi ottici [3].

Per le tecniche "non ottiche", possiamo individuare tre approcci principali: approccio elettromeccanico, elettromagnetico e approccio inerziale.

Per i metodi ottici, distinguiamo principalmente tra le tecniche che si avvalgono dell'utilizzo di marker e le tecniche markerless.

### **1.2.1 I metodi non ottici**

Nell'ambito dei sistemi MoCap non ottici, uno tra i primi approcci implementati fu quello elettromeccanico [10]: i sistemi basati su questo tipo di approccio consistono, ad oggi, in speciali tute indossabili che formano una sorta di "esoscheletro" per il soggetto in movimento, munite di dispositivi di misura in alcuni punti chiave (Figura 1.3). I sistemi di misura presenti sono generalmente costituiti da potenziometri uniti a speciali dispositivi scorrevoli (*sliding rods*) che misurano gli spostamenti tra i giunti della tuta indossabile, e da sensori (quali ad esempio elettrogoniometri [11]) in grado di misurare gli angoli tra i segmenti adiacenti dell'esoscheletro. Quando il soggetto si muove, l'azione viene rilevata attraverso piccoli cambiamenti di tensione ai capi dei potenziometri. Le tute ad oggi possono integrare anche sensori di tipo inerziale e forniscono risultati accurati (la nuova tuta-esoscheletro *Gypsy 7* misura gli angoli tra i segmenti dell'esoscheletro con una risoluzione di 0.125 gradi) ma condizionano il movimento perché sono pesanti e non permettono una piena libertà di azione: il principale problema di queste tute è infatti

proprio il loro andare a limitare o comunque a condizionare il movimento che si sta analizzando [2].



*Figura 1.3 Tuta indossabile MoCap elettromeccanica (a sinistra) e magnetica (a destra)*

Nella MoCap basata su approccio elettromagnetico, la libertà di movimento del soggetto migliora. Le tute indossabili realizzate rispetto a questo tipo di approccio (Figura 1.3) presentano sensori posizionati in punti chiave del corpo, dove ogni sensore è fondamentalmente costituito da un insieme di bobine orientate secondo le tre direzioni dello spazio [12]. Il funzionamento del sistema prevede inoltre la generazione di un campo magnetico esterno per cui alla fine sia possibile, incrociando i dati sul flusso del campo magnetico attraverso ciascuna delle bobine costituenti i sensori, ricostruire posizione e orientamento di ciascun sensore.

Le tute elettromagnetiche così costituite sono più comode e leggere di quelle elettromeccaniche [2], pertanto i movimenti del soggetto vengono meno condizionati.

Il problema principale legato a questi sistemi è la difficoltà d'impiego al di fuori dell'ambiente controllato di laboratorio, poiché in presenza di materiali ferromagnetici il campo viene distorto e le misure perdono in accuratezza [11]. Infine, in questo caso vi è una limitazione rispetto all'acquisizione di movimenti troppo ampi, data la necessità per i sensori di rimanere sempre all'interno dello spazio pervaso dal campo

magnetico imposto dal generatore [5]. Per quanto riguarda l'accuratezza, è possibile con le tecniche odierne ricostruire la posizione dei marker magnetici con un errore massimo di circa 2mm [13].

Una terza categoria di sistemi MoCap non ottici è quella che utilizza sensori inerziali: l'idea alla base di questi sistemi è quella di poter ricostruire il movimento del soggetto a partire dalla misura delle accelerazioni in gioco. Quando si parla di sensori inerziali, ci si riferisce comunemente ad accelerometri e giroscopi.

L'accelerometro è uno strumento in grado di tradurre un'accelerazione lineare in una grandezza elettrica. Il dispositivo è fondamentalmente costituito da una massa nota, libera di muoversi all'interno dello strumento per effetto delle forze inerziali: lo spostamento compiuto dalla massa è poi legato alla forza (e quindi all'accelerazione) per mezzo di una relazione nota, e viene rilevato attraverso l'effetto capacitivo o grazie all'uso di estensimetri. Ad oggi esistono accelerometri mono-, bi- e triassiali che, grazie alla tecnologia MEMS (Micro Electro-Mechanical Systems) possono essere integrati su un unico chip [14]. Il grado di accuratezza di questo tipo di accelerometro con tecnologia MEMS può arrivare fino a  $25\mu\text{g}$  (con g accelerazione di gravità) [15].

Il giroscopio è un dispositivo in grado di tradurre una velocità angolare in una grandezza elettrica secondo una relazione nota: la tipologia più comune si basa sulla legge di Coriolis. Grazie alla tecnologia MEMS, anche del giroscopio esistono attualmente versioni mono- o triassiali integrate su un unico chip. Il grado di accuratezza di un giroscopio di questo tipo può arrivare fino a  $0.05^\circ/\text{s}$  [15].

Il problema intrinsecamente legato alla MoCap basata su sensori inerziali è la sensibilità di queste tecniche rispetto a offset e derive, dato che per ricondursi a spostamenti e velocità a partire dalle accelerazioni occorre effettuare processi di integrazione numerica. Per questo motivo, ad oggi questi dispositivi vengono integrati in unità multi-sensore, in modo da poter sfruttare la ridondanza di informazione per contenere l'effetto degli errori [14].

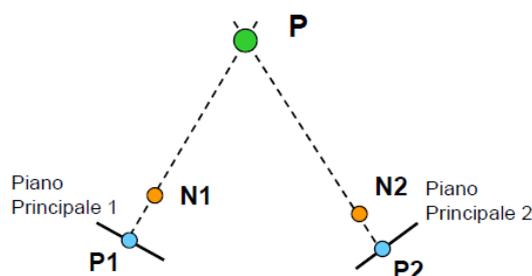
Nonostante tutte le tecniche MoCap appena presentate permettano, ad oggi, una buona accuratezza nella ricostruzione della posizione dei marker (o della configurazione dell'esoscheletro), ciò non garantisce si riesca a ottenere un'altrettanto accurata stima della cinematica articolare. Ad esempio, un artefatto in grado di inficiare l'accuratezza di queste tecniche è l'artefatto da tessuti molli. Maggiori dettagli su questo tipo di artefatto saranno forniti nel prossimo paragrafo.

### 1.2.2 La stereofotogrammetria

I sistemi MoCap ottici sono sistemi che si basano sull'utilizzo di una o di un set di telecamere sincronizzate per ricostruire in ogni istante la posa 3D di un soggetto in movimento [16].

Concettualmente, possiamo classificare i sistemi MoCap ottici a seconda del numero di punti di vista assunti durante l'acquisizione (sistemi monoculari o multi-view) e a seconda dell'impiego o meno di marker (siano questi attivi o passivi), distinguendo dunque due tipi di approccio: il marker-based e il markerless. Nei sistemi marker-based l'obiettivo è ricostruire la cinematica dei segmenti ossei attraverso l'individuazione della traiettoria 3D dei marker (piccoli oggetti applicati alla superficie corporea del soggetto in movimento), mentre le tecniche markerless mirano a estrarre i parametri relativi al movimento a partire da silhouettes o altre caratteristiche del soggetto nelle immagini (come ad esempio i contorni) [16].

In questo paragrafo ci occuperemo della descrizione delle tecniche multi-view e marker-based, mentre maggiori dettagli sulla tecnica markerless verranno forniti in seguito, nella sezione 1.3.



*Figura 1.4 Ricostruzione della posizione del punto P, note le sue proiezioni P1 e P2 sui piani principali delle telecamere e le posizioni N1 e N2 dei punti nodali delle telecamere*

Il funzionamento delle tecniche marker-based si basa fondamentalmente sul principio di triangolazione: quando almeno due telecamere riprendono contemporaneamente un punto dello spazio 3D allora è possibile, a partire dalle immagini formatesi sui piani principali delle telecamere, ricostruire la posizione di quel punto tramite semplici retroproiezioni (Figura 1.4). La tecnica MoCap ottica marker-based di questo tipo è detta anche stereofotogrammetria, dove il prefisso *stereo*- discende proprio dal principio appena esposto.

Per ricostruire la posizione di un punto tramite triangolazione è necessario aver definito a priori un sistema di riferimento globale. Occorre inoltre che i parametri dell'intero sistema stereofotogrammetrico siano noti, sia i parametri esterni (posizione e orientamento dei piani principali delle telecamere) che i parametri interni (come lunghezza focale, coordinate del punto nodale, coefficienti di distorsione delle telecamere). In altri termini, prima di poter effettuare le acquisizioni è necessario calibrare lo spazio all'interno del quale verrà eseguito il movimento.

La calibrazione avviene di norma in due fasi. Nella prima fase viene acquisito un oggetto di calibrazione composto da 3 assi ortogonali con marker attaccati in posizioni note, allo scopo di definire il sistema di riferimento globale e di ottenere posizione e orientamento dei piani delle telecamere. Nella seconda fase della calibrazione una barra rigida (*wand*) con marker attaccati in posizioni note viene mossa lungo le 3 direzioni dello spazio all'interno del volume di calibrazione per ottenere i parametri interni delle telecamere.

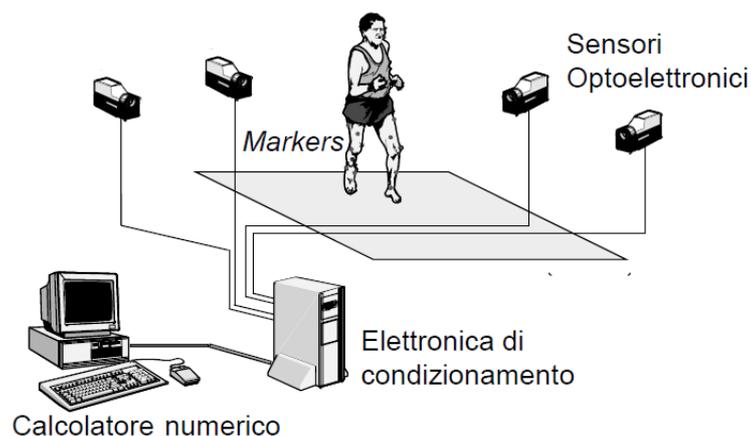


Figura 1.5 Architettura di un sistema stereofotogrammetrico

Una tipico esempio di architettura di un sistema stereofotogrammetrico è mostrato in Figura 1.5.

I marker utilizzati in stereofotogrammetria possono essere di due tipi: passivi e attivi. I marker passivi sono semplicemente piccole sferette leggere uniformemente ricoperte da materiale riflettente, mentre i marker attivi sono veri e propri LED in grado di emettere luce propria nello spettro infrarosso.

I sensori optoelettronici utilizzati sono telecamere che lavorano nello spettro infrarosso e integrano dispositivi per la rilevazione e trasduzione del segnale luminoso in un segnale elettrico, quali ad esempio i sensori CCD (charged coupled device) e CMOS (complementary metal oxide semiconductor). Indipendentemente dalla tecnologia, i sensori vengono solitamente organizzati in matrici di pixel sul piano immagine della telecamera [17].

Nel caso in cui il sistema lavori con marker passivi, le telecamere sono inoltre costruite in modo da emettere esse stesse fasci di luce infrarossa. Dato che i marker passivi sono ricoperti da materiale riflettente, il segnale luminoso di ritorno proveniente da questi sarà molto più intenso rispetto a un'eventuale riflessione di luce da parte del background, permettendo così una facile localizzazione dei marker. Impostando infine un'adeguata soglia di luminosità per le telecamere, la probabilità che vengano erroneamente riconosciuti segnali provenienti da pelle o vestiti del soggetto si riduce drasticamente [18].

Nel caso in cui vengano impiegati marker attivi non è necessario siano le telecamere a illuminare lo spazio, essendo gli stessi marker a emettere un segnale luminoso. Si noti che con questa seconda modalità di acquisizione si ha un minor consumo di potenza, dato che il segnale luminoso deve percorrere la metà dello spazio per incidere sul piano immagine della telecamera, rispetto al caso in cui si impiegano marker passivi. Un altro vantaggio legato ai marker attivi riguarda una miglior tracciabilità degli stessi rispetto ai marker passivi. Questa miglior tracciabilità è dovuta alla possibilità di diversificare i tempi di attivazione e le frequenze di pulsazione dei LED: potendosi basare

anche su questi parametri, il sistema riconosce più facilmente ciascun marker [19].

Nonostante queste considerazioni positive sui marker attivi, viene ancora tendenzialmente preferito l'uso dei sistemi a marker passivi, dato che in molti casi è ancora necessario il wiring per l'alimentazione dei marker attivi, il che va a limitare il movimento del soggetto [17] [3]. Con l'avvento del wireless, tuttavia, anche questo problema sta scomparendo.

Tra tutti i sistemi MoCap visti finora, quelli che hanno riscosso maggior successo sono i sistemi ottici. I principali motivi del successo riguardano innanzitutto il ridotto ingombro sul soggetto [3], e la possibilità di acquisire, con opportuni setup, anche movimenti ampi [5] come il cammino, la corsa, il ballo. Un altro importante vantaggio di questo tipo di MoCap sulle altre tecniche riguarda la flessibilità nella scelta della posizione e del numero dei marker a seconda del tipo di acquisizione [5], anche se numerosi protocolli sul posizionamento dei marker sono stati sviluppati (come il SAFLo, il VCM, il CAST). Infine, i sistemi ottici possono raggiungere frequenze di campionamento tendenzialmente più elevate rispetto alle altre tecniche [5], permettendo l'analisi anche dei movimenti rapidi, e presentano una buona accuratezza: nei sistemi marker-based a marker passivi, ad esempio, l'errore di ricostruzione della posizione dei marker è dell'ordine dei mm o inferiore.

Nonostante la popolarità, l'approccio ottico presenta alcune importanti limitazioni che devono essere considerate [20].

Innanzitutto, anche se l'ingombro è limitato se comparato agli altri sistemi, la presenza di marker attaccati alla pelle può influenzare il movimento del soggetto. Essendo inoltre i marker molto piccoli, è possibile che in determinate fasi del movimento un marker venga nascosto alle telecamere, facendo così perdere temporaneamente la sua traccia. In corrispondenza di questi eventi capita anche che le traiettorie di marker molto vicini vengano confuse. Questi problemi possono tuttavia essere parzialmente superati con l'utilizzo di un

maggior numero di telecamere [19] e un adeguato posizionamento dei marker.

Altri svantaggi riguardano il fatto che il posizionamento dei marker è dispendioso in termini di tempo, non completamente ripetibile [21], e la presenza di tessuti molli tra ossa e marker causa un moto relativo che introduce un notevole artefatto nella misura, chiamato in gergo ATM, “artefatto da tessuto molle” [22]. Gli errori indotti dagli artefatti da tessuto molle sono errori sistematici, tempo-varianti, variabili da persona a persona e variabili a seconda del punto del corpo di cui si analizza il movimento. Questi errori possono arrivare ad essere dell’ordine di alcuni cm, che si ripercuotono in errori di diversi gradi sul calcolo degli angoli articolari. Dato infine che il contenuto in frequenza degli errori da ATM è lo stesso del movimento che viene acquisito, l’effetto degli errori non può essere attenuato con tecniche di filtraggio senza perdita di informazione utile [23].

Per ottenere infine dati di alta qualità con le tecniche marker-based, occorre eseguire il movimento in un ambiente controllato [24], tendenzialmente un ambiente indoor e con un volume di acquisizione limitato, mentre strumenti come ad esempio i sensori inerziali possono essere utilizzati anche al di fuori dei laboratori, e per sessioni di acquisizione anche prolungate (anche di diverse ore).

Anche se numerose soluzioni sono state proposte in letteratura per superare queste limitazioni, diversi aspetti rimangono ancora irrisolti. Un interesse sempre maggiore nel trovare valide alternative a questi metodi sta dunque emergendo, e la nuova area di ricerca sulla Markerless Motion Capture ne è un esempio [3].

### **1.3 La tecnica markerless**

La possibilità di ricostruire il movimento umano a partire unicamente da sequenze di immagini è un argomento che, negli ultimi vent’anni, ha riscosso un notevole interesse nella comunità scientifica, soprattutto grazie all’ampio spettro di applicazioni che sarebbe possibile sviluppare con questa capacità. Non è infatti un caso se la

Markerless Motion Capture ha avuto origine dai campi della computer vision e dell'apprendimento automatico, piuttosto che essere nata con una prospettiva clinica [3].

In computer vision, algoritmi di riconoscimento e cattura del movimento articolato sono in continuo sviluppo, e una grande varietà di sistemi vision-based è stata proposta per la ricostruzione del movimento umano. Per i nostri scopi, una maggiore attenzione verrà data ai metodi il cui fine è la stima della posa 3D del soggetto e, in particolare, ai metodi che impiegano più viste utilizzando più telecamere. Di questi metodi è possibile fornire una classificazione in base all'utilizzo o meno di un modello per il soggetto, distinguendo tra i metodi model-free, quelli a uso indiretto del modello, e quelli a uso diretto [3].

Nei metodi model-free la ricostruzione della posa 3D del corpo avviene senza l'utilizzo della conoscenza a priori data da un modello del soggetto. Nei metodi a uso indiretto del modello, questo viene utilizzato unicamente come riferimento per guidare l'interpretazione dei dati misurati, mentre nei metodi a uso diretto il modello stesso viene assunto come rappresentazione del soggetto osservato, in grado di fornire in ogni istante qualunque informazione desiderata, inclusa la posa del soggetto in quell'istante [3].

La maggior parte degli approcci è di tipo model-based: in questo tipo di metodo un modello definito a priori, contenente le informazioni sia anatomiche che cinematiche necessarie, viene utilizzato per il matching con i dati, che possono avere o una rappresentazione bidimensionale sul piano delle telecamere o tridimensionale sottoforma, ad esempio, di *visual hull* [25], come verrà descritto in seguito. L'uso di un modello introduce importanti vantaggi, come la possibilità di gestire meglio eventuali errori e artefatti sui dati, e la semplicità con cui diversi vincoli cinematici possono essere incorporati nel sistema, limitando in questo modo lo spazio di ricerca delle possibili pose assunte dal soggetto a quelle fisicamente possibili [3].

Nella seguente sezione, verranno introdotti i principali concetti e le basi della tecnica markerless model-based con viste da più telecamere.

### 1.3.1 Descrizione della tecnica

Quasi tutti i sistemi markerless che come input prendono dei video necessitano di uno stadio iniziale che identifichi l'oggetto di interesse nella sequenza di immagini: questo processo viene tecnicamente chiamato *segmentazione*, ed è una fase particolarmente critica in molte applicazioni di computer vision [3].

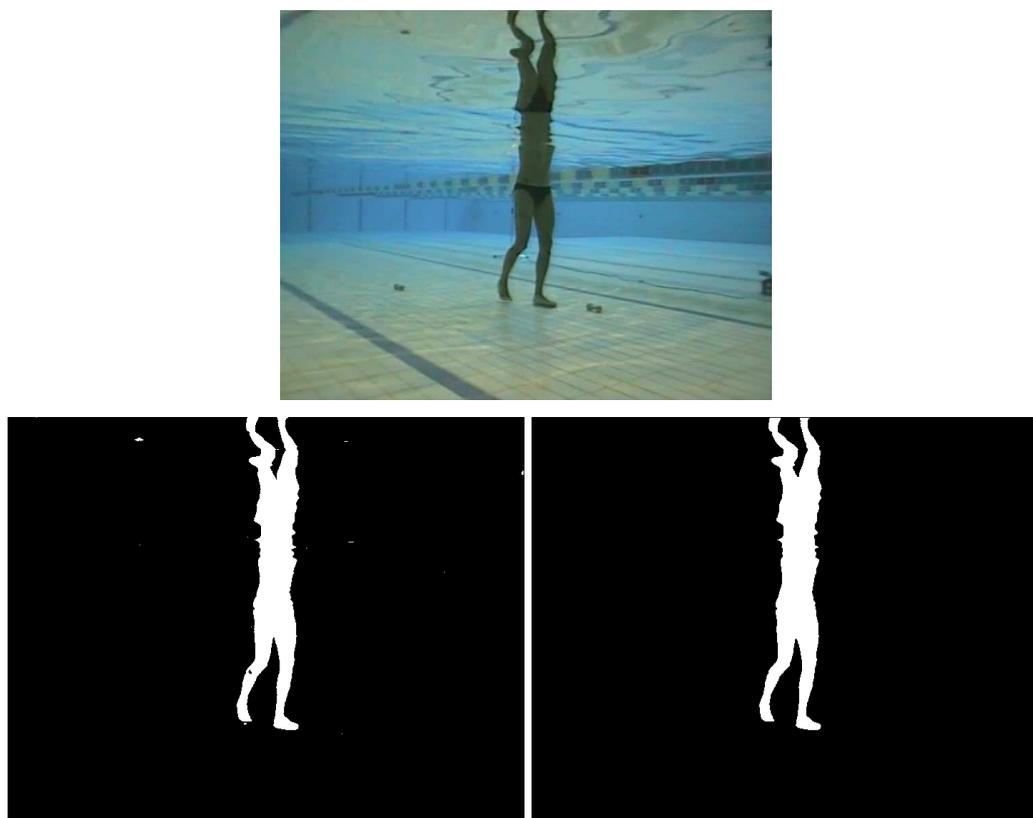
La situazione più favorevole in termini di segmentazione dell'immagine è quella che prevede che l'acquisizione avvenga in un ambiente controllato, dove i muri e il pavimento costituenti lo sfondo presentano un colore a tinta unita assente nel soggetto (solitamente blu o verde) [26]: in questo caso sarà sufficiente classificare come background tutti e solo i pixel che presentano il colore dello sfondo o un colore simile entro un fissato range di variabilità. La maggior parte delle applicazioni, tuttavia, coinvolge ambienti relativamente meno controllati, come ad esempio un laboratorio di analisi clinica del cammino oppure, nel caso di analisi del cammino in acqua, una piscina. Un approccio più complesso e generale è dunque solitamente necessario.

Un tipico metodo utilizzato per agevolare l'identificazione del soggetto è la sottrazione del background [27], ossia un'immagine che modella lo sfondo delle acquisizioni viene sottratta pixel per pixel a ognuno dei frame del video. In generale, l'effetto della sottrazione di due immagini è l'accentuazione delle differenze tra le due, il che significa nel nostro caso porre in evidenza il soggetto rispetto al background, facilitandone l'identificazione o tramite un semplice processo di sogliatura o tramite algoritmi di classificazione più complessi. Diversi modelli per il background sono stati ideati per trattare differenti problemi [28], e molti aspetti devono essere presi in considerazione nello sviluppo di un buon algoritmo di segmentazione del soggetto (il *foreground*) rispetto al background [27].

Al calcolatore, le immagini vengono semplicemente trattate come matrici di elementi, chiamati pixel (picture element). Se l'immagine è in bianco e nero, a ogni pixel sarà associato un numero che identifica il livello di grigio. Se l'immagine è a colori, a ogni pixel verranno

associati tre valori, corrispondenti ai livelli delle tre componenti di colore RGB (red, green, blue): sottrarre due immagini a colori significa dunque eseguire una sottrazione tra matrici tridimensionali, ossia per ogni pixel sottrarre separatamente le tre componenti di colore. La profondità di colore di un'immagine dipende dal numero di bit utilizzati per la codifica dei livelli di colore.

Indipendentemente dall'algoritmo utilizzato, il risultato della segmentazione è una binarizzazione dell'immagine, assegnando ad esempio il valore 0 (nero) a tutti i pixel del background e il valore 1 (bianco) a tutti i pixel del foreground. Solo l'informazione relativa alla sagoma del soggetto viene dunque alla fine mantenuta. Spesso, al termine della classificazione vengono applicati anche altri algoritmi correttivi che vadano a migliorare il risultato riempiendo eventuali piccoli buchi rimasti nella sagoma ed eliminando pixel spuri erroneamente classificati come foreground. Le immagini segmentate del soggetto così ottenute vengono chiamate *silhouettes* (Figura 1.6).



*Figura 1.6 Esempio di frame originale dell'acquisizione (in alto) ed estrazione della silhouette tramite segmentazione prima (a destra) e dopo (a sinistra) l'applicazione di un algoritmo correttivo*

Le silhouettes ottenute dalla segmentazione di tutte le differenti viste sincronizzate vengono impiegate per la generazione di un *visual hull* [29], che rappresenta un'approssimazione locale del volume istantaneamente occupato dal soggetto. La tecnica di ricostruzione del visual hull a partire dalle silhouettes fu originariamente introdotta da Laurentini [29]: note posizione e orientamento delle telecamere, a partire da ogni silhouette un generico cono può essere retroproiettato nello spazio; il volume compreso nell'intersezione di tutti i coni costituisce il visual hull (Figura 1.7).

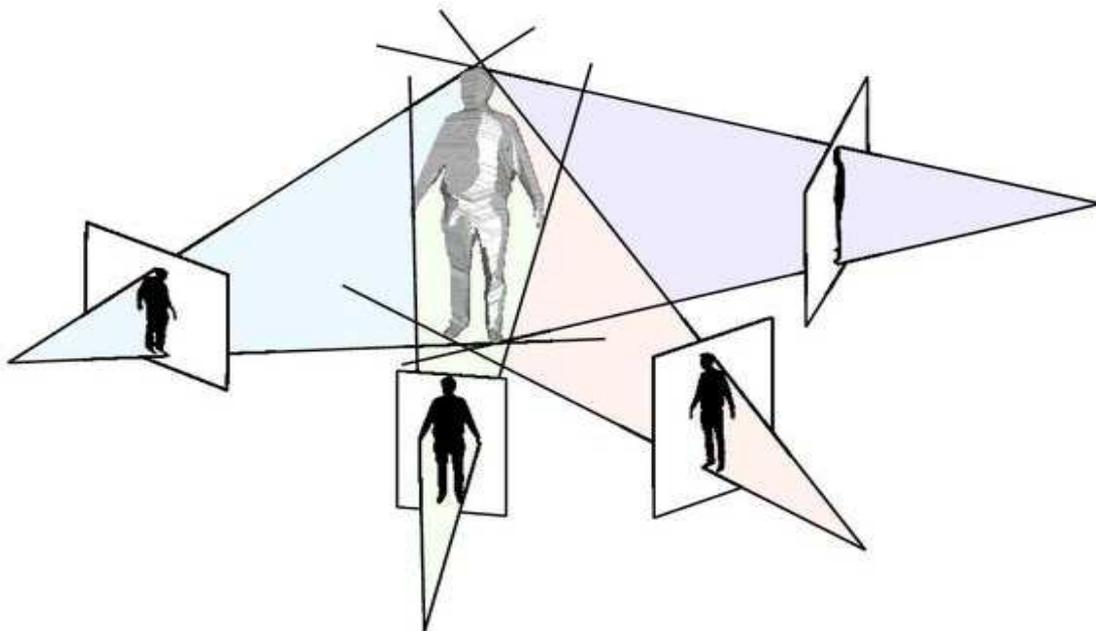


Figura 1.7 Generazione di visual hull a partire da 4 silhouettes

Il visual hull è costituito da un insieme di elementi discreti di volume, chiamati voxels (volumetric picture elements), ottenuti dalla partizione dello spazio calibrato in tanti piccoli cubetti: solo i voxels le cui proiezioni sui piani delle telecamere siano compresi nelle silhouettes faranno parte del visual hull. La risoluzione del visual hull dipende naturalmente dalla dimensione dei voxels [3].

Uno dei principali problemi legati all'uso dei visual hull è la possibile insorgenza, in alcuni frame, dei cosiddetti *phantom volumes*, volumi fantasma. I *phantom volumes* sono artefatti definiti come significative deviazioni locali rispetto alla vera superficie corporea del soggetto [30], che compaiono quando una porzione del volume di interesse è

occlusa alla vista delle telecamere. E' stato provato che un parametro critico nel controllo di questo problema è proprio il numero di telecamere impiegate [31]. Un esempio di ricostruzione di visual hull in cui compare un phantom volume è visibile in Figura 1.8.

A questo punto, si rende necessaria la definizione di un modello per il soggetto che guidi l'interpretazione dei dati ottenuti sperimentalmente. Il modello rappresenta infatti la conoscenza a priori sia sull'anatomia che sulla cinematica del soggetto e, con l'unica eccezione degli approcci model-free, il modello è in definitiva necessario per l'interpretazione dei dati sperimentali, per poter discriminare e riconoscere nel visual hull i diversi segmenti corporei, ridurre l'influenza degli artefatti nella ricostruzione del visual hull e ricavare i parametri del movimento, come gli angoli articolari. Occorre infatti ricordare che la nuvola di punti generata dalla ricostruzione del visual hull non contiene di per sé alcuna informazione riguardo ad esempio quali punti appartengono a una determinata regione corporea, e nemmeno contiene di per sé informazione sulla correlazione tra i punti del visual hull generati in un frame e quelli generati nei frame precedenti o successivi [3].



*Figura 1.8 Visual Hull affetto da phantom volume*

Nella realizzazione del modello, l'informazione anatomica può essere ottenuta tramite una scansione del soggetto utilizzando un laser scanner: il risultato della scansione è un oggetto tridimensionale rappresentante la superficie esterna del soggetto, descritta al calcolatore come mesh poligonale (o maglia poligonale). In generale,

una mesh poligonale è espressa da una lista ordinata di punti 3D rappresentanti i vertici dei poligoni e dalla lista di poligoni che indicano come i vertici sono connessi. Ogni vertice è unicamente identificato dalla sua posizione nella lista e definito dalle sue tre coordinate rispetto al sistema di riferimento, mentre ogni poligono è descritto dagli identificatori dei suoi vertici indicati in senso orario, guardando il poligono dall'esterno [3].

Le informazioni sulla cinematica possono essere rappresentate da un modello articolato del corpo umano. Il modello articolato per l'analisi del movimento di tutto il corpo consiste generalmente in 15 segmenti rigidi, modellanti i principali gruppi di ossa assimilabili a un unico segmento (testa, busto, bacino, braccia, avambracci, mani, cosce, gambe, piedi), connessi tra loro con articolazioni a 6 gradi di libertà (3 traslazioni e 3 rotazioni). Oltre a quello appena descritto possono essere utilizzati anche modelli diversi, più o meno articolati, a seconda delle necessità e degli obiettivi del caso. Ognuno dei segmenti corporei del modello ha un segmento padre (con l'unica eccezione della testa) e può avere più segmenti figli. Un sistema di riferimento tecnico viene a questo punto definito per ognuno dei segmenti, seguendo un criterio comune per cui il sistema di riferimento locale di ogni parte si trovi in una relazione nota con i sistemi di riferimento del segmento padre e dei segmenti figli. L'unica eccezione è costituita dal segmento testa, il cui sistema di riferimento è orientato come quello globale [3].

Così definito il sistema articolato forma una catena cinematica, e la posa dell'intero modello è univocamente identificabile noti posizione e orientamento di ogni segmento rispetto al segmento padre.

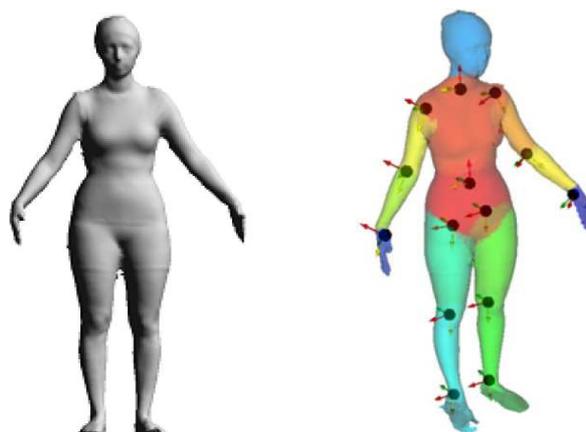
I vincoli imposti alle articolazioni sono inoltre parte del bagaglio di conoscenza a priori che può essere sfruttato e incorporato in questa fase della modellazione. Questo tipo di informazione permette di ammettere tra tutte le possibili configurazioni che andranno a fittare i dati sperimentali solo quelle anatomicamente sensate [3].

A questo punto del lavoro occorre generare il modello complessivo, specifico per il soggetto, che includa sia le informazioni anatomiche,

precedentemente ottenute con il laser scanner, che le informazioni cinematiche. In sostanza, ciò che occorre fare è adattare il modello a 15 segmenti rigidi alla mesh poligonale specificamente creata sul soggetto, identificando sulla mesh i 15 segmenti corporei e le posizioni dei centri articolari. Si riuscirà in questo modo nell'intento di creare un modello specifico per il soggetto e utilizzabile per l'interpretazione dei dati, che contenga al suo interno le informazioni di tipo sia morfologico che cinematico [3].

Recentemente, è stato sviluppato un algoritmo per la generazione automatica di un modello specifico per il soggetto a partire dalla sua rappresentazione superficiale come mesh poligonale [32], che permetta di includere direttamente le informazioni cinematiche nella mesh specifica del soggetto, ottenendo le posizioni dei centri articolari e la segmentazione della mesh nelle diverse parti del corpo. Il suddetto algoritmo può essere utilizzato con una qualsiasi forma umana, essendo stato sviluppato su un database di laser scan di persone, acquisite nella stessa posizione di riferimento [3].

Si mostra in Figura 1.9 un esempio di mesh triangolare di un soggetto acquisita con un laser scanner e il modello completo della segmentazione in 15 segmenti corporei, specifico per il soggetto dopo l'applicazione dell'algoritmo di generazione automatica del modello [32].



*Figura 1.9 Rappresentazione della superficie esterna del soggetto con mesh triangolare acquisita con laser scanner (a sinistra) e modello completo del soggetto dopo l'applicazione dell'algoritmo di generazione automatica del modello (a destra)*

E' stato provato che la procedura automatica [32] lavora bene anche quando si usa il visual hull come input invece che la mesh fornita dal laser scanner, che richiederebbe un costoso hardware dedicato allo scopo e non sempre disponibile. Nel caso in cui si voglia impiegare il visual hull come input, si richiede al soggetto di rimanere in piedi all'interno del volume di misura nella posizione di riferimento effettuando l'acquisizione per alcuni istanti. Il visual hull del soggetto in questa configurazione viene così utilizzato per generare il modello anatomico, che in questo caso risulta un po' più "abbondante" rispetto al soggetto vero, dato che il visual hull è un'approssimazione per eccesso del volume occupato dall'oggetto. Anche se il modello così creato è generalmente meno accurato rispetto all'utilizzo del laser scanner, può comunque rappresentare un input valido se il numero di telecamere utilizzate è sufficientemente elevato (in particolare in numero maggiore a 8 telecamere, possibilmente almeno 16) [3].

Una volta ottenuto il modello completo specifico per il soggetto, si è pronti per il suo utilizzo nell'interpretazione dei dati. Questo obiettivo viene raggiunto trovando, istante per istante, la configurazione spaziale (ossia la posa del modello) che meglio rappresenta i dati, tra tutte quelle permesse osservando i vincoli imposti. Ricordiamo che i dati sperimentali sono rappresentati in 3D dal visual hull. La giusta configurazione del modello può essere trovata facendo corrispondere ad ognuno dei punti del visual hull un punto del modello mesh, usando ad esempio l'algoritmo AICP [33] (Articulated Iterative Closest Point). Anche il matching nel senso opposto (modello sui dati) è tuttavia possibile: i due tipi di approccio differiscono solo nella loro robustezza rispetto ai dati corrotti [3]. Se nel primo caso (dati sul modello) l'algoritmo è meno robusto rispetto ai *phantom volumes* eventualmente presenti nel visual hull, nel secondo tipo di approccio le parti mancanti del visual hull (specialmente arti mancanti), dovuti per esempio a errori nella sottrazione del background, portano il modello ad assumere configurazioni completamente sbagliate [3].

L'approccio AICP [33] si basa sulla corrispondenza tra coppie di punti vicini su due mesh diverse: l'algoritmo agisce localmente, non considerando tutte le possibili combinazioni di punti corrispondenti sulle due mesh. Questo riduce sensibilmente il costo computazionale, al prezzo di un aumento della sensibilità rispetto al posizionamento iniziale del modello [3].

Fondamentalmente, alla fine il sistema si riduce ad essere un tipico problema ai minimi quadrati non lineare [3].

### **1.3.2 Vantaggi e problemi**

La markerless motion capture rappresenterebbe un'interessante potenziale via risolutiva nei confronti di molti di quei problemi che caratterizzavano i sistemi ottici marker-based. In primo luogo, con la rimozione di qualsiasi struttura applicata al corpo per l'acquisizione del movimento si elimina totalmente l'ingombro sul soggetto, permettendo l'acquisizione di movimenti naturali, liberi da costrizioni e eventualmente anche ampi, con l'utilizzo di opportuni setup. In secondo luogo, eliminando il bisogno di posizionare i marker, si riducono di molto anche i tempi di preparazione del paziente. Infine, la tecnica markerless presenta una potenziale applicabilità anche in contesti in cui l'applicazione di una tecnica marker-based sarebbe risultata più difficile: è possibile ad esempio impiegare la tecnica markerless per l'acquisizione del cammino in acqua.

Il principale limite presentato, ad oggi, rispetto all'impiego della tecnica markerless per applicazioni cliniche e biomeccaniche è ancora purtroppo la scarsa accuratezza dei metodi correnti [3]. L'accuratezza della markerless motion capture dipende tuttavia da molti fattori, tra cui la qualità della segmentazione iniziale, la risoluzione e soprattutto il numero di telecamere impiegate (parametro che influenza di molto la qualità della ricostruzione del visual hull) [34]: a seconda del dataset impiegato, infatti, la differenza tra la ricostruzione della posizione dei centri articolari con tecnica markerless e marker-based può passare da 7-8 cm a 1-2 cm, con l'algoritmo proposto da Corazza et al [35].

## **1.4 L'analisi del cammino in acqua con tecnica markerless**

Nel presente lavoro di tesi si fa riferimento a uno specifico uso della tecnica markerless: si impiega in particolare la tecnica nell'ambito dell'acquisizione del cammino in acqua. Nella seguente sezione si tenterà dunque di sottolineare l'importanza dello sviluppo di una tecnica che permetta di acquisire in modo accurato il movimento nella condizione subacquea, dati i numerosi benefici offerti dall'ambiente acquatico nell'ambito di un percorso riabilitativo.

### **1.4.1 Vantaggi offerti dall'ambiente acquatico sul processo riabilitativo**

In questa sezione si andranno a ricordare alcune proprietà fisiche dell'acqua (spinta idrostatica, viscosità e pressione idrostatica), evidenziandone gli effetti benefici nell'ambito di un eventuale percorso riabilitativo in seguito, ad esempio, a un infortunio o a un intervento chirurgico.

Dal principio di Archimede, ogni corpo immerso in un fluido riceve una spinta verticale dal basso verso l'alto uguale, in modulo, al peso della massa di fluido spostato. Ciò significa, nel caso di una persona immersa, riuscire a ridurre l'effetto della gravità sul corpo, scaricando progressivamente il peso dalle articolazioni immerse. La possibilità di decrementare il peso effettivo dell'individuo in proporzione al grado di immersione ha un'importante potenzialità terapeutica: dato infatti che il carico assiale sulla colonna vertebrale e il peso supportato dalle giunture, specialmente anca, ginocchio e caviglia si riduce aumentando il grado di immersione, monitorando le profondità alle quali vengono effettuati movimenti funzionali ed esercizi, l'effetto della gravità può essere progressivamente reintrodotta, promuovendo un rafforzamento graduale e controllato [36]. Grazie a questa proprietà, l'acqua permette di ritrovare un ambiente fisico a basso stress per le articolazioni, ideale quando le attività a pieno carico sono ancora premature, ma allo stesso tempo è auspicabile che il soggetto riacquisti mobilità. L'effetto della spinta idrostatica può essere inoltre sfruttato per portare le

articolazioni a lavorare con escursioni angolari più ampie che fuori dall'acqua. I vantaggi riguardanti la rapida ripresa della mobilità delle articolazioni sono ben documentati [37].

La viscosità è una grandezza fisica che quantifica la resistenza dei fluidi allo scorrimento, ossia la loro coesione interna. Quando un corpo si muove all'interno di un fluido, il corpo stesso è soggetto a una resistenza all'avanzamento (resistenza fluidodinamica) proporzionale alla dimensione e geometria del corpo, alla velocità di avanzamento e alla viscosità del fluido. Essendo l'acqua più viscosa dell'aria, la resistenza all'avanzamento offerta dall'ambiente acquatico è superiore rispetto all'avanzamento sulla terraferma, e l'effetto può risultare particolarmente utile a scopo riabilitativo per promuovere il rafforzamento dei muscoli. Inoltre, dato che la resistenza all'avanzamento nell'acqua aumenta all'aumentare della velocità di avanzamento e, quindi, allo sforzo esercitato dal soggetto, si riducono le possibilità di un re infortunio durante il processo riabilitativo [3].

Un'altra importante proprietà dell'acqua è la pressione idrostatica. Data infatti la legge di Stevino che afferma che in un fluido di densità costante  $\rho$ , la pressione esercitata da una colonna di fluido in un suo punto di profondità  $h$  (distanza dal pelo libero del fluido) è direttamente proporzionale a  $h$ , allora muovendosi dal pelo dell'acqua verso il fondo della piscina la pressione idrostatica aumenterà gradualmente. L'effetto di questo gradiente pressorio in un processo riabilitativo è la sua capacità di indurre la riduzione dei ristagni di liquido e la risoluzione degli edemi che normalmente si formano, ad esempio, dopo un intervento su un'articolazione. La pressione idrostatica è infine anche responsabile delle variazioni nella circolazione sanguigna osservate quando il corpo è immerso [3].

Camminare nell'acqua è considerata la principale terapia per pazienti con disturbi nella deambulazione, anche perché non richiede nessuna abilità particolare, come ad esempio nuotare. Come già spiegato, da un punto di vista meccanico i due principali vantaggi di una riabilitazione acquatica possono essere individuati nella riduzione del peso apparente sulle articolazioni, mentre allo stesso tempo aumenta la resistenza

all'avanzamento, che promuove un rafforzamento muscolare. Inoltre, i movimenti sono tipicamente eseguiti più lentamente rispetto all'ambiente esterno avendo dunque anche più tempo per controllarli. Anche se il cammino in acqua è stato riconosciuto di grande utilità sia in fase di allenamento che di riabilitazione, e nonostante il crescente interesse per l'analisi del cammino in acqua, ancora pochi studi sono stati condotti in questo senso. Comprendere invece come variano i parametri della locomozione a seconda del tipo di soggetto (sano o malato, età, grado di allenamento etc.), e quali siano esattamente gli effetti dell'ambiente acquatico sulla deambulazione, permetterebbe una prescrizione più adeguata del cammino in acqua come parte di programmi sia di allenamento che riabilitativi [3].

In questo senso, è dunque di grande importanza anche lo sviluppo e il perfezionamento delle tecniche che permettono l'acquisizione del movimento anche in un ambiente come quello acquatico. Tra queste tecniche, ci si occuperà in questo lavoro di tesi di una specifica fase dell'acquisizione della tecnica markerless.

## **1.5 Obiettivo della tesi**

Come già accennato nella sezione 1.3.1, la prima fase delle elaborazioni per l'acquisizione del movimento con tecnica markerless consiste solitamente nella segmentazione dell'oggetto di interesse rispetto al background, nella vista di ciascuna delle telecamere.

Nel presente lavoro di tesi, l'obiettivo sarà dunque lo sviluppo e la caratterizzazione di diversi algoritmi di segmentazione, nell'ambito della particolare applicazione del cammino in acqua. I principali particolari problemi presentati in questo tipo di applicazione sono innanzitutto, per quanto riguarda la segmentazione, la non completa staticità del background, a causa di riflessi e ombre a pelo dell'acqua che variano nel tempo, e la presenza del riflesso del soggetto sulla superficie inferiore del pelo dell'acqua che, avendo un colore molto simile al soggetto vero, viene facilmente erroneamente classificato. In questo lavoro di tesi si descriverà anche un algoritmo di miglioramento

della segmentazione, che ha il compito di riempire i piccoli “buchi” nella silhouette ed eliminare pixel spuri, erroneamente classificati come foreground. Per quanto riguarda problemi relativi alle fasi successive delle elaborazioni, occorrerà tenere conto del fatto che i visual hull generati da questi tipi di silhouettes mancano delle informazioni relative alla testa e alle braccia dei soggetti, con conseguente necessità di adottare, eventualmente, modelli opportuni per l’interpretazione dei dati.



## **2. Materiali e Metodi**

### **2.1 Materiale analizzato**

Il materiale impiegato per lo sviluppo e caratterizzazione degli algoritmi di segmentazione di cui al punto 1.4.2 consiste in una serie di video provenienti da acquisizioni del cammino in acqua alta di soggetti sani.

Le acquisizioni sono state effettuate chiedendo a quattro soggetti sani di camminare in una piscina con acqua alta 1.20 m, invitando i soggetti a tenere durante il movimento le braccia o fuori o al limite a pelo dall'acqua, in modo da non opporre ulteriore resistenza alla progressione. E' stato inoltre richiesto ai soggetti di scegliere autonomamente la velocità di avanzamento.

Il setup utilizzato per l'acquisizione del cammino subacqueo consiste in sei telecamere, ognuna delle quali con vista fissa sul fondo della piscina per tutta la durata delle acquisizioni. La disposizione delle telecamere prevedeva l'uso di una vista frontale (o posteriore, a seconda della direzione della camminata) e cinque laterali, tre da un lato e due dall'altro.

Sono state impiegate telecamere subacquee (TS-6021PSC, Tracer Technology Co. Ltd) a colori, digitali, con modello di colore RGB24 (24 bit codificano pixel, ossia 8 bit sono impiegati per codificare i livelli di colore di ognuna delle tre componenti), risoluzione 720x576 pixel e frequenza di campionamento 50 fps.

I video sui quali si è lavorato erano inoltre stati precedentemente elaborati con un algoritmo di correzione delle distorsioni, dovute sia alla visione subacquea che all'ottica delle telecamere.

Oltre alle riprese contenenti le acquisizioni del cammino dei soggetti, sono anche stati acquisiti i video contenenti unicamente le viste sul background delle sei telecamere.

## 2.2 Algoritmi sviluppati per la segmentazione del soggetto

Per il presente lavoro di tesi sono stati sviluppati e caratterizzati sette algoritmi per la segmentazione del soggetto, di cui nei paragrafi dal 2.2.1 al 2.2.7 si forniranno maggiori dettagli. Prima di descrivere gli algoritmi, si daranno inoltre nella sezione 2.2.0 informazioni preliminari riguardo la creazione delle immagini che modellano i background delle sei viste. Le immagini così create saranno infatti impiegate nell'ambito di alcuni dei successivi algoritmi.

Volendo introdurre una prima macro-classificazione, è possibile distinguere questi algoritmi innanzitutto a seconda dell'impiego o meno di un classificatore. In matematica e in apprendimento automatico, un classificatore è un oggetto in grado di eseguire una mappatura da uno spazio di attributi  $A$  ad un insieme di classi, o "etichette",  $C$ . Un classificatore può essere o prefissato o basato su apprendimento automatico, e in questo secondo caso l'apprendimento può essere supervisionato o non supervisionato, a seconda dell'utilizzo o meno di un "insieme di addestramento", in gergo *training set*.

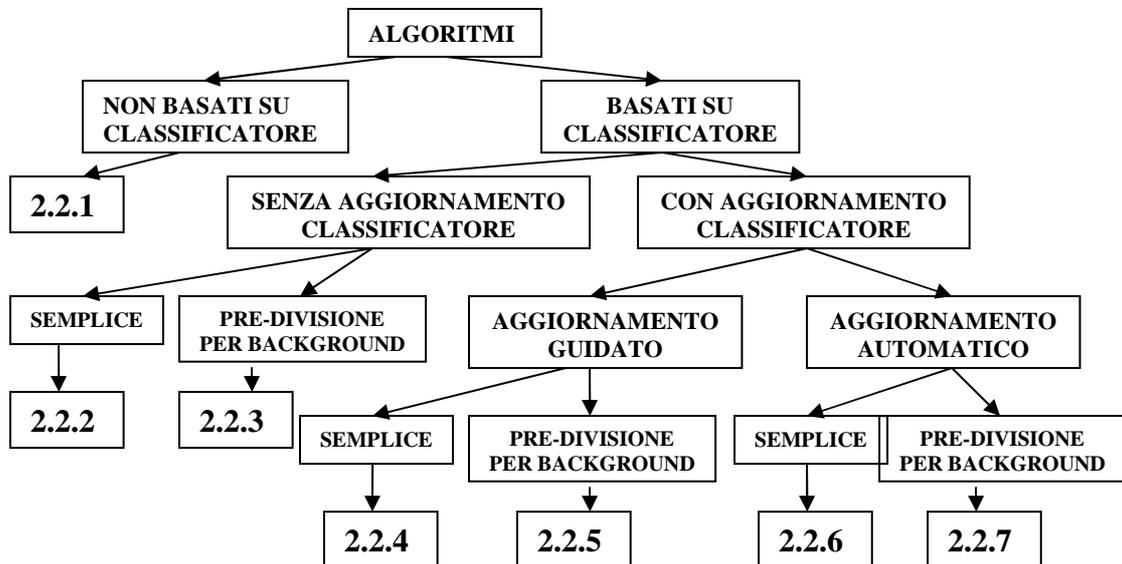


Figura 2.1 Schema ad albero riassuntivo delle caratteristiche degli algoritmi implementati

Gli algoritmi saranno inoltre distinti a seconda dell'introduzione o meno di un aggiornamento del classificatore fotogramma per fotogramma e, infine, a seconda che questi lavorino su immagini "non

trattate” oppure già preliminarmente divise per il background (allo scopo di aumentare il contrasto tra soggetto e sfondo).

Lo schema in Figura 2.1 ha lo scopo di riassumere visivamente quanto appena espresso.

### **2.2.0 Creazione di un modello per il background**

La creazione di un'immagine che modelli il background è una fase iniziale necessaria, dal momento che molti degli algoritmi che saranno descritti in seguito si basano su una preliminare divisione/sottrazione dello sfondo alle immagini delle acquisizioni.

Nel caso particolare in cui lo sfondo sia quello di una piscina occorre considerare, come già accennato, la non completa staticità del contesto a causa soprattutto della variabilità temporale delle ombre e dei riflessi presenti a pelo dell'acqua. Per attenuare l'effetto di questa variabilità, si è deciso di assumere come modello del background non un unico fotogramma scelto in modo casuale dalla sequenza video contenente le riprese del solo sfondo, ma piuttosto la media nel tempo di tutti i frame dell'acquisizione.

L'operazione di media temporale di tutti i fotogrammi produrrebbe accettabili risultati anche se applicata a sequenze video nelle quali compare anche il soggetto. Il motivo risiede nel fatto che il soggetto occupa determinate porzioni dell'immagine per un tempo molto più breve rispetto alla permanenza dello sfondo nella corrispondente posizione, pertanto il contributo del primo al colore medio nel tempo di un pixel non è in generale rilevante. L'unica eccezione, per cui la media temporale non produce buoni risultati, è rappresentata dalle telecamere con vista frontale o posteriore: in questi casi il soggetto permane nella posizione centrale dello schermo per quasi tutta l'acquisizione, per cui realizzando una media temporale di tutti i fotogrammi rimarrà sempre e comunque un alone corrispondente al soggetto al centro dell'immagine.

La media temporale dei fotogrammi viene fatta per ognuna delle sei telecamere, così da ottenere sei immagini che modellano gli sfondi delle acquisizioni (Figura 2.2).



*Figura 2.2 Immagini modellanti il background per ognuna delle sei telecamere, ottenute mediando nel tempo tutti i frame delle acquisizioni dei soli sfondi*

### **2.2.1 Sottrazione/divisione del background e Sogliatura**

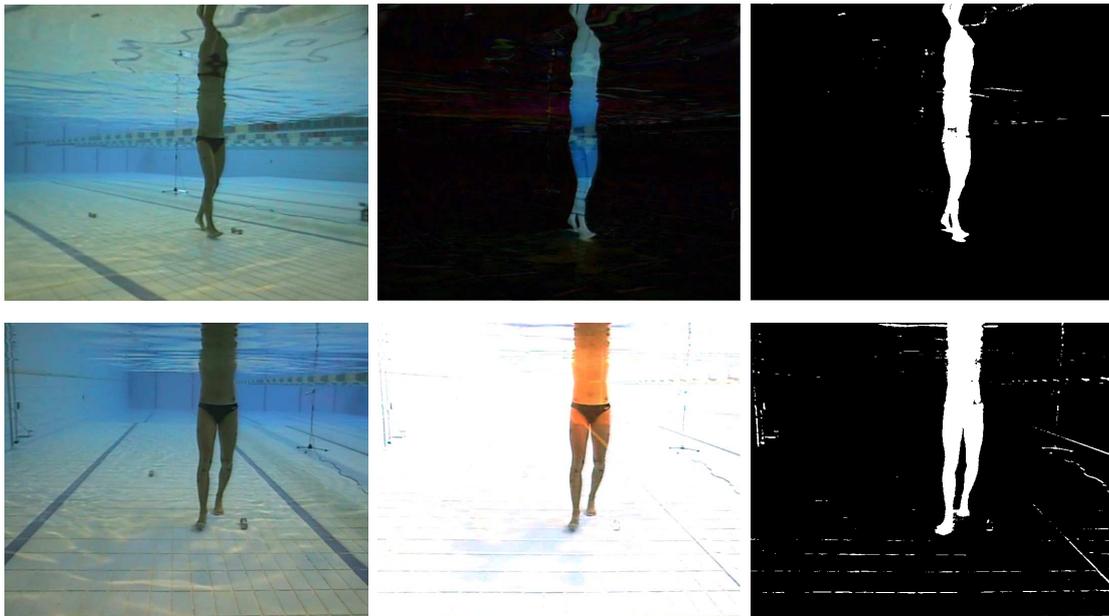
Questo algoritmo (che richiameremo in seguito con  $SD_B$ ) consiste nella semplice divisione o sottrazione dell'immagine modellante il background ad ognuno dei fotogrammi delle acquisizioni. Il motivo per cui si dividono o si sottraggono alle immagini i rispettivi background è cercare di ottenere un maggior contrasto tra soggetto e sfondo, dal momento che in generale la differenza/divisione tra due immagini sortisce l'effetto di un'accentuazione delle differenze tra le due. Una volta ottenuto più contrasto, l'algoritmo prevede una semplice operazione di sogliatura. La sogliatura è un processo per cui, scelto un determinato livello di grigio come soglia, tutti i pixel il cui livello di grigio sia inferiore alla soglia saranno posti al livello minimo (colore nero) mentre tutti i pixel con un livello di grigio superiore alla soglia saranno posti al livello massimo (colore bianco), o viceversa. Nel caso si lavori con immagini a colori, queste vengono prima convertite a immagini in scala di grigi, e successivamente sogliate.

Formalmente:

$$I_{binaria}(x, y) = \begin{cases} 0 & \text{se } I_{originale}(x, y) < \text{soglia} \\ 1 & \text{se } I_{originale}(x, y) \geq \text{soglia} \end{cases}$$

con x, y coordinate del pixel nell'immagine (indici di riga e di colonna dell'elemento in esame all'interno della matrice).

Per ognuna delle telecamere è stato empiricamente valutato se utilizzare la divisione o la sottrazione del background e quale valore di soglia scegliere, in base a una serie di prove preliminarmente effettuate.



*Figura 2.3 In alto: esempio di sottrazione del background (centro) all'immagine originale (sinistra) e sogliatura (destra). In basso: esempio di divisione per il background (centro), immagine originale (sinistra) e sogliatura (destra). Si noti che nel caso della divisione la sogliatura viene invertita.*

In queste prove si sono scelti tre fotogrammi per ognuna delle telecamere, corrispondenti a istanti diversi dell'acquisizione, effettuando per ognuno una sottrazione e una divisione del background. La soglia e il metodo (sottrazione o divisione) sono stati poi empiricamente determinati per ciascuna telecamera attraverso un'analisi qualitativa dei risultati che si ottenevano scegliendo diverse soglie (dove ogni nuova soglia veniva iterativamente proposta seguendo una sorta di "metodo di bisezione"), nella ricerca di una configurazione che permettesse di avere contemporaneamente un buon riconoscimento del foreground (silhouette "piena") ma relativamente poco rumore (pochi pixel spuri). La scelta della sottrazione o della

divisione e del valore di soglia utilizzata sono dunque dipendenti dalla telecamera, e non dal soggetto analizzato.

Una particolare attenzione, affinché le operazioni vadano a buon termine, va posta nel convertire tutte le immagini in doppia precisione prima di effettuare sia le sottrazioni che le divisioni, per non perdere l'informazione legata a eventuali decimali risultanti dalle operazioni. Inoltre, quando si sottraggono due immagini per accentuare le differenze, occorre ricordare che lo scopo è in realtà quello di evidenziare il valore assoluto della differenza, ossia la “distanza euclidea” tra i pixel nello spazio RGB; è pertanto importante ricordarsi di valutare il valore assoluto della differenza e non la semplice differenza, onde evitare che eventuali numeri negativi associati ai pixel come risultato della sottrazione vengano automaticamente appiattiti sul valore zero (colore nero). In Figura 2.3 sono mostrati esempi di sottrazione e divisione del background con successiva sogliaatura.

### 2.2.2 Naive Bayes Classifier

Il classificatore bayesiano è un particolare tipo di classificatore che sfrutta un approccio probabilistico ed eventualmente parametrico, ha un apprendimento supervisionato, e basa il suo funzionamento sul Teorema di Bayes.

In breve, a partire dalle caratteristiche dell'insieme di addestramento (training set) un classificatore bayesiano modella relazioni probabilistiche tra gli attributi e le etichette delle classi.

Per spiegare meglio il funzionamento di questi classificatori, ricordiamo brevemente il concetto di probabilità condizionata.

Si definisce probabilità condizionata di un evento A rispetto a un evento B,  $P(A|B)$ , la probabilità che si verifichi A, dato che si è già verificato l'evento B. Le probabilità condizionate  $P(A|B)$  e  $P(B|A)$  valgono rispettivamente:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad \text{e} \quad P(B|A) = \frac{P(A \cap B)}{P(A)}$$

mentre la formula di Bayes (Teorema di Bayes in forma elementare) indica semplicemente la relazione tra le probabilità condizionate di due eventi:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Definendo ora il vettore  $\mathbf{A}=(A_1,A_2,A_3,\dots,A_n)$  il vettore che descrive il set di attributi caratterizzanti gli oggetti da classificare e definendo  $C_i$  la variabile identificativa della classe, allora trattando  $\mathbf{A}$  e  $C_i$  come variabili aleatorie possiamo esprimere le relazioni probabilistiche tra  $C_i$  e  $\mathbf{A}$  attraverso la probabilità condizionata  $P(\mathbf{A}|C_i)$ . Sarà durante la fase di training che il classificatore apprenderà questo legame probabilistico,  $P(\mathbf{A}|C_i)$ .

Una volta noto  $P(\mathbf{A}|C_i)$ , un generico campione presentante l'insieme di attributi  $\mathbf{a}$  potrà essere classificato trovando la classe  $C_i=c$  che massimizza la probabilità a posteriori  $P(c|\mathbf{a})$ . Detto in altri termini, si assegnerà al campione con il set di attributi  $\mathbf{a}$  la classe  $C_i=c$  che massimizza la probabilità che il campione appartenga proprio a quella classe "c" tra tutte le classi definite.

Per calcolare la probabilità a posteriori  $P(C_i|\mathbf{A})$ , è sufficiente applicare la formula di Bayes, che esprime  $P(C_i|\mathbf{A})$  in termini di  $P(\mathbf{A}|C_i)$ ,  $P(C_i)$  e di  $P(\mathbf{A})$ :

$$P(C_i | \bar{\mathbf{A}}) = \frac{P(\bar{\mathbf{A}} | C_i) \cdot P(C_i)}{P(\bar{\mathbf{A}})}$$

ossia, in forma estesa

$$P(C_i | A_1 \cap A_2 \cap \dots \cap A_n) = \frac{P(A_1 \cap A_2 \cap \dots \cap A_n | C_i) \cdot P(C_i)}{P(A_1 \cap A_2 \cap \dots \cap A_n)}$$

Dato tuttavia che  $P(\mathbf{A})$  è indipendente dal tipo di classe considerata, massimizzare la probabilità a posteriori  $P(C_i|\mathbf{A})$  equivarrà a massimizzare il prodotto

$$\boxed{P(\bar{\mathbf{A}} | C_i) \cdot P(C_i)} \quad \text{ossia} \quad \boxed{P(A_1 \cap A_2 \cap \dots \cap A_n | C) \cdot P(C_i)}$$

rispetto alla classe.

Ora, i classificatori detti “naïve” (letteralmente tradotto “ingenui”), come il Naive Bayes Classifier, assumono come ipotesi di base che tutti gli attributi di  $\mathbf{A}$  siano statisticamente indipendenti.

Ricordando brevemente il significato di indipendenza statistica tra due eventi  $A$  e  $B$ :

$$P(A | B) = P(A) \quad \text{e} \quad P(B | A) = P(B)$$

o in maniera equivalente, per la formula di Bayes:

$$P(A \cap B) = P(A) \cdot P(B)$$

che significa che il fatto che si sia già verificato l’evento  $A$  (o in modo equivalente  $B$ ) non influenza la probabilità assoluta  $P(B)$  che  $B$  si verifichi (o che rispettivamente la probabilità  $P(A)$  che si verifichi  $A$ ). Data l’indipendenza statistica tra gli attributi  $A_j$ , fissata la classe  $C_i$ , è possibile allora di scrivere:

$$P(A_1 \cap A_2 \cap \dots \cap A_n | C_j) = P(A_1 | C_j) \cdot P(A_2 | C_j) \cdot \dots \cdot P(A_n | C_j)$$

e dunque massimizzare la probabilità a posteriori  $P(C_i | \mathbf{A})$  nel caso di indipendenza statistica degli attributi significherà, in questo caso, trovare la classe  $C_{i^*}$  tale per cui:

$$i^* = \arg \max_i P(C_i) \cdot \prod_{j=1}^n P(A_j | C_i)$$

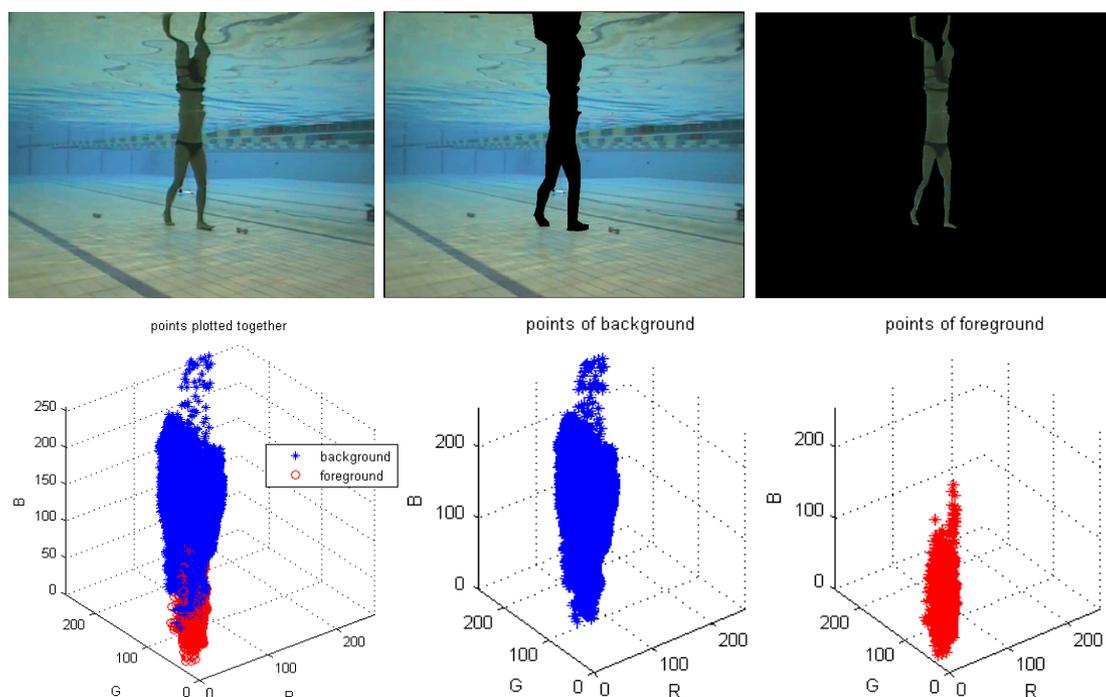
Per riuscire nello scopo, sarà necessario e sufficiente caratterizzare  $P(A_j | C_i)$  per ogni attributo  $A_j$  e per ogni classe  $C_i$ .

Nell’ambito dell’algoritmo descritto (che in seguito sarà richiamato come NBC) si sono definite due classi, 'background' e 'foreground', mentre gli attributi sui quali si basa la classificazione sono tre, ossia i tre valori delle componenti di colore rosso, verde e blu presenti in ogni pixel. Occorrerà pertanto nel nostro caso valutare sei distribuzioni di probabilità, cioè per ognuna delle due classi valutare le distribuzioni dei tre livelli di colore rosso, verde e blu, determinandole indipendentemente, essendo il classificatore “naïve”.

In questo algoritmo l'approccio è di tipo parametrico, e il modello adottato per descrivere la distribuzione di probabilità di ogni componente di colore nei pixel è la normale gaussiana. Pertanto, i parametri che sarà necessario valutare nella fase di training saranno la media  $\mu$  e la deviazione standard  $\sigma$  dei livelli quel colore in tutti i punti assegnati alla classe nell'insieme di addestramento (training set). La funzione densità di probabilità di ogni generica componente di colore  $x$  nella classe  $C_i$ , calcolati  $\mu$  e  $\sigma$  di quella componente nella classe risulterà dunque essere del tipo:

$$p(x | C_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Anche le probabilità assolute delle classi  $P(C_i)$ , dette anche *Prior*, sono facilmente ottenibili in fase di training. In particolare, si assume come probabilità assoluta di ogni classe il rapporto tra il numero di pixel appartenenti a quella classe e il numero totale di pixel presenti nel training set.



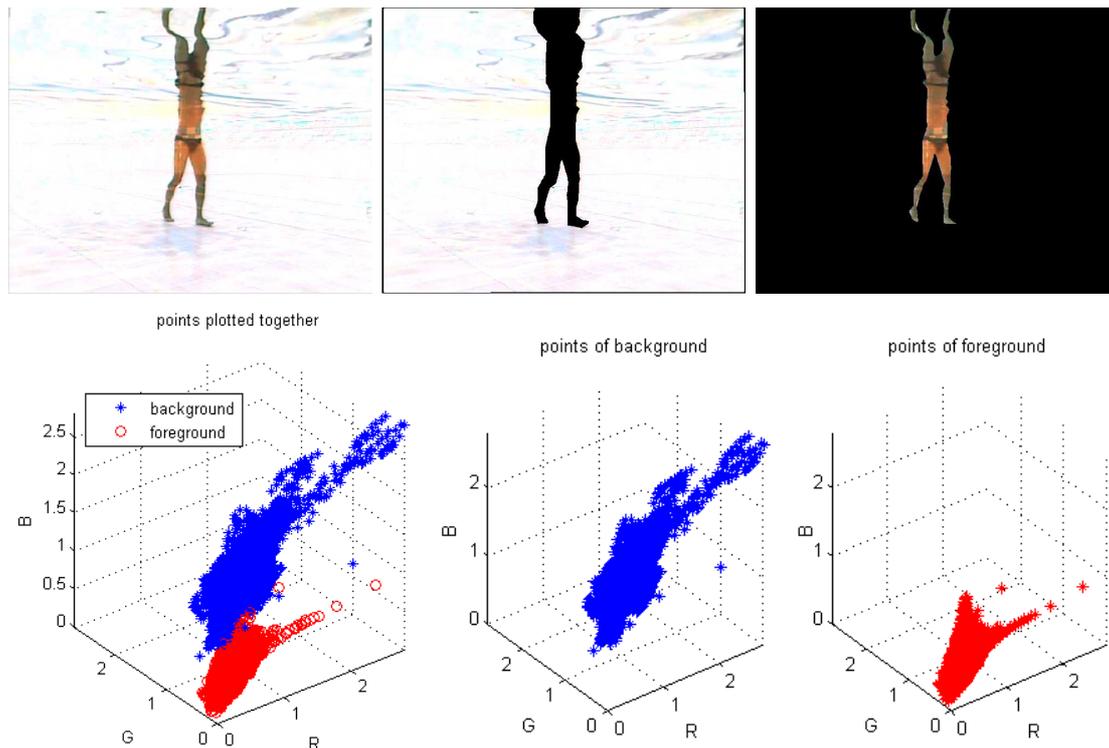
*Figura 2.4 In alto: Immagine scelta per la fase di training (sinistra), insieme di addestramento per la classe background (centro) e per la classe foreground (destra) In basso: nuvole di punti relative a background (blu) e foreground (rosso) nello spazio RGB*

Una volta noti i parametri  $\mu$  e  $\sigma$  per ogni componente di colore in ogni classe e le *Prior* delle classi  $P(\text{'background'})$  e  $P(\text{'foreground'})$ , si hanno tutti gli elementi a disposizione per la classificazione dei pixel come 'background' o come 'foreground', e dunque per la segmentazione delle immagini.

In Figura 2.4 è mostrato un esempio di immagine utilizzata in fase di training, gli insiemi di addestramento per background e foreground, e una visualizzazione delle nuvole di punti associate alle due classi nello spazio RGB.

### **2.2.3 Naive Bayes Classifier, applicato a immagini già divise per il background**

Questo algoritmo (che in seguito richiameremo con  $NBC_B$ ) sfrutta lo stesso tipo di classificatore descritto al punto 2.2.2. Tuttavia, invece di applicare il classificatore alle immagini non trattate, si decide di impiegarlo su immagini preventivamente divise per il background. Lo scopo della preliminare divisione per il background è quello di ottenere un maggior contrasto tra soggetto e sfondo, ossia di far sì che le nuvole di punti relative a background e foreground nello spazio RGB siano più distanziate. In questo modo, anche le gaussiane che descrivono le distribuzioni di probabilità di colore dovrebbero essere più "distanziate", il che significa teoricamente ridurre il rischio di un'errata classificazione dei pixel. Naturalmente, anche la fase di training andrà effettuata su un'immagine preliminarmente divisa per il background. In questo caso, essendo coinvolta una divisione tra immagini, oltre a utilizzare la doppia precisione occorre fare attenzione che nessuna componente di colore nei pixel dell'insieme di addestramento vada all'infinito per effetto della divisione per un valore nullo. Nel caso fossero presenti pixel con livelli di colore infiniti, infatti, calcolando  $\mu$  e  $\sigma$  anche questi parametri andrebbero all'infinito, rendendo il classificatore inutilizzabile. Per ovviare a questo problema, l'algoritmo prevede una correzione della matrice rappresentante l'immagine di training sostituendo a tutti i valori infiniti della matrice un valore finito comune, scelto pari a uno.



*Figura 2.5 In alto: immagine scelta per la fase di training (sinistra), insieme di addestramento per la classe background (centro) e per la classe foreground (destra) In basso: nuvole di punti relative a background (blu) e foreground (rosso) nello spazio RGB*

In Figura 2.5 un esempio di immagine utilizzata in fase di training, gli insiemi di addestramento delle due classi e le nuvole di punti relative a background e foreground nello spazio RGB.

#### **2.2.4 Naive Bayes Classifier, con aggiornamento guidato dei parametri del modello assumendo andamenti lineari dei parametri**

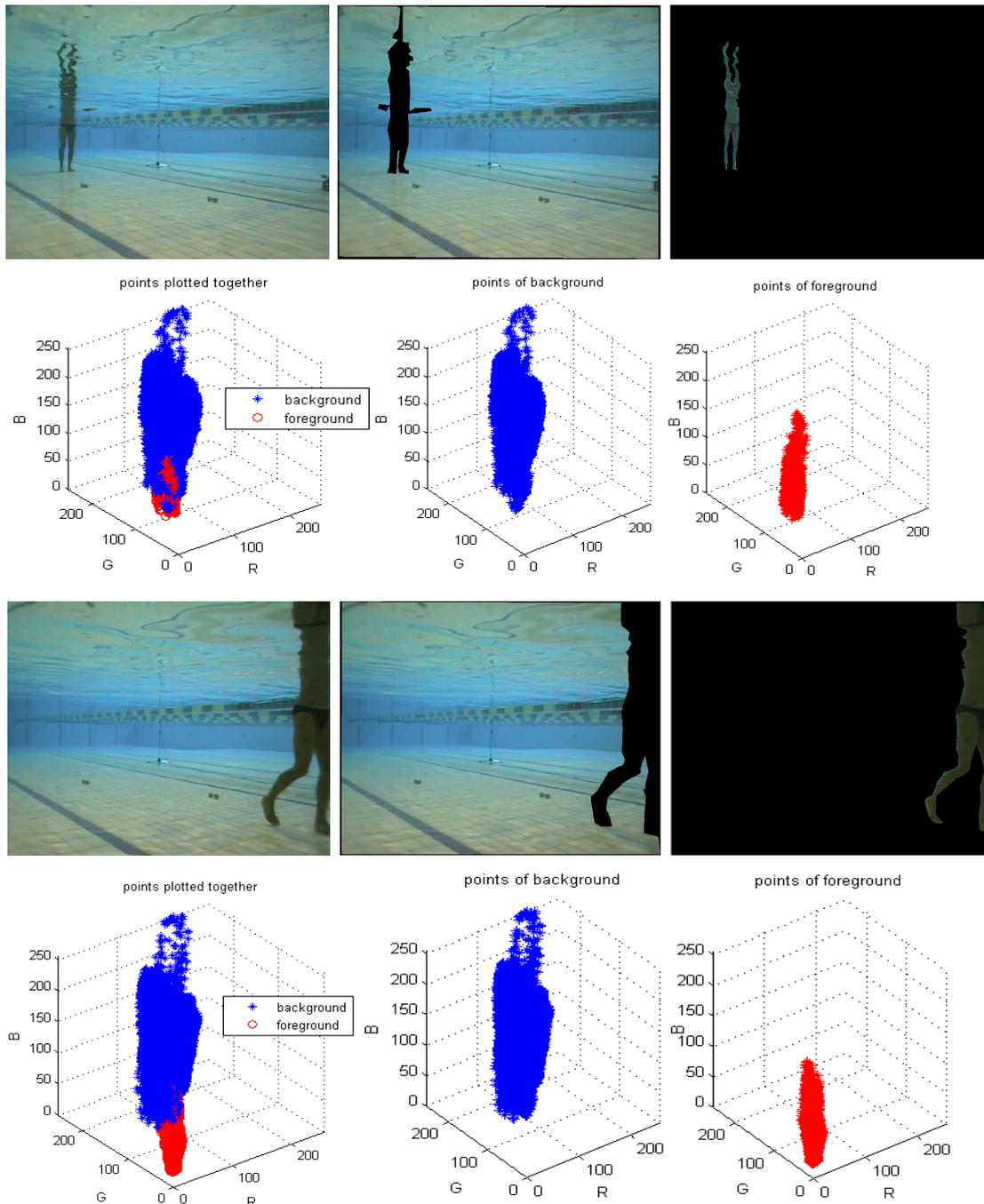
Questo algoritmo (che in seguito richiameremo con NBC<sub>AG</sub>) sfrutta ancora il Naive Bayes Classifier, descritto al punto 2.2.2, introducendo in più l'aggiornamento dei parametri del classificatore man mano che si procede con la segmentazione. Il motivo per cui si decide di inserire un aggiornamento deriva dall'osservazione per cui, eccezion fatta per le visioni puramente laterali, le caratteristiche del soggetto nelle immagini cambiano col procedere delle sequenze, sia in termini di dimensioni che di colore. In particolare, quando il soggetto è lontano dalla telecamera risulta più "piccolo", ed essendoci più acqua tra questo e il piano della telecamera il colore del soggetto è più tendente

al blu. Viceversa, quando il soggetto è più vicino alla telecamera, lo stesso appare più grande e c'è un maggior contrasto tra sfondo e soggetto. Ciò significa, in termini di parametri per il classificatore, che la media  $\mu$  della componente blu della classe 'foreground' sarà maggiore quando il soggetto è lontano e diminuirà man mano che il soggetto si avvicina, e viceversa. Inoltre, la probabilità assoluta (*Prior*) della classe 'foreground' sarà minore quando il soggetto è lontano per aumentare man mano che il soggetto si avvicina alla telecamera. Al contrario, per quanto riguarda la *Prior* della classe 'background' ci si può aspettare una leggera diminuzione man mano che il soggetto si avvicina, in virtù del fatto che più spazio nella scena è occupato dal soggetto, e quindi meno ne rimane per lo sfondo.

In questo algoritmo si ipotizza un andamento lineare per l'aggiornamento dei parametri, sia per le sei medie e deviazioni standard  $\mu$  e  $\sigma$  che per le due *Prior*. Si utilizzano pertanto due immagini di training, una verso l'inizio e una verso la fine dell'acquisizione, per poter valutare i coefficienti delle rette interpolanti e, dunque, l'incremento (o il decremento) che deve subire ogni parametro frame dopo frame. In particolare, dati  $nf1$  e  $nf2$  i numeri di fotogramma della prima e seconda immagine di training e  $x1$  e  $x2$  i valori del generico parametro  $x$  nella prima e nella seconda immagine, l'incremento  $\Delta x$  che il parametro subisce per ogni frame sarà:

$$\Delta x = \frac{x_2 - x_1}{nf_2 - nf_1}$$

In Figura 2.6 esempi di immagini di training all'inizio e alla fine dell'acquisizione, relativi insiemi di addestramento per le classi background e foreground e visualizzazione delle nuvole di punti associate alle classi. Si noti come le nuvole di punti si modificano dalla prima alla seconda immagine, notando in particolare come la nuvola di punti associata al foreground nella seconda immagine, quando il soggetto è più vicino, si sposta verso valori inferiori della componente di blu.



*Figura 2.6 Con la stessa organizzazione della Figura 2.4 e Figura 2.5, esempio riportante due immagini di training, una all'inizio e una alla fine dell'acquisizione*

### **2.2.5 Naive Bayes Classifier, con aggiornamento guidato dei parametri del modello assumendo andamenti lineari dei parametri, applicato a immagini già divise per il background**

Questo algoritmo (che in seguito richiameremo con  $NBC_{AG,B}$ ) prevede l'aggiornamento dei parametri del classificatore, man mano che la

segmentazione procede, con le stesse modalità descritte per il precedente algoritmo, nella sezione 2.2.4. L'unica differenza consiste nel lavorare con immagini preliminarmente divise per il background, sia per le fasi di training che di classificazione.

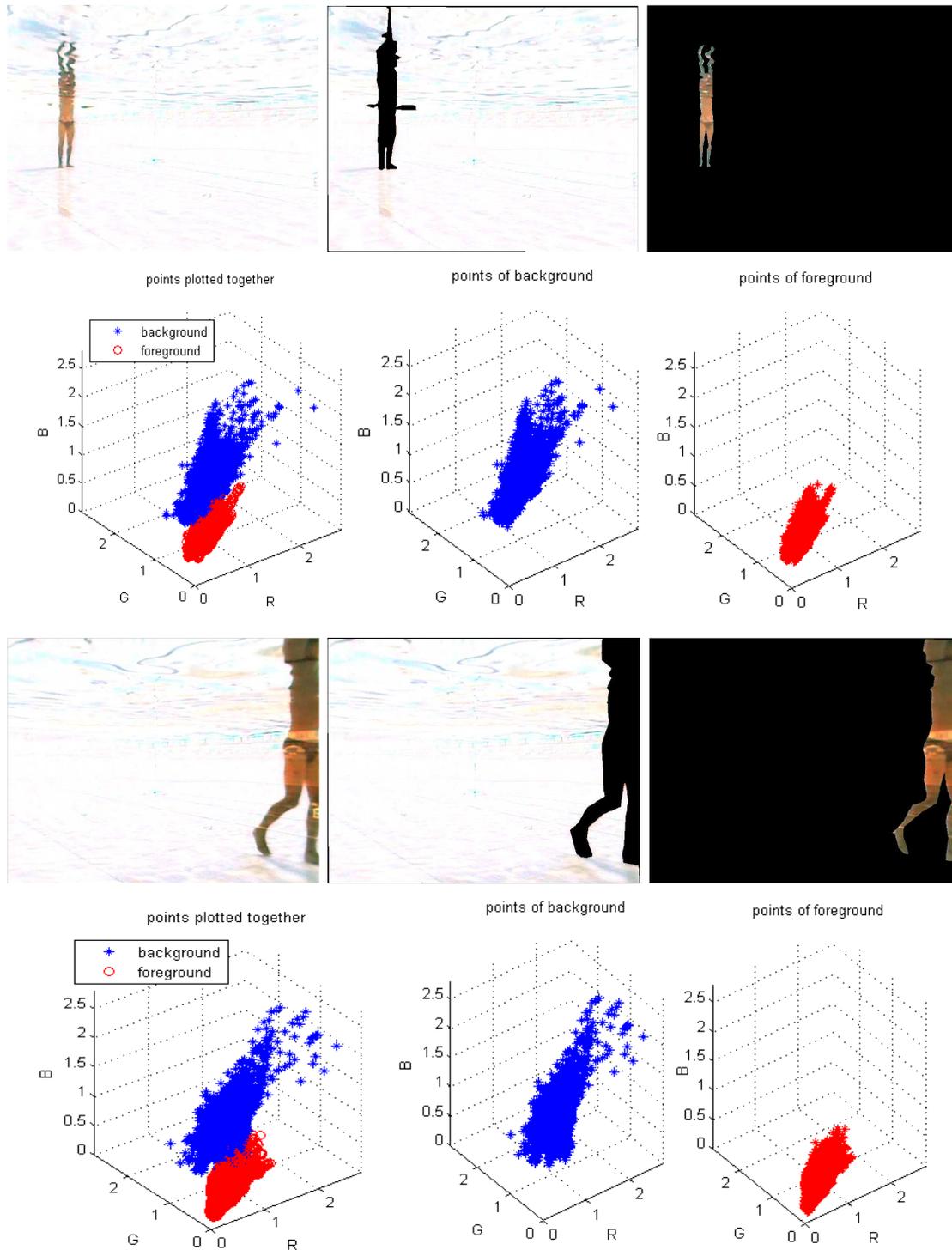


Figura 2.7 Stessa organizzazione della Figura 2.6 ma esempi di immagini di training per l'utilizzo dell'algoritmo 2.2.5

Per quanto riguarda le fasi di training, vale lo stesso accorgimento adottato nell'algoritmo 2.2.3, per cui si sostituiscono eventuali valori infiniti dei livelli di colore negli insiemi di addestramento con elementi di un valore finito prefissato, e scelto pari a uno.

Si mostrano in Figura 2.7 esempi di immagini di training all'inizio e alla fine dell'acquisizione, relativi insiemi di addestramento per le classi background e foreground e visualizzazione delle nuvole di punti associate alle classi.

### **2.2.6 Naive Bayes Classifier, con update automatico del modello ad ogni frame analizzato**

Nelle sezioni 2.2.2 e 2.2.3 si sono descritti gli algoritmi che impiegano il Naive Bayes Classifier, utilizzando un unico classificatore per tutti i frame. Negli algoritmi 2.2.4 e 2.2.5 si è introdotta l'idea di un'evoluzione del classificatore durante la segmentazione, attraverso un aggiornamento guidato dei suoi parametri con andamento lineare. In questo algoritmo (che in seguito richiameremo con  $NBC_{AA}$ ) si impiega ancora il Naive Bayes Classifier e si prosegue con l'idea di un aggiornamento del classificatore frame dopo frame, ma invece di guidare l'aggiornamento dei parametri si implementa l'algoritmo in modo che l'aggiornamento del modello avvenga automaticamente.

In questo algoritmo, ogni volta che il classificatore segmenta un'immagine, i pixel appena classificati costituiscono a loro volta gli insiemi di addestramento del classificatore per la segmentazione del fotogramma successivo, e così via. Per questo algoritmo basterebbe dunque una sola immagine per la fase di training del primo classificatore, scelta verso l'inizio della sequenza. Volendo però riutilizzare i dati relativi alle immagini di training di cui all'algoritmo 2.2.4 e per poter fare un confronto con il suddetto algoritmo, si riutilizzano anche in questo caso gli stessi insiemi di addestramento già definiti per l'algoritmo 2.2.4. Saranno dunque presenti anche in questo caso le due immagini di training, una all'inizio e una alla fine dell'acquisizione, e l'aggiornamento automatico del classificatore avverrà solo per tutti i fotogrammi compresi tra il primo e il secondo

training set. Si riutilizzano dunque gli stessi dati di prima, di cui un esempio è visibile in Figura 2.6.

### **2.2.7 Naive Bayes Classifier, con update automatico del modello ad ogni frame analizzato, applicato a immagini già divise per il background**

Questo algoritmo (che in seguito richiameremo con  $NBC_{AA,B}$ ) è l'analogo del precedente, descritto al punto 2.2.6, con la differenza che in questo caso si lavora con immagini già preliminarmente divise per il background. Anche in questo caso si riutilizzano i dati relativi alle due immagini di training, questa volta sfruttando i due insiemi di addestramento creati per l'algoritmo 2.2.5. Anche qui dunque si riutilizzano gli stessi dati di prima, di cui un esempio è visibile in Figura 2.7.

## **2.3 Algoritmi per il miglioramento della segmentazione**

Oltre ai sette algoritmi di segmentazione descritti, è stato impiegato anche un algoritmo per il miglioramento della segmentazione. Scopo di questo algoritmo è riempire piccoli “buchi” eventualmente rimasti nella silhouette ed eliminare i pixel spuri, erroneamente classificati come foreground in fase di classificazione. Nella seguente sezione, 2.3.1, si descriverà il funzionamento dell'algoritmo di miglioramento della segmentazione impiegato in questo lavoro di tesi.

### **2.3.1 L'algoritmo “regionGrowing”**

L'algoritmo *regionGrowing* è l'algoritmo di miglioramento della segmentazione impiegato per la maggior parte dei fotogrammi. Questo algoritmo prende in ingresso l'immagine segmentata che si intende migliorare e le due coordinate di un “punto iniziale”, espresse come indici di riga e di colonna di un particolare elemento della matrice associata all'immagine. Affinché l'algoritmo funzioni, questo “punto iniziale” deve essere scelto in modo da ricadere all'interno della silhouette, ossia dell'area bianca rappresentante la sagoma del

soggetto. Il funzionamento dell'algoritmo prevede che, partendo dal punto iniziale così indicato, si ricostruiscano progressivamente tutti i pixel associati alle parti bianche contigue, con un determinato grado di tolleranza circa la presenza di eventuali piccoli buchi; al termine dell'operazione, dell'immagine originale rimane dunque solo la regione bianca connessa circostante il punto indicato come punto iniziale. Come ultimo passaggio, l'algoritmo applica all'immagine così creata il filtro morfologico di chiusura [38].

Un esempio di immagine segmentata pre- e post-*regionGrowing* è visibile in Figura 2.8.

Essendo *regionGrowing* l'algoritmo di miglioramento della segmentazione impiegato per la maggior parte dei fotogrammi, è stato necessario implementare anche un metodo per il riconoscimento automatico di un punto iniziale che avesse le caratteristiche richieste.



*Figura 2.8 Esempio di immagine segmentata pre- (sinistra) e post-  
regionGrowing (destra)*

In particolare, il procedimento implementato sfrutta una funzione che, presa in ingresso l'immagine di cui si vuole migliorare la segmentazione (invertita con un 'not' logico), restituisce in uscita una matrice in cui a ogni pixel è associato il valore della distanza euclidea dal punto non nullo più vicino (ossia la distanza di ogni pixel dal più vicino pixel nero, essendo l'immagine invertita). Come conseguenza, nella matrice risultante il valore più elevato sarà associato al punto in assoluto più distante dai pixel neri, ossia il punto più interno della regione connessa bianca più estesa, che presumibilmente è proprio la

silhouette. Si prende dunque il punto così trovato come punto iniziale per la funzione *regionGrowing*.

Data la particolare implementazione della funzione *regionGrowing*, un'ulteriore condizione che il punto iniziale deve rispettare è l'essere lontano dai bordi dell'immagine per almeno due pixel. Non essendo scontato il rispetto di questa condizione, dato che il punto iniziale viene riconosciuto in modo automatico, si è messa a punto anche una procedura alternativa al *regionGrowing*. In particolare, si è implementato un metodo che come primo passaggio etichetta con una numerazione crescente tutte le regioni connesse presenti nell'immagine da migliorare. A questo punto, l'algoritmo elimina dall'immagine tutte le regioni connesse bianche eccetto la regione connessa bianca più estesa (che presumibilmente è la silhouette) e, con una serie di semplici passaggi logici, riempie i piccoli buchi neri presenti nella regione bianca aventi un'area inferiore a una soglia prefissata. In base ad alcune prove preliminarmente effettuate, l'area di soglia è stata scelta pari a 85 pixel.

## **2.4 Metodi di valutazione degli algoritmi**

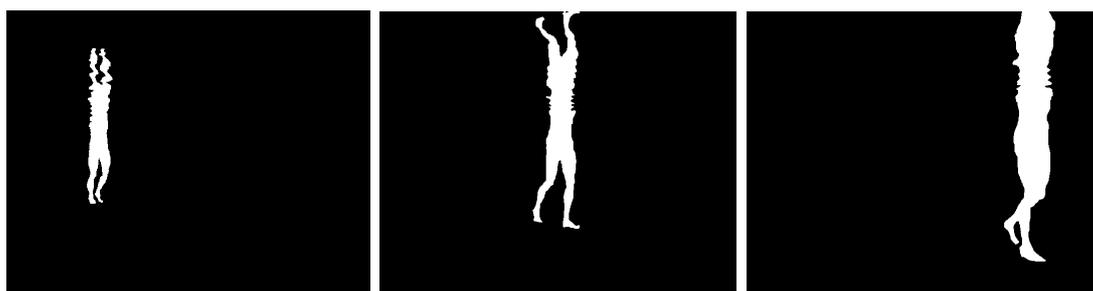
Nella seguente sezione si descriveranno i metodi che si sono impiegati per la caratterizzazione degli algoritmi di segmentazione, indicando da un lato i gold-standard utilizzati e dall'altro i parametri che si sono presi in considerazione per la valutazione della qualità degli algoritmi.

### **2.4.1 Gold-Standard e parametri per la valutazione**

Nel valutare la capacità di un algoritmo o in generale di un metodo di risolvere un determinato problema, occorre definire in principio quale sia il *gold-standard*, ossia quale sia quella soluzione che si assume come corretta o comunque la migliore ottenibile, e con la quale si confronteranno le risposte fornite dall'algoritmo che si intende caratterizzare.

In questo lavoro di tesi, i gold-standard utilizzati per la caratterizzazione degli algoritmi di segmentazione sono rappresentati

da immagini binarie segmentate a mano dall'operatore. In particolare, sono state definite 18 immagini gold-standard per ognuna delle acquisizioni, equivalenti a 3 per ogni telecamera. Le immagini per cui si è creato il gold-standard in ognuna delle telecamere sono state scelte cercando di selezionare la prima immagine verso l'inizio dell'acquisizione, la seconda circa a metà dell'acquisizione e la terza verso la fine. In questo modo, la qualità della segmentazione di un determinato algoritmo può essere valutata a partire da un numero maggiore di dati, e a partire da un set di prove che prende in considerazione una casistica più ampia di condizioni di segmentazione, permettendo una valutazione complessivamente più affidabile. In particolare, ogni valutazione finale potrà essere fatta utilizzando i risultati ottenuti su 72 immagini (4 soggetti x 6 telecamere x 3 gold-standard per telecamera).



*Figura 2.9 Esempio: i tre gold-standard definiti per l'acquisizione del Soggetto 1, telecamera 1*

In Figura 2.9 si riporta l'esempio dei tre gold-standard definiti per il soggetto 1, telecamera 1. Si noti come in queste immagini si sia deciso di mantenere all'interno della silhouette anche il riflesso ribaltato del soggetto a pelo d'acqua. Questa scelta è stata fatta poiché non sarebbe stato sensato, per come sono implementati gli algoritmi e dato che i colori del soggetto vero e del riflesso sono molto simili, chiedere che gli algoritmi riuscissero a distinguere le due parti. Se si fossero eliminati i riflessi nella definizione dei gold-standard, si sarebbero ottenuti per le segmentazioni risultati "forzatamente peggiorati", che non avrebbero rispecchiato la reale capacità di segmentazione degli algoritmi né permesso un corretto confronto tra i metodi.

Si rimanda pertanto il problema dell'eliminazione del riflesso a passi successivi dell'elaborazione delle immagini, solo una volta che siano state ottenute le prime segmentazioni.

Per quanto riguarda i parametri che saranno impiegati per la caratterizzazione degli algoritmi, questi saranno principalmente:

- TB, numero di *true background*
- FB, numero di *false background*
- FF, numero di *false foreground*
- TF, numero di *true foreground*

da cui si ricavano:

- percentuale di pixel del background riconosciuti correttamente
- percentuale di pixel del foreground riconosciuti correttamente

Il TB, ossia il numero di *true background* è il numero di pixel riconosciuti come background sia dal classificatore che nel gold-standard.

Il FB, ossia il numero di *false background* è il numero di pixel riconosciuti come background dal classificatore, ma come foreground nel gold-standard.

Il FF, ossia il numero di *false foreground* è il numero di pixel riconosciuti come foreground dal classificatore, ma come background nel gold-standard.

Il TF, ossia il numero di *true foreground* è il numero di pixel riconosciuti come foreground sia dal classificatore che nel gold-standard.

		GOLD STANDARD	
		Back	Fore
CLASSIFICATORE	Back	TB	FB
	Fore	FF	TF
		Nback	Nfore

Tabella 2.1 Schema riassuntivo dei 4 parametri TB, FB, FF, TF

Da questi quattro parametri segue che:

- **Nback**, il numero totale di pixel del background nel gold-standard è pari a  $TB+FF$

- **Nfore**, il numero totale di pixel del foreground nel gold-standard è pari a  $TF+FB$

Da cui segue facilmente che la percentuale di pixel del background riconosciuti correttamente è esprimibile come:

$$\%background\_corretti = \frac{TB}{Nback} \cdot 100 = \frac{TB}{TB + FF} \cdot 100$$

mentre la percentuale di pixel del foreground riconosciuti correttamente è:

$$\%foreground\_corretti = \frac{TF}{Nfore} \cdot 100 = \frac{TF}{TF + FB} \cdot 100$$

Per ognuno degli algoritmi saranno in particolare presi in considerazione i parametri  $\%background\_corretti$  e  $\%foreground\_corretti$ , valutati prima e dopo l'impiego della funzione *regionGrowing*. Essendo poi 72 i gold-standard complessivamente a disposizione, ognuno dei due parametri avrà sulle diverse prove una sua distribuzione, la cui descrizione permette di valutare la qualità dell'algoritmo stesso.

Per descrivere le distribuzioni si impiegheranno dunque due tipi di indice:

- la mediana "m", come indice di posizione delle distribuzioni
- due valori, che chiameremo LB e UB (lower bound e upper bound), come indici di dispersione della distribuzione

dove definiamo nel nostro caso gli indici LB e UB come quei valori per cui:

$$P(X < LB) = P(X > UB) = 0.05 = 5\%$$

ossia

$$P(LB \leq X \leq UB) = 0.9 = 90\%$$

con X variabile aleatoria associata al parametro  $\%background\_corretti$  (o a  $\%foreground\_corretti$ ).

pre-regionGrowing:			post-regionGrowing:		
<b>%background_corretti</b>			<b>%background_corretti</b>		
m=	%	UB= %	m=	%	UB= %
		LB= %			LB= %
<b>%foreground_corretti</b>			<b>%foreground_corretti</b>		
m=	%	UB= %	m=	%	UB= %
		LB= %			LB= %

Tabella 2.2 Formattazione della tabella con la quale si visualizzeranno i risultati sintetici

Escludere il 5% dei dati esterni in entrambi i lati significa, essendo 72 le prove a disposizione, escludere 4 dati (approssimazione di 3.6) sia a un estremo che all'altro; si sono dunque presi come limiti LB e UB ogni volta rispettivamente il quinto e il quintultimo dei 72 risultati, dopo averli ordinati in modo crescente.

Per riportare i valori di mediana, LB e UB delle distribuzioni dei due parametri, nel capitolo dei risultati i dati saranno organizzati visivamente come in Tabella 2.2.

Oltre ai risultati sintetici, per ognuno degli algoritmi si mostreranno anche gli istogrammi associati alle distribuzioni dei parametri *%background\_corretti* e *%foreground\_corretti* nelle 72 immagini di prova.

Infine, per permettere un immediato confronto visivo tra le distribuzioni, saranno aggiunti dei box-plot. In generale, un box-plot è un diagramma costituito da una serie di rettangoli affiancati (le "scatole"), dove i limiti di ogni scatola sono definiti dai valori del primo e dal terzo quartile (i valori al di sotto dei quali si trovano, rispettivamente, il 25% e il 75% delle osservazioni) di quella distribuzione, e ogni rettangolo è tagliato da una linea interna orizzontale, rappresentante la mediana. Agli estremi dei rettangoli si estendono inoltre dei "baffi" (*whisker*) che raggiungono, insieme agli *outlier* (valori distanti dal resto della distribuzione) l'intera estensione della distribuzione. In particolare, nei box-plot presentati si

considereranno come *outlier* tutti quei valori per cui, definiti  $q_1$  e  $q_3$  il primo e il terzo quartile della distribuzione:

$outlier > q_3 + w \cdot (q_3 - q_1)$  oppure  $outlier < q_1 - w \cdot (q_3 - q_1)$ , con  $w=1.5$ .

Gli *outlier* saranno disegnati nei box-plot con il simbolo “+” rosso.

Per i parametri da noi considerati, *%background\_corretti* e *%foreground\_corretti*, le distribuzioni hanno caratteristiche tanto migliori quanto più la mediana si avvicina al valore 100%, e quanto più è limitata la dispersione. Ciò significa, in termini di box-plot, avere “scatole” il più possibile strette e con un valore di mediana elevato.

Per valutare l’influenza dei diversi fattori sulla qualità della segmentazione in termini di *%background\_corretti* e *%foreground\_corretti* sarà inoltre eseguito il test statistico ANOVA (ANalysis Of Variance). In particolare, per ognuno dei due parametri considerati si effettueranno due test. Il primo test sarà un ANOVA a 2 vie, dove si utilizzeranno tutti i dati a disposizione e si considereranno i seguenti fattori:

- tipo di algoritmo utilizzato (“Sottrazione/Divisione del background e Sogliatura” oppure “Naive Bayes Classifier”)
- applicazione del regionGrowing (applicazione o no del regionGrowing)

Nel secondo caso si eseguirà invece un test ANOVA a 3 vie solo sui dati ottenuti dai Naive Bayes Classifier, e sulla base dei fattori:

- tipo di aggiornamento del classificatore (assenza di aggiornamento, aggiornamento guidato o aggiornamento automatico)
- presenza di una preliminare divisione per il background (assenza o presenza della soppressione del background)
- applicazione del regionGrowing (applicazione o no del regionGrowing)

Il fattore “applicazione del regionGrowing” viene ripetuto anche nel secondo test, per poter poi permettere a posteriori un confronto tra gli algoritmi Naive Bayes Classifier anche combinando i diversi fattori.

Il test ANOVA consiste fundamentalmente in un'estensione del t-test al caso di più popolazioni campionarie. Questo test consente di capire se variando gli attributi di un fattore si riscontrano delle differenze significative sulla media della variabile, tuttavia se un fattore (ad es. "aggiornamento") presenta più di due attributi (ad es. "no aggiornamento", "aggiornamento automatico", "aggiornamento guidato") il test non consente di individuare quali siano i particolari attributi del fattore a causare tale differenza. L'ipotesi nulla alla base del test su ognuno dei fattori è infatti che tutte le popolazioni campionarie associate agli attributi provengano da una distribuzione con la stessa media. Per eseguire un confronto tra i singoli attributi è dunque necessaria una comparazione multipla successiva, che in questo lavoro di tesi sarà realizzata attraverso il test di Bonferroni [39]. Nel prossimo capitolo, oltre a una descrizione a parole dei risultati ottenuti dal test ANOVA, si aggiungeranno nei box-plot delle parentesi quadre indicanti il P-value ottenuto dal confronto della due distribuzioni indicate. Ricordando che il P-value è il più basso livello di significatività per cui i dati osservati portano a rifiutare l'ipotesi nulla  $H_0$ , più basso è il P-value più è probabile sia falsa l'ipotesi nulla, e dunque specularmente è più probabile che le distribuzioni siano significativamente differenti.

Affinché il test ANOVA fornisca risultati significativi occorre che le distribuzioni dei dati in ingresso siano il più possibile rappresentabili come normali gaussiane. Non essendo tali nel nostro caso le distribuzioni dei parametri *%background\_corretti* e *%foreground\_corretti*, applichiamo preliminarmente ai dati una trasformazione logaritmica di questo tipo:

$$x' = -\log\left(1 - \frac{x}{100}\right)$$

ed eseguiamo il test ANOVA sui dati trasformati invece che sui dati originali. In Figura 2.10 un esempio di come varia la distribuzione dei dati dopo la trasformazione logaritmica indicata.

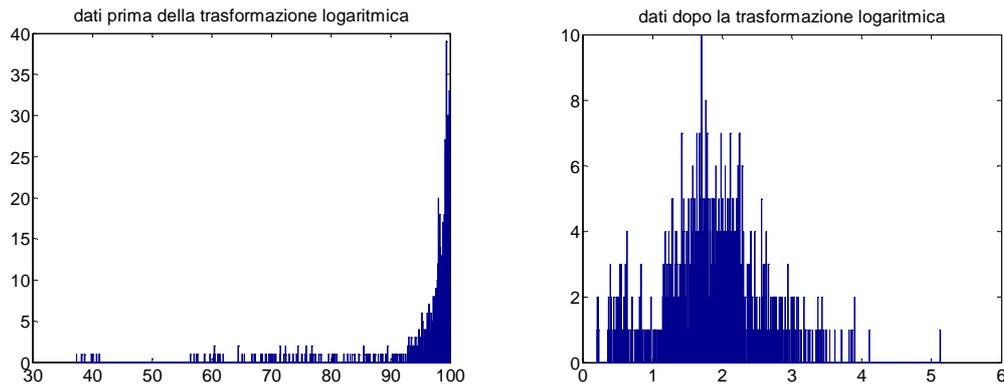


Figura 2.10 Distribuzione dei dati prima (sinistra) e dopo (destra) la trasformazione logaritmica indicata

Oltre alle misure precedentemente citate, si eseguirà per i due algoritmi presentanti i risultati migliori un ulteriore test, che consiste nella valutazione della distanza di Hausdorff tra i contorni delle silhouette stimati dall'algoritmo e i corrispondenti contorni nei gold-standard.

In generale, la *distanza di Hausdorff* è uno strumento utile per misurare la distanza tra due insiemi di punti. In particolare, dato  $R$  l'insieme di punti  $r$  di riferimento (ottenuti dal gold-standard) ed  $E$  l'insieme di punti  $e$  stimati con l'algoritmo in esame, si definisce:

- distanza massima di  $E$  rispetto a  $R$ :

$$H_{re}(E, R) = \max_{r \in R} d(r, E)$$

- distanza massima di  $R$  rispetto a  $E$ :

$$H_{er}(E, R) = \max_{e \in E} d(e, R)$$

dove si ricorda che, in generale, in uno spazio 2-D la distanza di un punto  $a$  da un insieme  $B$  è definita come:

$$d(a, B) = \min_{b \in B} \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

La distanza di Hausdorff tra gli insiemi  $E$  ed  $R$  è allora definita come:

$$H(E, R) = \max\{H_{re}(E, R), H_{er}(R, E)\}$$

In altri termini, la distanza di Hausdorff rappresenta la massima tra le distanze massime del primo insieme rispetto al secondo e del secondo insieme rispetto al primo [40].

Nel nostro particolare caso, verrà valutata ogni volta la distanza di Hausdorff tra i punti del contorno della silhouette nel gold-standard e i corrispondenti punti stimati dall'algoritmo di segmentazione. Alla fine, si valuterà per ognuno dei due algoritmi considerati il valore di mediana della distanza di Hausdorff sulle 72 immagini per cui è stato definito il gold-standard come indice di posizione, mentre per dare un'idea della dispersione del dato si aggiungeranno i valori di UB e LB (upper bound e lower bound), valutati nello stesso modo che per i parametri *%background\_corretti* e *%foreground\_corretti*. Oltre che alla distanza di Hausdorff (che rappresenta la massima distanza tra la due curve), sarà preso in considerazione anche il parametro MAD (*mean absolute deviation*), rappresentante di volta in volta la media distanza tra le due curve. Anche in questo caso, essendo 72 in totale le prove anche la MAD avrà una sua distribuzione, di cui riporteremo ancora la mediana come indice di posizione, LB e UB come indici di dispersione. Tutti i risultati ottenuti saranno poi riassunti e organizzati come in Tabella 2.3.

<b>Distanza di Hausdorff</b>	
mediana=	pixel
UB=	pixel
LB=	pixel
<b>MAD</b>	
mediana=	pixel
UB=	pixel
LB=	pixel

*Tabella 2.3 Tabella riassuntiva le distanze di Hausdorff valutate*

Come già introdotto, questa ulteriore valutazione sarà fatta sui due algoritmi che presentano i risultati migliori, dove il criterio con cui saranno selezionati i due algoritmi sono:

- mediana *%background\_corretti* > 95%
- mediana *%foreground\_corretti* più elevata possibile

considerando i risultati post-regionGrowing.

Nel prossimo capitolo si riporteranno i risultati delle segmentazioni ottenute, evidenziando per ogni algoritmo i valori dei parametri appena descritti. Per semplificare la lettura, ricordiamo di seguito la simbologia che permette di identificare velocemente i sette algoritmi:

$SD_B$	Sottrazione/Divisione del background e Sogliatura
$NBC$	Naive Bayes Classifier
$NBC_B$	Naive Bayes Classifier, applicato a immagini già divise per il background
$NBC_{AG}$	Naive Bayes Classifier, con aggiornamento guidato
$NBC_{AG,B}$	Naive Bayes Classifier, con aggiornamento guidato e applicato a immagini già divise per il background
$NBC_{AA}$	Naive Bayes Classifier, con aggiornamento automatico
$NBC_{AA,B}$	Naive Bayes Classifier, con aggiornamento automatico e applicato a immagini già divise per il background



### 3. Risultati

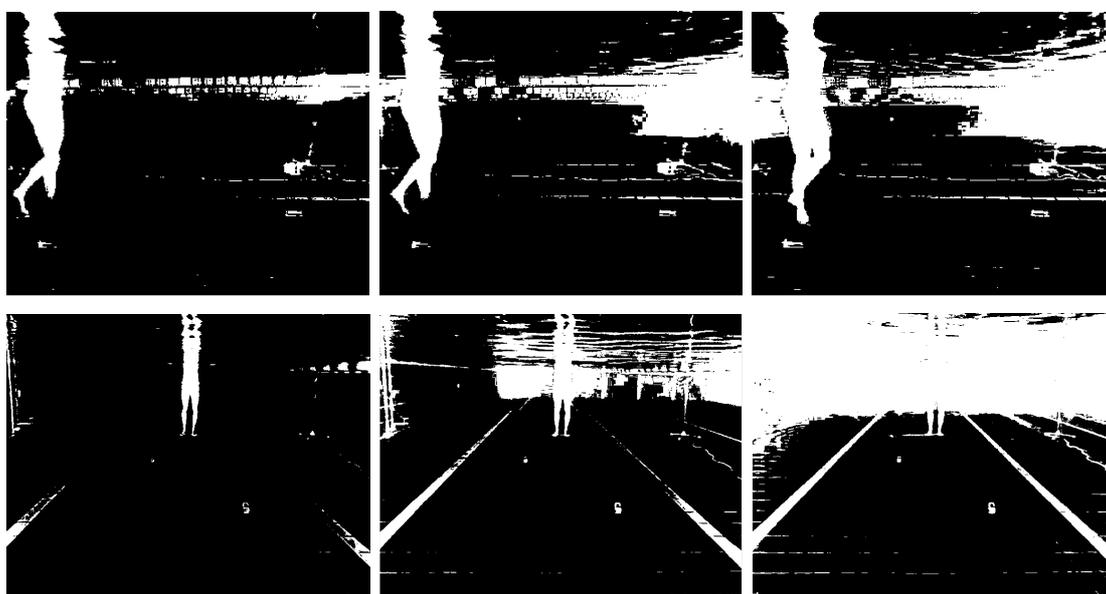
In questo capitolo si presenteranno i risultati ottenuti con ognuno degli algoritmi circa la qualità della segmentazione. In particolare, si mostreranno nella sezione 3.1 i risultati ordinandoli per algoritmo, e mostrando per ognuno sia le tabelle che evidenziano mediana, LB e UB delle due distribuzioni *%background\_corretti* e *%foreground\_corretti* nelle 72 immagini di prova, che gli istogrammi relativi alle distribuzioni dei parametri, pre- e post-regionGrowing. Si riporteranno in seguito, nella sezione 3.2, i risultati delle ulteriori valutazioni (comprendenti i valori di mediana, UB e LB della distanza di Hausdorff e della MAD sulle 72 immagini) effettuate sui due algoritmi che presentano i migliori risultati post-regionGrowing.

Diamo invece di seguito una breve descrizione dei risultati ottenuti, utile per l'interpretazione dei dati numerici a seguire, e supportata da una serie di box-plot alla fine del paragrafo. I box-plot presentati permetteranno infatti un immediato confronto visivo tra gli algoritmi e, grazie all'aggiunta di particolari simboli grafici in alcuni casi, evidenzieranno alcuni dei risultati ottenuti con i test ANOVA.

L'algoritmo  $SD_B$  presenta una distribuzione del parametro *%background\_corretti* comparabile rispetto agli altri algoritmi (escludendo il  $NBC_{AA}$ ), mentre per quanto riguarda la *%foreground\_corretti* è possibile notare dai box-plot (Figura 3.4) una dispersione significativamente maggiore, sia prima che dopo l'applicazione del regionGrowing. Il dato indica che la qualità del riconoscimento del foreground per l'algoritmo  $SD_B$  è piuttosto variabile a seconda del fotogramma considerato. Questa maggior dispersione è verificabile anche osservando gli istogrammi o analizzando gli indici LB e UB nella sezione 3.1.

Particolare attenzione va posta sui risultati dell'algoritmo  $NBC_{AA}$ . In particolare, per questo algoritmo la distribuzione del parametro  $\%background\_corretti$  presenta una dispersione significativamente maggiore rispetto agli altri algoritmi, unitamente a un valore inferiore della mediana (Figura 3.3). In effetti, da un punto di vista più qualitativo si nota che per l'algoritmo  $NBC_{AA}$  la segmentazione diventa velocemente "rumorosa", con una rapida crescita del numero di FF (*false foreground*) già pochi fotogrammi dopo l'inizio dell'aggiornamento del modello.

Per meglio chiarire questo concetto si osservino le immagini nella Figura 3.1.

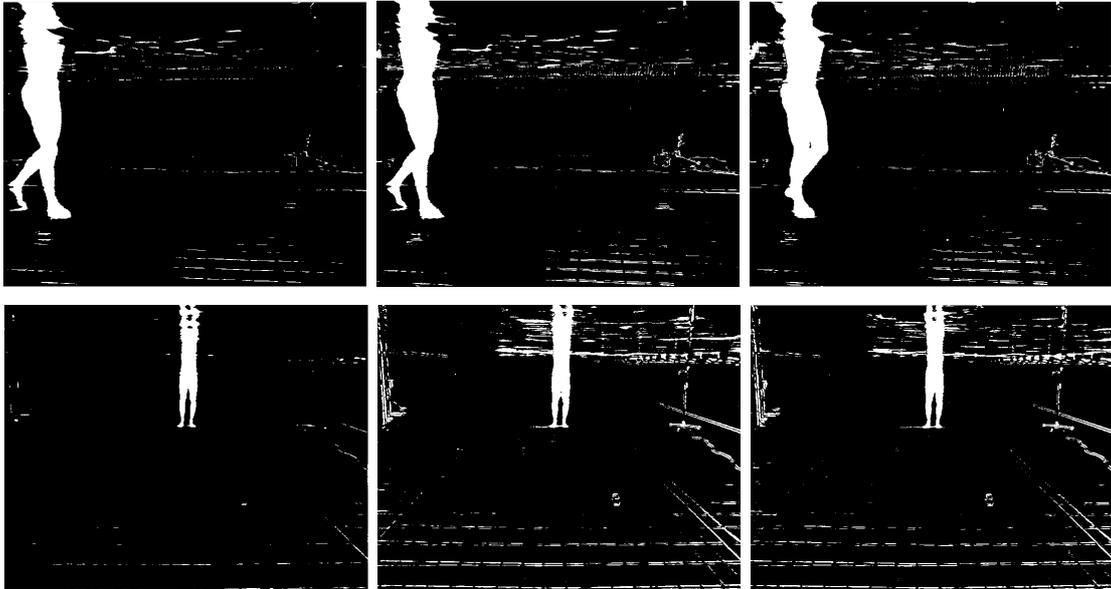


*Figura 3.1 Esempio di rapida crescita del rumore nell'algoritmo  $NBC_{AA}$  pre-regionGrowing. In alto, soggetto 1, cam2 e frame 309 (sinistra), 313 (centro) e 326 (destra). In basso, soggetto 1, cam3 e frame 1 (sinistra), 18 (centro) e 25 (destra). Si noti come in pochi fotogrammi (<20) il rumore aumenta rapidamente nelle immagini segmentate.*

L'effetto di crescita del rumore è stato riscontrato anche, seppur in maniera più limitata, per la versione del suddetto algoritmo che sfrutta una preliminare divisione per il background, il  $NBC_{AA,B}$ . In effetti, anche se non particolarmente evidente, dall'osservazione dei box-plot in Figura 3.3 e Figura 3.5, dagli istogrammi e dai dati sintetici della sezione 3.1, si rileva che la distribuzione di  $\%background\_corretti$

presenta una mediana inferiore agli altri algoritmi (escluso l'algoritmo  $NBC_{AA}$ ), anche dopo l'applicazione dell'algoritmo `regionGrowing`.

Per dare un'idea del diverso tasso di crescita del rumore nell'algoritmo  $NBC_{AA,B}$  rispetto all'algoritmo  $NBC_{AA}$  si confrontino le immagini in Figura 3.2 con quelle in Figura 3.1.



*Figura 3.2 Esempio per l'algoritmo  $NBC_{AA,B}$  pre-regionGrowing. I fotogrammi sono gli stessi considerati in Figura 3.1, per mostrare la differenza del tasso di crescita del rumore rispetto al precedente algoritmo.*

I due algoritmi che rispettano i requisiti di cui al punto 2.4.1 e per cui viene effettuato anche il test della distanza di Hausdorff risultano essere  $NBC_{AG,B}$  e  $NBC_{AA,B}$ . In particolare (sezione 3.2), si ottiene un valore di mediana della distanza di Hausdorff tra i contorni del soggetto nel gold-standard e i contorni stimati dall'algoritmo che per  $NBC_{AA,B}$  è circa 3.46 volte quella ottenuta con  $NBC_{AG,B}$ . Anche osservando le distanze medie tra le due curve (MAD) si mantiene lo stesso rapporto. In particolare, il rapporto tra la mediana della MAD dell'algoritmo  $NBC_{AA,B}$  e  $NBC_{AG,B}$  è 3.70.

Un altro accento va posto ancora sull'efficacia dell'algoritmo `regionGrowing` nel migliorare effettivamente la qualità della segmentazione. In particolare, è possibile notare dai box-plot (Figura 3.7) che passando dal pre- al post-`regionGrowing` la qualità `%background_corretti` migliora sempre per tutti gli algoritmi (valore

della mediana aumenta, dispersione tendenzialmente diminuisce), mentre ciò non è sempre vero considerando il parametro *%foreground\_corretti* (Figura 3.8). In particolare, il `regionGrowing` provoca un peggioramento complessivo della distribuzione del parametro *%foreground\_corretti* (diminuzione del valore della mediana, tendenziale aumento della dispersione) negli algoritmi  $SD_B$ , NBC,  $NBC_{AG}$ ,  $NBC_{AA}$  (Figura 3.8). Osservando inoltre gli istogrammi nella sezione 3.1, si nota per gli algoritmi NBC,  $NBC_{AG}$ ,  $NBC_{AA}$  la comparsa, solo dopo il `regionGrowing`, di fotogrammi in cui la *%foreground\_corretti* si azzerava. Risulta infine, per i tre appena citati algoritmi, che in alcuni fotogrammi segmentati pre-`regionGrowing` l'area bianca più estesa nelle immagini non è il soggetto: in particolare, il fenomeno avviene con maggiore frequenza in corrispondenza della telecamera 4.

Quanto appena detto sull'evoluzione dei parametri *%background\_corretti* e *%foreground\_corretti* negli algoritmi dal pre- al post-`regionGrowing` è riscontrabile anche osservando i risultati sintetici e gli istogrammi nella sezione 3.1.

Per quanto riguarda invece gli algoritmi  $NBC_B$ ,  $NBC_{AG,B}$ ,  $NBC_{AA,B}$ , oltre a notarsi un miglioramento della distribuzione del parametro *%background\_corretti*, non si osservano significative variazioni della distribuzione del parametro *%foreground\_corretti* dal pre- al post-`regionGrowing` (Figura 3.7 e Figura 3.8).

Per quanto riguarda i test statistici ANOVA i risultati ottenuti sono, considerando la *%background\_corretti*:

- l'utilizzo dell'algoritmo  $SD_B$  produce risultati significativamente diversi rispetto all'utilizzo degli algoritmi NBC (P-value<0.01)
- l'utilizzo del `regionGrowing` produce cambiamenti significativi nelle distribuzioni (P-value<0.01)
- l'aggiornamento automatico produce risultati significativamente diversi negli algoritmi NBC rispetto all'aggiornamento guidato o all'assenza di aggiornamento (P-value<0.01). Non vi sono invece

significative differenze tra l'aggiornamento guidato e l'assenza di aggiornamento (P-value=0.47)

- la preliminare divisione per il background produce risultati significativamente differenti negli algoritmi NBC rispetto all'impiego di immagini non trattate (P-value<0.01)

Per quanto riguarda invece i risultati dei test statistici ANOVA ottenuti considerando la *%foreground\_corretti*:

- l'utilizzo dell'algoritmo  $SD_B$  produce risultati significativamente diversi rispetto all'utilizzo degli algoritmi NBC (P-value<0.01)
- l'utilizzo del `regionGrowing` non produce cambiamenti significativi nelle distribuzioni (P-value=0.68)
- l'aggiornamento automatico produce risultati significativamente diversi negli algoritmi NBC rispetto all'aggiornamento guidato o all'assenza di aggiornamento (P-value<0.01). Non vi sono invece significative differenze tra l'aggiornamento guidato e l'assenza di aggiornamento (P-value=0.4914)
- la preliminare divisione per il background produce risultati significativamente differenti negli algoritmi NBC rispetto all'impiego di immagini non trattate (P-value<0.01)

Parte di queste considerazioni può essere ritrovata anche dall'osservazione dei box-plot a seguire.

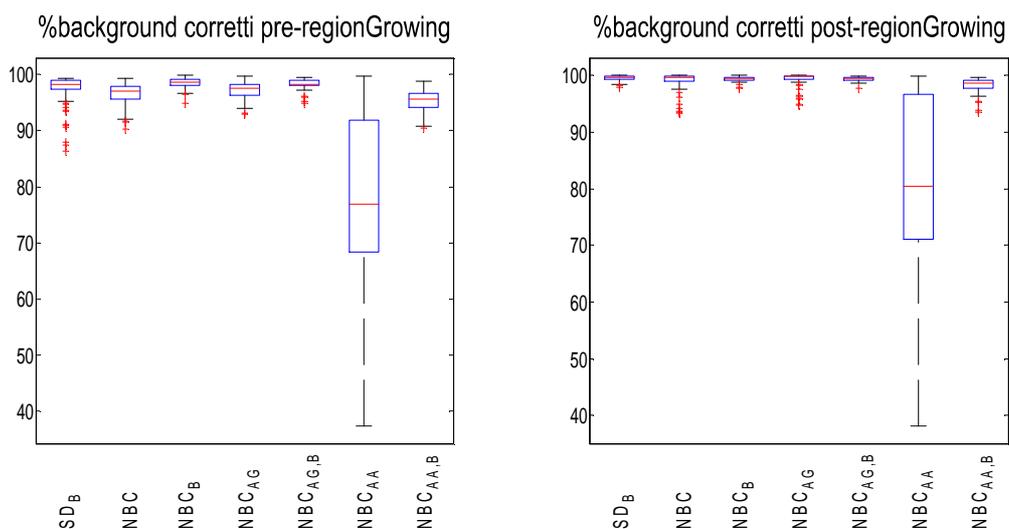


Figura 3.3 Visualizzazione di come variano le distribuzioni di *%background\_corretti* nei 7 algoritmi dal pre- al post-regionGrowing

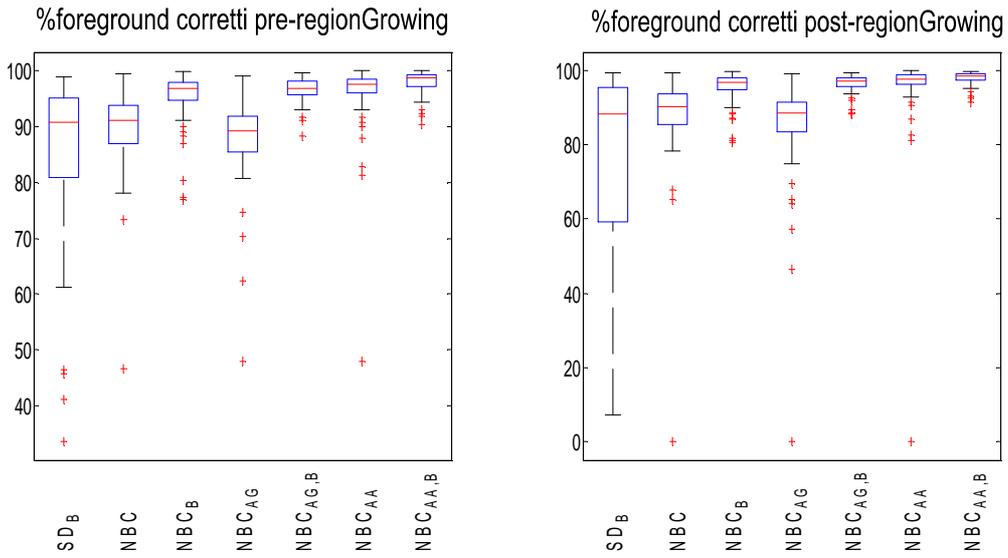


Figura 3.4 Visualizzazione di come variano le distribuzioni di %foreground\_corretti nei 7 algoritmi dal pre- al post-regionGrowing

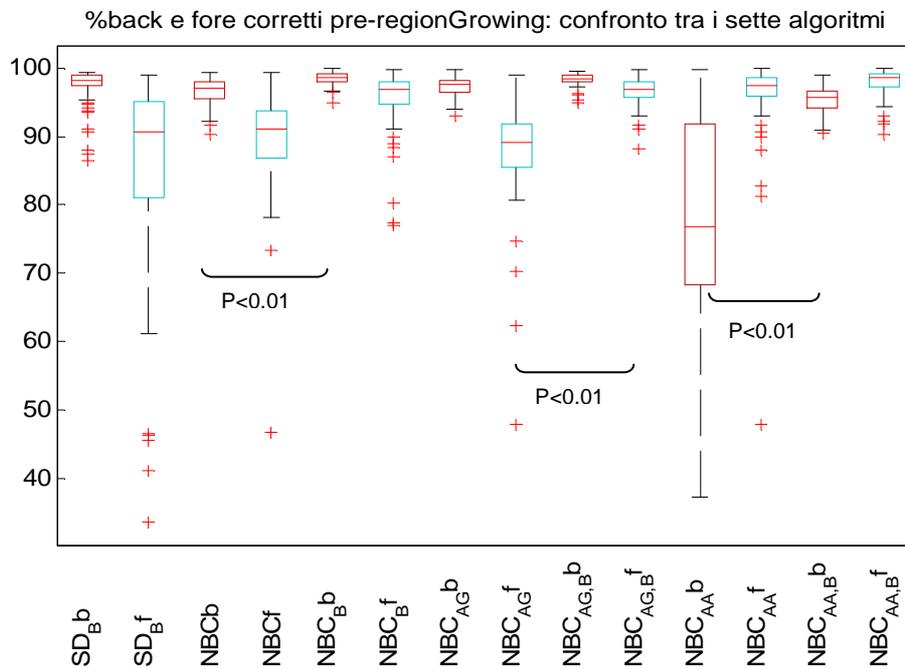


Figura 3.5 Confronto tra i 7 algoritmi, in termini di distribuzioni dei parametri %background\_corretti (rosso) e %foreground\_corretti (azzurro), prima della funzione regionGrowing

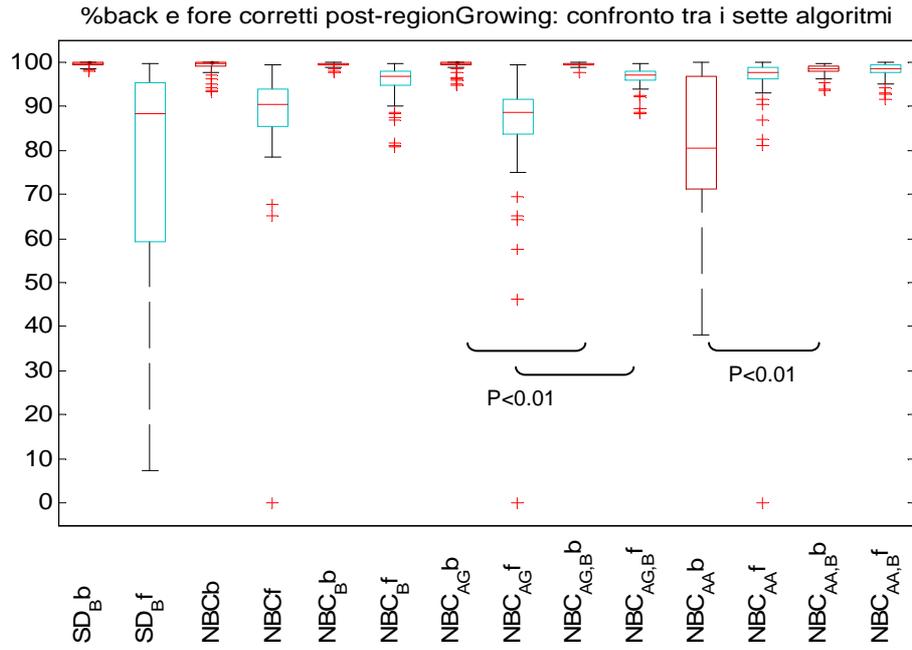


Figura 3.6 Confronto tra i 7 algoritmi, in termini di distribuzioni dei parametri %background\_corretti (rosso) e %foreground\_corretti (azzurro), dopo la funzione regionGrowing

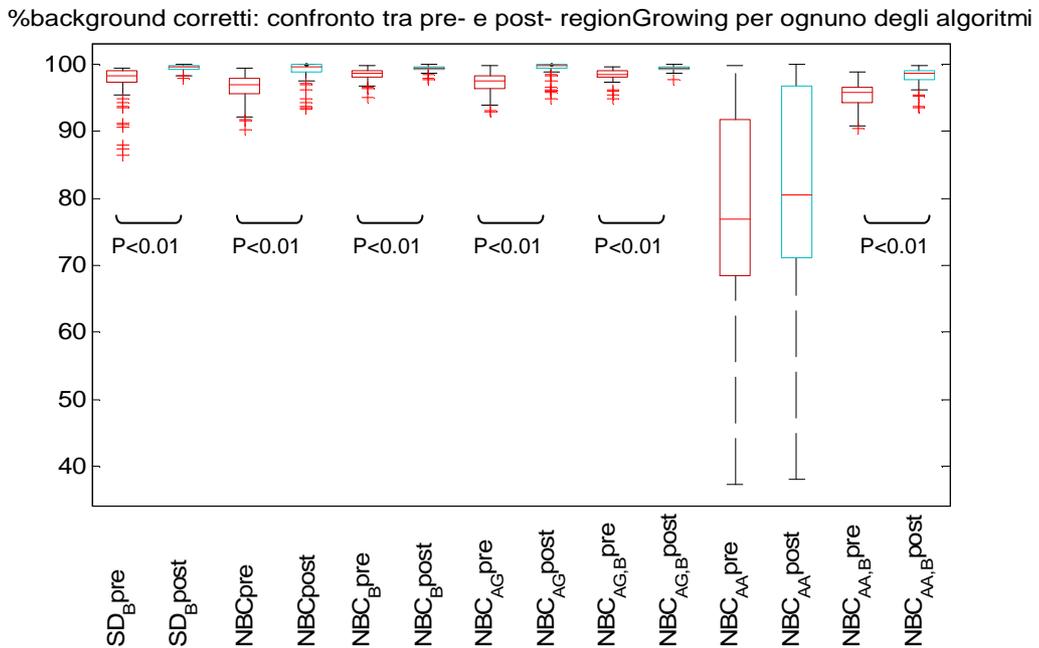


Figura 3.7 Visualizzazione di come variano le distribuzioni di %background\_corretti dal pre- (rosso) al post-regionGrowing (azzurro), affiancando i risultati per i sette algoritmi

%foreground corretti: confronto tra pre- e post- regionGrowing per ognuno degli algoritmi

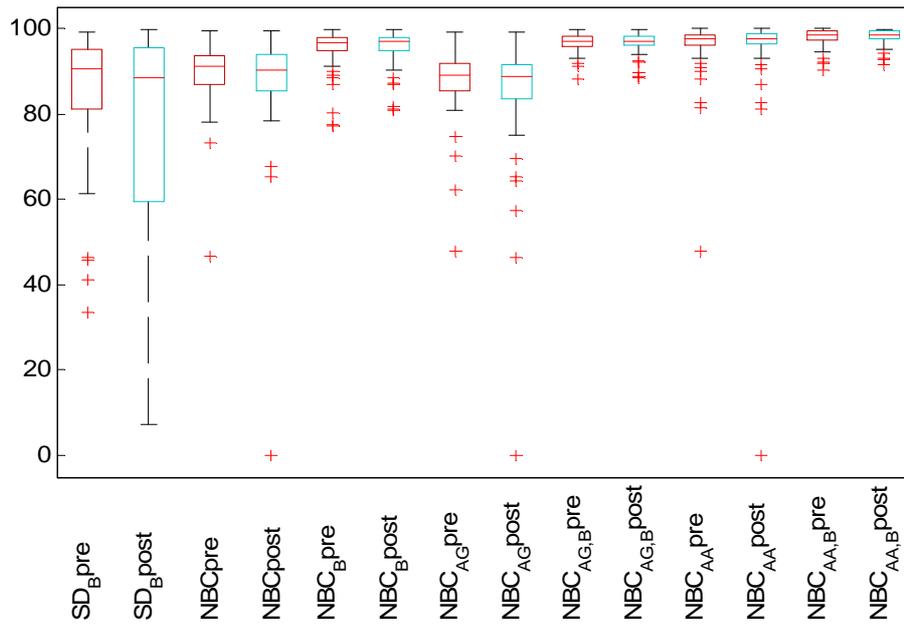


Figura 3.8 Visualizzazione di come variano le distribuzioni di %foreground\_corretti dal pre- (rosso) al post-regionGrowing (azzurro), affiancando i risultati per i sette algoritmi

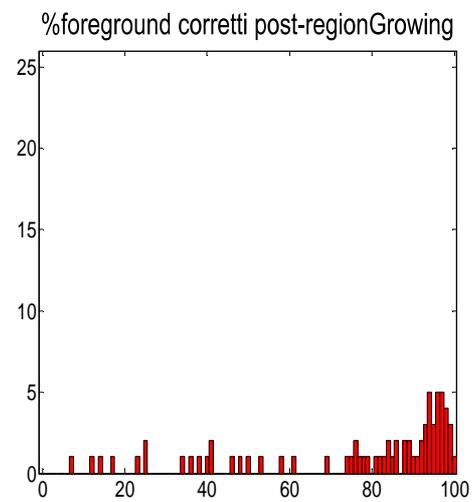
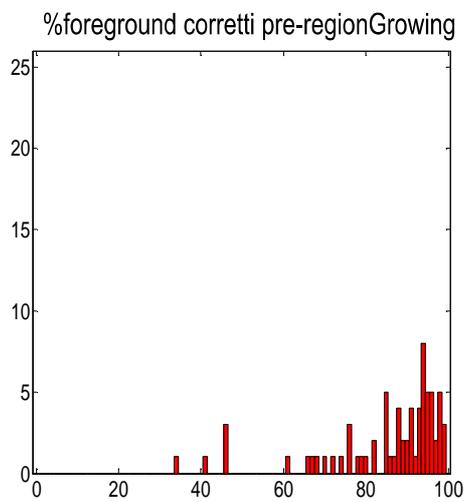
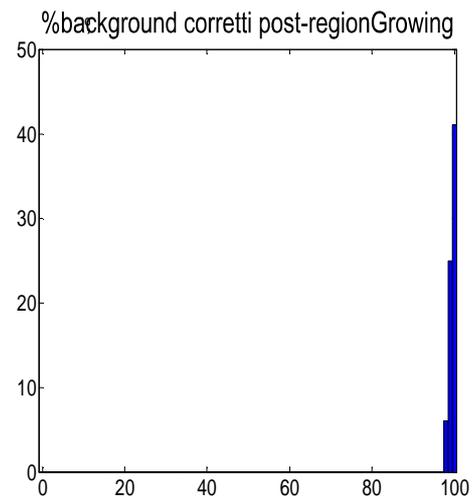
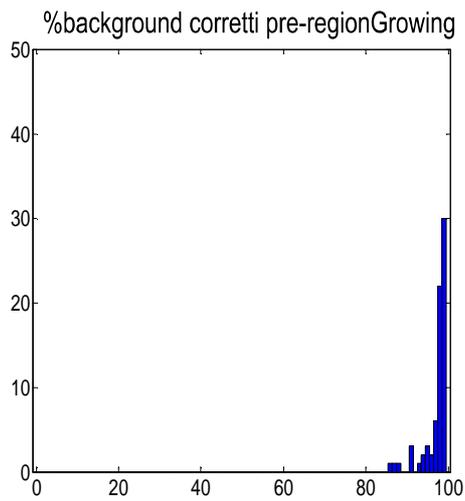
### 3.1 Tabelle riassuntive e istogrammi

In questa sezione si mostreranno, per ognuno degli algoritmi, sia le tabelle mostranti i valori di mediana, LB e UB delle distribuzioni dei due parametri considerati, che 4 istogrammi sempre nello stesso ordine: in alto e in blu %background\_corretti, in basso e in rosso %foreground\_corretti, con a sinistra le versioni pre-regionGrowing e a destra le post-regionGrowing. Si noti l'utilizzo di scale diverse per i due tipi di istogramma: in particolare i numeri in ordinata andranno da 0 a 50 per le %background\_corretti e da 0 a 26 per le %foreground\_corretti.

### 3.1.1 Algoritmo SD<sub>B</sub>

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	98.21%	UB=	99.19%
		LB=	91.04%
<b>%foreground_corretti</b>			
m=	90.67%	UB=	98.21%
		LB=	46.47%

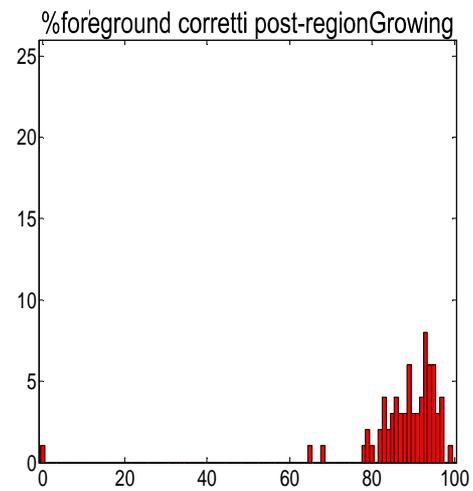
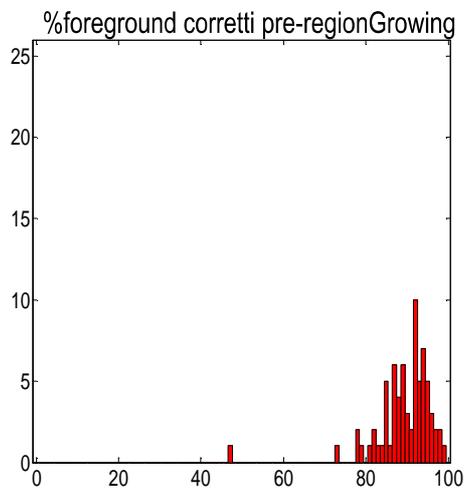
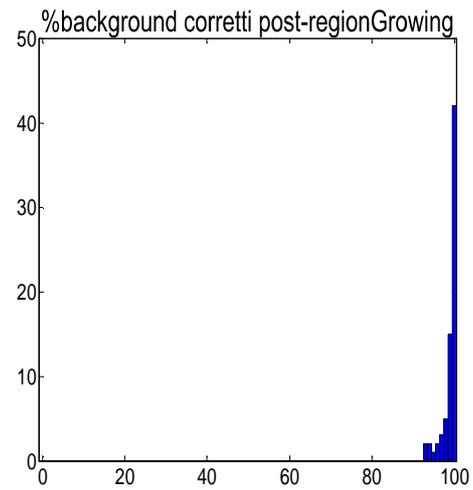
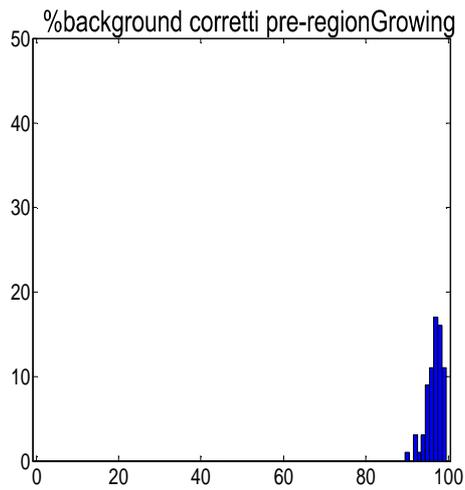
post-regionGrowing:			
<b>%background_corretti</b>			
m=	99.60%	UB=	99.97%
		LB=	98.37%
<b>%foreground_corretti</b>			
m=	88.25%	UB=	98.43%
		LB=	23.10%



### 3.1.2 Algoritmo NBC

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	96.96%	UB=	99.08%
		LB=	93.12%
<b>%foreground_corretti</b>			
m=	91.13%	UB=	96.91%
		LB=	78.99%

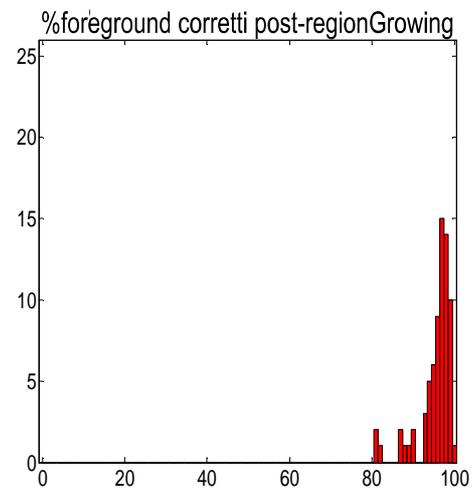
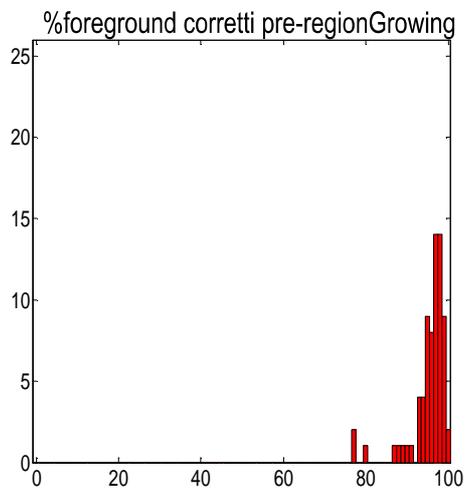
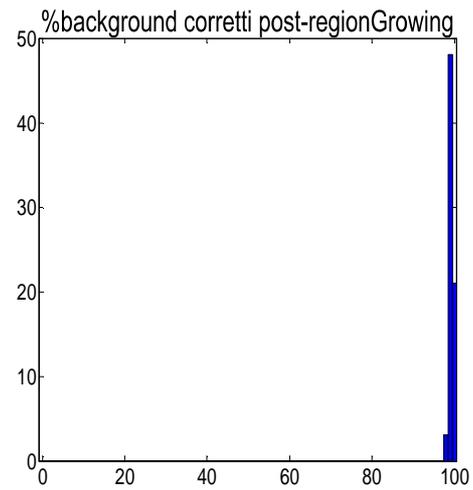
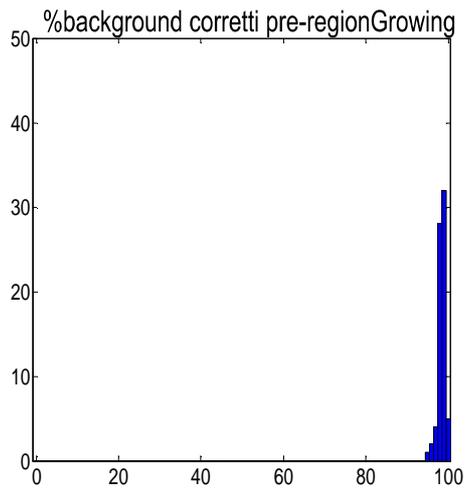
post-regionGrowing:			
<b>%background_corretti</b>			
m=	99.64%	UB=	99.97%
		LB=	94.90%
<b>%foreground_corretti</b>			
m=	90.35%	UB=	96.84%
		LB=	78.74%



### 3.1.3 Algoritmo NBC<sub>B</sub>

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	98.57%	UB=	99.50%
		LB=	96.82%
<b>%foreground_corretti</b>			
m=	96.76%	UB=	99.31%
		LB=	88.32%

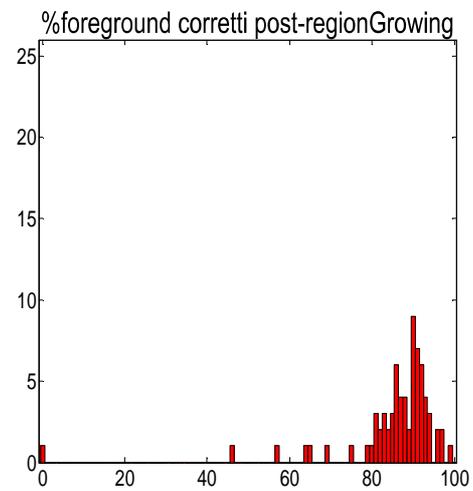
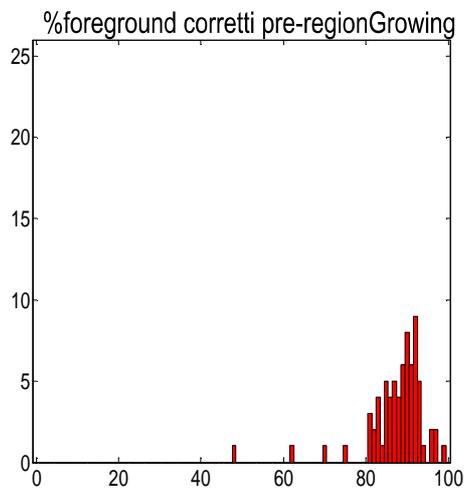
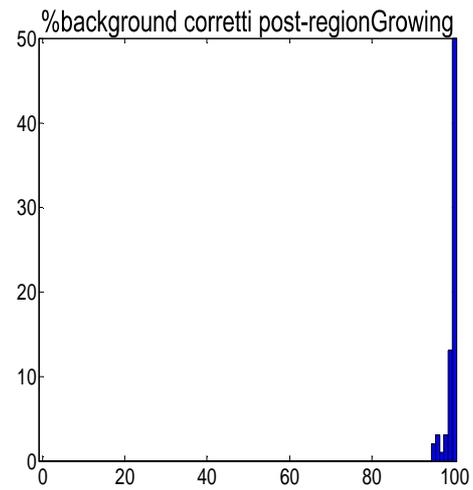
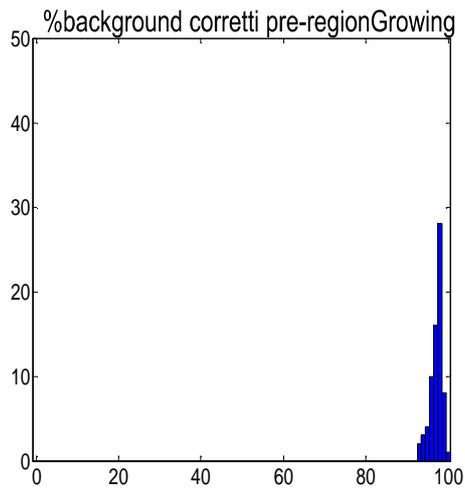
post-regionGrowing:			
<b>%background_corretti</b>			
m=	99.40%	UB=	99.90%
		LB=	98.88%
<b>%foreground_corretti</b>			
m=	96.78%	UB=	99.14%
		LB=	87.31%



### 3.1.4 Algoritmo NBC<sub>AG</sub>

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	97.56%	UB=	99.07%
		LB=	94.35%
<b>%foreground_corretti</b>			
m=	89.16%	UB=	96.04%
		LB=	80.71%

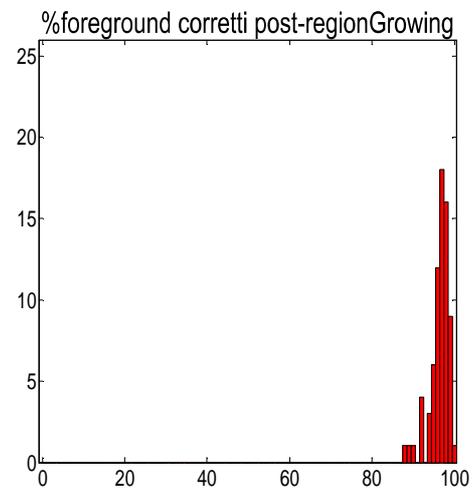
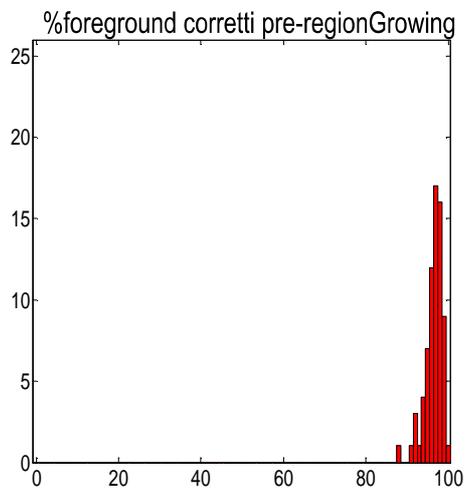
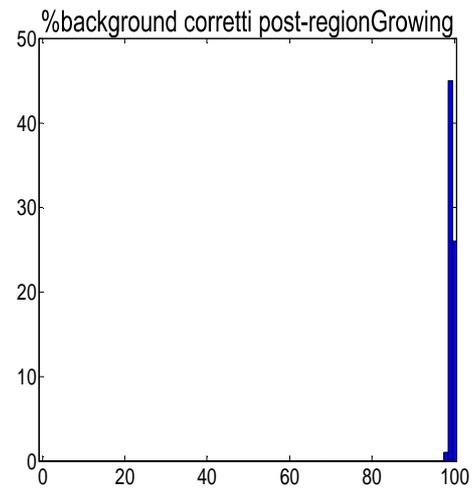
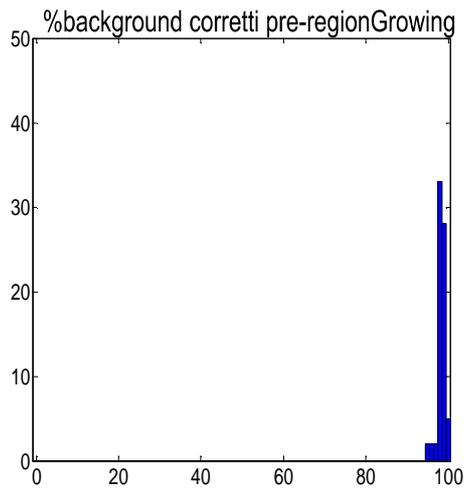
post-regionGrowing:			
<b>%background_corretti</b>			
m=	99.77%	UB=	99.96%
		LB=	96.26%
<b>%foreground_corretti</b>			
m=	88.62%	UB=	96.03%
		LB=	65.18%



### 3.1.5 Algoritmo NBC<sub>AG,B</sub>

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	98.36%	UB=	99.52%
		LB=	97.25%
<b>%foreground_corretti</b>			
m=	96.86%	UB=	98.94%
		LB=	91.73%

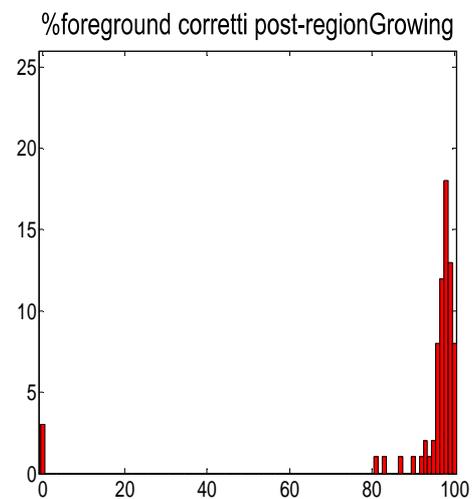
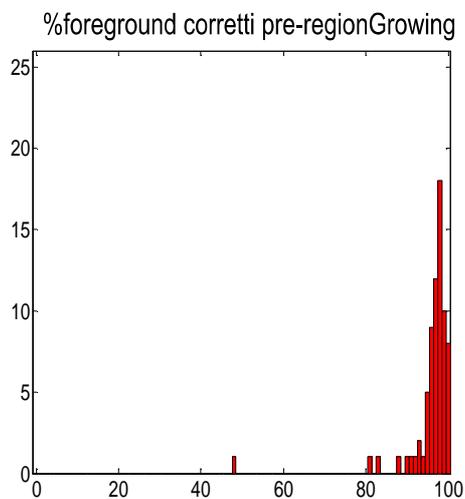
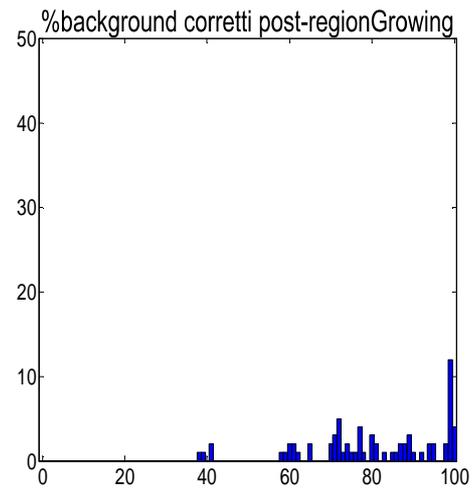
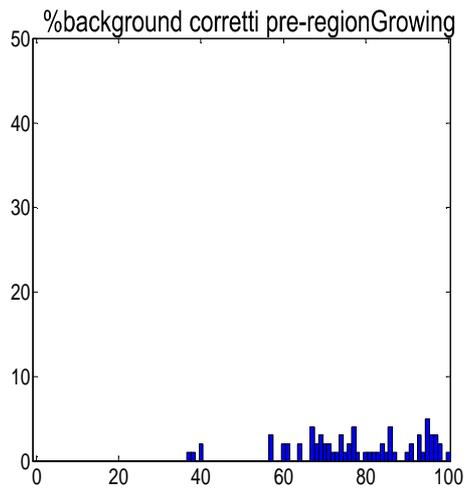
post-regionGrowing:			
<b>%background_corretti</b>			
m=	99.41%	UB=	99.84%
		LB=	98.87%
<b>%foreground_corretti</b>			
m=	97.05%	UB=	98.78%
		LB=	91.98%



### 3.1.6 Algoritmo NBC<sub>AA</sub>

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	76.84%	UB=	97.21%
		LB=	56.52%
<b>%foreground_corretti</b>			
m=	97.48%	UB=	99.66%
		LB=	90.01%

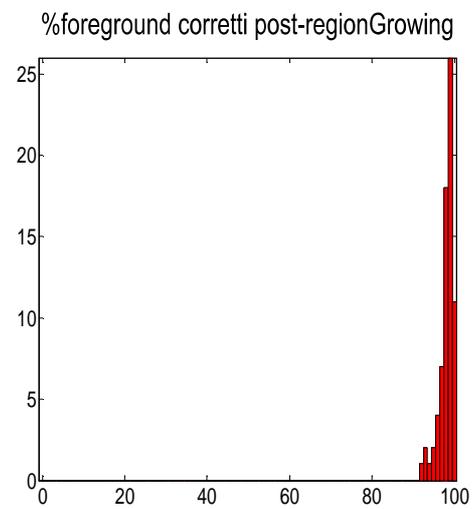
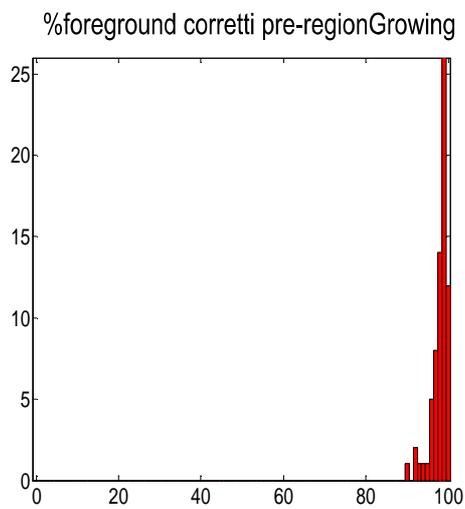
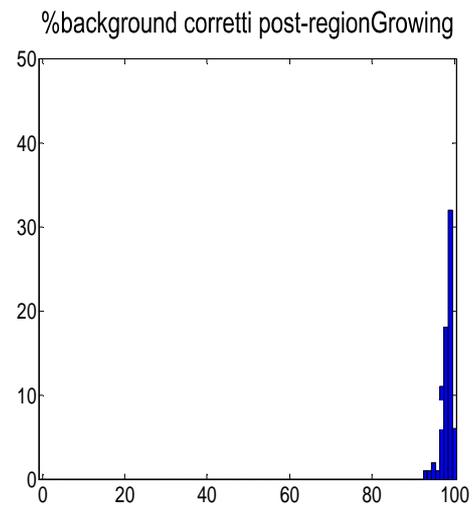
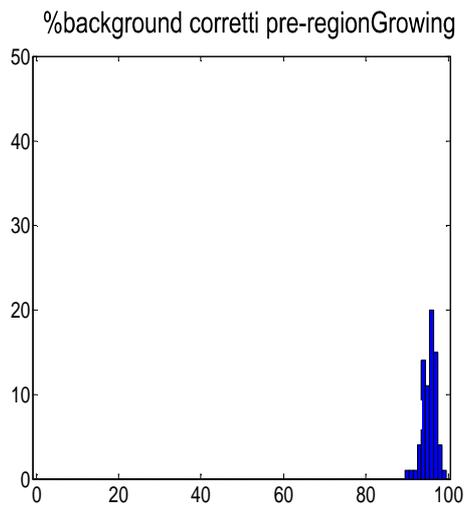
post-regionGrowing:			
<b>%background_corretti</b>			
m=	80.45%	UB=	99.48%
		LB=	57.68%
<b>%foreground_corretti</b>			
m=	97.65%	UB=	99.82%
		LB=	82.53%



### 3.1.7 Algoritmo NBC<sub>AA,B</sub>

pre-regionGrowing:			
<b>%background_corretti</b>			
m=	95.70%	UB=	97.51%
		LB=	93.25%
<b>%foreground_corretti</b>			
m=	98.60%	UB=	99.78%
		LB=	94.38%

post-regionGrowing:			
<b>%background_corretti</b>			
m=	98.57%	UB=	99.54%
		LB=	96.26%
<b>%foreground_corretti</b>			
m=	98.54%	UB=	99.65%
		LB=	95.05%



### 3.2 Distanza di Hausdorff

I due algoritmi che presentano, nei risultati post-regionGrowing, un valore della mediana di  $\%background\_corretti > 95\%$  e i maggiori valori di mediana di  $\%foreground\_corretti$  sono  $NBC_{AG,B}$  e  $NBC_{AA,B}$ . Si riportano pertanto in questa sezione i dati relativi agli ulteriori test (comprendenti la valutazione della distanza di Hausdorff e della MAD tra i contorni stimati della silhouette e i corrispondenti contorni nel gold-standard) eseguiti su questi due algoritmi. Tutti i risultati presentati sono stati valutati complessivamente sulle 72 immagini per cui si sono definiti i gold-standard.

▪  $NBC_{AG,B}$

<b>Distanza di Hausdorff</b>	
mediana=	30.3064 pixel
UB=	150.0533 pixel
LB=	11.4018 pixel
<b>MAD</b>	
mediana=	2.5821 pixel
UB=	14.5426 pixel
LB=	1.1594 pixel

▪  $NBA_{AA,B}$

<b>Distanza di Hausdorff</b>	
mediana=	104.7400 pixel
UB=	330.5420 pixel
LB=	18.9737 pixel
<b>MAD</b>	
mediana=	9.5516 pixel
UB=	63.7657 pixel
LB=	1.8678 pixel



## 4. Discussione dei risultati

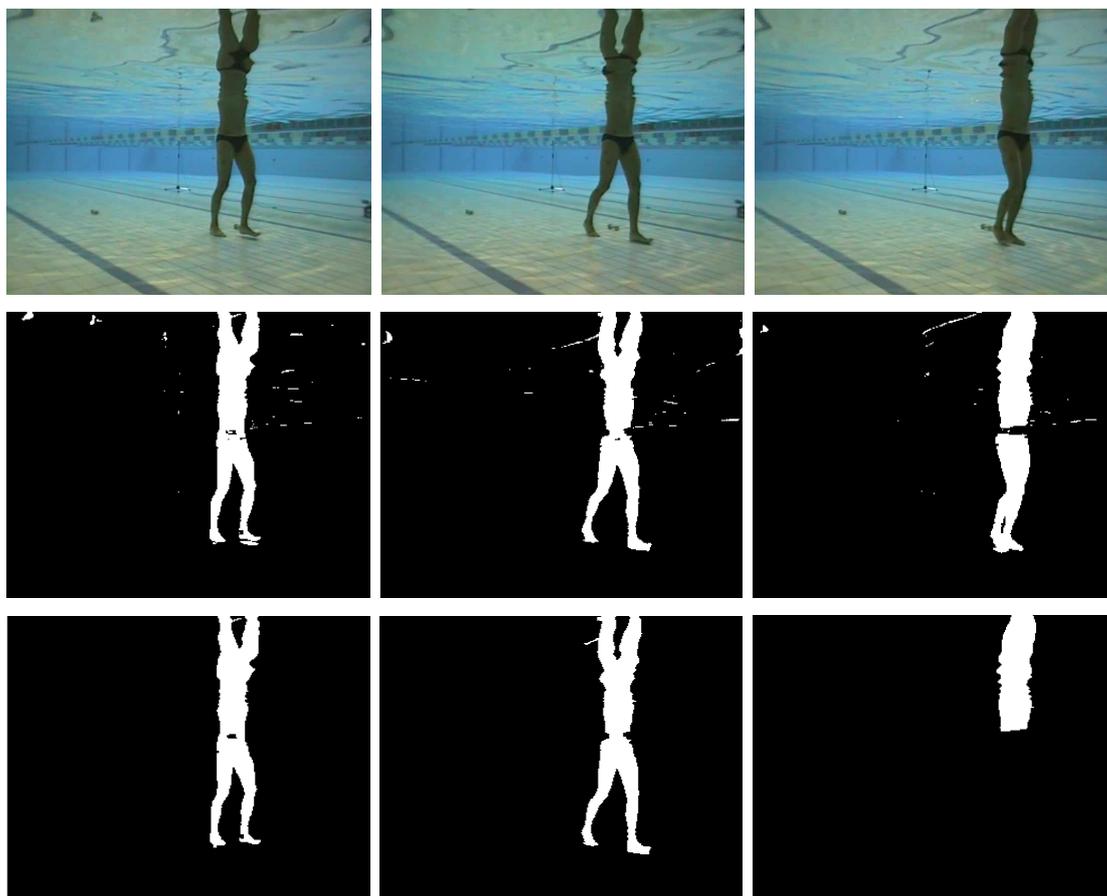
In questo capitolo si discuteranno i risultati ottenuti e mostrati al capitolo precedente. In particolare, nelle prime tre sezioni si tenterà di illustrare i motivi che portano ai risultati osservati in precedenza, nella sezione 4.4 si discuterà quale tra gli algoritmi implementati produca il risultato complessivamente migliore, infine nella sezione 4.5 si cercherà di rapportare il presente lavoro di tesi alla letteratura.

### 4.1 L'algoritmo Sottrazione/Divisione e Sogliatura

L'algoritmo  $SD_B$  è quello che sia nel pre- che nel post-regionGrowing presenta il valore di dispersione più elevato della distribuzione *%foreground\_corretti*, oltre che un valore di mediana tra i più bassi. Ciò significa che l'algoritmo  $SD_B$  è quello presentante la maggior variabilità nella qualità di segmentazione del soggetto, e il motivo è da ricercarsi nell'utilizzo di un valore di soglia unico per tutti i pixel. Quando infatti si sottrae (in valore assoluto) a una determinata immagine il suo background, il risultato è un'immagine in cui più i pixel dell'immagine di prova sono distanti nello spazio RGB dai pixel dello sfondo, e più sarà alto il valore finale associato a quel punto: scelto dunque un valore di soglia per la distanza in colore tra i pixel si rende possibile una discriminazione tra background e foreground. Il problema nasce tuttavia nel momento in cui l'immagine modellante lo sfondo non è uniforme (come nel nostro caso). In tal caso infatti il soggetto, muovendosi, può andare ad assumere posizioni nell'immagine in cui lo sfondo presenta colori più simili al soggetto stesso. E' dunque possibile che, per alcuni punti, la distanza in colore tra pixel del soggetto e pixel dello sfondo sia troppo poco elevata, e in particolare rimanga al di sotto del valore di soglia scelto per quella telecamera. In questo caso, si assiste alle genesi di "buchi" nella silhouette (non necessariamente di pochi pixel), dove i buchi

compaiono in funzione del numero del fotogramma segmentato con una relazione difficilmente individuabile, essendo questi dipendenti sia dall'istante in cui il soggetto assume una determinata posizione, che dalle caratteristiche dello sfondo.

Un esempio di quanto appena esposto è osservabile nelle immagini sottostanti, Figura 4.1.



*Figura 4.1 Esempi di immagini originali del soggetto 1, telecamera 1 (in alto), corrispondenti immagini segmentate con l'algoritmo  $SD_B$  pre- (al centro) e post-regionGrowing (in basso). Si noti come in questo caso la presenza dei buchi nella silhouette dipenda dal colore del galleggiante nello sfondo, e come questi buchi possano anche provocare (caso più a destra) un peggioramento della segmentazione del soggetto dopo l'applicazione del regionGrowing.*

## **4.2 Algoritmi con aggiornamento automatico del classificatore**

Nel precedente capitolo si è mostrato come l'algoritmo  $NBC_{AA}$  sia quello presentante la maggior dispersione del parametro

*%background\_corretti* sia prima che dopo l'applicazione del `regionGrowing`, unitamente a un valore di mediana nettamente più basso. Si è inoltre correlata la distribuzione di questo parametro all'effetto osservato di rapida crescita del "rumore" nelle immagini segmentate (Figura 3.1), ossia di rapida crescita del numero di FF a partire dal fotogramma in cui inizia l'aggiornamento del modello. Il motivo di questo comportamento è da ricercarsi nella procedura di aggiornamento automatico stessa. Ogni fotogramma segmentato di volta in volta, infatti, va a costituire il training set che istruisce il classificatore per la segmentazione successiva. Questo implica che anche solo pochi pixel FF (*false foreground*) nella prima immagine, presi come corretti dal classificatore, sono potenzialmente in grado di portare a una rapida crescita del rumore nelle segmentazioni successive, perché di volta in volta l'errore sulla classificazione dei FF si propaga amplificandosi. Essendo questo effetto legato alla procedura stessa di aggiornamento automatico, si riscontra anche nel  $NBC_{AA,B}$  (che è la versione del  $NBC_{AA}$  applicato a immagini già divise per il background). In questo caso tuttavia l'effetto di amplificazione dell'errore è limitato perché, grazie alla preliminare divisione, c'è più contrasto tra soggetto e sfondo, ossia c'è più distanza tra le nuvole iniziali di punti del background e del foreground (e quindi tra le gaussiane che modellano le distribuzioni di colore nelle classi). Nel caso di  $NBC_{AA,B}$ , se il numero di punti FF all'inizio è limitato, le gaussiane che modellano la distribuzione di colore della classe foreground non si "estendono" significativamente (cioè non aumentano significativamente i valori di deviazioni standard  $\sigma$  delle tre componenti di colore), e l'effetto di propagazione dell'errore rimane limitato.

In entrambi gli algoritmi considerati si nota ancora una buona distribuzione della *%foreground\_corretti*, ma occorre ricordare che ciò è da imputarsi al fatto che in realtà gli algoritmi classificano come soggetto "più pixel del dovuto".

Un'eventuale evoluzione di questi algoritmi potrebbe consistere nell'applicazione di un algoritmo di miglioramento della

segmentazione (come il `regionGrowing`) non a segmentazione ultimata bensì a ogni immagine segmentata, in modo da fornire come nuovo training set ogni volta un'immagine "ripulita" dei pixel FF, che sono i responsabili dell'amplificazione del rumore.

### 4.3 Efficacia dell'algoritmo `regionGrowing`

Per gli algoritmi `SDB`, `NBC`, `NBCAG` e `NBCAA` l'applicazione del `regionGrowing` produce un peggioramento per la distribuzione di `%foreground_corretti`. In particolare poi, per gli algoritmi `NBC`, `NBCAG` e `NBCAA` compaiono nel post-`regionGrowing` anche fotogrammi in cui la `%foreground_corretti` si annulla. Questo comportamento osservato è da imputarsi sia alla capacità dell'algoritmo considerato di riconoscere il soggetto, sia alla particolare implementazione del `regionGrowing`. In effetti, si può verificare che laddove si è presentato il problema il `regionGrowing` non ha funzionato correttamente perché:

- o la selezione automatica del punto iniziale, pur rispettando i requisiti richiesti (distanza dal bordo dell'immagine > 2 pixel), è scorretta poiché il punto non ricade all'interno della silhouette del soggetto
- oppure, se il punto iniziale viene scelto troppo vicino al bordo dell'immagine, la procedura alternativa di miglioramento della segmentazione fallisce perché viene selezionata e riempita un'area connessa diversa dalla silhouette

In entrambi i casi, il motivo del fallimento dipende dal fatto che la silhouette, verosimilmente, non è la regione connessa bianca più estesa nell'immagine in esame. Questo significa che l'algoritmo di segmentazione preliminarmente impiegato ha lasciato troppi "buchi" nella silhouette (numero eccessivo di FB, *false background*), oppure che troppi pixel dello sfondo sono stati erroneamente classificati come foreground (troppi FF, *false foreground*), creando macchie bianche nell'immagine più estese della silhouette stessa. Partendo dunque da una selezione errata del punto centrale (o della regione connessa più ampia), l'algoritmo di miglioramento va a individuare e a "riempire"

una regione dell'immagine diversa dalla silhouette, il che provoca l'annullamento del parametro *%foreground\_corretti*.

In Figura 4.2 portiamo un esempio relativo a questo caso. Nel caso specifico riportato in figura, la segmentazione finale risulta scorretta perché fallisce la procedura alternativa di miglioramento della segmentazione. In questo caso, infatti, il punto più distante dai pixel neri, scelto come punto iniziale, è il punto nell'angolo in alto a sinistra dell'immagine; essendo questo pixel troppo vicino al bordo, non viene applicato l'algoritmo *regionGrowing* ma la procedura alternativa. A questo punto, si giunge a un fallimento della segmentazione perché non è il soggetto l'area bianca più estesa.



*Figura 4.2 Immagine originale (sinistra), immagine segmentata dall'algoritmo NBC pre-regionGrowing (centro) e post-regionGrowing (destra)*

Eventualmente, se i fotogrammi per i quali si presenta il problema sono relativamente pochi, un intervento manuale dell'utente per l'identificazione del punto iniziale del *regionGrowing* può migliorare il risultato finale.

Osservando nel complesso i risultati ottenuti, è possibile constatare che l'algoritmo *regionGrowing* non porta tanto a un miglioramento della *%foreground\_corretti* (dove si ha invece o un peggioramento o, nei casi migliori, una distribuzione che rimane invariata), quanto più porta a un'evoluzione positiva del parametro *%background\_corretti*. Si può dunque concludere che per ottenere una buona qualità finale della segmentazione occorre partire da un algoritmo che già sia in grado di riconoscere bene i pixel del foreground.

Anche se non è in grado di produrre un grosso miglioramento della *%foreground\_corretti*, l'applicazione dell'algoritmo *regionGrowing* è

comunque in definitiva necessaria. Questo perché nelle fasi successive delle elaborazioni, quando si procederà alla creazione del visual hull, è necessario avere a disposizione immagini in cui l'unica regione bianca connessa sia quella della silhouette, la quale deve presentare nelle diverse viste una morfologia coerente (non devono essere ad esempio presenti piccoli falsi buchi neri dovuti a un cattivo riconoscimento del soggetto).

#### 4.4 Il risultato migliore

Dato che, come detto nella sezione 4.3, è comunque necessario come ultimo passaggio applicare un algoritmo di miglioramento, per determinare quale tra gli algoritmi implementati produca la segmentazione migliore considereremo i risultati post-regionGrowing. Osservando questi risultati, emerge che tutti gli algoritmi tranne il  $NBC_{AA}$  hanno buone distribuzioni di *%background\_corretti*. Per quanto riguarda, invece, gli algoritmi che presentano le migliori distribuzioni di *%foreground\_corretti*, ritroviamo innanzitutto il  $NBC_{AA,B}$  e in secondo luogo il  $NBC_{AG,B}$ . La scelta dell'algoritmo migliore ricadrà dunque tra questi ultimi due. Più in dettaglio, il primo,  $NBC_{AG,B}$ , presenta una *%background\_corretti* migliore del secondo, mentre il secondo,  $NBC_{AA,B}$ , una migliore *%foreground\_corretti*, ma una *%background\_corretti* tra le peggiori relativamente agli altri algoritmi. Riportiamo qui per comodità i dati relativi a *%background\_corretti* e *%foreground\_corretti* nei due algoritmi, nel caso post-regionGrowing.

<b><math>NBC_{AG,B}</math> post-regionGrowing:</b>	
<b>%background_corretti</b>	
m= 99.41%	UB= 99.84%
	LB= 98.87%
<b>%foreground_corretti</b>	
m= 97.05%	UB= 98.78%
	LB= 91.98%

<b><math>NBC_{AA,B}</math> post-regionGrowing:</b>	
<b>%background_corretti</b>	
m= 98.57%	UB= 99.54%
	LB= 96.26%
<b>%foreground_corretti</b>	
m= 98.54%	UB= 99.65%
	LB= 95.05%

Anche se il deficit in termini di *%background\_corretti* del secondo algoritmo rispetto al primo sembra non essere significativo, in realtà occorre ricordare che:

- il numero di punti del background è mediamente molto più elevato di quello del soggetto, quindi una differenza di un punto percentuale nel caso di *%background\_corretti* corrisponde a più pixel che nel caso *%foreground\_corretti*
- in realtà, per una buona ricostruzione del visual hull occorre anche una certa precisione sui bordi della silhouette, in modo da produrre un visual hull coerente nel momento in cui si retroproiettano tutte le sagome, e con meno *phantom volumes*

Per spiegare meglio quanto appena espresso, riportiamo a titolo di esempio per ognuno dei due algoritmi un'immagine segmentata per ogni telecamera nella Figura 4.3, per permettere un confronto di tipo più qualitativo. Come visibile in Figura 4.3, l'algoritmo NBC<sub>AA,B</sub> riconosce bene tutti i veri punti del foreground (TF), ma allo stesso tempo aggiunge alla silhouette diversi FF, ossia pixel del background erroneamente classificati come foreground. In altre parole, si potrebbe dire che uno dei motivi per cui la *%foreground\_corretti* è così elevata è anche che in realtà vengono riconosciuti come foreground “più pixel del dovuto”, perdendo così spesso la precisione dei contorni. In particolare, questi pixel FF si concentrano principalmente in due punti: da un lato si creano delle estensioni del soggetto a pelo dell'acqua, in corrispondenza del riflesso, dall'altro vengono spesso riconosciute come foreground anche le ombre sulla pavimentazione della piscina che si trovano in continuità con il piede (e che in questo modo non riescono ad essere eliminate dal regionGrowing). Ora, mentre le estensioni del soggetto a pelo d'acqua sono un problema che comunque verrebbe risolto in fasi successive delle elaborazioni (dal momento che comunque sia occorrerà anche una fase in cui si elimina il riflesso ribaltato a pelo d'acqua), la rimozione delle ombre ai piedi del soggetto erroneamente riconosciute come foreground necessita di un ulteriore specifico step, utilizzando ad esempio uno degli algoritmi proposti in letteratura. Questi algoritmi si possono in generale

suddividere tra quelli agenti sulla base di una preliminare osservazione/modellazione delle caratteristiche delle ombre stesse (come ad esempio [41], [42], [43], [44], [45]) e quelli che effettuano una rimozione a posteriori, una volta ottenute le segmentazioni del foreground, ombre eventualmente comprese (come ad esempio [46]).

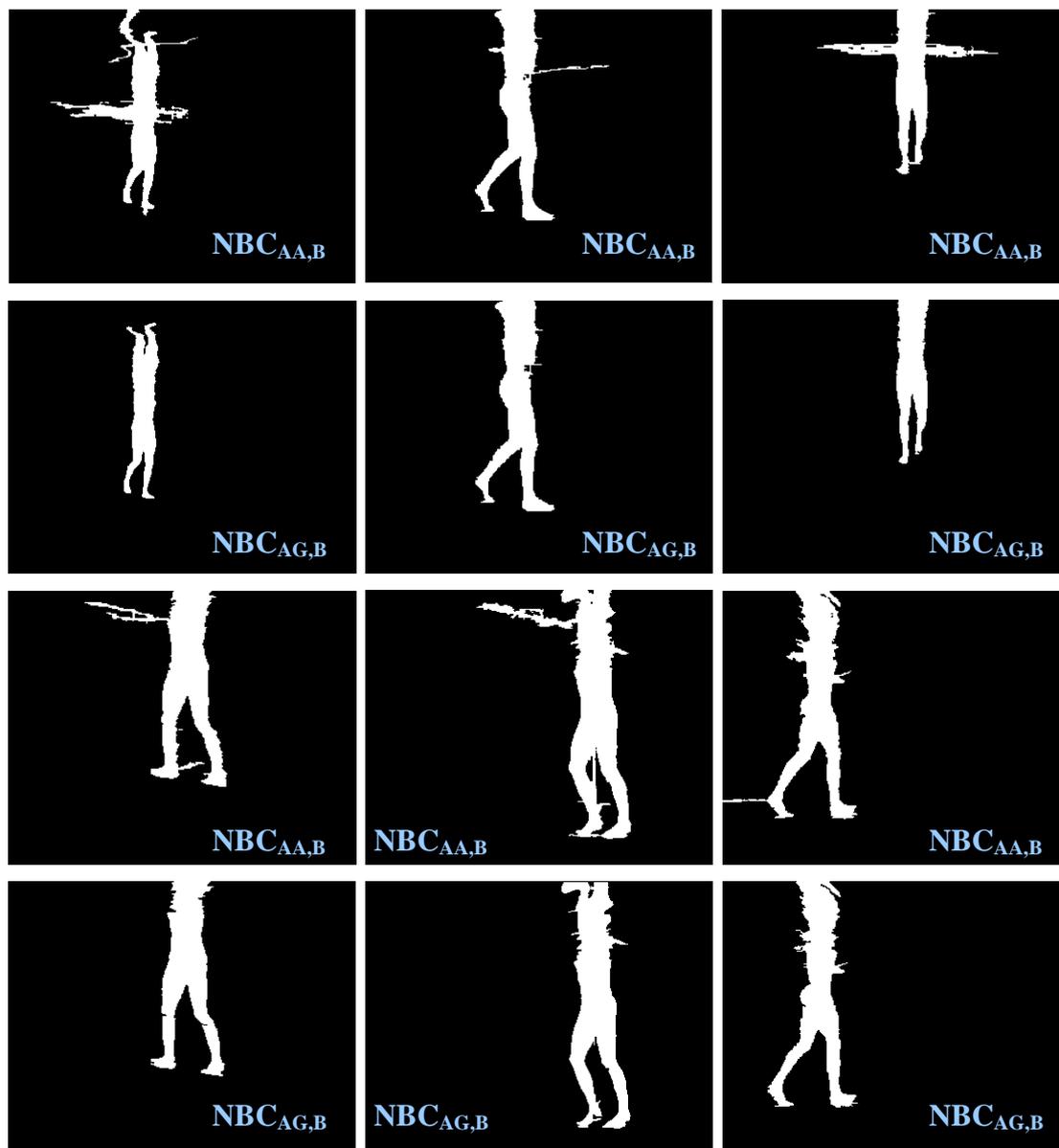


Figura 4.3 Righe 1 e 3: algoritmo  $NBC_{AA,B}$  *post-regionGrowing* Righe 2 e 4: corrispondenti immagini con l'algoritmo  $NBC_{AG,B}$  *post-regionGrowing*

Si ritiene ad ogni modo preferibile un algoritmo come il  $NBC_{AG,B}$  che, nonostante presenti una buona  $\%foreground\_corretti$ , è anche più selettivo del  $NBC_{AA,B}$  e produce un numero inferiore di FF. Un ulteriore motivo a favore della scelta di quest'ultimo algoritmo deriva

inoltre dall'osservazione per cui i punti FF aggiunti dall'algoritmo  $NBC_{AA,B}$  alla silhouette (che rendono questa "abbondante" e con i bordi piuttosto "frastagliati") vengono aggiunti in misura del fatto che l'algoritmo riconosce parti del background circostanti il soggetto come foreground. A seconda, dunque, delle caratteristiche locali del background in prossimità del soggetto saranno aggiunti più o meno FF, in maniera variabile e non facilmente prevedibile. In altre parole, oltre ad aggiungere punti FF alla silhouette, non vi sarà nemmeno coerenza e continuità sul riconoscimento dei bordi della silhouette né temporale (frame dopo frame) né tra le diverse telecamere nello stesso istante.

La considerazione appena espressa sulla presenza di numerosi FF nel  $NBC_{AA,B}$  appare anche osservando la distribuzione dei valori della distanza di Hausdorff tra i contorni delle silhouette segmentate con i due algoritmi e i corrispondenti contorni nelle immagini gold-standard. Il valore di mediana della distanza di Hausdorff ottenuta con l'algoritmo  $NBC_{AA,B}$  è infatti circa 3.46 volte quella ottenuta con  $NBC_{AG,B}$  (analogamente, il rapporto tra le mediane delle MAD è 3.70 circa).

Un ulteriore aspetto a favore dell'algoritmo  $NBC_{AG,B}$  in confronto al  $NBC_{AA,B}$  sono i minori tempi di computazione. In effetti, mentre nel  $NBC_{AA,B}$  l'aggiornamento automatico obbliga a una segmentazione sequenziale frame dopo frame (dato che ogni fotogramma segmentato è il training set del fotogramma successivo), nel  $NBC_{AG,B}$ , una volta valutate le caratteristiche del classificatore nella prima e nella seconda immagine di training, è possibile calcolare a priori quali valori dovranno assumere i parametri del classificatore, in funzione del numero del fotogramma. In altri termini, con il secondo algoritmo non si è obbligati a una segmentazione sequenziale, ma è possibile trattare ogni fotogramma indipendentemente dagli altri. Questo permette, ad esempio, di utilizzare il *parallel computing* (esecuzione simultanea dell'algoritmo su più microprocessori o su più core dello stesso processore), riducendo i tempi computazionali.

Si ritiene dunque infine il migliore, tra gli algoritmi implementati, il Naive Bayes Classifier con aggiornamento guidato dei parametri (e

andamento dei parametri supposto lineare), applicato a immagini già divise per il background.

#### **4.5 Contestualizzazione del presente lavoro di tesi**

Se guardiamo alla letteratura, relativamente pochi studi sono stati effettuati per l'analisi e caratterizzazione del cammino in acqua, e gli studi sono ancora meno se consideriamo quelli effettuati utilizzando la tecnica markerless. La maggior parte dei lavori presenti, qualora includano anche la valutazione della cinematica dei segmenti ossei e non solo informazioni di altra natura (come ad esempio i pattern di attivazione muscolare via EMG, l'andamento delle forze di reazione piede-suolo, la frequenza cardiaca durante il cammino ecc) sono stati effettuati con tecniche basate sull'utilizzo di marker (come ad esempio [47], [48], [49], [50]).

Tra i pochi lavori di analisi del cammino in acqua con tecnica markerless ricordiamo quello di A. Mantoan [3] nel quale si impiega come tecnica di segmentazione del soggetto una modellazione del background basata sul metodo di combinazione di gaussiane (*gaussian mixture*) ideato da Grimson et al. [51]. In particolare, in [3] vengono utilizzate per la modellazione iniziale del background le riprese della piscina vuota dalle diverse telecamere. Il metodo basato sulla combinazione di gaussiane [51] sfrutta un approccio probabilistico, e valuta indipendentemente la distribuzione di colore di ogni pixel. In particolare, per ognuno dei pixel si considera tutta la storia passata in termini di livelli di colore RGB precedentemente assunti. A questo punto, l'obiettivo è fornire un modello probabilistico per i valori dei livelli di colore RGB per ogni singolo pixel del background. Il modello dei dati assunto per ogni pixel è quello di una media pesata di gaussiane dove ognuna avrà un peso  $w_i$  e una deviazione standard  $\sigma_i$  in base alla persistenza in quel pixel di una certa combinazione di livelli RGB. La modellazione avviene in modo non supervisionato. Le gaussiane di tutta l'immagine vengono ordinate ad ogni fotogramma sulla base del parametro  $w_i/\sigma_i^2$  e si assumono come gaussiane

modellanti il background solo quelle presentanti un valore del parametro  $w_i/\sigma_i^2$  superiore a una soglia prestabilita. Nel momento in cui si analizza un nuovo fotogramma, per ogni pixel vengono confrontati i livelli RGB da analizzare con il modello di colore del pixel stesso ottenuto fino a quel momento; solo i pixel i cui livelli di colore RGB siano compresi entro  $2\sigma$  dalla media di ogni gaussiana modellante il background in quel pixel saranno classificati come tali, altrimenti il pixel sarà classificato come foreground. In questo modo concorrono inoltre all'aggiornamento del modello del background (in termini di aggiornamento delle  $K$  gaussiane modellanti ognuno dei pixel) solo selettivamente i pixel di volta in volta classificati come background. Il vantaggio di un modello del background di questo tipo consiste nel riuscire a tenere conto della non staticità dello sfondo, ossia nel nostro caso modellare meglio i riflessi di luce variabili a pelo dell'acqua e i corrispondenti riflessi sul fondo della piscina. Questo algoritmo è in grado di fornire una buona classificazione sia del background e quindi, per esclusione, anche del foreground. Rimane tuttavia il problema del riflesso ribaltato del soggetto a pelo d'acqua e l'inclusione delle ombre del soggetto sulla pavimentazione, entrambi significativamente diversi dal modello dello sfondo di un generico pixel. Anche se non utilizzato in [3], per quanto riguarda il problema delle ombre una versione leggermente modificata dell'algoritmo appena citato, ad opera di Kaewtrakulpong e Bowden [28] propone una soluzione. Volendo proporre un confronto qualitativo (non avendo a disposizione dati quantitativi) con questo algoritmo, considerando il risultato migliore nel presente lavoro di tesi ( $NBC_{AG,B}$ ) si potrebbe dire di essere riusciti comunque ad ottenere una buona segmentazione sia del background che del foreground (mediane di entrambi i parametri  $\%background\_corretti$  e  $\%foreground\_corretti > 96\%$ ) pur utilizzando per lo sfondo un modello meno complesso (semplice media temporale tra tutti i fotogrammi dei video contenenti le riprese del solo background). Vale inoltre lo stesso discorso già introdotto sui tempi computazionali: laddove l'aggiornamento del modello sia vincolato alla sequenzialità dei fotogrammi (come in questo caso) non è

possibile sfruttare strumenti come il parallel computing per velocizzare la computazione. Rimangono tuttavia anche nel nostro caso, per il  $NBC_{AG,B}$ , i problemi del riflesso ribaltato a pelo d'acqua e delle ombre del soggetto sulla piscina.

Altri approcci per la modellazione del background sono stati proposti, come ad esempio la modellazione con gaussiane singole delle distribuzioni di colore di ogni pixel sulla base dei valori RGB degli ultimi  $n$  fotogrammi [52], oppure ancora la sottrazione del background dove il modello del background subisce fotogramma dopo fotogramma un'evoluzione assegnando a ogni pixel del modello il valore di mediana ottenuto tra i valori RGB di quel pixel negli ultimi  $m$  fotogrammi [53]. Queste tecniche hanno il vantaggio di poter essere utilizzate anche non avendo a disposizione un video con le acquisizioni del solo background, tuttavia sarebbero probabilmente inefficaci nelle viste frontali e posteriori, in cui il soggetto rimane nella parte centrale dello schermo per tutto il tempo.

Un altro approccio per la segmentazione del soggetto è quello proposto da Laurentini et al. [54], che a sua volta trae spunto da Yamada et al. [55]. Anche in questo caso si sfrutta un modello iniziale del background, ottenibile da un video contenente il solo sfondo di circa 200 fotogrammi. Al termine della modellazione, ad ogni pixel sarà associato un indice di posizione e un indice di dispersione della distribuzione di colore del background in quel pixel. A questo punto, si esegue un confronto tra i fotogrammi delle acquisizioni e il modello del background tramite una sottrazione pixel a pixel. Tuttavia, invece di lavorare nello spazio RGB si esegue una conversione allo spazio YUV. I pixel vengono dunque sottoposti a sogliatura in base alla distanza nello spazio YUV. Il processo di sogliatura proposto è di tipo adattivo, per cui non si sceglie per tutti i pixel un unico valore di soglia ma la soglia viene determinata per ogni pixel sulla base di un determinato criterio. Il vantaggio di passare dallo spazio RGB allo spazio YUV risiede nel fatto che nel secondo tipo di codifica la prima componente (Y) codifica la luminosità del pixel, mentre la cromaticità è associata alle componenti U e V. Osservando che le ombre prodotte

dal background hanno lo stesso colore ma diversa luminosità rispetto allo sfondo, l'unica differenza tra un pixel del background senza e uno con un'ombra risiederà nella componente  $Y$ . E' in questo modo possibile distinguere le ombre dal vero soggetto. Anche se ancora non ci sono prove dell'efficacia di questo algoritmo nell'ambito applicativo dell'analisi del cammino in acqua, si può ipotizzare che un algoritmo di questo tipo possa produrre risultati anche migliori rispetto agli algoritmi studiati in questo lavoro di tesi, anche per la sua capacità di riconoscere le ombre come parte del background in un unico passaggio. Data la scarsità di studi specifici sul cammino in acqua con tecnica markerless non è semplice fornire un confronto oggettivo tra gli algoritmi implementati e caratterizzati in questo lavoro di tesi e quelli proposti in letteratura. Si può tuttavia in generale commentare che è stato utilizzato un tipo di approccio generalmente diverso: mentre nella maggior parte degli algoritmi descritti in letteratura ci si concentra infatti su come fornire una buona modellazione del background [53], al fine di classificare il foreground quasi "per esclusione", in questo lavoro di tesi (a parte per l'algoritmo  $SD_B$ ) si cerca di sfruttare una modellazione sia delle caratteristiche del background che del foreground. Uno dei motivi per cui in letteratura si può notare uno sforzo più propenso alla modellazione del background è probabilmente da ricercarsi nel fatto che nella maggior parte dei casi si conosce quali siano le caratteristiche del background, ma non a priori quali saranno quelle del foreground (esempio: tracking di autovetture nel traffico urbano, sistemi di sorveglianza ecc). Nel particolare caso di acquisizione del cammino di un particolare soggetto, invece, si possono sfruttare le conoscenze in più che si hanno a priori sul soggetto per ottenere segmentazioni migliori. Ci si può inoltre in questo modo "permettere" di utilizzare un modello per il background eventualmente più approssimato (come ad esempio la media temporale su tutti i fotogrammi), e quindi dal costo computazionale più basso. Anche il numero di gaussiane necessarie per descrivere le caratteristiche del background e del foreground è inferiore, e in particolare se si assume anche l'indipendenza statistica delle

componenti di colore, sono sufficienti in totale 6 distribuzioni gaussiane (3 per il background e 3 per il foreground), invece che  $K$  distribuzioni gaussiane per ogni pixel. Un altro risultato raggiunto in questo lavoro di tesi consiste nella prova che una classificazione eseguita su immagini preliminarmente divise per il background porta a risultati significativamente migliori, grazie all'aumento di contrasto tra soggetto e sfondo. Una preliminare elaborazione delle immagini in questo senso può dunque portare a un notevole miglioramento delle performances della segmentazione. Infine, si è provato che un aggiornamento del classificatore (e quindi delle gaussiane che modellano le distribuzioni di colore nelle classi) è in grado di migliorare la segmentazione, a patto che l'aggiornamento sia guidato e non automatico. Nel caso si volesse in futuro implementare un aggiornamento automatico, occorre fare in modo di "ripulire" l'immagine ad ogni step con un algoritmo come il regionGrowing, prima di utilizzare l'immagine appena segmentata come training set del classificatore per l'immagine successiva. Un ulteriore miglioramento potrebbe inoltre essere forse raggiunto se si rinunciassero all'ipotesi di indipendenza statistica delle distribuzioni delle componenti di colore nelle classi.

Nessuno degli algoritmi implementati in questo lavoro di tesi risolve il problema della rimozione delle ombre e del riflesso a pelo dell'acqua. Per quanto riguarda l'eliminazione delle ombre si potrebbe pensare di utilizzare come ulteriore step l'applicazione di un algoritmo come il già citato [46]; alternativamente, si potrebbe in lavori futuri testare l'effetto di una classificazione delle immagini nello spazio YUV invece che nello spazio RGB.

Per quanto riguarda l'eliminazione del riflesso, si potrebbe pensare di implementare un algoritmo che agisca sulla base di considerazioni geometriche e di prospettiva, specifico per ognuna delle telecamere. In alternativa, si potrebbe testare l'effetto di una classificazione dell'immagine utilizzando tre classi invece di due, creando cioè una classe in più appositamente per il riflesso a pelo dell'acqua. Si potrebbe in alternativa pensare a una classificazione in due step: la

prima per discriminare il foreground (riflesso compreso) dal background nelle immagini, e la seconda classificazione per discriminare il riflesso dal soggetto vero.



# Conclusioni

Nel presente lavoro di tesi si sono sviluppati e caratterizzati sette algoritmi di segmentazione per l'analisi del cammino in acqua con tecnica markerless. In particolare, possiamo suddividere gli algoritmi implementati in due grandi categorie: il primo, ossia l'algoritmo di Sottrazione/Divisione del background e Sogliatura e gli altri sei algoritmi, che rappresentano tutte possibili variazioni sul classificatore bayesiano Naive Bayes Classifier semplice. In particolare, gli ultimi sei algoritmi si sono differenziati in base a:

- preliminare elaborazione delle immagini consistente nella divisione per il background
- aggiornamento o meno del classificatore

I risultati ottenuti indicano nell'algoritmo di Sottrazione/Divisione del background e Sogliatura uno tra quelli mostrandoti i risultati peggiori, con una qualità della segmentazione finale anche piuttosto variabile a seconda del fotogramma analizzato. L'algoritmo ha rappresentato in effetti una soluzione di primo tentativo, essendo sia la più semplice dal punto di vista computazionale e peraltro anche una delle più "antiche" proposte in letteratura. Per quanto riguarda gli algoritmi basati sul Naive Bayes Classifier, i risultati ottenuti si sono mostrati anche piuttosto differenti a seconda dell'algoritmo considerato, e in base a quanti e quali gradi di complessità venivano aggiunti di volta in volta. In particolare si è ottenuto che:

- i Naive Bayes Classifier che lavorano su immagini preliminarmente divise per il background forniscono risultati significativamente migliori (le differenze tra le distribuzioni sono risultate evidenti dagli esiti dei test statistici ANOVA)
- l'aggiornamento automatico del classificatore produce risultati significativamente diversi dall'aggiornamento guidato o dall'assenza di aggiornamento

Alla fine, il risultato migliore è stato quello ottenuto con l'algoritmo Naive Bayes Classifier, con aggiornamento guidato del classificatore (assumendo andamento lineare dei parametri) e applicato a immagini già divise per il background. In particolare, questo algoritmo ha prodotto immagini segmentate in cui sia la *%background\_corretti* che la *%foreground\_corretti* presentavano valori di mediana  $> 96\%$  prima dell'applicazione del regionGrowing e arrivavano a oltre il  $97\%$  dopo l'applicazione del regionGrowing. Infine, anche la variabilità nella qualità della segmentazione sulle diverse immagini è rimasta contenuta (il  $90\%$  delle immagini analizzate post-regionGrowing ha presentato una *%background\_corretti* compresa tra  $98.87\%$  e  $99.84\%$ , e una *%foreground\_corretti* compresa fra  $91.98\%$  e  $98.78\%$ ).

I problemi che rimangono irrisolti anche dopo l'utilizzo del migliore di questi algoritmi sono:

- la classificazione come foreground anche delle ombre prodotte dal soggetto sulla pavimentazione che rimangono in continuità con il piede
- la classificazione come foreground anche del riflesso ribaltato del soggetto a pelo dell'acqua

Per quanto riguarda il primo problema, la soluzione potrebbe consistere nell'applicazione di un algoritmo di rimozione delle ombre come quello proposto da Xiao et al. in [46]. In alternativa, eventuali lavori futuri potrebbero testare l'effetto di una classificazione nello spazio YUV piuttosto che nello spazio RGB.

Per quanto riguarda la rimozione dei riflessi a pelo dell'acqua, potrebbero essere eventualmente implementati algoritmi di post-processing delle immagini che rimuovono i riflessi sulla base di considerazioni geometriche e di prospettiva, essendo le viste sui background fisse. In alternativa, si potrebbe pensare o a un ulteriore step di classificazione del foreground segmentato tra soggetto vero e riflesso, oppure a una classificazione che da subito sfrutti tre classi invece di due, includendo eventualmente anche la classe "riflesso del soggetto a pelo dell'acqua".

Infine, si potrebbe eventualmente in futuro valutare se la rinuncia all'ipotesi di indipendenza statistica delle tre componenti di colore nelle classi comporta o meno un miglioramento complessivo della segmentazione.



# Bibliografia

- [1] A. Menache, *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann, 2000.
- [2] J. Bray, «Markerless Based Human Motion Capture: A Survey», *Vision and VR Group, Brunel University*, 2001.
- [3] A. Mantoan, «Underwater gait analysis: a markerless approach», 18-Apr-2011. [Online]. Available: <http://tesi.cab.unipd.it/28262/>. [Accessed: 11-Set-2012].
- [4] J. D. Shutler, M. S. Nixon, e C. J. Harris, «Statistical gait recognition via velocity moments», in *2000/018), IEE Colloquium on Visual Biometrics (Ref.No, 2000*, pagg 10/1 –10/5.
- [5] P. D. F. J. Perales, *Human Motion Analysis & Synthesis using Computer Vision and Graphics Techniques. State of Art and Applications*. \* . .
- [6] R. Baker, «The history of gait analysis before the advent of modern computers», *Gait & Posture*, vol 26, n° 3, pagg 331–342, Set 2007.
- [7] E. Muybridge, Sterling, e F. C. A. Institute, *Animal locomotion*. Published under the auspices of the University of Pennsylvania, 1887.
- [8] E. Muybridge, *The Human Figure in Motion*. Dover Publications, 1955.
- [9] G. Johansson, «Visual perception of biological motion and a model for its analysis», *Attention, Perception, & Psychophysics*, vol 14, n° 2, pagg 201–211, 1973.
- [10] M. Gleicher, «Animation from observation: Motion capture and motion editing», *SIGGRAPH Comput. Graph.*, vol 33, n° 4, pagg 51–54, Nov 1999.
- [11] N. Protto, «Applicazioni della Hierarchical Cluster Analysis nella valutazione biomeccanica del piede diabetico», 12-Dic-2011. [Online]. Available: <http://tesi.cab.unipd.it/37656/>. [Accessed: 27-Ott-2012].
- [12] F. H. Raab, E. B. Blood, T. O. Steiner, e H. R. Jones, «Magnetic Position and Orientation Tracking System», *IEEE Transactions on Aerospace and Electronic Systems*, vol AES-15, n° 5, pagg 709 –718, Set 1979.
- [13] S. Hashi, M. Toyoda, S. Yabukami, K. Ishiyama, Y. Okazaki, K. I. Arai, e H. Kanetaka, «Wireless magnetic motion capture system using multiple LC resonant

- magnetic markers with high accuracy», *Sensors and Actuators A: Physical*, vol 142, n° 2, pagg 520–527, Apr 2008.
- [14] L. Chiari e A. Cappello, *Microsistemi indossabili per l' Ambient Assisted Living*. 2008.
- [15] M. De Agostino, «I Sensori Inerziali di Basso Costo per la Navigazione Geodetica», Politecnico di Torino, 2009.
- [16] G. Guerra-Filho, «Optical motion capture: theory and implementation», *Journal of Theoretical and Applied Informatics*, vol 12, n° 2, pagg 61–89, 2005.
- [17] A. Cappello, A. Cappozzo, e P.E. Prampero, *Bioingegneria della Postura e del Movimento*. Patron, 2003.
- [18] V. Medved, *Measurement of human locomotion*. CRC, 2000.
- [19] L. Chiari, U. D. Croce, A. Leardini, e A. Cappozzo, «Human movement analysis using stereophotogrammetry: Part 2: Instrumental errors», *Gait & Posture*, vol 21, n° 2, pagg 197–211, Feb 2005.
- [20] M. Donati, «3-D reconstruction of the human skeleton during motion».
- [21] U. Della Croce, A. Leardini, L. Chiari, e A. Cappozzo, «Human movement analysis using stereophotogrammetry: Part 4: assessment of anatomical landmark misplacement and its effects on joint kinematics», *Gait & Posture*, vol 21, n° 2, pagg 226–237, Feb 2005.
- [22] A. Leardini, L. Chiari, U. Della Croce, A. Cappozzo, e others, «Human movement analysis using stereophotogrammetry. Part 3. Soft tissue artifact assessment and compensation.», *Gait & posture*, vol 21, n° 2, pag 212, 2005.
- [23] J. Fuller, L.-J. Liu, M. C. Murphy, e R. W. Mann, «A comparison of lower-extremity skeletal kinematics measured using skin- and pin-mounted markers», *Human Movement Science*, vol 16, n° 2–3, pagg 219–242, Apr 1997.
- [24] L. Mündermann, S. Corazza, e T. P. Andriacchi, «The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications», *Journal of NeuroEngineering and Rehabilitation*, vol 3, n° 1, pag 6, Mar 2006.
- [25] D. M. Gavrilu e L. S. Davis, «3-D model-based tracking of humans in action: a multi-view approach», in *Proceedings CVPR '96, 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1996*, 1996, pagg 73–80.
- [26] J. Jackman, *Bluescreen Compositing*. Burlington, USA: Focal Press, 2007.

- [27] S. Cheung e C. Kamath, «Robust techniques for background subtraction in urban traffic video», in *Visual Communications and Image Processing*, 2004, vol 5308, pagg 881–892.
- [28] P. KaewTraKulPong e R. Bowden, «An improved adaptive background mixture model for real-time tracking with shadow detection», in *Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems*, 2001, vol 25, pagg 1–5.
- [29] A. Laurentini, «The visual hull concept for silhouette-based image understanding», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 16, n° 2, pagg 150–162, Feb 1994.
- [30] S. Corazza, L. Mündermann, A. M. Chaudhari, T. Demattio, C. Cobelli, e T. P. Andriacchi, «A markerless motion capture system to study musculoskeletal biomechanics: Visual hull and simulated annealing approach», *Annals of Biomedical Engineering*, vol 34, n° 6, pagg 1019–1029, 2006.
- [31] L. Mündermann, S. Corazza, A. M. Chaudhari, E. J. Alexander, e T. P. Andriacchi, «Most favorable camera configuration for a shape-from-silhouette markerless motion capture system for biomechanical analysis», pagg 278–287, Gen 2005.
- [32] S. Corazza, E. Gambaretto, L. Mündermann, e T. P. Andriacchi, «Automatic Generation of a Subject-Specific Model for Accurate Markerless Motion Capture and Biomechanical Applications», *IEEE Transactions on Biomedical Engineering*, vol 57, n° 4, pagg 806–812, Apr 2010.
- [33] L. Mündermann, S. Corazza, e T. P. Andriacchi, «Accurately measuring human movement using articulated ICP with soft-joint constraints and a repository of articulated models», in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07, 2007*, pagg 1–6.
- [34] L. Mündermann, A. Mündermann, A. M. Chaudhari, e T. P. Andriacchi, «Conditions that influence the accuracy of anthropometric parameter estimation for human body segments using shape-from-silhouette», pagg 268–277, Gen 2005.
- [35] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, e T. P. Andriacchi, «Markerless Motion Capture through Visual Hull, Articulated ICP and Subject Specific Model Generation», *International Journal of Computer Vision*, vol 87, n° 1–2, pagg 156–169, Set 2009.

- [36] J. T. Woolfenden, «Aquatic physical therapy approaches for the extremities», *Orthopedic physical therapy clinics of North America: aquatic therapy*. Cirullo JA (Eds). Philadelphia: WB Saunders. p, vol 209, 1994.
- [37] R. Harrison, M. Hillman, e S. Bulstrode, «Loading of the Lower Limb when Walking Partially Immersed: Implications for Clinical Practice», *Physiotherapy*, vol 78, n° 3, pagg 164–166, Mar 1992.
- [38] R. A. Lotufo e E. R. Dougherty, *Hands-On Morphological Image Processing*. SPIE Press, 2003.
- [39] B. A. Rosner, *Fundamentals of Biostatistics*. Cengage Learning, 2011.
- [40] L. Tersi, «Methodological improvement of 3D fluoroscopic analysis for the robust quantification of 3D kinematics of human joints», PhD Thesis, Alma Mater Studiorum - Università degli studi di Bologna, 2011.
- [41] T. Tanaka, A. Shimada, D. Arita, e R. Taniguchi, «Non-parametric Background and Shadow Modeling for Object Detection», in *Computer Vision – ACCV 2007*, Y. Yagi, S. B. Kang, I. S. Kweon, e H. Zha, Cur Springer Berlin Heidelberg, 2007, pagg 159–168.
- [42] A. Bevilacqua e M. Roffilli, «Robust denoising and moving shadows detection in traffic scenes», in *IEEE CVPR*, 2001, pagg 1–4.
- [43] S. Nadimi e B. Bhanu, «Physical models for moving shadow and object detection in video», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 26, n° 8, pagg 1079 –1087, Ago 2004.
- [44] O. Schreer, I. Feldmann, U. Golz, e P. Kauff, «Fast and robust shadow detection in videoconference applications», in *Video/Image Processing and Multimedia Communications 4th EURASIP-IEEE Region 8 International Symposium on VIPromCom*, 2002, pagg 371 – 375.
- [45] G. S. K. Fung, N. H. C. Yung, G. K. H. Pang, e A. H. S. Lai, «Effective moving cast shadow detection for monocular color image sequences», in *11th International Conference on Image Analysis and Processing, 2001. Proceedings*, 2001, pagg 404 –409.
- [46] M. Xiao, C.-Z. Han, e L. Zhang, «Moving shadow detection and removal for traffic sequences», *Int J Automat Comput*, vol 4, n° 1, pagg 38–46, Gen 2007.
- [47] A. M. F. Barela, S. F. Stolf, e M. Duarte, «Biomechanical characteristics of adults walking in shallow water and on land», *Journal of Electromyography and Kinesiology*, vol 16, n° 3, pagg 250–256, Giu 2006.

- [48] A. M. F. Barela e M. Duarte, «Biomechanical characteristics of elderly individuals walking on land and in water», *Journal of Electromyography and Kinesiology*, vol 18, n° 3, pagg 446–454, Giu 2008.
- [49] T. Miyoshi, T. Shirota, S.-I. Yamamoto, K. Nakazawa, e M. Akai, «Effect of the walking speed to the lower limb joint angular displacements, joint moments and ground reaction forces during walking in water», *Disability & Rehabilitation*, vol 26, n° 12, pagg 724–732, Gen 2004.
- [50] L. C. Carneiro, S. M. Michaelsen, H. Roesler, A. Hauptenthal, M. Hubert, e E. Mallmann, «Vertical reaction forces and kinematics of backward walking underwater», *Gait Posture*, vol 35, n° 2, pagg 225–230, Feb 2012.
- [51] C. Stauffer e W. E. L. Grimson, «Adaptive background mixture models for real-time tracking», in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, 1999, vol 2, pag 2 vol. (xxiii+637+663).
- [52] C. R. Wren, A. Azarbayejani, T. Darrell, e A. P. Pentland, «Pfinder: real-time tracking of the human body», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 19, n° 7, pagg 780 –785, Lug 1997.
- [53] R. Cucchiara, C. Grana, M. Piccardi, e A. Prati, «Detecting moving objects, ghosts, and shadows in video streams», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 25, n° 10, pagg 1337 – 1342, Ott 2003.
- [54] A. Bottino e A. Laurentini, «A Silhouette Based Technique for the Reconstruction of Human Movement», *Computer Vision and Image Understanding*, vol 83, n° 1, pagg 79–95, Lug 2001.
- [55] M. Yamada, K. Ebihara, e J. Ohya, «A new robust real-time method for extracting human silhouettes from color images», in *Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998. Proceedings*, 1998, pagg 528 –533.