
Università degli studi di Padova

Facoltà di Ingegneria

Tesi di Laurea Magistrale in

INGEGNERIA INFORMATICA

# Costruzione di modelli 3D a colori con videocamere e sensori a tempo di volo.

Relatore

Prof. Pietro Zanuttigh

Correlatore

Ing. Carlo Dal Mutto

Correlatore

Prof. Guido Maria Cortelazzo

Candidato

Enrico Cappelletto

Anno Accademico 2011/2012

*Alla mia Famiglia che*
*mi ha sempre sostenuto e*
*agli Amici che mi hanno*
*accompagnato nella vita.*

# Sommario

La tesi tratta la ricostruzione di modelli tridimensionali a colori a partire dai dati acquisiti da un sensore a tempo di volo e da una videocamera.

La scelta dei sensori è atta a garantire la possibilità di riprendere oggetti in tempo reale o scene dinamiche che cambiano velocemente, e quindi richiedono una frequenza di acquisizione molto alta, nell'ordine di 20, 30 fps(frame per secondo).

Utilizzando il solo sensore a tempo di volo si può ricostruire la geometria della scena, ma riuscendo a fondere anche le informazioni di colore è possibile rendere la ricostruzione molto più significativa e apprezzabile aprendo nuove frontiere nell'utilizzo di questa tecnologia mista.

Il contributo principale di questa tesi è lo studio e la realizzazione software di un un sistema che permetta l'acquisizione dei dati dai sensori, la loro fusione e la ricostruzione del modello 3D a colori.

La tesi è stata suddivisa in 6 capitoli:

Il capitolo 1 descrive alcuni scenari di utilizzo di questa tecnologia, evidenziando come l'inserimento dell'informazione di colore in una ricostruzione tridimensionale permetterà in un futuro applicazioni nuove per l'esperienza offerta e fruibili anche da semplici utenti. Vengono descritti i problemi principali da affrontare per fondere le informazioni e per costruire il modello.

Il capitolo 2 introduce la tecnologia su cui si basano i dispositivi a tempo a di volo, evidenziandone pregi e i difetti. Inoltre viene presentata la fase di acquisizione dei dati di profondità, l'utilizzo che si può farne e il postprocessing necessario per ridurre il rumore dei dati.

Nel capitolo 3 viene presentato il sistema di acquisizione misto composto da sensore ToF e videocamera analizzando il modo per fondere correttamente l'informazione della geometria e del colore. Partendo dalla considerazione generale che la videocamera e il sensore a tempo di volo avranno un punto di vista della scena differente viene descritto il meccanismo di gestione delle parti occluse.

Il capitolo 4 descrive il problema della registrazione di viste consecutive risolto con l'utilizzo dell'algoritmo Iterative-Closest-Point(ICP). Verrà spiegato il metodo con la quale vengono estratte le coppie di punti corrispondenti necessarie per l'allineamento di due viste, e introdotto un nuovo meccanismo di estrazione dei punti rilevanti.

Il capitolo 5 presenta i risultati sperimentali sui test dei diversi moduli implementati e sui modelli tridimensionali generati. Inoltre verranno analizzati alcuni casi limite.

Il capitolo 6 ,infine, contiene le conclusioni di questo lavoro e una panoramica sui fronti di ricerca rimasti aperti per eventuali futuri miglioramenti.

L'appendice A contiene la descrizione a grandi blocchi dell'implementazione del tool di acquisizione ed elaborazione implementati durante la tesi e descrive l'interfaccia grafica.

# Indice


**1 Introduzione** **1**
- 1.1 Contesto applicativo . . . 2
- 1.2 Descrizione del problema . . . 2
- 1.3 Descrizione dell'architettura del software . . . 3

**2 Sensori matriciali Time-Of-Flight(ToF)** **5**
- 2.1 La tecnologia ToF . . . 5
- 2.2 Limiti e vantaggi della tecnologia TOF . . . 6
  - 2.2.1 Ventaglio di tecnologie alternative . . . 8
- 2.3 Sistema di acquisizione ToF . . . 10
- 2.4 Post-processing dei dati acquisiti . . . 11
  - 2.4.1 Rimozione dello sfondo . . . 11
  - 2.4.2 Sfruttare mappa confidenza e ampiezza . . . 12
  - 2.4.3 Calcolo della curvatura . . . 12
  - 2.4.4 Filtro bilaterale . . . 13

**3 Acquisizione combinata di forma e colore** **15**
- 3.1 Acquisizione informazioni colore . . . 15
- 3.2 Fusione informazione geometria e colore . . . 17
- 3.3 Gestione delle occlusioni . . . 19
  - 3.3.1 Algoritmo z-buffer . . . 21
  - 3.3.2 Algoritmo Scanline per il filling di triangoli . . . 21

**4 Costruzione modello 3D a colori** **25**
- 4.1 Calcolo della rototraslazione ottima . . . 26
- 4.2 Allineamento sfruttando solo i punti rilevanti . . . 27
- 4.3 Merging delle viste . . . 32
- 4.4 Cleaning modello finale . . . 32
- 4.5 Rendering del modello . . . 34

**5 Risultati sperimentali** **35**
- 5.1 Fusione della geometria e del colore . . . 35
- 5.2 Estrazione dei punti salienti . . . 38
- 5.3 Algoritmo ICP . . . 41

5.4 Ricostruzione di un oggetto . . . . . 41

**6 Conclusioni** **47**

**A Descrizione del sistema software** **49**
A.1 Tool di acquisizione . . . . . 49
A.2 Tool di elaborazione . . . . . 52

# Capitolo 1

# Introduzione

Fin dall'antichità l'uomo ha avuto l'esigenza di mantenere ricostruzioni di scene di vita, ambienti o oggetti della realtà per conservarne la memoria nel tempo. La qualità delle ricostruzioni è aumentata con lo sviluppo tecnologico dell'uomo, che è passato dalle pitture rupestri dell'epoca preistorica, ai disegni sui papiri egizi fino ad arrivare agli inizi dell'ottocento all'invenzione della fotografia, con la quale si poteva finalmente ottenere delle registrazioni permanenti e statiche di un'immagine della scena reale, proiettata da un sistema ottico in uno strato fotosensibile, o negli ultimi decenni in una memoria digitale.

Nell'ultimo decennio l'evoluzione tecnologica ha portato a dispositivi che potranno rappresentare una base per una ulteriore evoluzione della fotografia, poiché invece di registrare una proiezione bidimensionale della realtà, acquisiscono la geometria 3D della realtà stessa. La ricostruzione della geometria da un'idea della scena, permette di cogliere il senso della profondità degli oggetti, la loro reale dimensione ma è solo con l'aggiunta dell'informazione del colore che la ricostruzione può essere veramente coinvolgente ed efficace [19]. Il colore permette di identificare meglio gli oggetti della scena aggiungendo un grado di dettaglio che la pura forma non può dare. Inoltre spesso il colore può determinare il dettaglio essenziale di una scena, ad esempio si pensi ad una tela/quadro, ad una cartina geografica o ad un paesaggio.

L'idea generale a cui far riferimento è quella di un sistema che permetta l'acquisizione in modo semplice e intuitivo come se si disponesse di una normale videocamera. Spostando il sistema di acquisizione o inquadrando una scena dinamica, si otterranno delle viste differenti della geometria e del colore, che il sistema elabora in tempo reale, fondendo via via le viste in un unico modello in bassa qualità per dare l'idea a chi sta acquisendo se ci sono problemi evidenti con l'illuminazione o la distanza, oltre ad avere un feedback su quali zone siano state o meno riprese (possibilità di riempire buchi). Terminata l'acquisizione, il sistema rielabora i dati e con una disponibilità

computazionale maggiore per raffinare la ricostruzione creando un modello 3D a colori con la miglior qualità possibile. Una volta ottenuto il modello digitale lo si può visualizzare su uno schermo, interagendo con esso varindo la posizione e l'orientazione del punto di vista all'interno della scena e infine lo si potrà salvare.

## 1.1 Contesto applicativo

I Campi di applicazione principali sono nell'ambito della video-sorveglianza, intrattenimento, costruzione di ambienti virtuali per intrattenimento e visite guidate, ma anche modellazione di oggetti reali come nuova alternativa alla fotografia bidimensionale, modellazione di stanze e ambienti per permettere visioni virtuali fatte in casa e nell'ambito dell'arte e istruzione(digitalizzazione di modelli 3d di statue, quadri,).

Tutte queste applicazioni erano possibili anche senza la fusione del colore, ma con questa aggiunta l'impatto su un utente sarà maggiore e il colore permetterà sicuramente di identificare meglio gli oggetti in una scena oltre a garantire una dose di realismo maggiore. Inoltre in questo modo potrebbe aumentare anche il pathos delle persone verso una ricostruzione in ambito consumer.

## 1.2 Descrizione del problema

Si elencano brevemente i problemi affrontati durante questo lavoro:

0. Gestione dei limiti intrinseci dei diversi sensori (le acquisizioni del ToF sono molto rumorose sulle zone di colore scuro o in punti ad alto grado di curvatura mentre la videocamera subisce il problema della riflesso sugli oggetti). Questi dati rumorosi non sempre sono affidabili per una ricostruzione precisa per cui devono essere gestiti.

1. Date le due viste di una stessa scena acquisite rispettivamente dal ToF e dalla videocamera il problema è come fondere le informazioni di colore e geometria in una singola vista.

2. Considerando una sequenza di n viste acquisite in istanti di tempo consecutivi è necessario registrarle in modo da ottenere un unico modello tridimensionale sfruttando al meglio le informazioni spesso ridondanti di colore e geometria per aumentare la qualità del modello.

3. Data la disponibilità dell'informazione di colore per ogni vista, cercare di sfruttarla non solo per aumentare la qualità del modello, ma anche per rendere più robusto l'algoritmo di registrazione.

## 1.3 Descrizione dell'architettura del software

In questo paragrafo sarà descritta la pipeline del sistema software implementato in questo lavoro di tesi. La natura di questo paragrafo è puramente introduttiva e ogni blocco che qui verrà solo illustrato brevemente troverà il suo spazio nei prossimi capitoli per esser analizzato e approfondito.

La pipeline delineata in questo lavoro si è ispirata alla 3D Model acquisition pipeline [12] che descrive la sequenza di operazioni di base che vanno dall'acquisizione dei dati 3D alla visualizzazione del modello corrispondente. A quella pipeline sono stati aggiunti diversi passi per realizzare la fusione nel modello dell'informazione di colore proveniente dalla camera.

Il sistema software è suddiviso in due parti distinte, un tool di acquisizione per salvare in tempo reale i frame catturati dalla camera e dal ToF e un tool per l'elaborazione offline dei dati.

Nell'interfaccia grafica del tool di acquisizione (Vedi appendice A) si possono impostare i parametri del ToF e delle videocamere collegate, avviare la scheda di sincronizzazione hardware e infine acquisire e salvare un dato numero di frame.


**Fig. 1.1:** Pipeline della fase di acquisizione

L'altra parte del sistema è il tool di elaborazione che permette di ricostruire offline il modello a partire dai frame della scena acquisiti in precedenza.

La figura 1.1 sottolinea due aspetti importanti della pipeline. Il primo riguarda il ruolo dell'utente nel programma: l'intervento dell'operatore è necessario solo nella fase di caricamento dei frame, e alla fine del processo quando potra interagire con il modello. Il secondo aspetto importante è che all'interno della pipeline c'è un grande ciclo di operazioni che viene eseguito per ogni frame: una prima parte del ciclo riceve come ingresso l'immagine a colori e i dati di profondità e si occupa di processare i dati geometrici restituendo la nuvola di punti 3D ripulita dai punti più romorosi , che nella seconda parte della pipeline vengono fusi con l'informazione di colore per ottenere una vista di punti 3D a colori pronta per esser aggiunta al modello che frame dopo frame prende forma ricostruendo la scena. Il frame dev'esse-

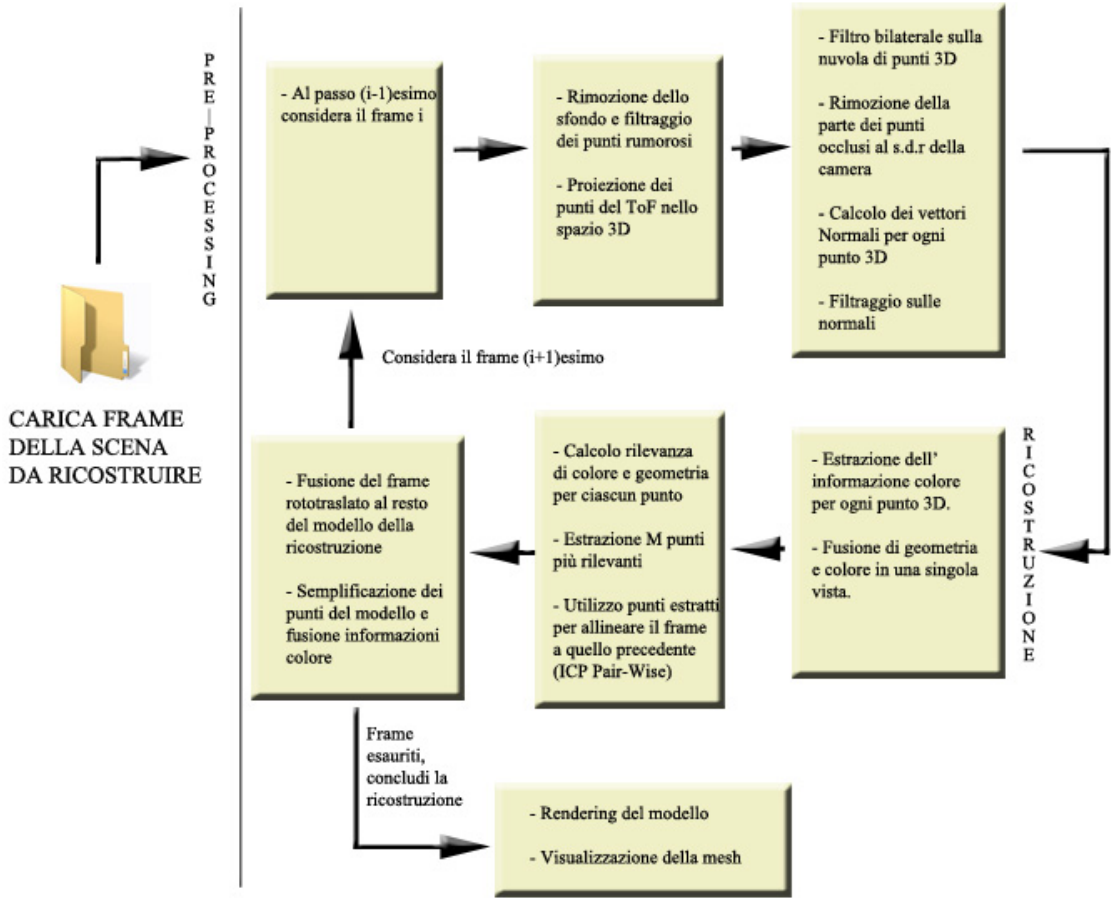



**Fig. 1.2:** Pipeline della fase di elaborazione dei dati offline

re aggiunto al modello in modo corretto, sovrapponendo le parti in comune il frame si integra perfettamente con il resto. Questa fase importantissima si chiama registrazione ed è direttamente collegata alla qualità del modello finale. Al termine del ciclo, le nuove informazioni sono state integrate al modello e la pipeline considera il frame successivo se è presente oppure termina l'elaborazione renderizzando il modello ottenuto e mostrandolo a video.

# Capitolo 2

# Sensori matriciali Time-Of-Flight(ToF)

## 2.1 La tecnologia ToF

I sistemi a tempo di volo si basano su un principio di funzionamento fondamentalmente semplice e simile a quello dei radar: stimano la profondità degli oggetti emettendo un segnale infrarosso sinusoidale e misurando il tempo intercorso tra l'istante della trasmissione del segnale e la sua ricezione. Più precisamente misurano la differenza di fase tra il segnale inviato e il segnale riflesso. Per la tesi è stato impiegato un sensore Mesa SwissRanger4000 [3] (figura 2.1) che è composto da una matrice di trasmettitori sincronizzati che emettono un fascio di segnali infrarosso e una matrice di ricevitori integrati in un sensore CCD/CMOS.

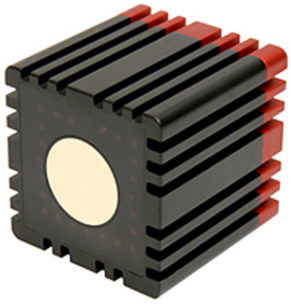


**Fig. 2.1:** Sensore Mesa SR4000

Questo sensore può raggiungere una frequenza di 54 frame per secondo, e lavora correttamente su oggetti in un intervallo di distanze comprese tra 1 e 5 metri. L'accuratezza della misura di questo tipo di sensori dipende dalla

precisione con la quale avvengono le misure temporali e dalla qualità della riflessione del segnale sull'oggetto.

Per quanto riguarda le prestazioni nominali, Mesa, il costruttore del sensore SR4000, dichiara che la ripetibilità delle acquisizioni è caratterizzata da una deviazione standard inferiore a 5mm e la sua precisione assoluta può avere al massimo uno scarto di 1 cm rispetto all'effettiva distanza dell'oggetto, nel caso in cui le misure vengano effettuate nelle condizioni di illuminazione e riflettanza ideali. Il comportamento dello SR4000 in scene reali, risulta essere molto meno ideale rispetto alla descrizione nominale fornita dal costruttore, sia in termini di ripetibilità che di accuratezza assoluta. Queste considerazioni legate ai limiti di operabilità dei sensori ToF vanno tenuti ben presenti durante il loro utilizzo e per una migliore comprensione dei risultati ottenuti.

## 2.2 Limiti e vantaggi della tecnologia TOF

La facilità di utilizzo di un sensore matriciale TOF, semplice quanto usare una comune videocamera, e l'alto framerate raggiunto da questo tipo di sensori promettono un futuro roseo a questa tecnologia nell'ambito della ricostruzione di scene 3D e oggetti. Tuttavia il metodo utilizzato per rilevare la profondità di un punto nella scena intrinsecamente porta a delle problematiche non trascurabili che inficiano sulla qualità delle misurazioni ottenute. In particolare bisogna tenere in considerazione:

1. Risoluzione molto bassa

   Le camere ToF disponibili oggi sul mercato hanno risoluzioni molto basse, ( il Mesa SwissRanger4000 presenta una risoluzione di 176x144) molto più bassa rispetto agli standard per le videocamere. Questa caratteristica si traduce in una generale povertà di informazione sulla geometria della scena e delle superfici

2. Scarsa accuratezza e rumore

   Oltre al rumore aleatorio sui dati di profondità (rumore termico, rumore di quantizzazione della distanza, rumore shot tipico dei dispositivi a ricezione di fotoni), i sensori TOF sono affetti anche da errori sistematici sulle misurazioni.

Il modello di rilevazione della distanza, si basa sull'assunzione che il segnale trasmesso viaggi direttamente dai led emettitori del sensore all'oggetto, la cui superficie, parallela al sensore, rifletta completamente il segnale che ritorna al ricevitore avendo compiuto esattamente il doppio della distanza tra la camera e l'oggetto. Se la superficie dell'oggetto ha un angolo di curvatura diverso da quella ideale una parte significativa del segnale potrebbe non venire riflessa nella direzione del ricevitore che quindi può ricevere segnali

di ritorno molto deboli, o non riceverli affatto nel caso in cui il segnale trasmesso venga deviato totalmente in un'altra direzione.

Questo comporta che la stima lungo i bordi di un oggetto sia tanto più rumorosa tanto più il raggio di curvatura è elevato, fino a rendere inaffidabili alcune misure.

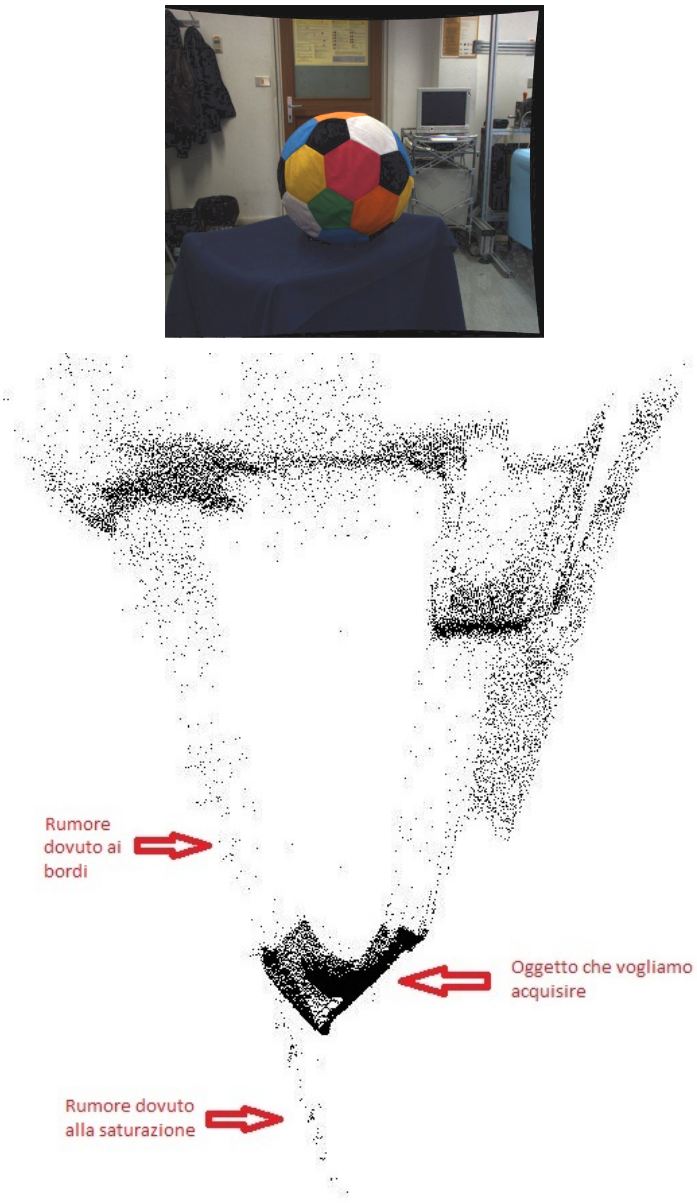



**Fig. 2.2:** Sopra foto della scena ripresa con la camera, sotto la vista dall'alto della nuvola di punti acquisita con il sensore a tempo di volo

La figura 2.2 mostra l'acquisizione di una scena composta da un pallone di stoffa sopra un tavolino ricoperto da un telo blu. Come si può notare nell'immagine di destra, che è una vista dall'alto della nuvola di punti acquisita dal ToF, le misurazioni lungo i bordi del tavolo e del pallone sono molto rumorose e generano delle pericolose scie di punti che sono completamente inesistenti nella realtà.

Oltre al rumore dovuto all'attenuazione dei segnali, nel caso la scena sia composta da più oggetti, possono esser disposti in modo tale che il segnale percorra una direzione diversa da quella ideale: il raggio emesso può esser riflesso da un primo oggetto prima di raggiungere l'oggetto di interesse. Il risultato finale sarà una sovrastima della distanza della distanza. Questo fenomeno è chiamato multipath error. Un'altra situazione comune che porta ad un multipath error è quando si cerca di misurare porzioni di oggetti con geometrie concave come mostrato in figura 2.2

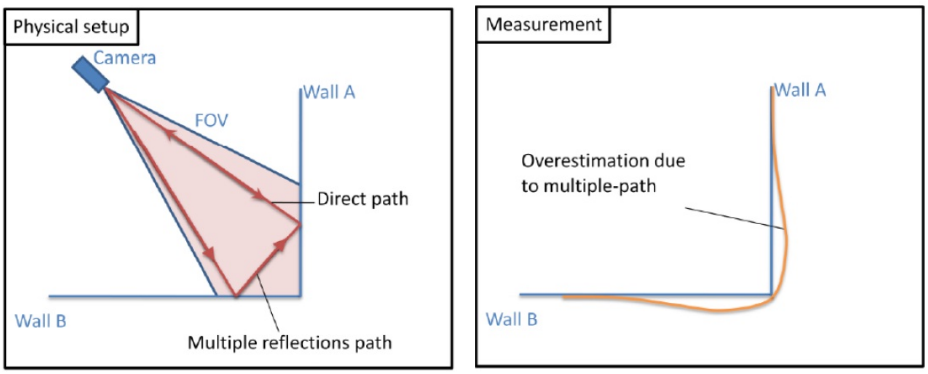



**Fig. 2.3:** Scattering sensori TOF

Oltre al problema di sovrastima, si possono verificare due tipi di saturazione: la prima relativa all'eccessiva potenza associata all'illuminazione di fondo e una relativa ad una eccessiva ampiezza del segnale ricevuto dovuto all'eccessiva riflettanza o vicinanza dell'oggetto (Vedi figura 2.3). Superfici con alto grado di assorbimento come la stoffa o affini o le superfici scure attenuano eccessivamente il segnale rendendo la misura della distanza particolarmente rumorosa. Questi problemi rendono difficile delineare accuratamente la geometria di oggetti di dimensioni ridotte.

### 2.2.1 Ventaglio di tecnologie alternative

Considerando gli obiettivi e le premesse di questo lavoro, in questo capitolo sono analizzate le tecnologie alternative presenti nel mercato. Anticipiamo che delle due tecnologie candidate i Laser Scanner hanno dei tempi di acquisizione troppo elevati e quindi non adatti ad un contesto dinamico mentre il

dispositivo Microsoft Kinect potrà rappresentare una valida alternativa da esplorare e vagliare.

**Laser scanner**: Indubbiamente la qualità di una singola vista, la ripetibilità delle acquisizioni e l'affidabilità delle misure sono parametri di punta per questa tecnologia che permette ricostruzioni di elevata qualità. Purtroppo però non è adatta ad acquisizioni di scene dinamiche poiché necessita di diversi secondi per l'acquisizione di ciascuna singola vista.

**Kinect**: Questo dispositivo [1], come vedremo, è una possible reale alternativa al ToF. La tecnologia su cui si basa è completamente differente riesce ad acquisire dati 3D della scena con una frequenza sufficientemente alta da essere adatta ad un contesto dinamico.

Questo dispositivo è stato rilasciato dalla Microsoft come accessorio della Xbox per permettere una nuova modalità di iterazione con l'utente.


**Fig. 2.4:** Sensore Microsoft Kinect

Tecnicamente è dotato di tecnologia basata su una telecamera in grado di misurare la distanza di oggetti e superfici in base ad una scansione effettuata attraverso raggi infrarossi. Vengono catturati a circa 40fps due immagini differenti da due telecamere VGA (risoluzione 640x480): una a colori e e un'altra nella quale viene misurata la profondità della scena ripresa, quest'ultima correda ogni pixel all'interno dello scatto catturato con un valore che ne rappresenta la distanza misurata. Le due immagini vengono sovrapposte per attribuire ai pixel colorati le rispettive profondità.

La misurazione della distanza avviene grazie ad un proiettore a infrarossi che genera una griglia di punti invisibili all'occhio umano e di un sensore CMOS monocromatico che cattura come l'ambiente riflette tali infrarossi e valuta come tali informazioni ritornano ad essa. Ha una distanza minima di funzionamento di 1.2 metri, e quella massima di 3.5 metri per un funzionamento ottimale mentre fino a 6 metri con prestazioni via via decrescenti.

Per il ToF l'errore è costante nel range di profondità, il Kinect invece passa da un basso rumore nelle immediate vicinanze (qualche millimetro) fino ad un rumore più alto allontanandosi dal sensore.

Nel confrontare le caratteristiche dei due dispositivi si tenga conto che il prezzo di un dispositivo ToF si aggira sui 3500 euro, mentre il costo del Kinect è decisamente inferiore: costa infatti circa 150 euro.

## 2.3 Sistema di acquisizione ToF

La camera ToF acquisisce frame con una frequenza massima di 54 fps, per ogni frame il dispositivo genera tre immagini che forniscono informazioni sulla profondità e sull'affidabilità dei dati stessi. Nel dettaglio, le tre immagini acquisite sono:

- Un'immagine di ampiezza, che descrive la riflettanza di ciascun pixel della scena alla lunghezza d'onda del segnale emesso
- Un'immagine di profondità, che fornisce pixel per pixel l'informazione della distanza del punto nella scena reale.
- Una mappa di confidenza che qualifica ciascun pixel con una stima della precisione della misura di profondità. Valori alti di confidenza indicano alta precisione nella misurazione della profondità, mentre bassi valori indicano bassa precisione.

Il sistema di acquisizione salva queste immagini in file di testo.

Per poter conoscere a quale punto nella scena corrisponde l'informazione associata ad un pixel è necessario eseguire la calibrazione del dispositivo per conoscere i parametri della matrice di proiezione (parametri intrinseci).

$$K = \begin{bmatrix} -fk_u & 0 & u_0 \\ 0 & -fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

Dove $(u_0, v_0)$ sono le coordinate del punto principale, $f$ è la lunghezza focale del sensore ToF e $K_u(K_v)$ è l'inverso della dimensione del pixel lungo la direzione $u(v)$. Visto che l'ottica non è ideale, introduce una distorsione che può essere modellata e corretta. Il modello utilizzato è lo stesso che vedremo nel prossimo capitolo per la videocamera.

Il metodo di calibrazione del sensore ToF va oltre lo scopo di questa tesi ma possiam brevemente dire che è identico al metodo utilizzato per una videocamera standard con l'unica particolarità che viene utilizzata l'immagine di ampiezza anzichè l'immagine a colori. Per approfondire il metodo di calibrazione [13] e per la geometria proiettiva [5].

Mediante la seguente relazione:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z} \quad K \quad \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

si può passare dalle coordinate nel piano immagine al piano focale o viceversa; in quest'ultimo caso se $(\hat{u}, \hat{v})$ sono le coordinate distorte nel piano

immagine di cui si conosce l'informazione di profondità z, le coordinate 3D $(x, y, z)$ del punto si ottengono tramite:

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = z \quad K^{-1} \begin{bmatrix} \hat{u} \\ \hat{v} \\ 1 \end{bmatrix}$$

Applicando la formula a tutti i punti di un frame si ottengono le rispettive coordinate 3D, che possono esser utilizzate per visualizzare la scena acquisita come una nuvola di punti, come ad esempio quella in figura 2.2.

## 2.4 Post-processing dei dati acquisiti

L'immagine di profondità acquisita in un singolo istante è composta da più di 25000 punti (176x144), ma a seconda della scena, una parte più o meno rilevante di questi punti contengono molto rumore dovuti ai problemi descritti nel paragrafo 2.2 e possono non essere abbastanza affidabili da esser utilizzati. Inoltre possono essere presenti punti relativi a zone non interessanti della scena. L'idea generale è di eliminare i punti che non servono a preservare gli edge e i particolari geometrici e di colore della parte della scena che ci interessa e nello stesso tempo ridurre il rumore nei dati che interessano. Queste operazioni consentono all'applicazione una stima più accurata degli allineamenti e un minor numero di dati da gestire per le successive operazioni. La fase di post-processing non deve escludere troppi punti, altrimenti la successiva operazione di allineamento non possiede sufficienti informazioni per eseguire correttamente le elaborazioni.

### 2.4.1 Rimozione dello sfondo

Il caso di una ricostruzione di un oggetto di dimensioni ridotte permette una semplificazione importante: i punti d'interesse sono tutti e soli quelli che riguardano l'oggetto e li si vuole isolare dai punti appartenenti ad altri oggetti o allo sfondo della scena. Questa opzione può essere implementata banalmente impostando delle soglie a priori di distanza minima e massima ad di fuori delle quali scartare i punti.

Questa strategia è ottimale se l'oggetto d'interesse ha una profondità significativamente diversa dal resto e quindi possono essere isolati i punti della geometria dell'oggetto utili alla ricostruzione, diminuendo il carico computazionale dell'applicazione e nel contempo incontrando il favore dell'utente. Il filtro così delineato ha complessita lineare nel numero dei punti di un singolo frame del ToF.

### 2.4.2 Sfruttare mappa confidenza e ampiezza

Per eliminare i punti più rumorosi la strada più veloce sarebbe implementare un filtro con delle soglie sui valori di confidenza e ampiezza. Come vedremo entrambe queste strade hanno sperimentalmente una buona controindicazione.

**Filtraggio sulla confidenza**: elimina i punti lungo i bordi, nel caso della ricostruzione di oggetti questa operazione è molto pericolosa, perché i bordi anche se rumorosi sono punti geometricamente rilevanti e spesso fondamentali per una buona registrazione.

**Filtraggio sull'ampiezza** : quando la superficie dell'oggetto da ricostruire è fatta di materiali che assorbono molto i segnali ToF, l'ampiezza del segnale ricevuto sarà molto bassa e filtrando per ampiezza si eliminano oltre ai punti poco affidabili (punti su una superficie con un angolo di curvatura rispetto al s.d.r del ToF elevato) vaste zone della scena che contengono informazioni notevoli.

Scartare troppi punti in un frame non sarebbe in se un problema se avessimo la garanzia che i dettagli della superficie erroneamente filtrati in un frame vengano considerati in quello successivo. Ma le zone che contengono dettagli geometrici come ad esempio il naso di un orsetto o le dita di una mano verranno filtrati da tutti i punti di vista perché la loro inclinazione o il materiale di cui sono composti non cambia e il filtro troppo aggressivo continuerebbe ad eliminarli.

Quindi la scelta fatta è di non filtrare i punti utilizzando ampiezza e confidenza (o comunque farla con una soglia altissima) ma di utilizzare un metodo più fine per eliminare i punti che verrà presentato nel prossimo paragrafo.

### 2.4.3 Calcolo della curvatura

Il problema in questione del ToF è la rumorosità delle misure acquisite lungo i bordi degli oggetti, ossia dove la superficie ha un angolo di curvatura elevato rispetto all'asse ottico (raggio trasversale al piano immagine che passa per il centro ottico).

Se ipotizzassimo di conoscere la direzione del vettore normale alla superficie in ogni punto potremmo filtrare i punti acquisiti la cui normale forma un angolo troppo elevato con l'asse ottico, evitando così di fidarci dei punti a rischio.

Considerato che questi dati non sono a disposizione, esiste però un modo alternativo per ottenere una versione approssimata della direzione della superficie in ogni punto. Considerando la nuvola di punti 3D, si può calcolare la normale in ogni punto considerando il piano che approssima localmente la superficie.

Considerando tutti punti della nuvola 3D in un intorno circolare, si cerca di calcolare il piano passante per i punti, ma considerando che generalmente saranno più di tre punti e quasi sempre non coplanari il problema è impossibile da risolvere correttamente. Il problema può essere formalizzato come un sistema lineare di equazioni sovradeterminato che non ha soluzione esatte, ma esiste una soluzione ai minimi quadrati. Il metodo implementato in questo sistema si basa sulla Singular Value Decomposition (SVD)[5].

Sperimentalmente in questo modo l'immagine di profondità ottenuta è più pulita senza perdita di informazione collaterale. Questo processo però ha un costo computazione superiore rispetto al semplice sbarramento sulla confidenza o ampiezza, ma una volta calcolate le normali sui punti scremare i punti è un'operazione lineare. Le normali calcolate, inoltre sono utili anche nella fase di cleaning e durante la visualizzazione della mesh al termine della ricostruzione.

### 2.4.4 Filtro bilaterale

La necessità che porta ad inserire il filtro bilaterale [8] nella pipeline è quella di ridurre il più possibile il rumore dei punti preservando le forme generali e i dettagli.

L'idea alla base di questo filtro è che il valore in un punto in una data posizione può essere sostituito da una funzione dei valori dei punti vicini. Nell'ipotesi che la profondità nello spazio vari molto lentamente per rimuovere il rumore sarebbe sufficiente sostituire ad ogni valore una media dei valori dei pixel vicini, tuttavia acquisendo dati di scene reali, l'ipotesi che punti vicini abbiano una profondità simile fallisce lungo i bordi di ogni oggetto incontrato. Il filtro bilaterale quindi è un filtro ugualmente semplice, ma che permette di livellare i punti preservando i bordi.

Il valore di profondità di un pixel viene rimpiazzato con la media pesata dei valori di profondità dei pixel vicini, ma il peso non dipende solamente dalla distanza euclidea ma anche dalla differenza di similarità espressa come differenza di profondità.

Sia $p$ il punto a cui viene applicato il filtro, e $p_{ij}$ uno dei punti nell'intorno di p e siano $z(p_{ij})$ e $z(p)$ rispettivamente i valori di profondità di $p$ e $p_{ij}$, allora il peso assegnato al punto $p_{ij}$ sarà:

$$w_{ij} = e^{\left(\frac{1}{2}\left(\frac{dist(p_{ij},p)}{\sigma_d}\right)^2 - \frac{1}{2}\left(\frac{z(p_{ij})-z(p)}{\sigma_r}\right)^2\right)}$$

Nella formula $\sigma_r$ e $\sigma_d$ sono due parametri che codificano rispettivamente il peso specifico della distanza euclidea rispetto alla differenza di profondità.

Questo filtro ha una complessità lineare, con una costante moltiplicativa che dipende dalle dimensioni della finestra applicata a ciascun punto (numero dei vicini considerati).

# Capitolo 3

# Acquisizione combinata di forma e colore

Nel precedente capitolo si è discusso la parte del sistema di acquisizione che riguarda il sensore time of flight tralasciando la videocamera. In questo capitolo vedremo in che modo aggiungere una videocamera al sistema di acquisizione, e in che modo sia possibile associare all'informazione di profondità di un punto il suo colore corrispondente estrandolo dalle immagini acquisite dalla videocamera .

## 3.1 Acquisizione informazioni colore

Per l'acquisizione dell'informazione di colore della scena è sufficiente utilizzare una comune videocamera digitale. In questo lavoro è stata scelta la videocamera Basler Scout A1000 perché già presente in laboratorio.

I dati acquisiti dalla camera digitale descrivono il colore di ogni pixel come una terne di interi compresi tra 0 e 255 , che rappresentano i valori delle componenti RGB. L'immagine acquisita altro non è che un array bidimensionale di terne di interi.

**Calibrazione geometrica**: per poter associare a ciascun pixel il corrispondente punto della scena (in realtà la direzione del punto) è necessaria la calibrazione della videocamera, con la quale viene stimata la relazione tra le coordinate di un pixel nell'immagine acquisita e la posizione del punto nella scena.

$$K = \begin{bmatrix} -fk_u & 0 & u_0 \\ 0 & -fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

Dove $(u_0, v_0)$ sono le coordinate del punto principale, $f$ è la lunghezza focale della videocamera e $K_u(K_v)$ è l'inverso della dimensione del pixel lungo la direzione $u(v)$.

Visto che l'ottica della videocamera non è ideale sarà completamente caratterizzata conoscendo sia i parametri intrinseci che i parametri di distorsione radiale.

$$d = [K_1, K_2, K_3, d_1, d_2]$$

Per la trattazione della fase di calibrazione e stima della distorsione si veda [7].

Una volta che la camera è calibrata, e i suoi parametri di distorsione conosciuti, si deve applicare la procedura di antidistorsione a tutte le immagini acquisite con quella camera in modo da compensare la distorsione radiale.

**Fig. 3.1:** Da sinistra a destra l'immagine prima e dopo l'antidistorsione

Come possiamo notare nella figura 3.1 l'immagine distorta presenta la tipica forma a botte dovuta alla lente che crea delle deviazioni sulle proiezioni rettilinee. Nell'immagini a destra si possono notare gli effetti più evidenti dell'antidistorsione: l'immagine è priva di tutte quelle aberrazioni ottiche, e infatti le linee che nella realtà sono parallele, come ad esempio le fessure delle ante dell'armadio, sono ritornate ad essere parallele.

**Calibrazione fotometrica**: mentre il comportamento del ToF non varia al cambiare delle condizioni di luminosità della scena, le acquisizione delle videocamera sono caratterizzate dalla funzione che mappa l'irradiazione della luce nei valori dei singoli pixel, per cui è necessario adattarla per ottimizzare la qualità delle acquisizioni effettuando la calibrazione fotometrica [21]. Il fattore principale che incide sulla funzione è ii tempo di apertura e la velocità dell'otturatore che possono essere misurate con il tempo di esposizione ed è regolabile a mano sulla videocamera, o attraverso il software Pylon Viewer in dotazione con le videocamere. Utilizzando lo stesso software e adoperando una palette in scala di grigi si devono inoltre bilanciare le componenti RGB per ottenere il bianco e il nero ed avere così una buona resa dei colori.

## 3.2 Fusione informazione geometria e colore

Il sistema di acquisizione preparato in laboratorio e utilizzato durante i test è quello mostrato in figura 3.2, come si può vedere è costituito da un sensore ToF e da due videocamere fissate alla stessa barra.


**Fig. 3.2:** Foto del sistema di acquisizione usato in laboratorio

Come mostra anche la figura, è possibile dotare il sistema di più videocamere e l'applicazione supporta l'acquisizione trinoculare (Sensore ToF combinato al sistema stereo), ma nella fase di elaborazione le camere devono essere usate in mutua esclusione.

Rispetto al capitolo precedente, quindi viene aggiunto al sistema di acquisizione formato dal solo sensore time of flight una videocamera digitale, per misurare il colore della scena e poterlo fondere con l'informazione geometrica della stessa [4].

Nella figura 3.3 sono riportati i principali output del sistema di acquisizione: l'immagine a colori è prodotta dalla videocamera di sinistra (rispetto al ToF) mentre la figura di destra è una stampa a video della nuvola di punti acquisiti con il ToF e visualizzata con il programma Meshlab[2]. Come si può notare dalla tipica forma a botte di entrambe le figure, le immagini sono già state antidistorte.

L'informazione geometrica e di colore proveniente dai sensori è una fotografia della scena presa dalla loro specifica posizione al momento dello scatto e quindi per poter fondere le informazioni è necessario conoscere la loro posizione relativa. Intuitivamente per associare l'informazione geometrica di un punto 3D al suo corrispondente valore di colore è necessario disporre di una mappa per passare dalle coordinate (x,y,z) di un punto 3D alle coordinate (u,v) nel piano immagine della videocamera.

Durante l'acquisizione di scene dinamiche, o se il sistema viene spostato durante l'acquisizione di una scena statica, si presenta il problema di sin-

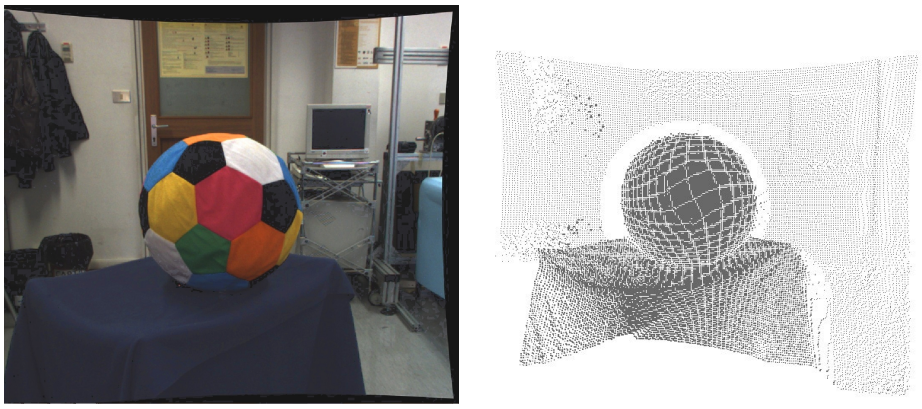


**Fig. 3.3:** Acquisizione di un pallone di stoffa sopra un tavolino

cronizzare perfettamente gli istanti in cui i dispositivi acquisiscono le loro immagini, in modo che abbiano in quell'istante la stessa versione della scena. Questo evita fastidiosi disallineamenti tra le informazioni geometriche e la corrispondente informazione di colore.

Per risolvere questo problema, i dispositivi vengono pilotati da una scheda hardware (che non viene trattata in questa tesi) che generando un'onda quadra sincronizza gli istanti di acquisizione di ogni frame del sensore time of flight e della videocamera. Modificando il periodo e il duty cycle dell'onda quadra si può controllare la frequenza con la quale vengono acquisiti i frame (Appendice A - Tool di acquisizione per i dettagli).

Se durante l'acquisizione il sensore ToF e la videocamera si muovessero indipendentemente, per ogni istante si dovrebbe conoscere la posizione assoluta di entrambi i sensori per poter calcolare la loro pozione relativa. Per semplificare il problema, la videocamera e il sensore ToF vengono fissati ad una barra di metallo mantenendo costante la loro posizione relativa che andrà quindi calcolata una sola volta. La stima della posizione relativa può essere formalizzata come il calcolo della matrice di rotazione e del vettore di traslazione t che lega i due sistemi di riferimento. La calibrazione del sistema restituisce la matrice:

$Rt_{SR-Cam} = [R_{3x3}T^{-1}]$

che permette quindi di passare dal s.d.r solidale al ToF al s.d.r standard della fotocamera. Più la matrice sarà stimata esattamente, più la fusione tra la geometria e i rispettivi colori sarà puntuale e affidabile.

Una volta che i punti ottenuti dal ToF sono stati rototraslati sul s.d.r della fotocamera, non è possibile fondere le informazioni semplicemente sovrapponendo le due viste, perché anche se l'inquadratura della scena fosse esattamente la stessa, la risoluzione dei due dispositivi è diversa.

La risoluzione del sensore time of light è 177x144 mentre la camera Basler

raggiunge una risoluzione molto più elevata di 1032x778 inquadrando una parte della scena molto più grande di quella acquisita dal sensore ToF. La parte dell'immagine acquisita dalla videocamera che non ha corrispondenza nelle immagini acquisite dal ToF occupa memoria inutilmente durante l'elaborazione visto nessun punto del ToF avrà un corrispondente li. Per evitare lo spreco, prima della calibrazione, è utile ridimensionare l'area inquadrata della videocamera abbassando la risoluzione, raggiungendo idealmente la risoluzione che minimizzi lo scarto di informazione inutile acquisita.

Non potendo banalmente sovrapporre le due viste per associare il giusto colore ai punti 3D, è necessario proiettare i punti 3D che erano stati proiettata nel s.d.r standard della videocamera al piano immagine.

Formalizzando quanto descritto:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z} \quad K \quad [Rt] \quad \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

dove $K$ è la matrice dei parametri intrinseci della camera, $[Rt]$ è la matrice di rototraslazione tra il Tof e il s.d.r standard della camera, $[u, v, 1]^T$ sono le coordinate nel piano immagine della camera riferite al pixel che è la proiezione del punto 3D $[x, y, z, 1]^T$ espresso in coordinate omogenee.

Le coordinate (u,v) calcolate sono generalmente non intere e quindi non puntano direttamente un pixel del piano immagine. Per l'assegnamento del colore si possono seguire diverse strade, ad esempio si potrebbero arrotondare le coordinate all'intero più prossimo assegnare al punto il colore del pixel più vicino. Un'altra strada, che è quella seguita in questa tesi, è quella di assegnare al punto (u,v) la media di tutti i pixel in una finestra di un raggio x (con ad esempio x=4) intorno al punto (u,v) e calcolare il colore come media non pesata dei colori dei pixel. Questa soluzione ha una costante maggiore nella complessità computazione , ma garantisce una certa robustezza al rumore nella scelta del colore.

Se consideriamo l'insieme dei pixel in cui cadono i punti 3D, questi formano un'immagine a colori a bassa risoluzione (la stessa del ToF) che può essere esattamente sovrapposta alla mappa di profondità fondendo l'informazione di colore e di profondità.

## 3.3 Gestione delle occlusioni

Durante l'ancoraggio del sensore a tempo di volo T e della videocamera C alla staffa, per quanto si cerchi di metterli il più vicino possibile, fisicamente sarà impossibile sovrapporli, e quindi C e T acquisiscono la scena da un punto di vista diverso.

Questo conduce ad un problema rilevante, infatti alcune parti della scena che sono visibili dal sensore ToF non lo sarannno dal punto di vista della videocamera e viceversa.

Senza l'eliminazione di queste parti occluse, i punti 3D ottenuti dal sensore ToF verrebbero ugualmente proiettati nel sistema di riferimento della camera e il loro colore sarebbe quello corrispondente al pixel in cui cadono. Il punto occluso e quello non occluso hanno coordinate diverse nel sistema di riferimento del ToF ma cadono nelle stesse coordinate nel sistema di riferimento del piano immagine della camera, questo porta ad assegnare un colore a tutti i punti che in realtà sarebbero occlusi.

L'approccio seguito per la rimozione delle parti occluse è quello di costruire la mesh 3D a partire dalla mappa di profondità e renderizzarla utilizzando come punto di vista quello della camera [20].

Dati i punti 3D del ToF, vengono proiettati nel sistema di riferimento della videocamera, la costruzione della mesh poi passa per la definizione delle primitive geometriche elementari, come ad esempio i triangoli, definiti dalle coordinate dei tre vertici che lo compongono.

In un contesto tridimensionale, perché le primitive (triangoli) siano visibili non è sufficiente che siano all'interno della finestra di visualizzazione, perché altre primitive, per la loro collocazione spaziale possono impedirne la vista.

Dato l'insieme delle superfici dei triangoli, Le tecniche di rimozione delle superfici nascoste possono essere ricondotte ad un problema di ordinamento: l'ordinamento delle primitive geometriche in profondità rispetto all'osservatore (s.d.r videocamera) permette di individuare le parti visibili e le parti nascoste. Il problema dell'ordinamento presenta una complessità computazionale quadratica nel numero di primitive. L'approccio seguito in questa tesi sarà, come si vedrà nei prossimi paragrafi, lineare nel numero di primitive.

La tecnica di HSR (Hidden Surface Removal) seguita appartiene alla categoria di algoritmi image-space: si determina per ogni punto significativo del piano di proiezione (in pratica per ogni pixel del piano immagine) il poligono della scena che l'osservatore (in pratica s.d.r della videocamera) vede attraverso quel punto. Dal punto di vista computazionale il problema si riconduce a calcolare per ogni pixel qual'è la primitiva più vicina (operazione lineare nel numero di primitive).

O(n x m x k) = O(k) dove: k = massimo numero di superfici in serie che si occludono l'un l'altra; n x m = dimensioni piano immagine;

L'algoritmo che si è deciso di implementare è lo z-buffer, forse quello più diffuso della categoria image-space.

### 3.3.1 Algoritmo z-buffer

La tecnica z-buffer fa uso di due buffer di memoria, il primo chiamato appunto z-buffer o depth-buffer e il secondo frame-buffer aventi le stesse dimensioni del piano immagine. Al termine dell'algoritmo per ogni posizione (x,y) lo z-buffer contiene il valore di profondità del punto corrispondente alla primitiva visibile, mentre il frame buffer conterrà l'identificativo della primitiva visible. Inizialmente tutti i valori dello z-buffer sono posti a profondità massima mentre i valori del frame buffer sono posti a primitiva nulla.

Per ogni primitiva, vengono calcolati i valori di profondità di tutti i punti che appartengono ad essa a partire dall'equazione del piano su cui giace la primitiva stessa ( utilizzando l'algoritmo scan-line). Per ogni primitiva esaminata, una volta determinata la profondità in corrispondenza di ogni pixel (u,v) in cui la primitiva viene mappata, se la profondità z corrispondente al punto (u,v) risulta inferiore alla profondità corrente memorizzata per quel pixel nello z-buffer allora lo z-buffer assume z come nuovo valore di profondità per quel pixel e segna nel corrispondente valore del frame-buffer a che primitiva appartiene il nuovo valore. A completamento del processo, lo z-buffer conterrà le profondità degli oggetti visibili mentre il frame-buffer avrà assunto gli id delle corrispondenti primitive.

Data la proiezione (u',v') nel piano immagine di un vertice (x,y,z) nel s.d.r TOF, il punto sarà visibile e quindi meritevole di esser associato all'informazione di colore se la sua z sarà uguale al valore nello z-buffer, altrimenti il punto non è di profondità minima e quindi non è visibile nel sistema di riferimento della videocamera.

### 3.3.2 Algoritmo Scanline per il filling di triangoli

In letteratura sono noti diversi algoritmi per il filling di poligoni, ad esempio...

Nel caso in questione i poligoni sono semplici triangoli e sono possibili alcune semplificazioni notevoli che hanno portato al seguente algoritmo:

INPUT: vertici del triangolo: $P1(x_1, y_1, z_1), P2(x_2, y_2, z_2) e P3(u_3, v_3, z_3)$ , ordinati per valore crescente di ordinata ($y_1 < y_2 < y_3$).

OUTPUT: Valore di profondità in ogni punto discreto contenuto nella superficie del triangolo di vertici P1, P2 e P3.

ALGORITMO

- Interpolare i vertici calcolando le coordinate dei punti lungo i bordi. Quindi si calcolano gli incrementi lungo i tre lati:

  $dx_{12} = \frac{x_2 - x_1}{|y_2 - y_1|}$ $\quad$ $dz_{12} = \frac{z_2 - z_1}{|y_2 - y_1|}$

  $dx_{13} = \frac{x_3 - x_1}{|y_3 - y_1|}$ $\quad$ $dz_{13} = \frac{z_3 - z_1}{|y_3 - y_1|}$

  $dx_{23} = \frac{x_3 - x_2}{|y_3 - y_2|}$ $\quad$ $dz_{23} = \frac{z_3 - z_2}{|y_3 - y_2|}$

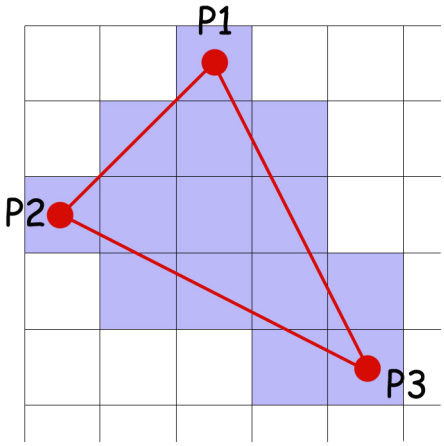



**Fig. 3.4:** Rasterizzazione del triangolo

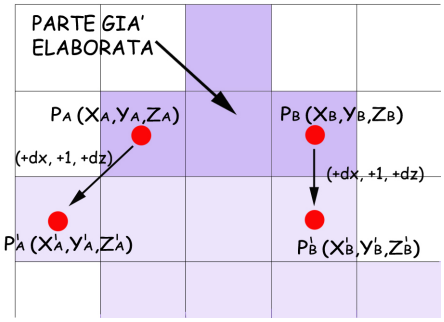



**Fig. 3.5:** Interpolazione lungo i bordi per righe crescenti

Arrivati alla riga i-esima, sono stati noti $P_A$ e $P_B$ i punti lungo i bordi di coordinate rispettivamente $(x_A, y_A, z_A)$ e $(x_B, y_B, z_B)$.

Alla riga (i+1)esima $P_A^I = (x_A + dx_{12}, y_A + 1, z_A + dz_{12})$ e $P_B^I = (x_B + dx_{13}, y_B + 1, z_B + dz_{13})$

- Dopo aver calcolato le coordinate dei punti ai due estremi di ciascuna riga, si calcolano le coordinate dei punti interni alla riga interpolando le coordinate dei bordi. Durante l'interpolazione della riga il valore dell'ordinata rimarrà invariata, il valore dell'ascissa crescerà con scatto unitario, mentre la z verrà incrementata di $dz_L = \frac{z_B^I - z_A^I}{|x_B^I - x_A^I|}$ ad ogni passo

Intuitivamente l'algoritmo calcola la profondità dei punti interni al triangolo scorrendolo per righe. Infatti, ad ogni riga dopo aver calcolato le coordinate

degli estremi (tramite interpolazione dei vertici) le interpola ottenendo i valori dei punti interni alla riga.

Questo algoritmo è lineare nel numero di pixel contenuti nell'area del triangolo, ossia lineare nella dimensione dell'area stessa O(area).

# Capitolo 4

# Costruzione modello 3D a colori

Non è possibile catturare completamente l'aspetto di un oggetto o di una scena con una singola vista, ma ne servono molte prese con angolazioni e da posizioni differenti in modo che ogni vista possa aggiungere alcuni dettagli non inquadrati nelle altre viste.

Le porzioni della superficie dell'oggetto (possiamo pensarle come nuvole di punti colorate) che si ottengono da ciascuna vista sono relative alla posizione esatta da cui è stata acquisita quella vista (posizione del sensore), quindi ogni vista ha un sistema di riferimento diverso. Non possedendo un sistema di tracking che tenga traccia del cambio di orientazione e posizione durante gli spostamenti del sistema di acquisizione è necessario portare tutte le viste nello stesso sistema di riferimento calcolando le opportune trasformazioni rigide.

La registrazione di più viste, è ottenuta dalla concatenazione di registrazioni di coppie di viste consecutive. Il problema intrinseco generato da questa scelta è l'introduzione di un errore incrementale nella stima delle rototraslazioni, che non conduce alla soluzione globale ottimale.

Per risolvere il problema di allineare una coppia di viste si è utilizzato l'algoritmo di allineamento ICP (Iterative Closest Point) [9] [10] [14].

Definendo più formalmente il problema: dati un insieme di punti P con coordinate $Ps_i$, i=1, ...,n rispetto ad un sistema di riferimento S, e di coordinate $Pt_i$, i=1, ...,n nel sistema di riferimento T, il problema è la stima della rototraslazione Rt del sistema di riferimento S rispetto a T, che minimizzi la distanza tra le due nuvole di punti. E' fondamentale notare che non sono note a priori le corrispondenze dei punti da allineare, ma ICP risolve simultaneamente il problema delle corrispondenze e la stima della trasformazione rigida. L'algoritmo ICP implementato si può riassumere così:

*Input*: Due nuvole di punti 3D: S (source) e T (target), i parametri: ns, max-RMS, max-iter

*Output*: Matrice di rototraslazione che allinea S su T

*Esecuzione :*

1. Calcola i centroidi del source e del target; Trasla i punti del source in modo da far coincidere i centroidi(questo passo velocizza la convergenza)

2. Scegli ns punti di S (il criterio di scelta sarà oggetto di studio nel paragrafo 4.2),

3. Per ognuno degli ns punti, trova il punto corrispondenti in T;

4. Date in ingresso le coppie registrate, rimuove quelle in cui la somiglianza (espressa in termini di vicinanza nello spazio dei colori CIE-LAB) è al di sotto di una certa soglia [22]

5. Con le corrispondenze rimaste, calcola la rototraslazione ottima.

6. Applica la trasformazione trovata nel passo precedente a tutti gli ns punti di S

7. Se la media dell'errore quadratico (RMS) è minore della soglia max-RMS o se l'algoritmo è giunto al suo numero di iterazioni massimo termina l'esecuzione, altrimenti vai al passo 3.

Questo metodo, come dimostrato da Besl e McKay, converge al più vicino minimo locale della somma delle distanze al quadrato tra i punti più vicini. La fase della ricerca del closest-point non è oggetto di questa tesi, visto che era già implementata nel sistema precedente, ma si noti che l'algoritmo utilizzato che sfrutta un Kd-tree[6] per mantenere i punti ha una complessita O(ns logn), dove n è la cardinalità di T.

## 4.1 Calcolo della rototraslazione ottima

L'algoritmo di horn [15] restituisce una soluzione in forma-chiusa al problema di orientazione relativo, minimizzando l'errore della distanza euclidea tra le coppie di punti corrispondenti:

$$\arg\min_{[M_L]} \sum_{i=1}^{n} || \, p_T^i \; - \; M_L \, p_S^i ||$$

L'algoritmo di Horn permette di ottenere la miglior roto-traslazione $M_L$ perché la soluzione in forma chiusa permette di evitare di cadere in un minimo locale, uno dei problemi principali dei metodi basati sul gradiente.

L'algoritmo di Horn è stato inserito in uno schema RANSAC [17] in modo da limitare il contributo dovuto a coppie di punti accoppiate malamente. Ad ogni iterazione vengono pescate casualmente n (con n deciso a priori) coppie di punti, sulle quali viene applicato l'algoritmo di Horn che calcola la roto-traslazione ottima. La roto-traslazione viene applicata a tutte le coppie e si contano il numero di coppie che hanno una distanza minore di sotto una certa soglia. Se quel numero è soddisfacente, la ricerca si ferma, altrimenti vengono pescate altre n coppie e si ripete il processo. Questa pipeline viene ripetuta finché non si raggiunge una roto-traslazione che supera la soglia oppure viene raggiunto un numero fissato a priori di iterazioni.

## 4.2 Allineamento sfruttando solo i punti rilevanti

La strategia con la quale vengono scelte le coppie di punti corrispondenti influisce sulla bontà della trasformazione che verrà calcolata; Se pensiamo alle coppie di punti corrispondenti come a dei fili elastici che vincolano il source al target, possiamo intuire come ci possano essere dei fili che valgono molto più di altri, ad esempio se si dovessero registrare due viste di una porta se tutte le coppie di punti fossero in zone piane della porta i vincoli sarebbero molto deboli, poiché la distanza minima tra le due viste sarebbe raggiunta anche con qualche traslazione errata. Se invece molti punti campionassero l'eventuale maniglia, i vincoli creati servirebbero molto di più.

La scelta dei punti del Source è quindi un aspetto rilevante in una implementazione efficiente di ICP, con riflessi sulla velocità di convergenza e sulla qualità dell'allineamento finale.

L'approccio banale sarebbe l'estrazione casuale dei punti, ma come possiamo facilmente intuire questo metodo è poco efficace. Tipicamente, infatti, nella nuvola di punti che rappresenta la scena ci sono molti punti a bassa curvatura che generalmente vincolerebbero poco una trasformazione rigida mentre ci sono un numero di diversi ordine di grandezza inferiore di punti che invece posseggono un informazione geometrica particolare e sarebbero molto più rilevanti per trovare l'allineamento corretto. Un campionamento casuale pescherebbe molti punti nelle grandi regioni a basso contenuto informativo, piuttosto che sulle piccole regioni ad alto contenuto. Per questo motivo si è deciso di implementare nel sistema un metodo proposto nel lavoro [11] per la campionatura di punti rilevanti.

L'approccio seguito pesa i punti in base alla loro rilevanza o distintività locale, ossia la dimensione dell'area intorno al punto in cui i punti sono simili al punto stesso. Se vogliamo calcolare la rilevanza geometrica di un punto rispetto ad un altro una buona strada è quella di misurarne il raggio di curvatura medio in un intorno.

Sia $p$ un punto della nuvola, allora l'area connessa a $p$

$$A_p = \left\{ q \in S \mid N_p^T N_q > T, p- > q \right\}$$

Dove $N_p$ e $N_q$ sono le normali della superficie S nei punti $p$ e $q$ e p -> q indica che c'è un percorso di punti che appartengono ad $A_p$ che va da $p$ $q$.

L'area sarà inversamente proporzionale alla curvatura, poichè lungo i bordi la regione si estenderà solo lungo una direzione, mantenendo una grado di larghezza minore dei punti nel piano in cui la regione si estenderà in tutte le direzioni. Quindi l'area è inversamente proporzionale a quanto la superficie possa vincolare la trasformazione localmente.

Il valore di T è un parametro molto importante e il suo significato è semplice: considerando la formula del prodotto interno di due vettori $N_p^T N_q = |N_p||N_q| \cos\theta$ con $|N_p| = |N_q| = 1$ (vettori normalizzati) l'espressione $N_p^T N_q > T$ indica che p e q sono simili se l'angolo tra le loro normali è inferiore ad un certo angolo $\theta_0$.

Nelle superfici piane l'area $A_p$ potrebbe crescere in modo incontrollato, per questo viene utilizzata una soglia massima D della distanza rispetto al punto $p$. Il calcolo della distintività coinvolgerà quindi al più $D^2$ punti.

Un punto sarà tanto più rilevante quanto più la sua distintività è bassa, ossia tanto meno è simile ai punti nell'area circostante.

Il concetto di similarità dipende da quello che vogliamo misurare, ad esempio la similarità geometrica tra un punto p e un punto q possiamo definirla come la misura nella quale sono simili le loro orientazioni.

Le figure sottostanti

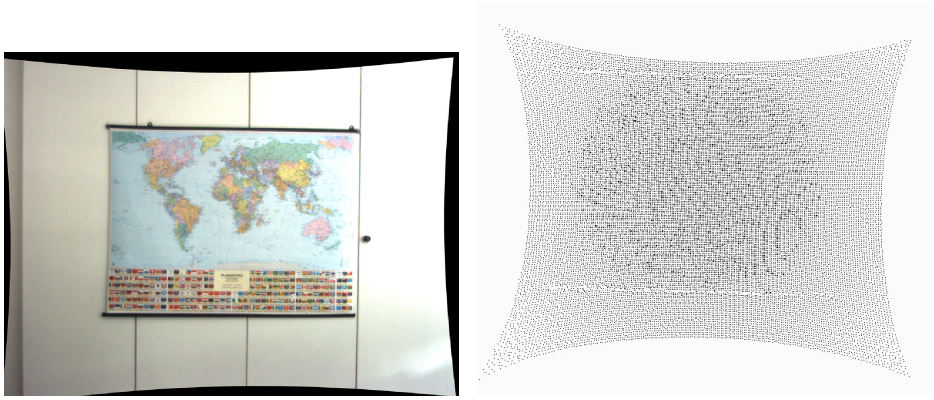


**Fig. 4.1:** Acquisizione di una cartina geografica TOF

mostrano un caso limite in cui la scena inquadrata è un armadio bianco con una cartina politica appesa. Le due figure sono una parte dell'output del sistema di acquisizione: l'immagine a colori proveniente dalla videocamera e la nuvola dei punti 3D proveniente dal ToF. Come si può facilmente intuire, se dovessimo registrare questa vista geometrica con un'altra non ci sarebbe nessun tipo di appiglio geometrico, poiché in qualunque modo vengano scelti i punti sulla cartina geografica avrebbero la stessa identica curvatura. In

questo caso l'algoritmo di registrazione produrrebbe una soluzione ottima a meno di una traslazione che nel caso della registrazione della cartina sarebbe un risultato probabilmente pessimo.

Per evitare questo problema si può sfruttare l'altra parte di informazione disponibile e associata a ciascun punto: il suo colore. Estendendo il ragionamento fatto per i punti rilevanti geometricamente, possiamo quindi pensare che anche nello spazio dei colori ci siano piccole zone (cambi netti di colore) in cui l'informazione di colore sia rilevante e può diventare l'unica fonte di vincoli per la registrazione in mancanza di appigli geometrici.

Definiamo la similarità di colore tra il punto p e q come la distanza euclidea tra le loro posizioni nello spazio di colori CIE-Lab. E' stato scelto questo specifico spazio di colori per la sua caratteristica di correlare punti vicini nello spazio (distanza euclidea ridotta) a colori con una bassa discrepanza nell'esperienza media di un osservatore umano. In questo modo punti percepiti vicini vengono considerati simili e punti che verrebbero umanamente percepiti distanti vengono considerati appunto diversi. Le tre coordinate nello spazio CIE-Lab rappresentano la luminosità del colore (L), la sua posizione tra il rosso/magenta e il verde ($a^*$) e infine la sua posizione tra il giallo e il blu ($b^*$) e permettono di rilevare efficacemente le transizioni da un colore all'altro. Tuttavia il fatto stesso che una delle tre componenti sia la luminosità implica che giochi di luce o di ombre su superfici omogenee nel colore, porti a individuare falsi punti rilevanti. Il problema è ancor più rilevante considerando che a meno di acquisire con una illuminazione artificiale ad arte, la disomogeneità della luminosità nella scena sarà sempre notevole. Per questo motivo, adottando una soluzione già utilizzata da altri in letteratura, si è deciso di ignorare la coordinata L, calcolando la distanza tra due colori solo nel piano $(a^*, b^*)$. Questa scelta risolve il problema ma non è priva di conseguenze negative note, infatti, nello spazio CIE-Lab i colori in scala di grigio hanno le medesime coordinate $a^*$ e $b^*$ e differiscono solo per la luminosità. Questo significa che per evitare che vengano presi falsi punti rilevanti, si rinuncia per fare un esempio a rilevare i bordi nelle celle di una scacchiera b/n.

Nella figura 4.2 l'immagine di sinistra e di destra rappresentano rispettivamente i risultati del calcolo della distintività geometrica e di colore: ad ogni pixel è stato assegnato un valore nella scala di grigi proporzionale al numero di punti simili nel suo intorno. Esemplificando ai due estremi, un punto con 0 elementi simili nel suo intorno e quindi molto rilevante, verrà mappato con il colore nero; in modo simmetrico un punto completamente irrilevante in cui $|Ap| = D^2$ e quindi simile a tutti i punti del suo intorno verrà mappato con il colore bianco.

Come possiamo notare dalle immagini il meccanismo di calcolo della distintività è efficace, infatti nell'immagine di sinistra sono stati estratti come punti rilevanti i bordi del tavolino e persino le ondulazioni della tovaglia, mentre nell'immagine di destra si possono notare come emergano i punti

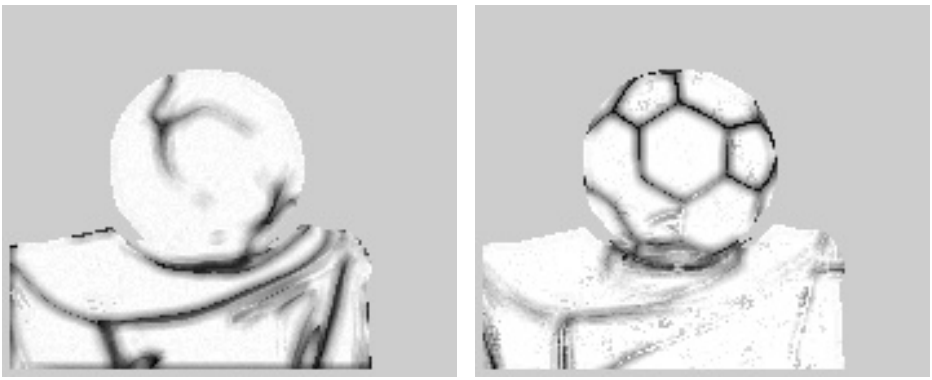


**Fig. 4.2:** Rappresentazione della distintività geometrica e di colore

lungo i bordi degli esagoni colorati del pallone, ossia nei punti di maggior contrasto. Ad uno sguardo più attento non potrà sfuggire, però, che la transizione tra l'esagono nero e quello bianco del pallone non è stato rilevato e la spiegazione la troviamo nella scelta di ignorare la componente della lightness.

Quindi per ogni punto abbiam a disposizione la distintività geometrica e la distintività di colore. Come possiamo fondere i due valori in un'unica metrica che riassuma la distintività nel punto? la soluzione pensata, e poi implementata, è quella di assumere come distintività del punto il minimo tra i due valori in modo da far emergere sempre la qualità più evidente.

Questo Approccio è possibile solo perché le due misure di distintività sono confrontabili, a patto ovviamente di calcolarle utilizzando la stessa soglia di distanza. Conoscendo la distintività di un punto, i punti che vincoleranno maggiormente la rototraslazione saranno i punti con la minor distintività.

```
Traccia dell'algoritmo:
    N numero di coppie per l'algoritmo ICP
    A array di N interi;
    S Source ( |S|>> N) ;
    for each punto p in S{
        Calcola distintività geometrica Dg per il punto p;
        Calcola distintività di colore Dc per il punto p;
        Distintività(p) = min(Dc, Dg);
        if (A non è ancora stato riempito)
            Inserisci il punto in A in modo ordinato
            rispetto al valore di distintività;
        else
            if (distintività(p)> A[N-1]) then
                Scarta il punto; (ci sono già N punti
                più rilevanti di p)
```

```
            else
                Elimina il punto alla posizione N-1;
                Inserisci p in A in modo ordinato;
    }
```

Al termine dell'algoritmo nell'array A si troverrano gli N punti(Con N uguale al numero di punti scelti per l'ICP) più rilevanti del Source.

In situazioni particolari può accadere che i punti più rilevanti siano concentrati in un'area ridotta rispetto all'intera scena acquisita lasciando senza nessun vincolo aree grandi. In particolar modo il problema nasce quando i punti più rilevanti appartengono ad uno stesso piano. In questi casi l'algoritmo ICP converge ad un minimo locale, riducendo la distanza media tra ogni coppia di punti corrispondenti, senza però raggiungere necessariamente un mimino globale.

Questo problema non si presentava nel caso della scelta casuale dei punti, poiché la casualità garantiva una distribuzione uniforme dei punti estratti. Per questo motivo è stata adottata un approccio ibrido, in cui una piccola percentuale dei punti utilizzati per la registrazione viene estratto casualmente, garantendo che vi siano vincoli anche in zone della scena che non sono rilevanti ma che contribuiscono a raggiungere un ottimo globale.

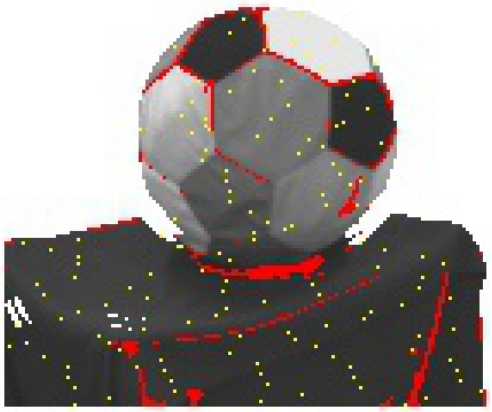


**Fig. 4.3:** Punti scelti per la registrazione

Nella figura 4.3 sono stati proiettati sull'immagine della videocamera, opportunamente convertita in scala di grigi, i punti della vista utilizzati per la registrazione: i punti colorati di rosso sono i 700 punti più rilevanti dell'intera vista, mentre i 200 colorati di giallo sono quelli scelti in modo random. Per rinforzare il concetto evidenziato in precedenza si può notare come i punti rilevanti sono spesso concentrati nelle regioni di maggior

interesse come i bordi del tavolo, i risvolti della stoffa che lo ricopre e sui cambi di colore nella texture del pallone tralasciando completamente tutto il resto della scena. I punti random permettono di aggiungere vincoli meno significativi rispetto a quelli generati con i punti rilevanti ma garantendo una distribuzione maggiore degli stessi.

La percentuale dei punti random da aggiungere a quelli estratti per rilevanza è un parametro importante che sarà oggetto di studio e di test, infatti dev'essere sufficientemente alto da produrre una copertura estesa della vista, ma allo stesso tempo non deve sovrastare..

## 4.3 Merging delle viste

Il processo di merging consiste nel fondere i dati 3D delle singole viste in una singola rappresentazione geometrica. Questa operazione viene fatta progressivamente dopo ciascuna fase di allineamento, quindi il modello tridimensionale all'iterazione zero è costituito solo dalla prima vista ed ad ogni iterazione la vista i-esima viene registrata sulla vista immediatamente precedente e i suoi punti vengono appesi al modello dopo esser stati opportunamente allineati. La fase di merging quindi consiste solamente nel mantenere in memoria il modello tridimensionale incrementale.

## 4.4 Cleaning modello finale

Consideriamo uno scenario in cui acquisiamo con una media di 30fps, nell'ipotesi di essere interessati ad un oggetto di dimensioni medie possiamo scartare tutti i punti relativi al resto della scena e arriviamo ad una stima realistica di appendere al nostro modello tridimensionale incrementale circa 15000 punti ad ogni frame (1/2 dei punti acquisiti).

Dopo un secondo il modello conterrà circa mezzo milione di punti e se proseguiamo l'acquisizione dopo una decina di secondi avremo già superato i 5 milioni. Considerando che per ogni punto oltre alle coordinate spaziali (x,y,z) mantenuti in memoria come float (32 bit) vengono mantenute anche le componenti del vettore normale (tre double) e l'informazione del colore (8 bit per componente) lo spazio complessivo occupato da un milione di punti è di circa 27 MB e viene generato in un paio di secondi.

Questi semplici calcoli servono sono solo per mostrare come l'informazione mantenuta dal modello tende ad aumentare vertiginosamente, e la stima è semplificata al caso di ricostruzione di un oggetto perché se invece fossimo interessati all'intera scena ad ogni frame dovremmo appendere al modello più di 25000 punti.

Questo problema può essere ridotto molto e la soluzione è la risposta alla seguente domanda: tutti questi punti che via via vengono appesi al modello

incrementale sono sempre utili ad aggiungere dettagli o qualità al modello stesso?

La risposta è un incoraggiante no, infatti molti dei punti che aggiungiamo appendendo una nuova vista sono già stati acquisiti anche da viste precedenti e quindi rappresentano dei duplicati costosi da mantenere che possono essere accorpati.

Per questo durante il processo di merging ad intervalli regolari è viene inserita una pulizia del modello che riesca a ridurre il numero di punti senza però eliminare dettagli geometrici preziosi.

L'algoritmo di pulizia implementato non solo riduce il numero di punti eliminando i punti che presentano le medesime coordinate (x,y,z) ma anche quelli che hanno una distanza inferiore ad una soglia impostata ( i punti eliminati vengono considerati dunque come punti duplicati).

Se geometricamente i punti duplicati sono simili e quindi non danno informazioni aggiuntiva, il valore di colore ad essi associato può essere diverso vista l'influenza dell'angolo di incidenza della luce durante l'acquisizione. Per cui se consideriamo un insieme di punti geometricamente equivalenti l'informazione di colore più rilevante sarà quella del punto con la normale più perpendicolare all'ottica della videocamera nell'istante dell'acquisizione.

L'algoritmo che ne deriva è questo:

INPUT:

- nuvola di punti 3D I $\{< p_i,\ N_i(x,y,z),\ rgb_i > coni = 1,\ldots,n\}$ dove $N_i$ e $rgb_i$ sono rispettivamente la normale e l'informazione di colore associate al punto $p_i$ mentre $n$ è la cardinalità di I.
- distanza D

OUTPUT:

- nuvola di punti O $\{< p_j,\ N_j(x,y,z),\ rgb_j > \mid d(p_i,p_j > d)\ \forall i \neq j \quad i,j = 1,\ldots,m\}$ con $m \leq n$

```
ALGORITMO:
 Crea nuvola di punti vuota O;
 for i=1 to n{
    if (in O NON c'è già un punto che dista da Pi meno di D) then
        inserisci il punto nella nuvola O (poiché nello spazio
        adiacente non c'è nessun altro punto);
    else
        Il punto non va inserito;
        if (Zp > Zb) then
            la normale del punto Pi è migliore di quella del punto
            Pb e quindi il colore è più affidabile quindi
            sostituisco i dati aggiuntivi di Pb con quelli di Pi;
 }
```

Durante l'operazione di eliminazione dei duplicati attorno al punto $p$ vengono fuse le informazioni di colore relative ai punti candidati ad esser eliminati con quelle del punto $p$. Sono possibili diverse strategie di merging del colore: il colore più reale sarà quello in cui ci saranno meno riflessi, e questa particolare situazione la si ottiene quando l'ottica della videocamera è parallela alla superficie nel punto corrispondente della scena. L'algoritmo implementato, considera il punto p e tutti i punti da eliminare ed estrae le informazioni di colore dal punto con la normale più perpendicolare rispetto alla superficie. In questo modo il colore del modello finale risulta essere il più vicino possibile al colore reale dell'oggetto.

## 4.5 Rendering del modello

In questa fase viene generata l'immagine di sintesi a partire dalla rappresentazione digitale dell'oggetto (nuvola di punti colorata). Questa parte non è stata oggetto di studio, infatti ci si è appoggiati alla libreria di visualizzazione VTK che si occupa di generare la mesh a partire dalla nuvola di punti, definire la vista voluta, backface culling e clipping sul volume di vista per eliminare le parti della scena esterne alla vista e infine applica le dovute trasformazioni per proiettare sul piano immagine per esser visualizzate.

# Capitolo 5

# Risultati sperimentali

La configurazione di prova per i test è data da un pc Dell XPS420 con le seguenti caratteristiche tecniche:

- Procesore Intel Core 2 Quad CPU Q6600 (2.40 GHz)
- Memoria RAM, 4GB DDR2
- Scheda video: ATI Radeon 3870 512 memoria integrata
- Sistema operativo: Windows 7 x64 Enterprise

Come si può notare, si tratta di una configurazione appartenente al mercato consumer di circa 5 anni fa, inferiore quindi agli standard odierni, specie per quanto riguarda il processore in dotazione.

I dataset utilizzati per i test sono stati acquisiti in laboratorio in condizioni di luce naturale. Per ciascuna scena è stata catturata una sequenza di 40-150 frame che, ad un framerate di 25fps, corrispondono ad acquisizioni di pochi secondi. Questa scelta è una diretta conseguenza dei limiti della ricostruzione di un modello 3D basato unicamente su un approccio ICP pair-wise: anche se i vincoli calcolati tra un frame e il successivo sono buoni, la catena di allineamenti genera un errore incrementale che porta ad un degeneramento della qualità della ricostruzione.

Nella preparazione delle scene, gli oggetti sono stati scelti variando la forma, le dimensioni, i materiali e il colore, in modo da avere un dataset piuttosto generale di oggetti.

## 5.1 Fusione della geometria e del colore

In questo paragrafo sono stati inseriti i risultati di alcune operazioni di fusione dell'informazione geometrica e del colore in un'unica vista.

Le scene analizzate sono un pallone di stoffa di diametro di circa 50 cm con gli esagoni di diverso colore, un set di oggetti posizionati sopra un tavolo, una persona e una parte della parete del Laboratorio LTTM.

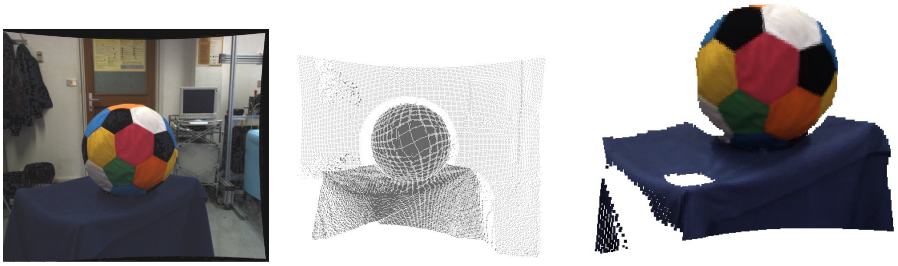


**Fig. 5.1:** Fusione della geometria e colore di un pallone di stoffa

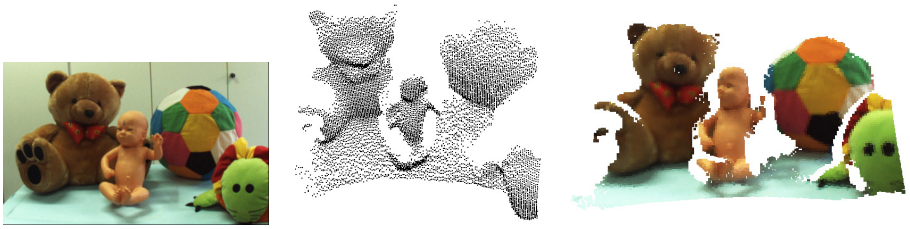


**Fig. 5.2:** Fusione della geometria e colore di un insieme di oggetti

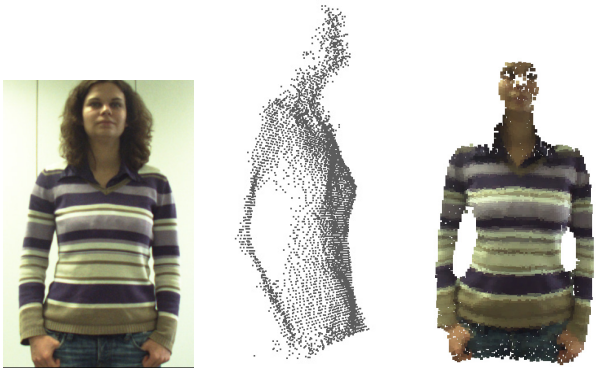


**Fig. 5.3:** Fusione della geometria e colore di una donna

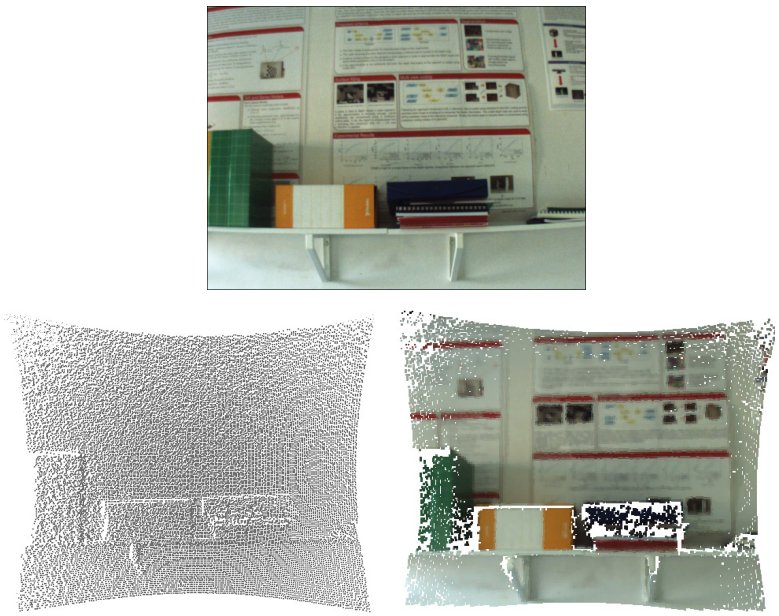


**Fig. 5.4:** Fusione geometria e colore di una parete

## 5.2 Estrazione dei punti salienti

Nel paragrafo 4.2 è stato descritto l'algoritmo per discernere in una scena i punti salienti o rilevanti sia dal punto di vista geometrico che da quello del colore. Il comportamento di questo algoritmo è fortemente influenzato dal parametro T che, rappresentando la soglia sotto la quale due punti sono simili, pesa la rilevanza geometrica e di colore di ciascun punto.

Più precisamente nel calcolo della somiglianza geometrica tra due punti $p$ e $q$, il parametro T indica l'angolo massimo tra i rispettivi vettori normali alla superficie perchè i due punti siano simili. In modo analogo nel calcolo della somiglianza di colore tra due punti, il significato del parametro è simile, poichè indica l'angolo massimo tra i vettori dei rispettivi punti nello spazio dei colori CIE-Lab.

Anche se il ruolo di T è analogo sia nel calcolo della rilevanza geometrica sia in quella del colore, gli spazi in cui consideriamo i vettori sono completamente differenti, per cui ad esempio un angolo di 5°tra due vettori normali alla superficie indica una somiglianza pressochè totale, mentre nello spazio dei colori indica una differenza notevole. Queste premesse servono a giustificare un diverso settaggio di T per i due algoritmi di calcolo di rilevanza.

Sono stati riportate due sequenze di immagini che mostrano i cambiamenti nel comportamento dell'algoritmo al variare di questo parametro fissando ovviamente la stessa scena.

La scena inquadrata (vedi figura 5.1)è composta da un tavolino su cui è appoggiato un pallone rotondo di stoffa.

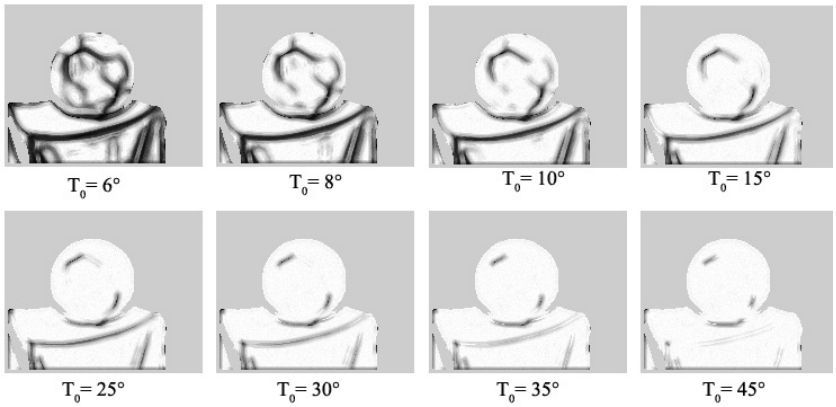



**Fig. 5.5:** Distintività geometrica al variare della soglia di rilevanza

Dalle figure 5.5 e 5.6 si può notare in quale modo il parametro T influisca nel far emergere i bordi del tavolo nella rilevanza geometrica, e i bordi degli esagoni del pallone nella rilevanza di colore. Nelle immagini in figu-

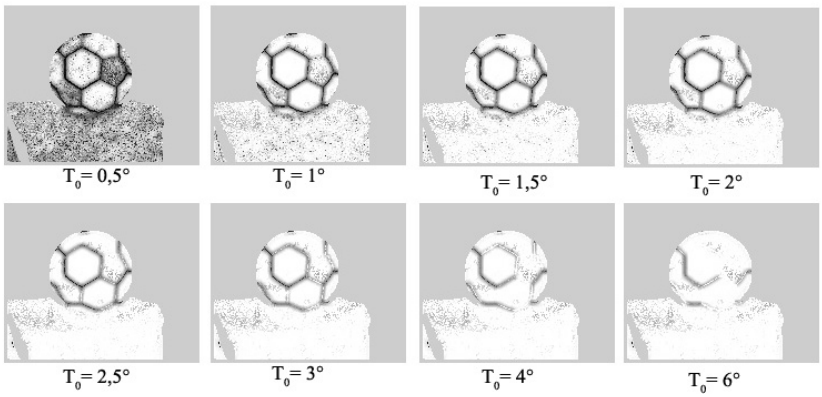



**Fig. 5.6:** Distintività del colore al variare della soglia di rilevanza

ra 5.5 si notano dei punti rilevanti sulla superficie del pallone che sembra essere totalmente omogenea. Questo sbaglio dell'algoritmo è dovuto al rumore esagerato prodotto dal ToF esattamente lungo i bordi degli esagoni che confonde l'algoritmo.

Nell'algoritmo per ogni punto viene preso il valore minimo tra la rilevanza geometrica e quella di colore per poter individuare la caratteristica migliore di ciascun punto. C'è quindi la necessità di calibrare il parametro $T_{Geometria}$ e $T_{Colore}$ in modo che i picchi di rilevanza geometrica e di colore siano confrontabili.

Per questo motivo sono stati scelti $T_{Geometria} = 15°$e $T_{Colore} = 1°$. I test nel resto del capitolo sono stati effettuati con questi valori.

Ecco alcuni esempi di come vengono rilevati i punti rilevanti nella geometria e nello spazio dei colori nelle diverse situazioni.

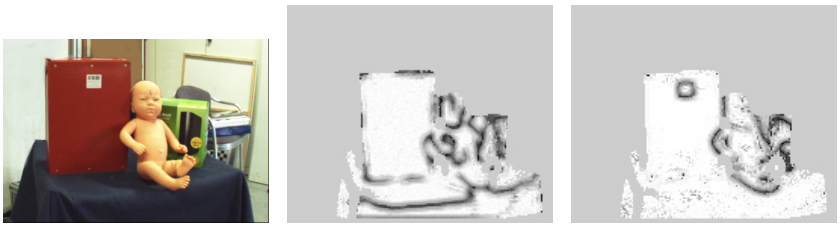


**Fig. 5.7:** Da sinistra a destra: (a) Immagine di un insieme di oggetti (b) immagine della rilevanza geometrica (c) rilevanza del colore

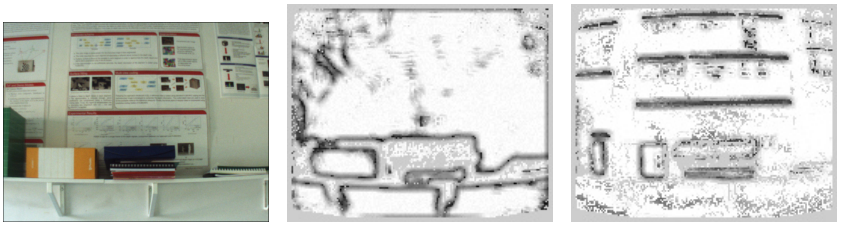


**Fig. 5.8:** Da sinistra a destra: (a)immagine della parete acquisita (b) rilevanza geometrica (c) rilevanza del colore

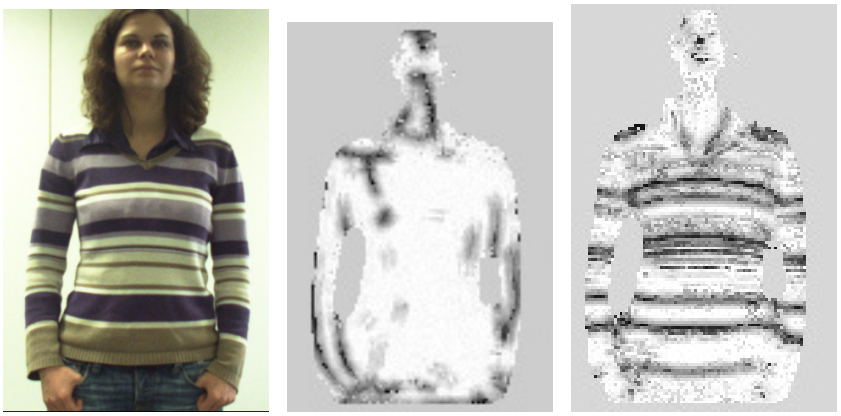


**Fig. 5.9:** Da sinistra a destra: (a) Immagine di una persona acquisita (b) immagine della rilevanza geometrica (c)immagine della rilevanza di colore

## 5.3 Algoritmo ICP

In questo paragrafo viene illustrato un esempio di allineamento di due frame utilizzando l'algoritmo ICP pair-wise.

In figura 5.10 sono visualizzati i due frame sovrapposti prima della fase di allineamento

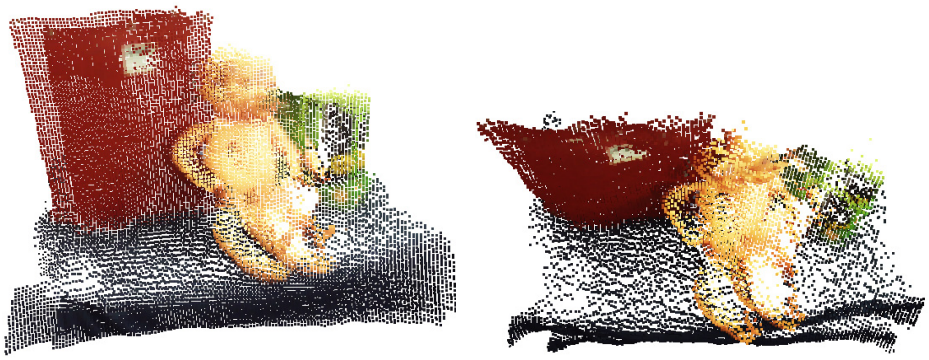


**Fig. 5.10:** Frame sovrapposti prima della registrazione

In figura 5.11 è visualizzata la fusione dei due frame dopo la fase di allineamento utilizzando il metodo Horn per il calcolo della rototraslazione.

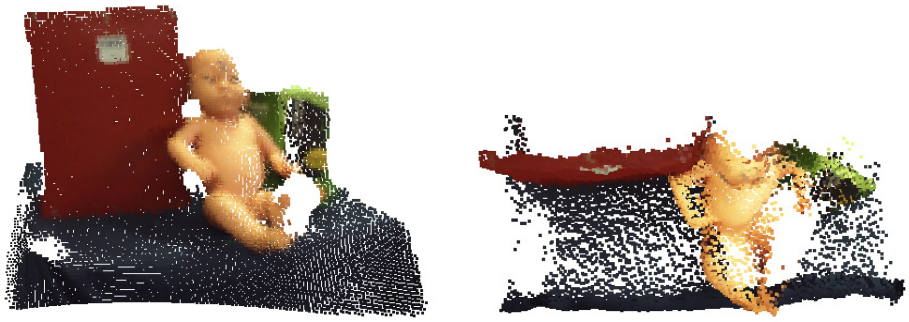


**Fig. 5.11:** Frame dopo la registrazione

## 5.4 Ricostruzione di un oggetto

In ques'ultimo paragrafo sono stati riportati alcuni esempi di ricostruzioni 3D a colori utilizzando la pipeline di ricostruzione descritta nella tesi.

Nell'immagine 5.12 sono stati riportati i risultati della ricostruzione di una persona ottenuti sfruttando a vari gradi il colore nell'allineamento.

In (a) il colore non viene sfruttato ne per l'estrazione dei punti rilevanti ne per il pruning delle false coppie di punti corrispondenti. Nell'immagine (b) è stato inserito il pruning delle coppie di punti corrispondenti che non avevano una somiglianza di colore ma non l'estrazione dei punti rilevanti. Infine nella ricostruzione (c) sono stati utilizzati entrambi i metodi.

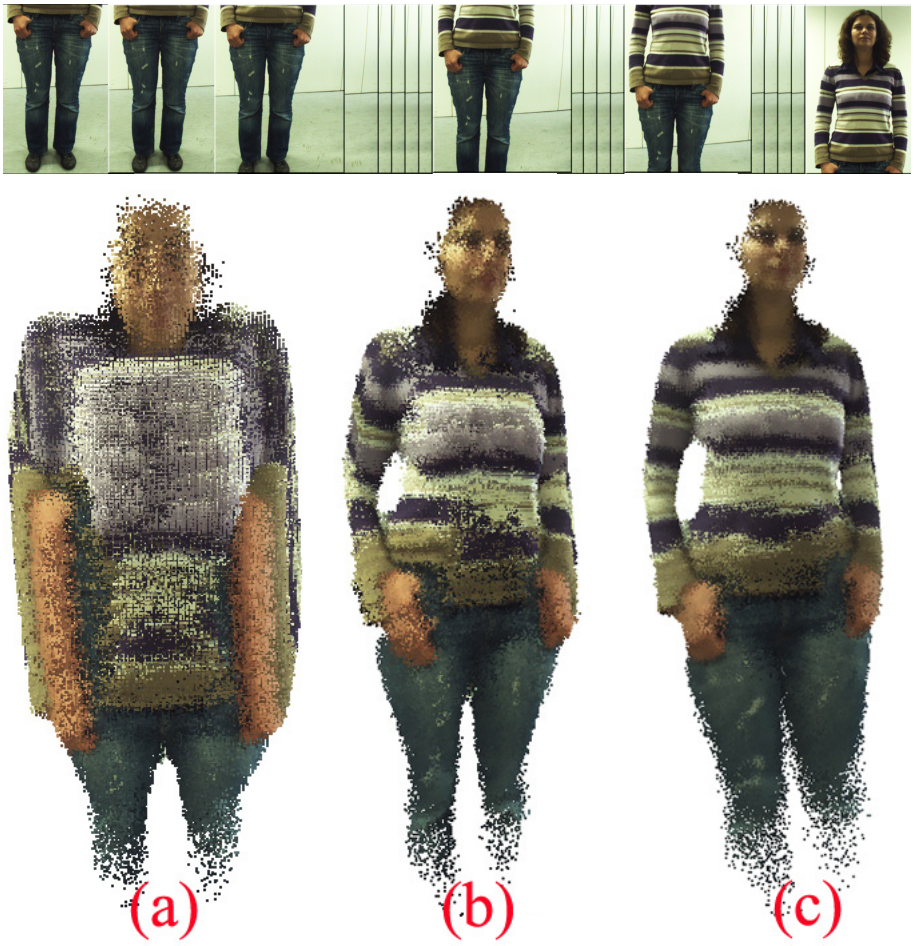



**Fig. 5.12:** Ricostruzione 3D di una persona. La ricostruzione è ottenuta dall'allineamento di 80 frame acquisiti con una sciata verticale del sistema di acquisizione inquadrando la silhouette come mostrato nella figura più in alto.

Questa ricostruzione è stata scelta con l'intento di mettere alla prova l'utilizzo del colore durante gli allineamenti, poiché pur essendo una forma geometrica complessa in realtà non offre particolari appigli visto che tutte le curve presenti hanno un raggio di curvatura ridotto e piuttosto omogeneo e non ci sono bordi netti. Proprio per questo in ricostruzioni come que-

sta lo sfruttamento dell'informazione del colore è fondamentale, ancor più considerando che il maglioncino di cotone e i jeans hanno una ricca texture.

Fatte queste considerazioni possiamo notare come l'algoritmo che non sfrutta il colore fallisca ogni allineamento (figura (a))fermandosi di volta in volta in minimi locali rappresentati da traslazioni verticali della silhouette.

Sfruttando il colore per eliminare le false coppie di punti corrispondenti si aiuta l'algoritmo a convergere verso una rototraslazione migliore, tuttavia non definendo vincoli particolarmente stringenti (i punti scelti sono casuali) molte registrazioni si fermano in minimi locali. Per questo motivo molti dettagli, come le mani o la texture del maglione, vengono persi o confusi.

Nella ricostruzione (c) invece, il colore viene sfruttato in entrambi i modi presentati in questa tesi, presenta dei miglioramenti in ogni allineamento, raggiungendo molto spesso un'ottima rototraslazione. Anche in questa ricostruzioni però sono presenti diversi allineamenti errati dovuti alla concentrazione dei punti rilevanti solo in alcune parti della scena (righe del maglione) che portano alcuni allineamenti a fermarsi in minimi locali. Oltre a questo si deve considerare l'effetto dell'errore incrementale.

**Fig. 5.13:** Vista frontale e dall'alto della ricostruzione di un pallone. La ricostruzione a 360°ha coinvolto 45 frame.

Le acquisizioni sono composte da una sequenza molto lunga di frame, ed un approccio di registrazione basato sull'ICP pairwise che minimizza l'errore di registrazione tra ogni frame e il precedente, non minimizza l'errore dell'intera ricostruzione. Infatti, l'errore si accumula allineamento dopo allineamento portando ad un disallineamento sempre più evidente. La vista dall'alto dell'immagine 5.13 descrive quest'ultimo concetto: al termine della ricostruzione a 360°l'ultimo frame non è allineato con il frame di partenza, perché l'errore accumulato è tale da aver perso all'incirca una ventina di gradi nella rotazione. Se la ricostruzione fosse stata diluita in più di 40 fra-

me, probabilmente il problema si noterebbe meno, ma il problema di fondo rimane ed è intrinseco nell'uso dell'algoritmo ICP pair-wise.

Per risolvere questo problema sarebbe necessario aggiungere una registrazione globale [**?**] al termine del calcolo degli allineamenti.

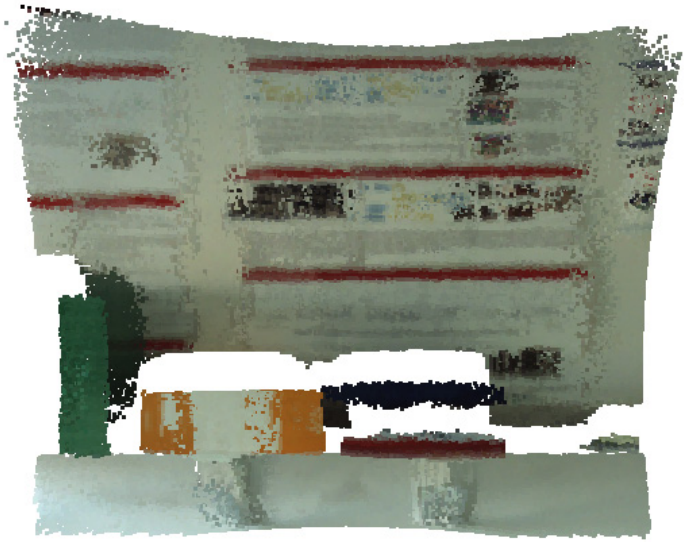


**Fig. 5.14:** Ricostruzione della parete ottenuta dalla fusione di 20 frame

Nelle ricostruzioni in figura 5.15 possiamo notare come, nonostante la fusione di 70 di frame i dettagli come il fiocco con i fiorellini gialli dell'orsetto di peluche o la texture sulla scatola di cartoncino verde siano rimasti evidenti. Questo fatto evidenzia come l'algoritmo si comporti meglio quando l'abbondanza di appigli geometrici o di colore è distribuita quasi uniformemente nella scena, e non concentrata in alcune zone come nel caso della parete. In ques'ultimo caso infatti, i punti rilevanti estratti sono concentrati lungo le striscie rosse orizzontali dei poster lungo la parete portando inevitabilmente a raggiungere dei minimi locali con delle traslazioni orizzontali rispetto all'allineamento ottimo.

**Fig. 5.15:** (sopra) Vista frontale della ricostruzione di una scena composta da un bambolotto di plastica, un raccoglitore di cartone rosso e una scatola di cartoncino riflettente. Per questa ricostruzione sono stati utilizzati 70 frame. (sotto) Vista frontale della ricostruzione di una scena composta da un orsetto di peluche, un bambolotto di plastica, un pallone di stoffa e una tartaruga di peluche. Questa ricostruzione è la fusione di 70 frame

# Capitolo 6

# Conclusioni

L'obiettivo principale di questo lavoro era riuscire a ricostruire dei modelli 3D a colori fondendo le informazioni di geometria e colore provenienti dai due diversi dispositivi in un'unica vista e costruendo iterativamente il modello aggiungendo di volta in volta il frame corrente.

Il sistema software creato implementa correttamente questi obiettivi, ma si possono fare molte considerazioni sui pregi e sui difetti di molti blocchi dell'elaborazione.

Il sistema di acquisizione descritto permette di acquisire dati fino a 25fps, anche se non raggiunge il framerate massimo a cui possono arrivare singolarmente la videocamera e il ToF, è sufficientemente alto per permettere di spostare l'apparato durante l'acquisizione in modo, ad esempio, da girare attorno ad un oggetto senza doversi preoccupare che la mano poco ferma infici la registrazione. Per il cameraman l'acquisizione è molto facile, perchè la sensazione è di muoversi con una videocamera broadcast.

Lo sfruttamento del colore è risultato fondamentale per riuscire a vincolare quelle scene in cui la geometria, da sola non riusciva a farlo. Tuttavia come risultato dai test sulla parete anche questo approccio al colore, da solo, non basta per vincolare totalmente un allineamento. Nell'ottica di sfruttare la geometria e il colore nell'allineamento l'introduzione del concetto di rilevanza ha portato a concentrare lo sforzo della registrazione nei punti più vincolanti.

Questo lavoro lascia aperte tantissime strade per migliorare il sistema di registrazione: la necessità di affinare la tecnica di estrazione dei punti rilevanti spingendone una distribuzione più omogenea nelle zone rilevanti della scena, la fusione nel modello finale dei punti vicini, sviluppo di un algoritmo icp globale[16] per il raffinamento delle matrici di rototraslazione tra i frame e infine lo studio di un modello per descrivere qualitativamente un allineamento tra due frame in modo da render meno soggettivi i test e più automatizzata la fase del parameter tuning.

Accanto a questi miglioramenti una strada parallela da considerare è la

sperimentazione della ricostruzione sui dati generati dal dispositivo Microsoft Kinect. Visti i costi molto contenuti e la larga diffusione attraverso la console, se questo dispositivo permettesse delle buone ricostruzioni potrebbe essere la tecnologia giusta per far fare il salto a questo genere di applicazioni nel panorama commerciale.

# Appendice A

# Descrizione del sistema software

## A.1 Tool di acquisizione

Il Tool di acquisizione è il software che permette di definire e impostare il sistema di acquisizione desiderato, scegliendo la configurazione adatta in base ai dispositivi collegati.

Il tool gestisce sistemi eterogenei composti da una o più camere e dal ToF, quindi principalmente permette l'acquisizione dal solo ToF , dalle camere in Stereo, dal sistema trinoculare composto da Stereo e ToF, e infine l'accoppiata camera e ToF utilizzata nella fase dei test.

Le funzionalità del software sono raggruppate nelle diverse sezioni dell'interfaccia grafica, che possono essere considerate dall'alto verso il basso e da sinistra verso destra una pipeline da seguire per poter fare le acquisizioni: selezione e preparazione dei dispositivi da utilizzare, impostazione dei parametri per la sincronizzazione, scelta del sistema di acquisizione e avvio della stessa.

#### Sezione 1

All'avvio del programma, nel box di testo vengono automaticamente presentate le videocamere disponibili per l'acquisizione. Selezionando ciascuna camera, sulla parte destra è possibile vedere le caratteristiche e modificare risoluzione, formato e dimensione dei pixel e soprattutto il tempo di esposizione. Una volta eseguita la configurazione, premere Set Sensore per salvare i parametri.

**Nota:** Se è stata precedentemente compiuta la calibrazione fotometrica della videocamera utilizzando ad esempio il software Pylon Viewer, questo passaggio dev'essere saltato per non sovrascrivere le impostazioni.

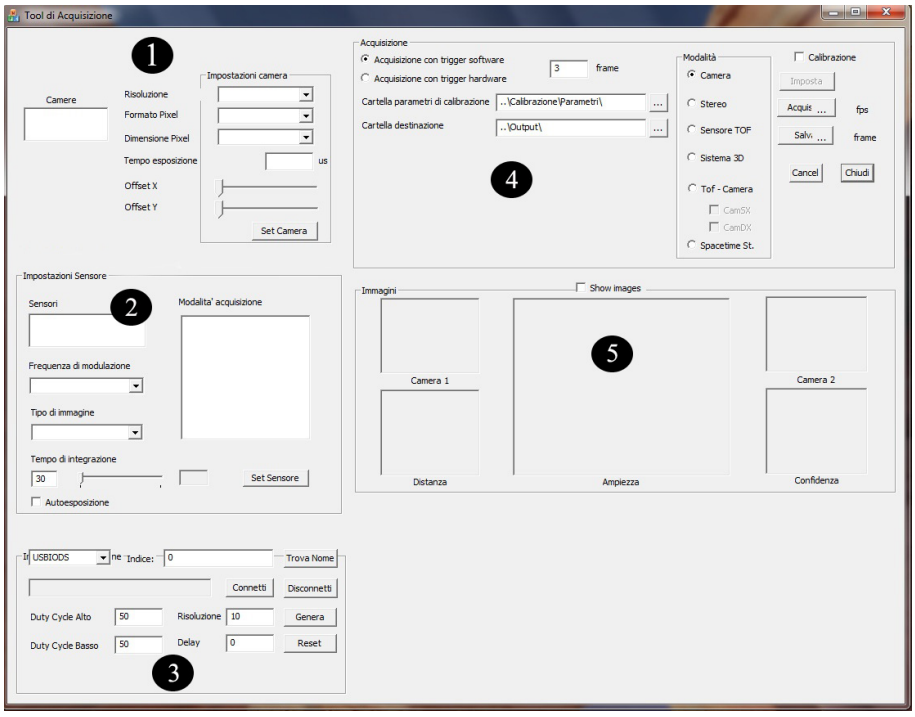



**Fig. A.1:** Interfaccia grafica del Tool di Acquisizione suddivisa in 5 sezioni

### Sezione 2

Questa sezione consente di inizializzare e configurare il sensore time of flight. E' possibile impostare la frequenza di modulazione del segnale inviato dal sensore, il tempo di integrazione e l'attivazione di alcune modalità di acquisizione che il sensore mette a disposizione.

Le opzioni tipiche selezionate sono AM_COR_FIX_PTNR, AM_CONV_GRAY, AM_CONF_MAP per generare ad ogni frame l'immagine di confidenza.Nel caso si utilizzi la scheda di sincronizzazione è necessario selezionare anche l'opzione AM_HW_TRIGGER.

**Nota:** Il tempo di integrazione, ossia il tempo di esposizione del sensore, dev'essere settato opportunamente perché incide molto sulla deviazione standard dell'errore e quindi in modo inverso sulla ripetibilità delle misure. Aumentare il tempo di integrazione fa diminuire la deviazione standard ma, se il tempo di integrazione è troppo elevato, c'è il rischio che il sensore saturi. Quindi la ripetibilità è ottimizzata se il tempo di integrazione è più lungo possibile, senza che però sia presente saturazione. Allo stesso tempo, bisogna considerare che il valore del tempo di integrazione implica direttamente il numero massimo di frame che si possono acquisire in un secondo.

### Sezione 3

Questa sezione è esclusiva al caso in cui si voglia utilizzare la scheda di sincronizzazione hardware per allineare gli istanti di acquisizione dei vari dispositivi connessi. In questa sezione è possibile configurare il duty cicle dell'onda del clock che controllerà i dispositivi definendo il numero di frame per secondo. Per ulteriori dettagli si veda [18].

**Nota:** Per raggiungere i fps stabiliti, è necessario che il tempo di esposizione delle videocamere e il tempo di integrazione del ToF siano più piccoli del duty cicle.

### Sezione 4

Questa è la sezione principale del Tool, dove si possono definire le specifiche dell'acquisizione. Anzitutto è necessario impostare il numero di frame da acquisire e la cartella di destinazione dell'output del programma. Il passo più importante però è la scelta della modalità di acquisizione, per i nostri scopi l'opzione migliore è ToF - Camera che permette anche nel caso siano collegate entrambe le camere di scegliere quale usare. Dopo aver scelto la modalità di acquisizione, è necessario cliccare il bottone Imposta e poi avviare l'acquisizione.

La sezione 5, se viene spuntata l'opzione, permette di visualizzare in tempo reale i frame che vengono acquisiti dai vari dispositivi. Al termine dell'acquisizione è possibile salvare le immagini relative alla videocamera e i dati provenienti dal sensore ToF(dati di profondità, immagini di ampiezza e confidenza). Come spiegato nel primo capitolo, i dati che vengono acquisiti potranno essere utilizzati offline dal tool di Elaborazione per la ricostruzione.

Se viene spuntata l'opzione calibrazione i frame vengono acquisiti uno alla volta indipendentemente dalle impostazioni precedenti. Il nome dell'opzione deriva dal fatto che questa opzione viene utilizzata quasi esclusivamente nella fase di acquisizione delle immagini per la calibrazione.

## A.2 Tool di elaborazione

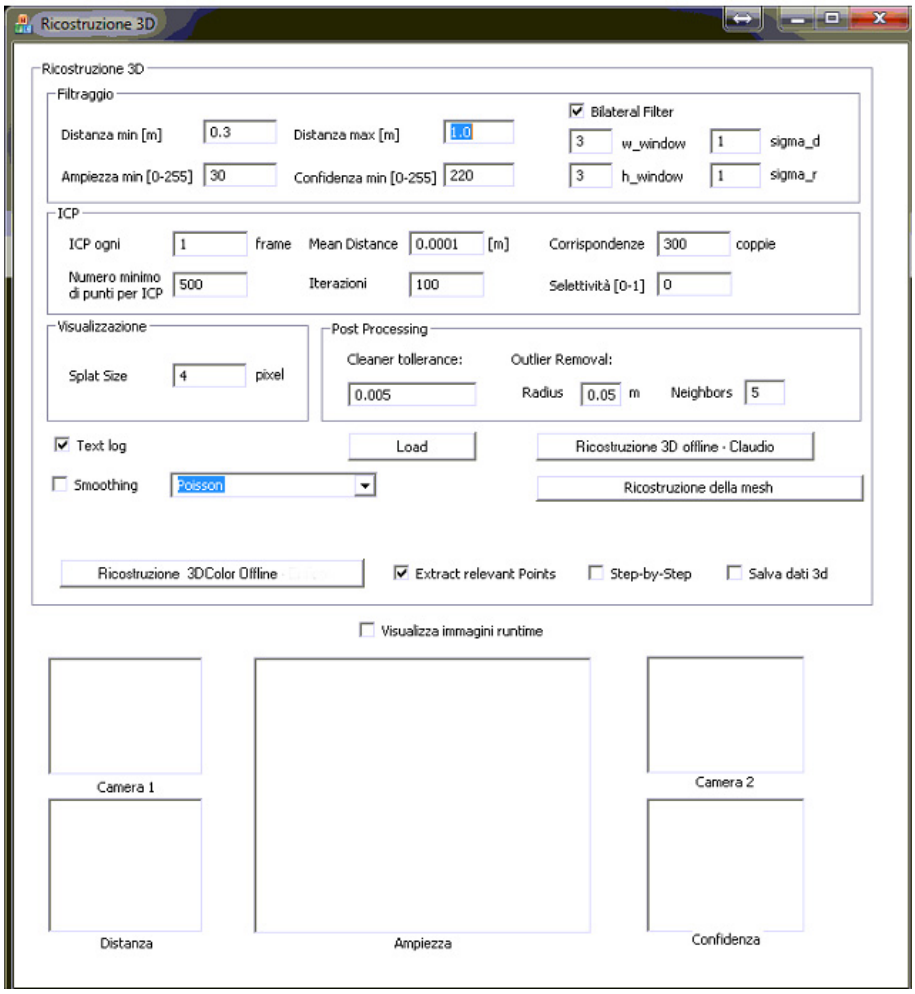



**Fig. A.2:** Interfaccia grafica del Tool di Acquisizione suddivisa in 5 sezioni

La parte superiore dell'interfaccia contiene i campi utilizzati nella fase di pre processing di ciascuna vista: la distanza minima e la distanza massima dei punti che si vogliono elaborare. Viene dunque definito uno spazio immaginario utile al di fuori del quale i dati semplicemente non vengono considerati e tutte le elaborazioni successive verranno effettuate in riferimento ai punti che hanno superato il filtraggio. Una scelta accurata di questi parametri permette tempi di esecuzione più rapidi e una migliore accuratezza in fase di ricostruzione.

Una seconda soglia di filtraggio è data dai valori minimi di ampiezza e confidenza, ma come descritto nel paragrafo 2.4 il loro utilizzo è sconsigliato

perché comportano l'eliminazione di molta informazione.

Inoltre è possibile attivare il filtro bilaterale (implementato via software) e aumentando le dimensioni della finestra si aumenta l'efficacia del filtro ma allo stesso tempo il tempo di esecuzione.

Il secondo blocco riguarda l'algoritmo di allineamento ICP, ed è possibile selezionare una varietà di parametri per meglio adeguare l'algoritmo alle particolarità della ricostruzione o per velocizzare l'esecuzione a scapito della qualità. L'ICP implementato può allineare frame successivi o frame distanti k frame l'uno dal l'altro, con k scelto dall'utente. Poichè infatti frame successivi sono fortemente sovrapposti, l'allineamento di frame successivi può risultare ridondante per piccole rotazioni della camera (o dell'oggetto), viene dunque data all'utente la libertà di scelta sulla frequenza dell'allineamento. Impostando k>1 i frame intermedi tra quelli da allineare verranno semplicemente scartati dal ciclo.

Il *numero minimo di punti di ICP* presente nella GUI fa riferimento al minimo numero di punti che un frame deve possedere per essere considerato valido: ad esempio se il volume considerato (impostando le soglie di distanza) contiene pochi punti il contenuto informativo del frame è considerato insufficiente per il processo di allineamento. Il programma gestisce questa situazione fermando il processo di registrazione delle viste e riattivandolo al nel momento in cui un frame torna ad avere un numero sufficiente di punti. Ovviamente è importante che la posizione relativa del frame con la quale si rientra nel processo di allineamento presenti una traslazione e rotazione piccola rispetto a quella del frame a cui dev'esser registrato. In caso contrario l'algoritmo ICP potrebbe fallire visto che è disegnato per far convergere nuvole di punti vicine. Se vedendo la ricostruzione ci si accorge di questo problema è necessario rifare un'altra acquisizione.

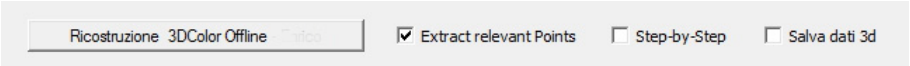
Sempre nella GUI l'utente può forzare la convergenza di ciascun allineamento in un determinato numero di iterazioni, o più semplicemente forzare la convergenza ad ottenere un errore (espresso in metrica *Root Mean Square*) inferiore alla soglia impostata. Inoltre è molto importante impostare il numero massimo di corrispondenze di punti per la stima della rototraslazione tramite il metodo di Horn in un contesto RANSAC. La variazione di questi parametri influenza la robustezza a scapito della velocità del processo di allineamento dei frame.

Per quanto riguarda la fase di visualizzazione con *Splat size* si intende la grandezza con cui vengono rasterizzati i punti.

La fase di cleaner avviene alla fine del processo di ricostruzione, ma anche ogni qual volta il modello incrementale diventa troppo pesante per esser gestito correttamente dalla libreria VTK. La *cleaner tollerance* è la distanza (espressa come percentuale della diagonale della bounding box del modello) entro la quale punti vicini vengono considerati come un unico punto e dunque fusi. Invece radius e neighbors controllano i parametri dell'algoritmo di

eliminazione degli outlier (un punto è outlier se, in un intorno sferico di raggio radius, ha un numero di vicini inferiore a *neighbors*).

La parte principale dell'interfaccia è:




Il bottone *Ricostruzione 3DColor Offline* permette di effettuare la ricostruzione 3D a colori a partire da dati di geometria e colore. In seguito al click si apre una finestra da cui è possibile navigare nelle directory e selezionare le immagini per la ricostruzione.

Il bottone *Extract relevant Points*, selezionato per default, setta l'utilizzo dei punti rilevanti in base alla geometria e al colore per l'allineamento dei frame Source nel Target. In caso contrario i punti del Source da utilizzare per la registrazione vengono estratti casualmente.

L'opzione *salva dati 3D* è una feature preziosa in fase di analisi e debug in quanto salva la nuvola di punti a colori generata dalla fusione del target e del source dopo l'applicazione della rototraslazione e inoltre salva la nuvola completa ottenuta al termine della ricostruzione.

L'opzione *Step-by-Step* permette di visualizzare al termine di ciascun allineamento la fusione delle nuvole Source e Target e il modello generato fino a quel punto, con la possibilità di iteragire via mouse per ispezionare la qualità della mappatura di allineamento e fusione geometria e colore.

# Bibliografia


[1] Kinect for windows. www.microsoft.com/en-us/kinectforwindows.

[2] Meshlab. http://meshlab.sourceforge.net.

[3] Overview mesa sr 4000. www.mesa-imaging.ch/prodview4k.php.

[4] *Registration and Integration of Texture 3-D Data*, Sept 1996.

[5] A.Fusiello. Visione computazionale - appunti delle lezioni. http://ilmiolibro.kataweb.it/schedalibro.asp?id=387047. Pubblicato a cura dell'autore, Giugno 2008.

[6] A.W.Moore. Chapter 6 kd-trees for cheap learning. 1991.

[7] G.M Cortelazzo C. Dal Mutto, P.Zanuttigh. *Time-of-flight Cameras and Microsoft KinectTM*. Springer, to Appear in 2012.

[8] R.Manduchi C.Tomasi. Bilateral filter for gray and color images. In *Proceedings of the 1998 IEEE International Conference on Computer Vision*, pages 839–846, jan 1998.

[9] Y.Chen e G.Medioni. Object modelling by registration of multiple range images. *Image Vision Comput*, 10:145–155, April 1992.

[10] P.Besl e N. McKay. A method for registration of 3-d shapes. *IEEE transactions on pattern Analysis and Machine Intelligence*, 14:239–256, 1992.

[11] A. Albarelli E. Rodola, A. Torsello. Sampling relevant points for surface registration. *2011 International COnference on 3D imaging*, pages 290–295, May 2011.

[12] H. Rushmeier F. Bernardini. The 3d model acquisition pipeline. *Computer Graphics Forum*, 21(2):149–172, 2002.

[13] P.Zanuttigh G.M.Cortelazzo, C. Dal Mutto. A probabilistic approach to tof and stereo data fusion. *3DPVT(Paris, France)*, May 2010.

[14] G.Xu H.Fukai. Fast and robust registration of multiple 3d point clouds. In *RO_MAN, 2011 IEEE*, pages 331–336, Aug 2011.

[15] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternion. *Journal of the Optical Society of America*, 4:629–642, March 1987.

[16] K.Pulli. Multiview registration for large data sets. *International Conference on 3D Digital imaging and modeling*, page 160, 1999. http://www-graphics.stanford.edu/ kapu/3dim99.pdf.

[17] M. A Fischler and R.C Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Readings in computer vision: issues, problems, principles and paradigms*, 1:726–740, March 1987.

[18] Alessio Marzio. Rilevamento della posizione della mano tramire immagini e dati 3d, 2011. Tesi di Laurea.

[19] M.Scherer. The 3d-tof-camera as an innovative and low-cost tool for recordings, surveying and visualisation - a short draft and some experiences -. *22nd CIPA Symphosium, YEAR = 2009, month = October 11-15, note = Kyoto, Japan.*

[20] C.Montani R.Scopigno R.Scateni, P.Cignoni. *Fondamenti di grafica tridimensionale interattiva.* McGraw-Hill, Milano, first edition, 2005.

[21] R.Szeliski. *Computer Vision: Algorithms and Application.* Springer, 2010.

[22] A.Crosnier S.Druon, M.J.Aldon. Color constrained icp for registration of large unstructured 3d/color data sets. In *2006 IEEE International Conference on Information Acquisition*, pages 249–255, Aug 2006.