



UNIVERSITÀ POLITECNICA DELLE MARCHE

FACOLTÀ DI INGEGNERIA

Corso di Laurea triennale in Ingegneria Informatica e dell'Automazione

**ANALISI E IMPLEMENTAZIONE DI SISTEMI
FOTOGRAMMETRICI IN CLOUD PER LA DOCUMENTAZIONE
SOTTOMARINA**

**ANALYSES AND IMPLEMENTATION OF PHOTOGRAMMETRICS
SYSTEMS FOR CLOUD SUBMARINE DOCUMENTATION**

Relatore:

Prof. Scaradozzi David

Tesi di laurea di:

Di Rado Simone

A.A. 2020 / 2021

Indice

CAPITOLO 1	3
1.1 INTRODUZIONE	3
1.2 PROBLEMA IN STUDIO E STATO DELL'ARTE	6
CAPITOLO 2	13
2.1 TEORIE ALLA BASE DELLA RICOSTRUZIONE 3D OTTICA.....	13
CAPITOLO 3	31
3.1 SOFTWARE COMMERCIALI A CONFRONTO	31
CAPITOLO 4	49
4.1 APPLICAZIONE DELLE TEORIE DI RICOSTRUZIONE E STUDIO SU PROBLEMATICHE PRESENTI NELL'AMBIENTE MARINO	49
BIBLIOGRAFIA	59
RINGRAZIAMENTI	60

CAPITOLO 1

1.1 INTRODUZIONE

Come suggerito dal titolo principale, la qui presente tesi tratterà di software di fotogrammetria ponendo particolare attenzione all'ambiente sottomarino.

La fotogrammetria rappresenta un insieme di metodi volti all'analisi ed alla ricostruzione 3D di un determinato ambiente, mediante l'utilizzo di sensori ottici. È dunque importante dedicare particolare attenzione alla calibrazione ed alla posizione nell'ambiente degli strumenti di cattura delle immagini.

La rappresentazione dello spazio 3D in maniera simmetrica è il fulcro della fotogrammetria ed è ormai da diverso tempo oggetto di studio. Ad essa, infatti, si diede un importante contributo già nella prima metà del '400 con l'introduzione della prospettiva centrale di Brunelleschi che pone delle regole geometrico-matematiche al fine di raffigurare oggetti tridimensionali su superfici piane, consentendo quindi un netto miglioramento nel passaggio dalla visione dell'oggetto alla sua rappresentazione

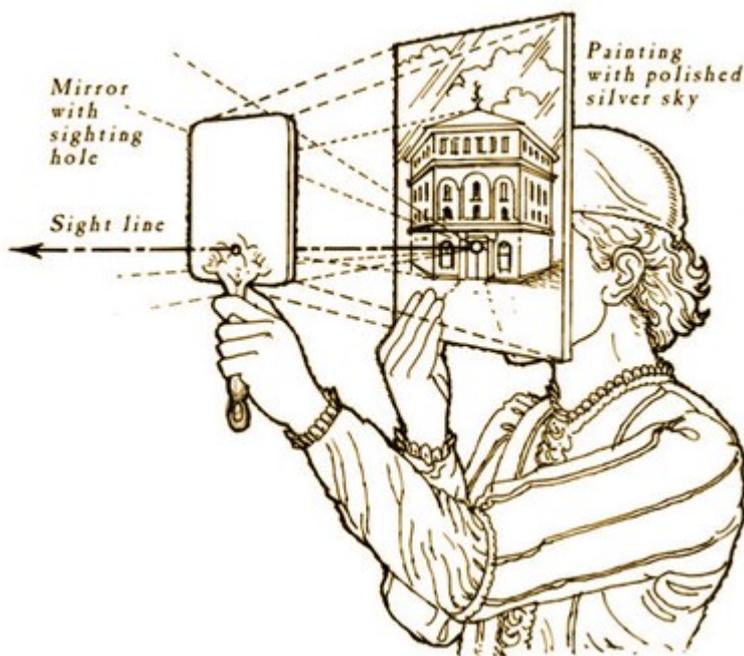


Figura 1.1.1, Esperimento di Brunelleschi (Fonte: www.botteghe dellearti.it)

Un paio di secoli dopo l'introduzione della prospettiva, arriva un altro importantissimo contributo da parte di Keplero nel campo dell'ottica geometrica. L'ottica è la branca dell'elettromagnetismo che descrive il comportamento della luce, le sue proprietà e come essa interagisce con la materia. Nel saggio "Ad Vitellionem Paralipomena" Keplero elabora una teoria riguardante la fisica della visione (teoria delle lenti, della riflessione, della rifrazione, costruzione delle immagini, concetti riguardanti il meccanismo della visione).

Prima di Keplero, altri avevano già formulato delle loro teorie riguardo il funzionamento dell'occhio; in particolare Pitagora nel VI secolo a.C., il quale sosteneva che l'occhio inviava dei raggi, come rette, per esplorare l'esterno, ma tale teoria non era sostenuta da filosofi come gli atomisti o Democrito, i quali ritenevano invece che dei simulacri emessi dagli oggetti venissero ricevuti dall'occhio.

Intorno al '300 a.C. Euclide riorganizzò le teorie di Pitagora in un trattato e spiegò le leggi della riflessione, dando quindi vita all'ottica geometrica.

Nei secoli successivi si cominciò a parlare di ottica geometrica sia in occidente, sia nel mondo islamico con, ad esempio, Alhazen (da lui la teoria che ogni punto emette infiniti raggi luminosi, ma solo i raggi che convergono nell'occhio determinano la visione). A portare le teorie di Alhazen in occidente fu Vitellione, un monaco polacco. Grazie al lavoro di Alhazen, l'abate Francesco Maurolico di Messina riuscì a spiegare il funzionamento dell'occhio. Nel 1604 Keplero riprende il lavoro di Francesco Maurolico nel suo "Ad Vitellionem Paralipomena", introducendo l'ottica geometrica moderna.

Si giunge così alla formulazione dell'inversione dell'immagine sulla retina, proprio come accade in una camera oscura, e dunque ai vari studi riguardo il capovolgimento delle immagini da parte del cervello.

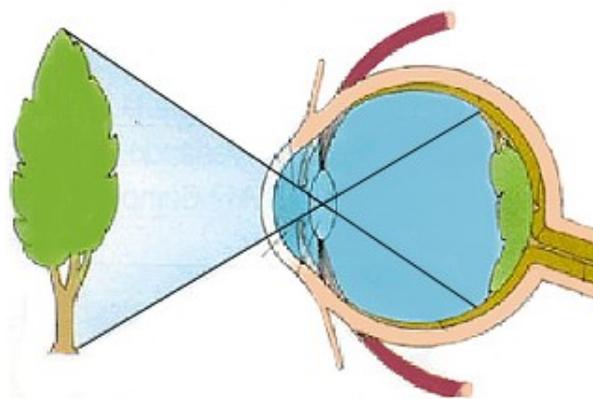


Figura 1.1.2, Immagine sulla retina (Fonte: www.occhioumano.weebly.com)

A seguito del periodo di rivoluzione informatica che ci ha poi portati nell'era dell'informazione, inevitabilmente ci si è chiesti come rendere un'immagine, catturata ad esempio da una macchina fotografica, comprensibile ad un calcolatore per poi essere elaborata in vari modi. Entriamo così nell'ambito della Computer Vision: un insieme di tecniche per la modellazione basata su immagini. In particolare, la Computer Vision riguarda la costruzione di modelli tridimensionali approssimati del mondo reale a partire dalle sue proiezioni bidimensionali, ovvero dalle sue immagini. Lo scopo principale della Computer Vision è dunque quello di riprodurre la vista umana servendosi di un sistema di visione.

Per sistema di visione si intende un apparato elettronico che integra una o più telecamere, sistema di illuminazione, sistema di acquisizione ed elaborazione delle immagini e chiaramente un software.

Ci ritroviamo quindi, dopo importanti scoperte, a parlare di analisi e di tecniche per l'implementazione di sistemi fotogrammetrici.

L'obbiettivo di questa tesi sarà:

- illustrare le caratteristiche principali della fotogrammetria, trattando specialmente la fotogrammetria sottomarina,
- illustrare le problematiche in studio e l'importanza nel portare avanti una ricerca in quest'ambito,
- descrivere i concetti teorici alla base della ricostruzione 3D ottica introducendo, dunque, anche le varie tecniche matematiche,
- analizzare e confrontare vari software fotogrammetrici commerciali più conosciuti,
- illustrare esempi di applicazione e possibili sviluppi futuri.

1.2 PROBLEMA IN STUDIO E STATO DELL'ARTE

1.2.1 FOTOGRAMMETRIA: DEFINIZIONE E CARATTERISTICHE

La fotogrammetria si definisce come una tecnica che permette di acquisire determinate caratteristiche di un oggetto o di un ambiente, principalmente forma e posizione, attraverso l'acquisizione di dati. Fin dalle sue origini, tale tecnica veniva utilizzata in ambito bellico e nella cartografia, mentre oggi viene utilizzata in diversi ambiti: architettura, ingegneria, geologia, archeologia, operazioni di polizia, cinematografia, ambito videoludico e persino per l'individuazione del petrolio.

Fino a poco tempo fa, il costo degli strumenti necessari non ne permettevano il pieno utilizzo. Lo sviluppo di calcolatori capaci di gestire un'ingente mole di dati e della grafica computerizzata l'hanno resa semplice da utilizzare e hanno ridotto di molto i costi, tanto che cominciò ad essere utilizzata anche in ambiti in cui raramente lo era, rendendo obsolete le apparecchiature ottiche usate fino a quel momento.

Al giorno d'oggi la fotogrammetria è uno degli strumenti per l'acquisizione dei dati più affidabili, precisi ed economici.

I vari tipi di fotogrammetria si differenziano tra loro secondo due criteri:

- **Distanza della ripresa:** a seconda della distanza da cui catturiamo l'oggetto, ottenendone quindi delle immagini bidimensionali che lo raffigurano, dividiamo la fotogrammetria in: microfotogrammetria, fotogrammetria degli oggetti vicini, fotogrammetria architettonica, fotogrammetria terrestre, fotogrammetria aerea, fotogrammetria satellitare, fotogrammetria da UAV.
- **Tipo di dato in output:** a seconda del tipo di dato che il processo fornisce in output dividiamo la fotogrammetria in: fotogrammetria tradizionale che fornisce in output un dato disponibile su pellicola fotografica e fotogrammetria digitale che fornisce in uscita un dato digitalizzato (tipologia trattata in questa tesi).

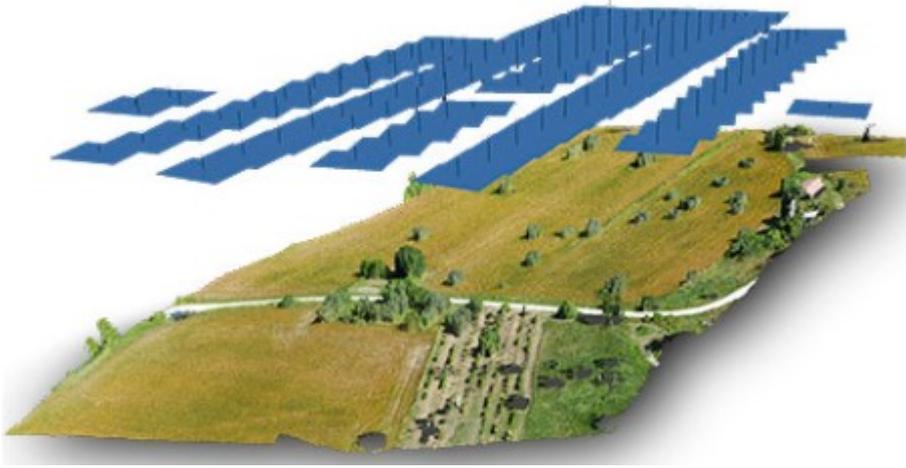


Figura 1.2.1.1, Esempio di fotogrammetria aerea con drone (Fonte: www.aeromatic.com)

1.2.2 FOTOGRAMMETRIA SOTTOMARINA

Dopo aver dato una definizione di fotogrammetria e aver illustrato alcune principali caratteristiche È possibile ora introdurre quello che è il punto focale di questa tesi: la fotogrammetria sottomarina. Si tratta della branca della fotogrammetria che, come suggerisce il nome, si occupa principalmente dell'analisi e la modellazione degli ambienti sottomarini.

Nel 1973 ci fu una crisi petrolifera che comportò l'avvio di alcune ricerche nel campo ingegneristico. Tali ricerche dovevano supportare le nuove tecnologie di prefabbricazione e assiemaggio di superpetroliere e piattaforme industriali da installare in mare aperto.

I governi: USA, Francia, Polonia, Germania Est e Gran Bretagna; insieme alle compagnie: Shell, BP, Exxon e Agip; iniziarono delle attività di rilevamento fotogrammetrico per la verifica dimensionale di elementi prefabbricati sulla terraferma e da assemblare poi in alto mare, spesso non in buone condizioni atmosferiche.

In seguito, l'agenzia Agip avviò un proprio programma di ricerca, per risolvere alcuni problemi legati al controllo dimensionale, per sostenere le piattaforme di produzione, che portò alla realizzazione di un'apparecchiatura installabile sia su nave, sia su elicottero.

Dunque, è questo il primo importante passo verso lo studio delle potenzialità della fotogrammetria marina. È stato proprio lo sviluppo delle attività industriali, e quindi il sempre più frequente

bisogno di controlli dimensionali sulle installazioni in mare aperto, ad aver favorito lo sviluppo e l'utilizzo della fotogrammetria in ambiente marino.

Quando si lavora in ambiente marino vi sono determinate problematiche da tenere in conto:

- La ripresa avviene mediante due camere di ripresa sincronizzate fra loro. Esse sono vincolate al mezzo (elicottero, nave, drone, robot sottomarino ecc.), ma non possono essere né livellate, né puntate verso l'oggetto con assi ottici esattamente imposti.
- Gli oggetti fotografati, trovandosi sott'acqua o comunque in superficie, potrebbero essere soggetti ad un lento e magari costante movimento o angolare o lineare esattamente come la superficie dell'acqua. Non è comunque una difficoltà che si presenta in ogni caso, dal momento che l'oggetto da fotografare potrebbe essere fermo o fissato al fondale marino.

Detto ciò, ci si rende subito conto che la fotogrammetria sottomarina non è oggetto di facile studio e presenta di per certo le sue difficoltà, ma è anche vero che ha da offrire notevoli e utili applicazioni nell'ambito marino, ad esempio, nel campo dell'archeologia subacquea.

Fin dal XV secolo l'uomo ha sempre trovato interesse nell'esplorare i fondali marini, dunque, ebbe inizio lo studio per la realizzazione di strumenti adatti all'esplorazione e al lavoro sott'acqua. Nel corso del tempo furono le scoperte e i recuperi di diverse opere d'arte greca a mettere in risalto l'importanza dell'archeologia subacquea, ma le risorse a quel tempo non erano sufficienti a rendere meno complessi gli studi dei reperti sott'acqua. È solo alla fine degli anni Settanta, grazie a diversi errori commessi e diverse conoscenze acquisite, che l'uomo riuscì ad entrare in possesso di tecniche per la raccolta di dati e informazioni di assoluto rigore scientifico.



Figura 1.2.2.1, Foto scattata durante l'esplorazione di un relitto (Fonte: www.books.openedition.org)

Gli scavi si moltiplicarono, infatti si poteva ormai contare su strumenti di lavoro efficaci ad esplorare gli ambienti subacquei e avanzati sistemi di documentazione grafica, fotografica e soprattutto fotogrammetrica. Negli anni Novanta l'interesse nel ricercare strumenti e tecnologie sempre più performanti spinge l'occhio dei ricercatori a prestare una particolare attenzione alla fotogrammetria e all'informatica. Rimaneva comunque indispensabile l'uso della Quadrettatura, tecnica che consiste nel suddividere l'area subacquea in tante aree di forma quadrata per poi passare alla documentazione grafica e fotografica di ognuna.

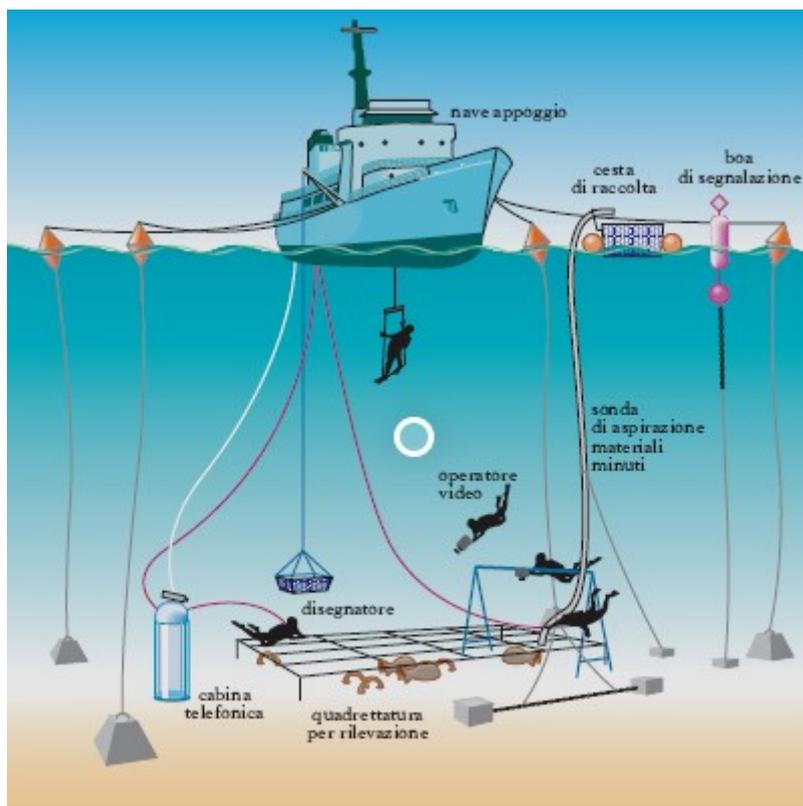


Figura 1.2.2.2, Rappresentazione schematica di un'analisi ambientale subacquea (Fonte: www.treccani.it)

Lo sviluppo di sistemi informatici ha dato il via a nuovi metodi di gestione, elaborazione e archiviazione delle immagini proprio come la fotogrammetria. Essa utilizza coppie di immagini stereoscopiche, elaborandole e restituendo in output determinati dati. La stereoscopia è una tecnica fotografica che consiste nel catturare due immagini del soggetto da punti di vista differenti con l'obiettivo di metterne in risalto la tridimensionalità.



Figura 1.2.2.3, Immagine stereoscopica del Colosseo (Fonte: Wikipedia)

Nel tempo la fotogrammetria è diventata, dunque, uno strumento talmente indispensabile per l'analisi e la ricerca in ambienti marini e non solo, da essere tutt'ora studiata. Lo scopo della ricerca

non è solo quello di migliorare e sperimentare nuovi metodi che mirino al miglioramento ed alla semplificazione della tecnica, ma è anche quello di trovare nuovi campi d'impiego.

Non è una novità che i reperti e i vari siti archeologici, giacenti sul fondale marino, siano da tempo oggetti di furto. Si pensi all'utilità che avrebbe una tecnica come la fotogrammetria nell'individuazione di un furto archeologico in ambiente subacqueo. Prima di concentrarsi sullo studio di un singolo reperto, si potrebbe pensare infatti di utilizzare la fotogrammetria per mappare l'intera area e controllare periodicamente se i reperti presenti sul fondale coincidano con quelli presenti sulla mappatura effettuata all'inizio. In questo modo, chiaramente, identificare un furto diventa estremamente più semplice e veloce.



Figura 1.2.2.4, Drone per la fotogrammetria subacquea (Fonte: www.archeomatica.it)

L'individuazione di un furto archeologico è solamente uno dei molteplici impieghi per la fotogrammetria a poter essere oggetto di studio. Da qualche anno si sta pensando anche all'impiego della fotogrammetria per monitorare le colonie coralline e la loro crescita, andando ad evidenziare i vantaggi e le possibili soluzioni a problematiche che l'utilizzo della tecnica potrebbe portare. L'idea sarebbe quella di mappare periodicamente le colonie, tenendo conto della velocità di crescita di ogni singolo corallo, e andare a confrontare le differenze tra ogni singola mappatura.

In sostanza, ci troviamo di fronte ad una materia pressoché giovane, almeno per quanto concerne l'ambiente subacqueo, e investire nella ricerca e nello sviluppo di software fotogrammetrici potrebbe regalare diverse soddisfazioni e risultare prospicuo in diversi campi.

È proprio sui software fotogrammetrici che voglio riporre particolare attenzione. Parlando di fotogrammetria terrestre, aerea, satellitare ecc. sono diversi i software che potremmo utilizzare sia open-source, sia close-source.

La mancanza di un software completamente dedicato alla fotogrammetria sottomarina è uno dei primi problemi da affrontare. Essendo la fotogrammetria sottomarina un argomento da approfondire e ancora in fase di studio, probabilmente ci vorrà del tempo poiché vi siano vari software completamente dedicati ad essa, così come già lo è per la fotogrammetria aerea ad esempio.

Proprio per questo durante l'attività di tirocinio ci si è posti, innanzitutto, l'obiettivo di comprendere le leggi matematiche dietro la fotogrammetria, al fine di evidenziare al meglio caratteristiche, punti di forza e svantaggi, per poi selezionare un gruppo di software, tra quelli più utilizzati, ed analizzarli. La comprensione dei concetti matematici permette anche di avere un'idea sui vari algoritmi implementabili per lo sviluppo di un software fotogrammetrico.

Lo scopo dell'analisi dei software è fondamentalmente quello di selezionare un vincitore, prendendo come parametro di valutazione delle immagini scattate sott'acqua e cercando di riprodurre un buon modello tridimensionale da queste.

CAPITOLO 2

2.1 TEORIE ALLA BASE DELLA RICOSTRUZIONE 3D OTTICA

La fotogrammetria sottomarina mira in particolare all'analisi ed alla ricostruzione 3D dei siti sottomarini. Pertanto, essa presenta le seguenti problematiche:

- Modellazione basata su immagini provenienti da una camera stenopeica;
- Calcolo dei parametri intrinseci e calibrazione degli strumenti di cattura con lo scopo di ottenere buone informazioni in 2D;
- Estrazione della forma geometrica tridimensionale degli oggetti;
- Rilevamento della posizione (relativa e assoluta) degli oggetti;
- Errori di misurazione dovuti alla precisione ed alla accuratezza della strumentazione;
- Programmi ed algoritmi da utilizzare;
- Uso esterno di altri modelli come game engine o stampante 3D.

2.1.1 COMPUTER VISION

Nel capitolo precedente, la computer vision è stata definita come l'insieme di tecniche per la modellazione basata su immagini. Essa riguarda infatti la costruzione di modelli tridimensionali approssimati del mondo reale a partire dalle sue immagini, definibili come sue proiezioni bidimensionali.

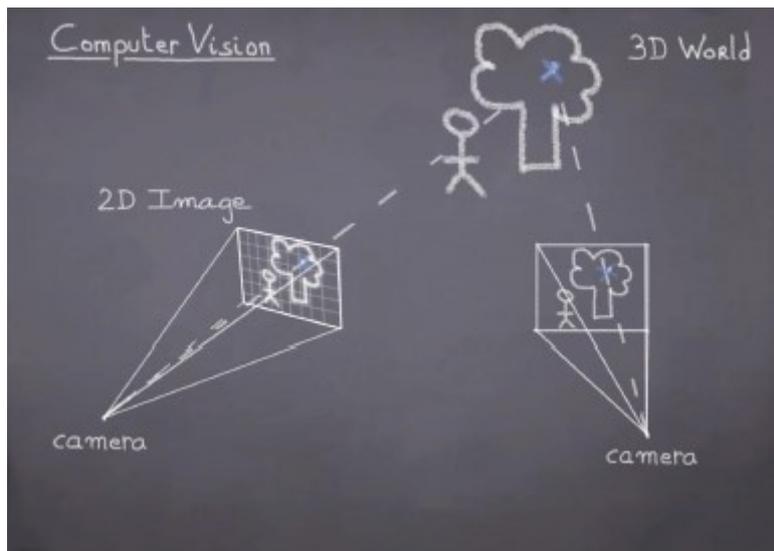


Figura 2.1.1.1

Ogni punto della scena, infatti, “influenza” diversi punti del piano immagine ed ogni punto del piano immagine è “influenzato” da diversi punti della scena.

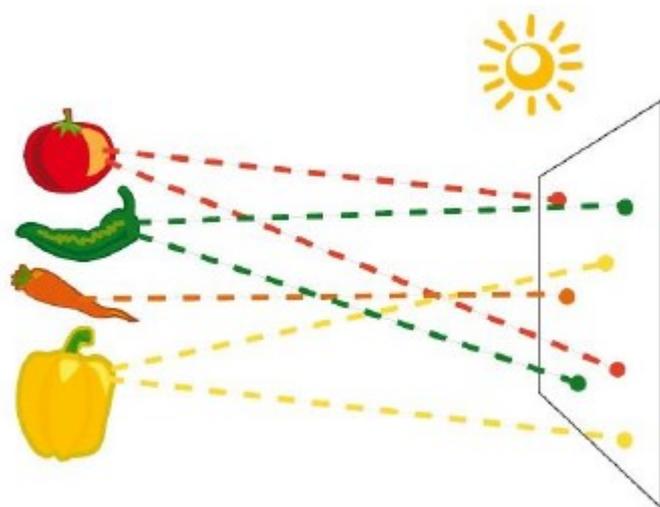


Figura 2.1.1.2

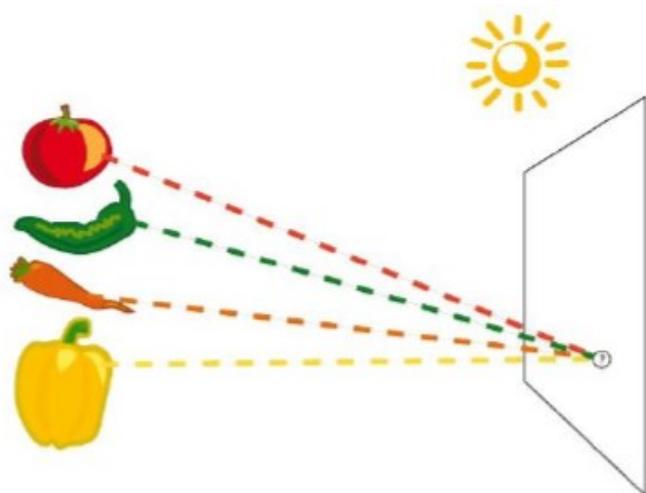


Figura 1.1.1.3

2.1.2 CAMERA STENOPEICA ED EQUAZIONE DELLALENTE SOTTILE

Qui di seguito vi è un modello semplificato di camera stenopeica.

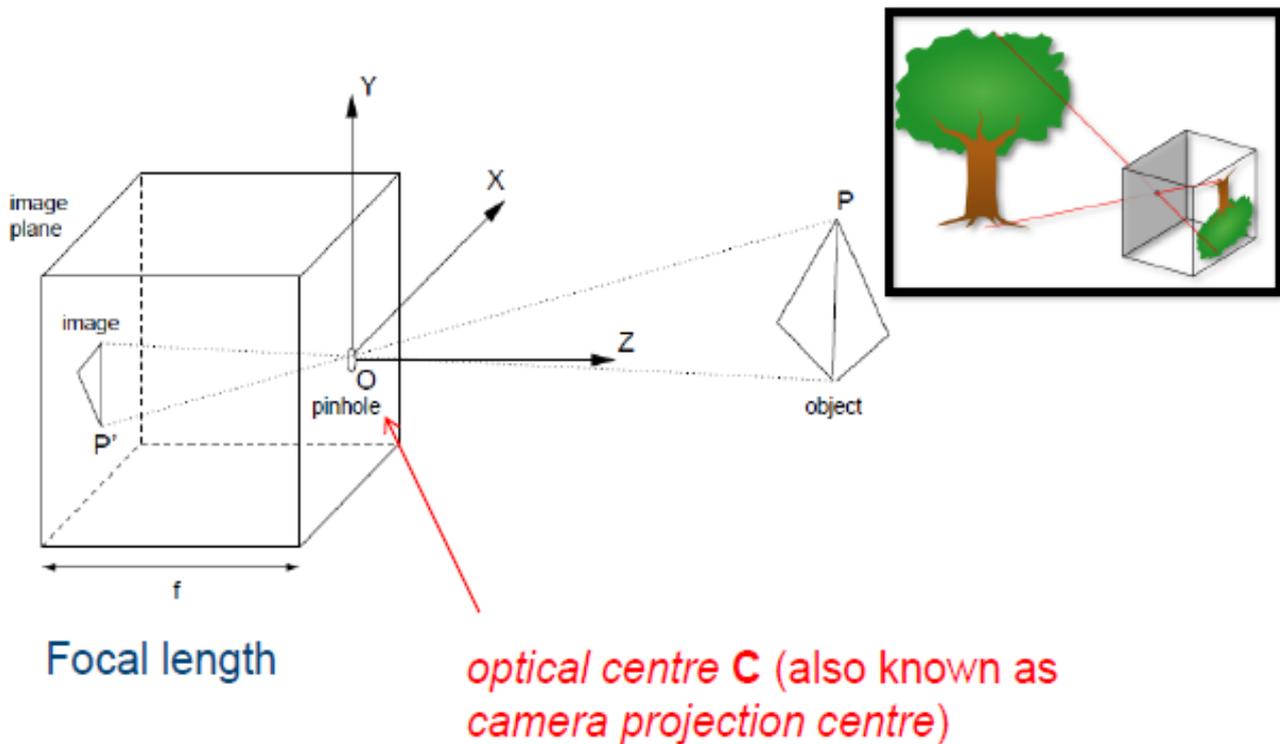


Figura 2.1.2.1

La camera presenta un foro stenopeico in grado di proiettare luce su un piano fotosensibile, creando un'immagine.

Profondità di campo, nitidezza e tempo di esposizione dipendono dalla dimensione del foro. Diminuire le dimensioni del foro implica un netto miglioramento della profondità e della nitidezza, ma un aumento del tempo di esposizione. Questo perché logicamente in un foro più piccolo passa una quantità di luce minore rispetto ad un foro di dimensione maggiore. Dunque, se l'obiettivo è quello di far passare più luce per diminuire il tempo di esposizione, bisogna aumentare le dimensioni del foro.

Volendo essere più precisi, il tempo di esposizione dipende dalla sensibilità del supporto fotografico utilizzato e dal rapporto del diaframma. È possibile calcolare quest'ultimo dividendo la lunghezza focale per il diametro del foro.

La lunghezza focale è un parametro importante che possiamo sfruttare anche nell'equazione della lente sottile. Prendiamo come grafico di riferimento la seguente figura.

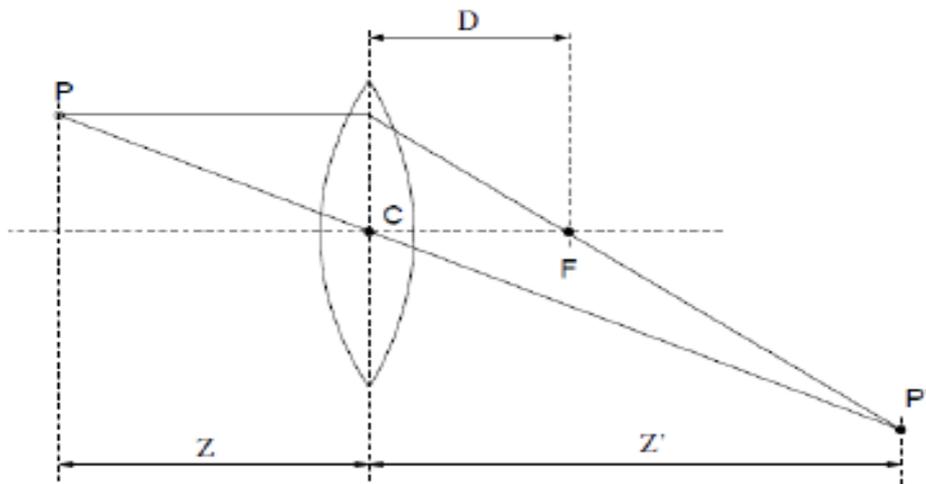


Figura 2.1.2.2

L'equazione della lente sottile descrive la relazione tra la distanza dell'oggetto Z e la distanza della sua immagine Z' dalla lente:

$$\frac{1}{Z} + \frac{1}{Z'} = \frac{1}{D}$$

In questo caso, con D si vuole indicare la distanza focale.

Al variare di Z o Z' , otteniamo un'immagine fuori fuoco, un cerchio chiamato centro di confusione. Se il centro di confusione non è più grande del sensore, l'immagine è a fuoco.

Introdotta la camera stenopeica e l'equazione delle lenti, ci si pone ora il problema di come poter digitalizzare le immagini catturate con la camera.

Assieme ad una camera stenopeica viene in genere utilizzato un dispositivo di accoppiamento di carica, chiamato anche CCD (Charge-Coupled Device).

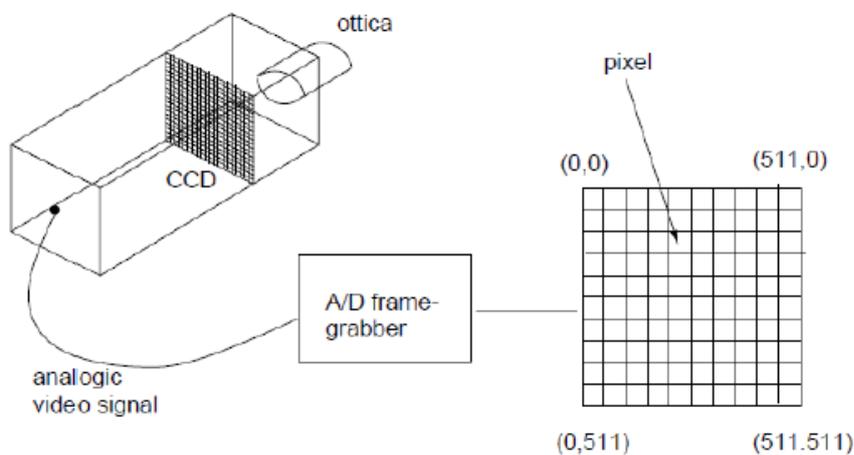


Figura 2.1.2.3

Si tratta di un circuito integrato composto da una griglia di semiconduttori in grado di accumulare una carica elettrica pari all'intensità della radiazione elettromagnetica che li colpisce. Gli elementi sono accoppiati in modo che ognuno, sollecitato da un impulso elettrico, possa trasferire la propria carica a quello adiacente. Inviando al dispositivo una sequenza temporizzata d'impulsi, si ottiene in uscita un segnale elettrico con il quale è possibile ricostruire la matrice dei pixel che compongono l'immagine proiettata sulla superficie del CCD stesso.

2.1.3 MATRICI DI PROIEZIONE E CALIBRAZIONE DELLA CAMERA

L'ambiente ed i punti immagine possono essere rappresentati da vettori omogenei e la proiezione della prospettiva può essere espressa come una moltiplicazione tra matrici:

$$\begin{pmatrix} fx \\ fy \\ z \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} * \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

La matrice che descrive la mappa è chiamata matrice di proiezione della camera **P**:

$$z * m = P * M \quad M = (x \ y \ z \ 1)^T, \quad m = \left(\frac{fx}{z} \quad fy \quad z \quad 1 \right)^T$$

Cambiare coordinate nello spazio è equivalente a moltiplicare la matrice **P** per la matrice 4x4 **G**:

$$G = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}$$

Le righe di **R** sono vettori che, insieme al centro ottico, definiscono il frame di riferimento della camera espresso in coordinate del sistema.

G descrive la posizione e l'orientamento della camera rispetto alle coordinate del sistema esterno. Essa dipende da sei parametri chiamati parametri estrinseci della camera.

Se si vuole un modello realistico, bisogna tener conto della pixelizzazione, quindi di forma e dimensione della matrice CCD. Ciò significa sostanzialmente, in termini matematici, moltiplicare alla matrice **P** la seguente matrice **K**:

$$K = \begin{pmatrix} \frac{f}{s_x} & \frac{f}{s_x \cdot \cot \theta} & o_x \\ 0 & \frac{f}{s_y} & o_y \\ 0 & 0 & 1 \end{pmatrix}$$

K viene detta matrice di calibrazione della camera e dipende dai seguenti parametri estrinseci della camera: distanza focale, punto principale (centro immagine) e angolo tra gli assi ottici.

In conclusione, è possibile definire la matrice **P** come il prodotto di tre matrici:

$P = K[I|O]G = K[R|t]$, con **K** dipendente da parametri intrinseci e **G** da parametri estrinseci.

Un importante step è quello della calibrazione della camera, ovvero il processo di stima dei parametri intrinseci e/o estrinseci appena descritti. Conoscere i parametri intrinseci è, infatti, essenziale nel campo della computer vision 3D. La calibrazione si divide in tre aspetti principali:

- Approssimazione basata su considerazioni fisiche;
- Calibrazione fuori dall'acqua;
- Calibrazione in acqua alla distanza operativa dal fondo.

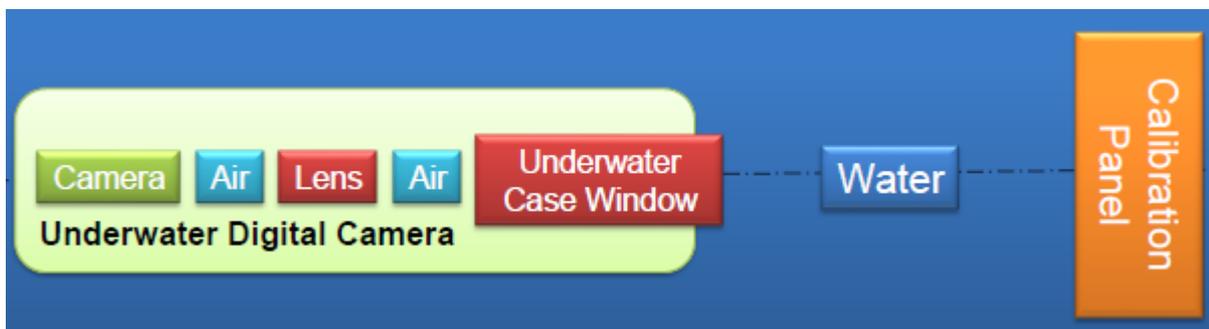


Figura 2.1.3.1

Inoltre, un corretto processo di calibrazione della camera non può evitare di prendere in considerazione il fenomeno della distorsione radiale. Si tratta di una distorsione, causata dalla forma delle lenti, della reale posizione del punto sotto osservazione. L'entità della distorsione è tanto

maggiore quanto più la proiezione del punto è lontana dal punto centrale dell'immagine. In genere, l'effetto causato da questo tipo di distorsione è quello di far apparire curve le linee rette presenti sulla scena. Può essere legata anche all'uso di lenti non di buona qualità o alla necessità di dover ampliare il campo visivo della scena, utilizzando una camera con distanza focale corta.

2.1.4 ESTRAZIONE DELLA FORMA GEOMETRICA 3D

La percezione 3D di una scena o di un oggetto sotto osservazione può essere ottenuta attraverso una visione stereoscopica oppure da una *SFM* (*Structure From Motion*). Tra le tecniche, utilizzate per la ricerca e l'associazione dei punti in multiple immagini, vi sono ad esempio la Correlazione e la *SIFT* (*Scale Invariant Feature Transform*).

La stereoscopia è una tecnica utilizzata per individuare la posizione 3D degli oggetti presenti nello spazio da uno o più punti di vista della scena. I vantaggi nell'utilizzo di tale tecnica per i dispositivi mobili robotici sono:

- Affidabilità delle informazioni ricevute riguardo l'ambiente circostante (implementazione real-time su hardware di basso costo);
- Passività del sensore, ovvero esso non disturba o interferisce con gli altri sensori;
- Può essere facilmente integrato con altre routine di visione (riconoscimento degli oggetti, tracking).

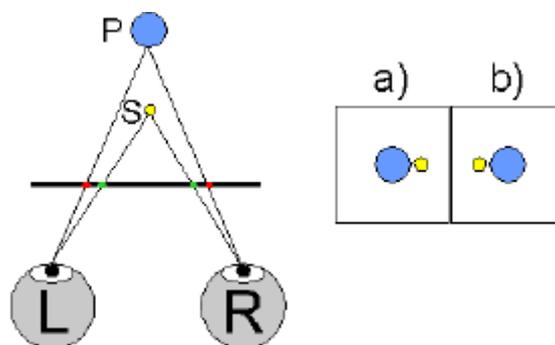


Figura 2.1.4.1

La *SFM* è invece la stima della struttura 3D di un oggetto, solitamente rigido, e del movimento della camera, a partire da immagini 2D scattate da quest'ultima, quando i parametri esterni della camera sono sconosciuti e calcolati in tempo reale. In determinate occasioni può capitare che anche i parametri interni della camera siano sconosciuti.

L'*SFM* può essere vista come la risoluzione simultanea di due problemi ben noti:

- Lo studio di una struttura sconosciuta, a partire da una nota posizione della camera;
- La determinazione delle sole posizioni o movimento di camera, a partire da punti noti o dalla posizione delle camere in tempo reale.

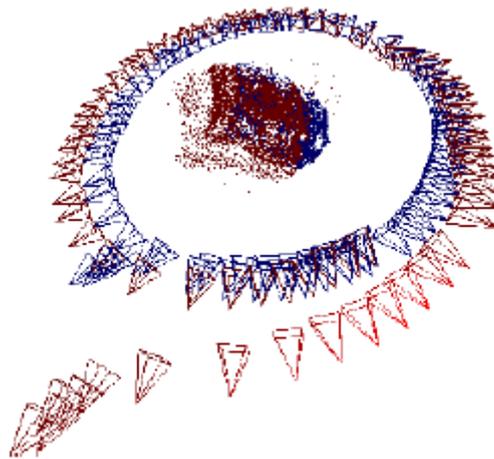


Figura 2.1.4.2

Parlando ora di tecniche per la ricerca e l'associazione dei punti in multiple immagini, una correlazione è una relazione tra due elementi tale che a ciascun valore della prima corrisponda un valore della seconda, ovviamente, secondo un determinato criterio. Quello che si fa è, dunque, prendere una prima immagine come riferimento, misurare un punto e, secondo la misurazione effettuata, ricercare la corrispondenza nella seconda immagine. Il processo si ripete per ogni possibile coppia di immagini.

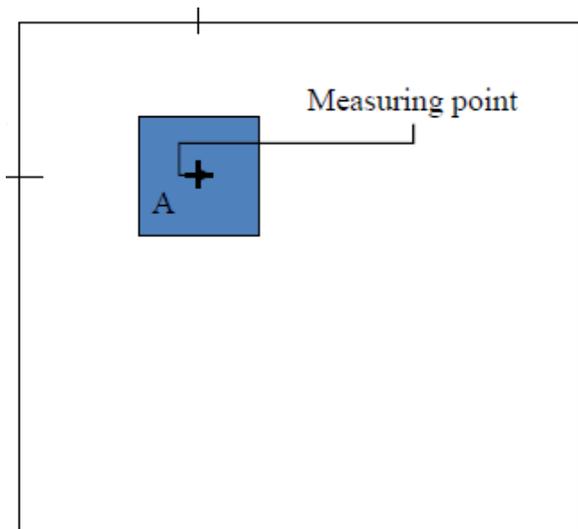


Figura 2.1.4.3

Immagine di riferimento

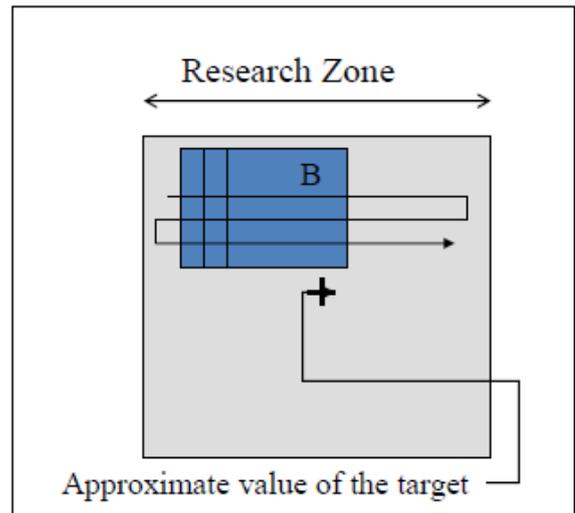


Immagine sulla quale cerchiamo il punto

La correlazione è però una tecnica che presenta varie problematiche nel caso in cui le immagini fossero diverse per scala e parametri di traslazione e rotazione. Il motivo lo si può intravedere dall'immagine di seguito raffigurata

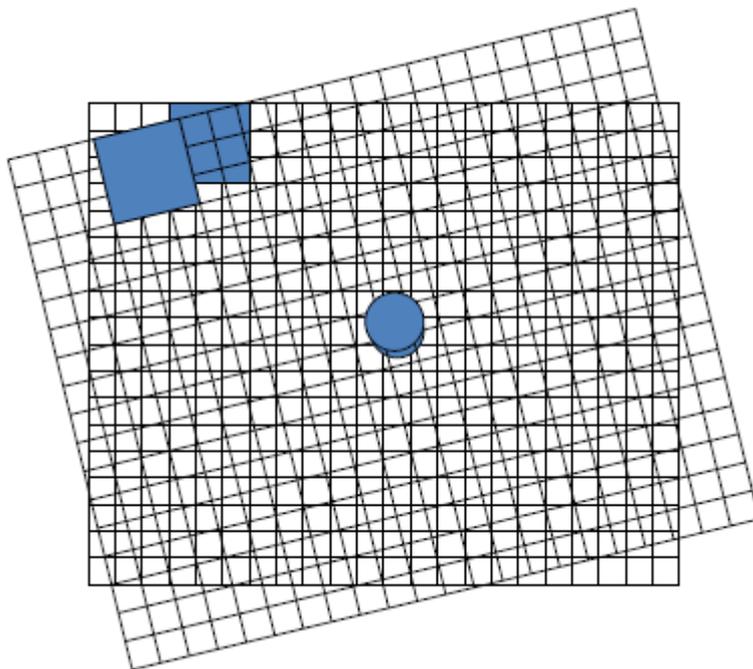


Figura 2.1.4.4

Metodo alternativo alla correlazione ed efficace nonostante cambi di luce e parametri di scala, traslazione e rotazione differenti è il **SIFT**. Esso si divide nei seguenti step:

- Rilevazione estremi nello spazio-scala (differenza di gaussiana);
- Posizione dei punti chiave (rimozione degli oggetti instabili);
- Assegnamento dell'orientazione (basato su un intorno del punto chiave);
- Descrittore dei punti chiave (basato sui gradienti locali intorno al punto chiave);
- Confronto dei punti chiave (i punti chiave vengono confrontati per trovare gli altri punti a loro più vicini).

Dunque, il primo passo consiste in un'implementazione efficiente della Differenza di Gaussiana con lo scopo di identificare potenziali punti chiave, invarianti rispetto a dimensione di scala e orientazione, costruendo una piramide avente dei ricampionamenti per ogni livello.

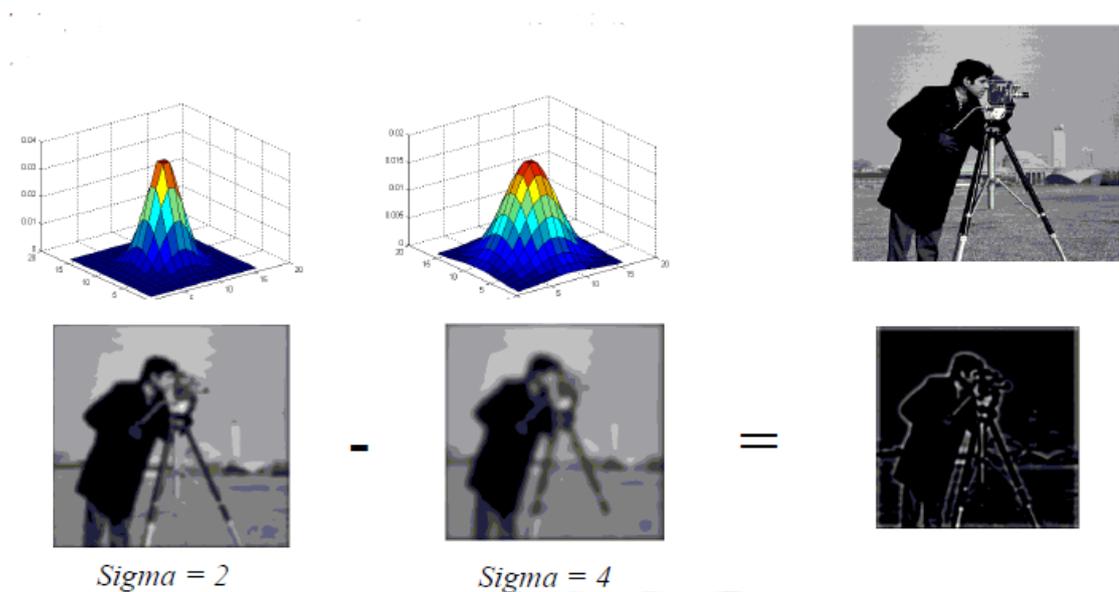


Figura 2.1.4.5

L'immagine originale è convolta in una serie di funzioni gaussiane, ottenendo così un'immagine differente.

Per ogni punto candidato, si sviluppa un modello dettagliato per determinare la sua posizione e la scala. La selezione dei punti chiave è basata su misurazioni della loro stabilità:

- Rilevazione degli estremi sulla stessa scala;
- Verifica della stabilità su molteplici scale.

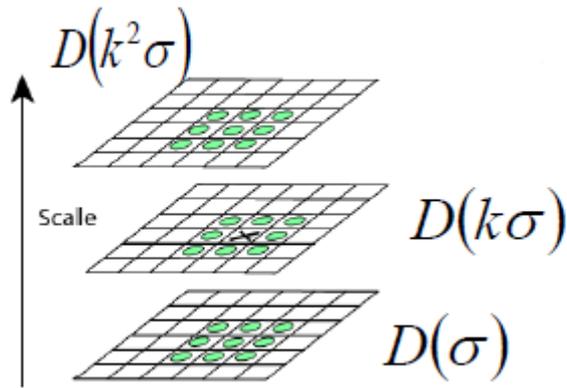


Figura 2.1.4.6

Si giunge ora all'assegnamento dell'orientazione. Una o più orientazioni vengono assegnate ad ogni singolo punto chiave basandosi sulle direzioni del gradiente dell'immagine locale. Tutte le future operazioni saranno effettuate sui dati dell'immagine che sono stati modificati rispetto all'orientazione assegnata, alla scala ed alla posizione, in maniera da fornire invarianza a queste trasformazioni.

Per ogni estremo della Differenza di Gaussiana, calcoliamo la direzione associata in questo modo:

$$\theta(x, y) = \tan^{-1} \left(\frac{G_y^\sigma(x, y)}{G_x^\sigma(x, y)} \right)$$

$$G_x^\sigma(x, y) = \frac{\partial}{\partial x} G(x, y, \sigma) = I(x, y) * \frac{\partial}{\partial x} g_\sigma(x, y)$$

I gradienti dell'immagine locale sono misurati sulla scala selezionata nella regione intorno ad ogni punto chiave. I descrittori *SIFT* associati con i punti di vista sono istogrammi di orientazioni locali intorno al punto chiave.

- Si divide lo spazio intorno ad ogni punto chiave (x, y) in una N^2 quadrati 4×4 ;
- Si calcola il gradiente $[G_x(a, b, s), G_y(a, b, s)]$ per i $4 \times 4 * N^2$ punti (a, b) ;

- Per ogni quadrato 4×4 , si calcola un istogramma dell'orientazione quantizzata in otto direzioni tenendo conto del gradiente del modulo e dell'inverso della distanza da ogni punto chiave (x, y) ;
- Per essere invariante rispetto alla rotazione, l'orientazione locale del punto chiave viene utilizzata come origine degli istogrammi.

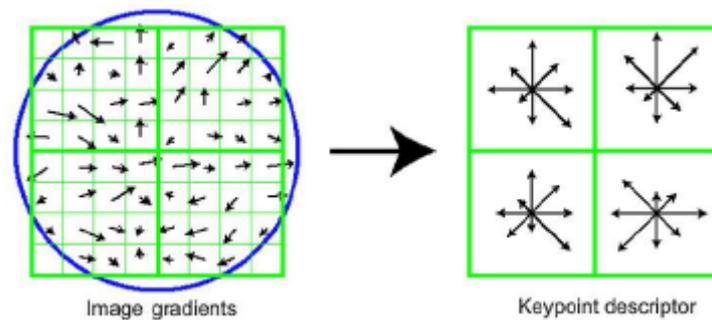


Figura 2.1.4.7

Per introdurre il calcolo della corrispondenza bisogna prima introdurre il fenomeno della stereopsi. Si tratta dell'impressione di profondità che si percepisce guardando la scena con entrambi gli occhi (visione binoculare).

La visione binoculare crea due immagini leggermente differenti della scena nei due occhi a causa delle loro diverse posizioni sulla testa. Questa differenza, chiamata disparità oculare, è utilizzata dal cervello per calcolare la profondità della visuale della scena.

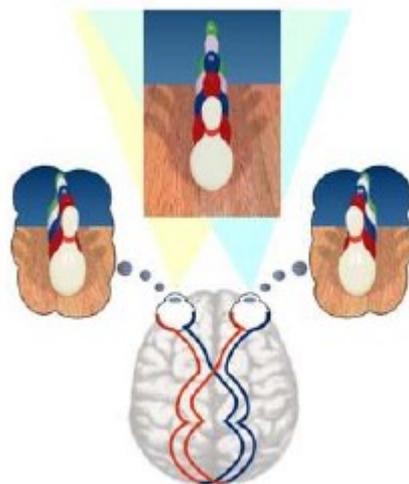


Figura 2.1.4.8

Per il calcolo della corrispondenza si utilizza la geometria epipolare.

Ogni punto della scena 3D $M = (x, y, z, 1)^T$, è proiettato a sinistra e a destra della vista rispettivamente come $m_l = (u_l, v_l, 1)^T$ e $m_r = (u_r, v_r, 1)^T$

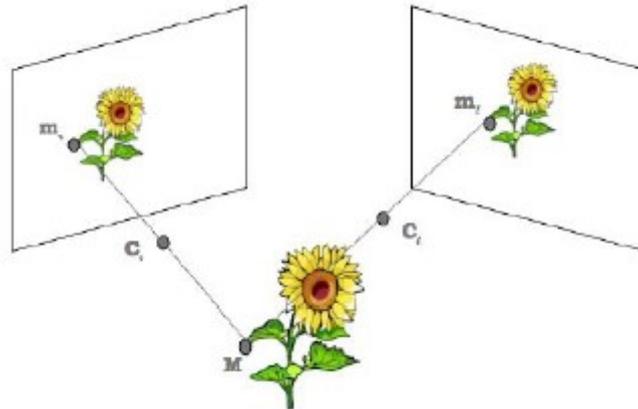


Figura 2.1.4.9

Per costruzione, la linea epipolare sinistra è l'immagine del centro di proiezione della camera destra sul piano immagine sinistro e, dunque, la linea epipolare destra risulta essere l'immagine del centro di proiezione della camera sinistra sul piano immagine destro.

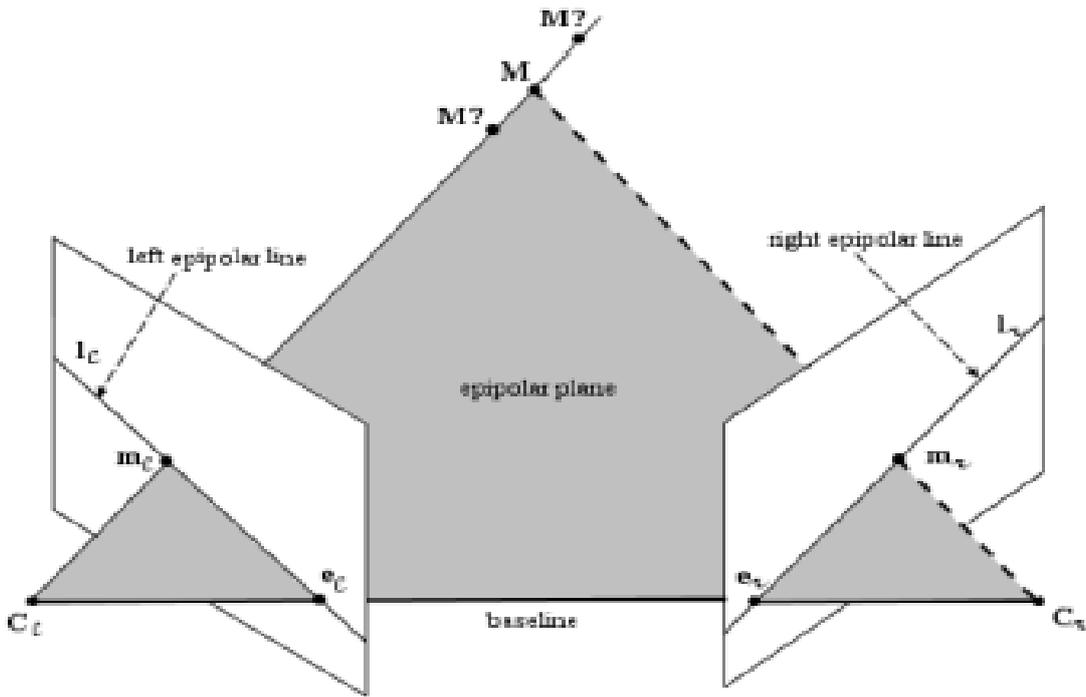


Figura 2.1.4.10

In riferimento alla seguente figura:

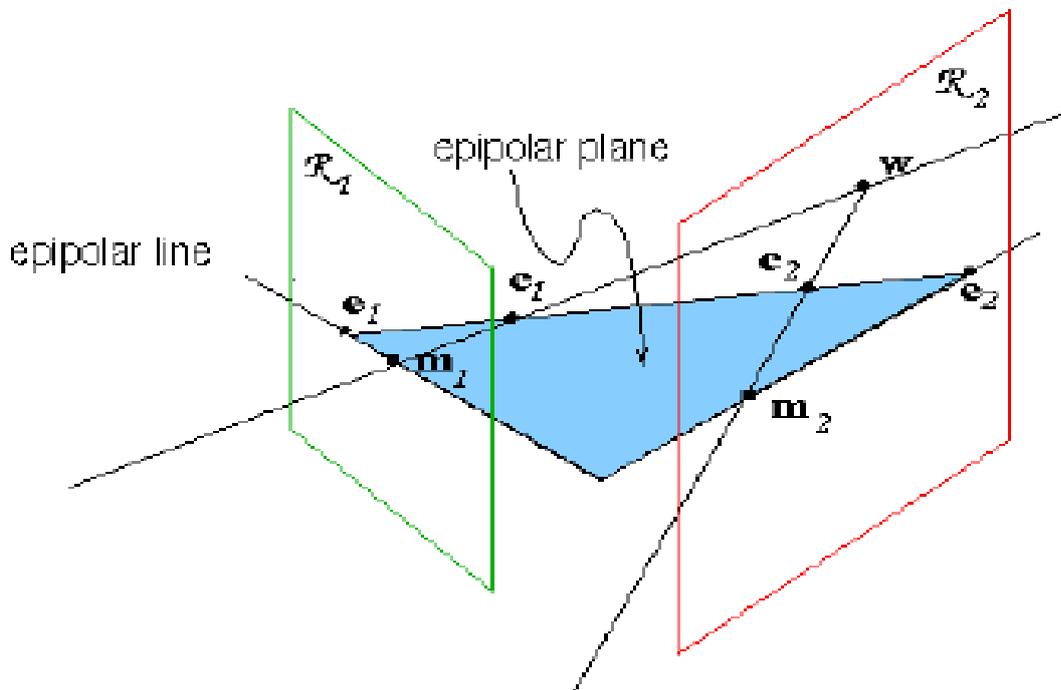


Figura 2.1.4.11

Dato un sistema di riferimento assoluto:

$$\begin{cases} \tilde{m}_1 \cong \tilde{P}_1 * \tilde{W}_1 \\ \tilde{m}_2 \cong \tilde{P}_2 * \tilde{W}_2 \end{cases} \quad \text{con} \quad \tilde{W} = \begin{bmatrix} c_1 \\ 1 \end{bmatrix} + \lambda \begin{bmatrix} P_1^{-1} \tilde{m}_1 \\ 0 \end{bmatrix} := \text{Raggio ottico di } m_1$$

La linea epipolare relativa a \mathbf{m}_2 è la proiezione del raggio ottico di \mathbf{m}_1 sulla base di \mathbf{P}_2 . Tale linea corrisponde a:

$$\tilde{\mathbf{m}}_2 = \lambda P_2 P_1^{-1} \tilde{\mathbf{m}}_1 + \tilde{\mathbf{e}}_2$$

Sia $F = [\tilde{\mathbf{e}}_2] \wedge P_2 P_1^{-1}$ ($\tilde{\mathbf{e}}_2$ è una matrice asimmetrica), la linea epipolare di \mathbf{m}_1 è rappresentata dal vettore

$$\tilde{\mathbf{e}}_2 \wedge P_2 P_1^{-1} \tilde{\mathbf{m}}_1 = F \tilde{\mathbf{m}}_1$$

F è detta Matrice Fondamentale.

La Matrice Fondamentale include tutte le informazioni relative alla geometria epipolare ed è quindi utile a disegnare le linee epipolari. È una matrice 3×3 che mette in relazione i punti corrispondenti ad un'immagine stereo. Nella geometria epipolare, con coordinate omogenee relative all'immagine, \mathbf{x} e \mathbf{x}' , di punti corrispondenti ad un paio di immagini stereo, F_x descrive una linea epipolare su cui il corrispondente punto \mathbf{x}' sull'altra immagine giace. Ciò implica che per tutte le coppie di punti vale l'equazione:

$$\mathbf{x}' F \mathbf{x} = 0$$

Dalla Matrice Fondamentale ricaviamo anche la Matrice Essenziale:

$$E = P_2' F P_1$$

Essa descrive una relazione simile a quella descritta dalla Matrice Fondamentale. Rappresenta infatti un oggetto metrico pertinente alle camere calibrate, mentre la Matrice Fondamentale descrive la corrispondenza, attraverso termini generali e fondamentali, della prospettiva geometrica.

2.1.5 RILEVAMENTO DELLA POSIZIONE RELATIVA E ASSOLUTA

Riguardo all'orientazione, è possibile classificarne due tipi: relativa e assoluta.

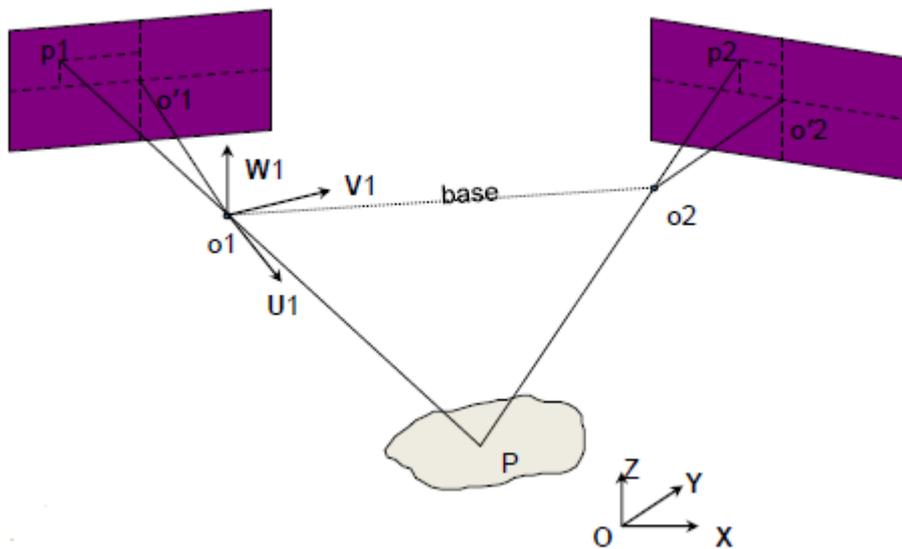


Figura 2.1.5.1

- **Posizione relativa:**
 $P1O1, P2O2$ stesso piano $(O1P1 \wedge O2P2) O1O2 = 0$
- **Posizione assoluta:**
 Adattamento conformazione 3D $(o1, u1, v1, w1) (O, X, Y, Z)$

2.1.6 REGOLAZIONE DEL FASCIO E ANALISI DI PRECISIONE

Altri importanti step sono la regolazione del fascio e l'analisi di precisione, tra cui troviamo l'accuratezza del segnale dell'immagine, rapporto tra basi e profondità, errore ellissoide nello spazio.

L'obiettivo diventa quindi:

- Incrementare le basi rispetto alla profondità;
- Ottenere una sfera al posto dell'ellisse;
- Incrementare l'accuratezza.

Prima della regolazione del fascio e dell'analisi di precisione:

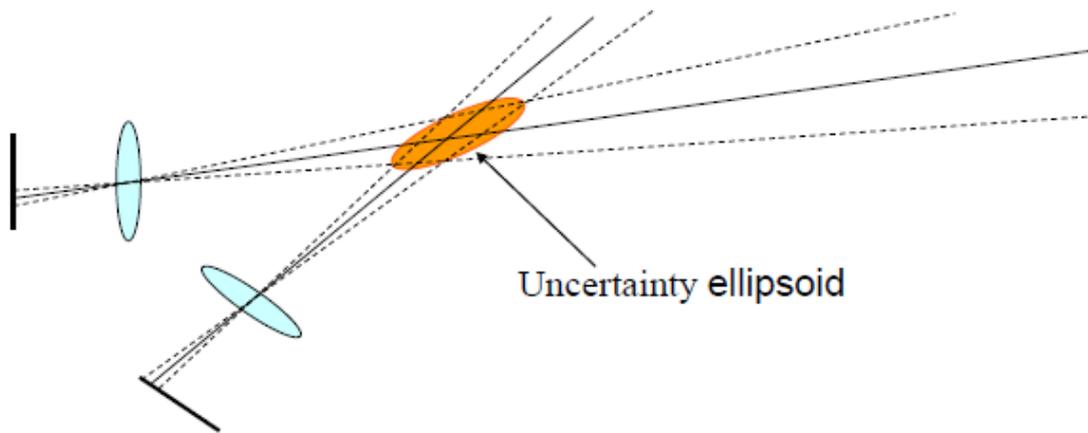


Figura 2.1.5.2

Dopo la regolazione del fascio e dell'analisi di precisione:

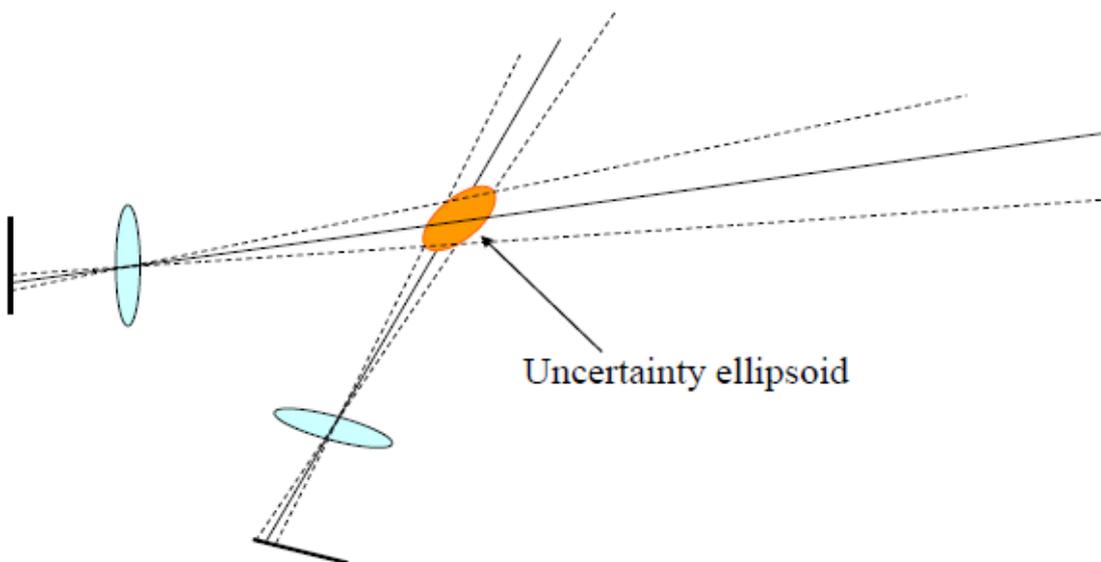


Figura 2.1.5.3

2.1.7 PROCESSO DI RICOSTRUZIONE 3D SFM

Rilevamento caratteristiche e confronto: L'obiettivo è trovare le stesse caratteristiche in immagini differenti e confrontarle tra loro, attraverso il recupero della struttura e del movimento.

Recupero struttura e movimento: Consiste nel recuperare la struttura e il movimento della scena (coordinate 3D delle caratteristiche rilevate: posizione, orientazione e parametri della camera nella posizione di cattura delle immagini).

Stereo mapping: Serve a creare una mappa densa per il confronto. Proprio come nel recupero della struttura e del movimento, qui non è possibile creare una mappa densa della profondità della scena.

Modelling: Include procedure necessarie alla ricostruzione di un modello tridimensionale della scena (modelli mesh, mapping delle texture). L'intero processo è anche chiamato Struttura del Movimento.

CAPITOLO 3

3.1 SOFTWARE COMMERCIALI A CONFRONTO

Teoria, tecnicismi e concetti matematici, affrontati nel capitolo precedente, sono di fondamentale importanza per lo sviluppo e l'implementazione di un software fotogrammetrico.

Un software fotogrammetrico è un software che permette di elaborare dei modelli tridimensionali, a partire da immagini bidimensionali. Deve, dunque, implementare algoritmi che tengano conto di tutti gli aspetti teorici visti. Negli ultimi anni questo tipo di tecnologia ha subito una grande diffusione, dovuta allo sviluppo di software a basso costo e semplici da utilizzare e soprattutto alla possibilità di poter utilizzarli anche con un pc di medie prestazioni.

Un software fotogrammetrico perfettamente funzionante restituisce in uscita un modello 3D metricamente corretto, ciò significa che il modello dev'essere nella giusta scala e su di esso devono potersi effettuare misurazioni precise e con colori realistici (quest'ultima caratteristica è necessaria o meno, a seconda dell'utilizzo che si vuole fare del modello 3D).

I software fotogrammetrici elaborano le immagini bidimensionali attraverso determinati passaggi (tali passaggi sono gli stessi per ogni software) ed è, quindi, importante conoscerli anche per sapere come scattare le fotografie da elaborare e come intervenire nel caso di eventuali errori in un processo. I passaggi sono i seguenti:

- **STRUCTURE FROM MOTION (SFM) E MULTIVIEW STEREO RECONSTRUCTION (MVS):**

Da un punto di vista dell'elaborazione software, si tratta della fase più lunga e complessa di tutto l'intero processo. In questa fase vengono costruite la geometria di presa delle fotografie e la nuvola di punti densa. Attraverso le tecniche, viste nel precedente capitolo, si individuano i punti chiave tra le immagini per cercare corrispondenze tra loro ed allinearle tutte correttamente. Trovati i punti, attraverso un procedimento di triangolazione fotogrammetrica a stelle proiettive (bundle adjustment), viene calibrata la fotocamera calcolando, ad esempio, la distanza focale ed i parametri esterni ed interni. Ogni punto chiave possiede delle coordinate spaziali e tutti insieme vanno a formare una nuvola di punti chiave. A tale nuvola vengono poi aggiunti altri punti, ricavati da un'analisi a partire dall'intorno di ogni singolo punto chiave, giungendo così alla nuvola densa.

- **MESH RECONSTRUCTION:**

A partire dalla nuvola densa, viene generato un insieme di poligoni triangolari facendo corrispondere i punti della nuvola densa con i vertici dei triangoli generati. Tali triangoli, i quali vertici, a questo punto, possiedono delle coordinate (x,y,z) , compongono la mesh. Una volta applicata la mesh alla nuvola, si passa da un insieme di punti ad una superficie continua.

- **ATTRIBUZIONE DEL COLORE:**

La mesh in origine non presenta nessun colore. Esso dev'essere assegnato ai poligoni e ciò può essere fatto o tramite COLOR-PRE-VERTEX o tramite TEXTURE MAPPING. Il COLOR-PRE-VERTEX consiste nel trasferire il colore dei punti della nuvola ad ogni poligono corrispondente, visto che i vertici dei poligoni triangolari corrispondono ai punti della nuvola densa. La risoluzione del modello corrisponde al numero di poligoni che lo compongono e, generalmente, al numero di punti presenti nella nuvola densa, dunque, maggiore sarà la risoluzione e migliore sarà anche la qualità visiva dei colori. Il metodo del TEXTURE MAPPING utilizza invece le immagini di partenza. Poiché le immagini sono orientate rispetto al modello attraverso la structure from motion, possono essere proiettate sul modello e applicate ai poligoni triangolari. Così facendo, le immagini vanno a comporre una texture che ricopre poi l'intero modello. Il vantaggio nell'utilizzare quest'ultima tecnica è che la qualità visiva dipende dalla qualità delle fotografie e non dalla risoluzione del modello. Indipendentemente dalla risoluzione del modello, dunque, la qualità visiva sarà sempre la stessa.

Introdotti gli step fondamentali di un software di fotogrammetria, si può ora procedere con l'analisi di alcuni software molto utilizzati, al fine di confrontarli e capire quale tra questi potrebbe risultare un buon programma da utilizzare nell'ambito della fotogrammetria sottomarina.

I programmi analizzati saranno i seguenti:

- **VisualSFM:** Si tratta di un programma non completo. Ciò che si vuole intendere è che VisualSFM permette di costruire la nuvola densa e la mesh del modello 3D, ma ha bisogno del supporto di un secondo programma per applicare la mesh al modello (Nel corso del tirocinio, a supporto di VisualSFM, è stato utilizzato il programma MeshLab);
- **CloudCompare:** Anche in questo caso si tratta di un software incompleto dal punto di vista dell'intero processo fotogrammetrico. Al contrario di VisualSFM, CloudCompare permette di lavorare su una nuvola di punti, ma non di generarla a partire da immagini bidimensionali;
- **Agisoft Metashape:** A differenza dei precedenti, non ha bisogno di alcun supporto ed è provvisto di diverse funzioni;

- **Meshroom:** Programma OpenSource, completo e, quindi, da tenere in considerazione, nel caso si cercasse qualcosa di completamente gratuito.

3.1.1 VISUALSFM

VisualSFM è una GUI application (applicazione grafica). GUI è una sigla che sta per *Graphical User Interface*.

È utilizzato per la ricostruzione 3D attraverso la Structure From Motion (SFM) e integra diverse funzionalità come, ad esempio, la SIFT on GPU, metodo utilizzato per elaborare i pixel contemporaneamente alla costruzione delle piramidi gaussiane e al calcolo della differenza delle gaussiane dei punti d'interesse, al fine di ottenere le loro orientazioni e descrittori.

Presenta anche una PBA (*Parallel Bundle Adjustment*), un'implementazione necessaria per la regolazione del fascio per sistemi multicore.

Per la ricostruzione della densità, il programma utilizza invece il tool PMVS/CMVS di Yasutaka Furukawa:

- **CMVS (*Clustering Views for Multi-View Stereo*):** Prende l'Output di un software SFM come Input, in seguito decompone le immagini Input in un set di gruppi di immagini di dimensione regolabile.
- Un software **MVS** è in genere utilizzato per elaborare ogni singolo gruppo di immagini indipendentemente e in parallelo. Dall'unione delle ricostruzioni di ogni singolo gruppo, non dovrebbe mancare alcun dettaglio presente all'interno dell'intero set di immagini.

Qui di seguito, un'immagine che raffigura la barra del menu di VisualSFM con alcune funzioni più importanti messe in evidenza.



Figura 3.1.1.1

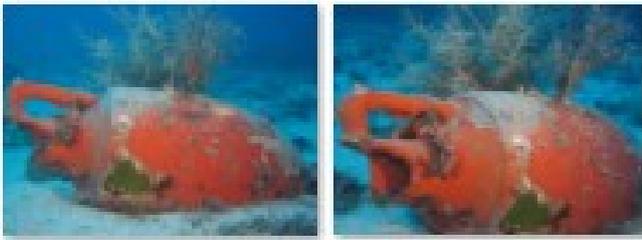


Figura 3.1.1.2

Importando queste due immagini all'interno di VisualSfM ed utilizzando la sequenza di comandi evidenziata nell'immagine precedente, otteniamo la seguente nuvola densa di punti:

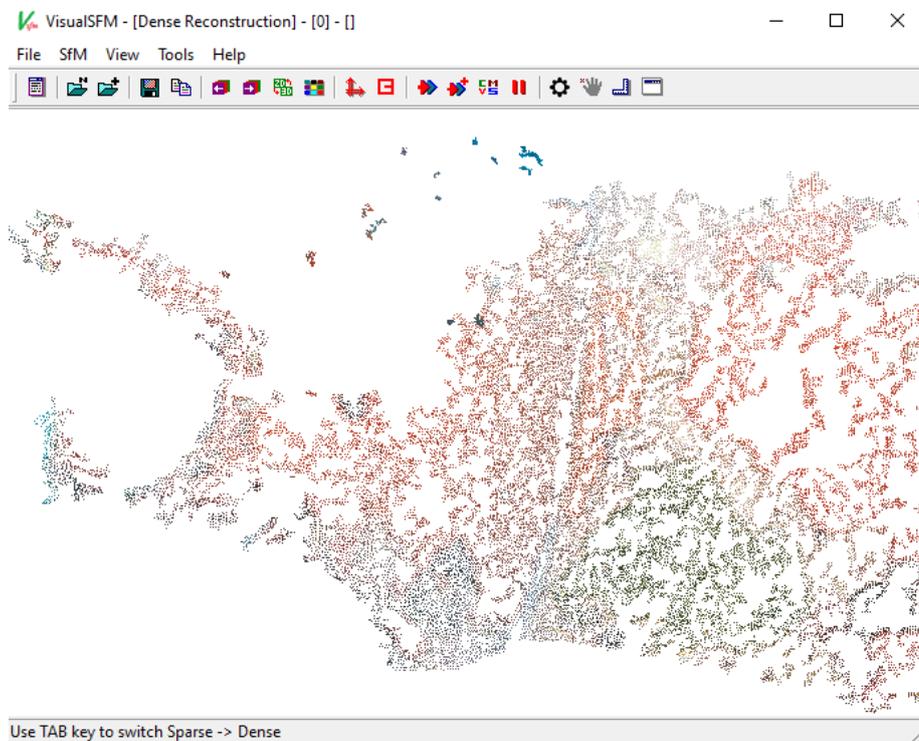


Figura 3.1.1.3

```
Log Window
Totally 1.000 seconds used

Run dense reconstruction, begin...
Save to File_Per_Presentazione.nvm ... done
#####
Undistorting 2 images in model #0
param_undistortion_thread_num = 0
NOTE: using 4 threads...
#00: IMG_4147 -> 00000000.jpg, 4.603 sec
#00: IMG_4148 -> 00000001.jpg, 4.532 sec
-----

Running Yasutaka Furukawa's CMVS tool...
cmvs "C:\Users\JTENTE\Desktop\Nuova cartella
(2)\File_Per_Presentazione.nvm.cmvs\00\\" 50 4
genOption "C:\Users\JTENTE\Desktop\Nuova cartella
(2)\File_Per_Presentazione.nvm.cmvs\00\\" 1 2 0.700000
7 2 4
1 cluster is generated for pmvs reconstruction.
pmvs2 "C:\Users\JTENTE\Desktop\Nuova cartella
(2)\File_Per_Presentazione.nvm.cmvs\00\\" option-0000
This may take a little more time, waiting...
54 seconds were used by PMVS
Loading option-0000.ply, 47435 vertices ...
#####
You can manually remove bad MVS points:
1. Switch to dense MVS points mode;
2. Hit F1 key to enter the selection mode;
3. Select points by dragging a rectangle;
4. Hit DELETE to delete the selected points.
#####
Save to File_Per_Presentazione.nvm ... done
Save C:\Users\JTENTE\Desktop\Nuova cartella
(2)\File_Per_Presentazione.0.ply ...done
-----

Run dense reconstruction, finished
Totally 67.000 seconds used
```

Figura 3.1.1.4

Questa invece è la Log Window di VisualSFM, una sorta di terminale in cui è possibile visualizzare tutte le operazioni, il tempo trascorso fino al loro completamento ed eventuali errori, nel caso ve ne fossero.

È possibile scaricare tale programma accendendo al sito ccwu.me.

Sul sito è inoltre possibile visualizzare tutte le informazioni riguardanti il programma. Tra queste è importante sapere che non si tratta di un OpenSource. Come infatti si legge all'indirizzo ccwu.me/vsfm/README VisualSFM è stato sviluppato da Chanchang Wu con il contributo di Sameer Agarwal, Brian Curless e Steven M. Seitz ed è esclusivamente concesso all'utilizzo personale, no profit o accademico.

Difficile dire se, come progetto, VisualSFM sia vivo o morto. Visitando il sito di Chanchang Wu o la sua pagina YouTube, l'idea che potremmo farci è quella di un progetto ormai giunto al termine, dal momento che dal 2014 non vi sono più aggiornamenti corposi o video tutorial caricati. Egli scrisse inoltre nella sua tesi di ricerca che in futuro avrebbe voluto esplorare e ricercare nuovi metodi più funzionali per il confronto dei punti chiave.

L'ultima versione scaricabile di VisualSFM è la 5.0.26 che, rispetto alle versioni precedenti, apporta piccoli cambiamenti come l'aggiunta di nuove funzioni e la correzione di alcuni bug.

Durante l'attività di tirocinio la nuvola di punti, creata con VisualSFM a partire da due immagini e mostrata precedentemente, è stata poi importata sul programma MeshLab per procedere con l'applicazione della mesh.

MeshLab è uno strumento OpenSource che presenta funzioni per l'editing, applicazione di mesh e pulizia di modelli tridimensionali.

E' sicuramente un programma indispensabile e utile ad affiancarne altri come, ad esempio, VisualSFM. Da quest'ultimo, infatti, MeshLab prende in Input i file *bundle.rd.out*, il documento di testo *list* e infine la mesh generata come file di tipo *.ply*. Di seguito, il risultato ottenuto:

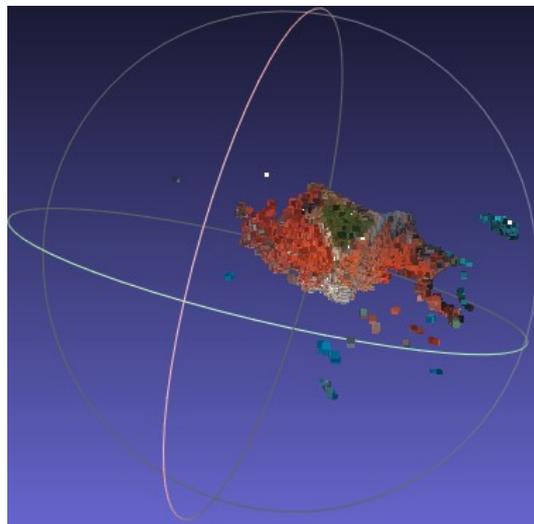


Figura 3.1.1.5

A questo punto il modello non è completo, se non si ricostruisce la superficie.

Per la costruzione della superficie MeshLab mette a disposizione l'algoritmo di Poisson. Tutto quello che si deve fare è cercare la voce *Surface Reconstruction: Screened Poisson*. In alcune guide può capitare di trovare la dicitura *Surface Reconstruction: Poisson*, ma si tratta della stessa voce (il nome

venne cambiato in seguito ad un aggiornamento che apportò varie modifiche, tra cui anche l'aggiunta ed il miglioramento di alcune funzioni).

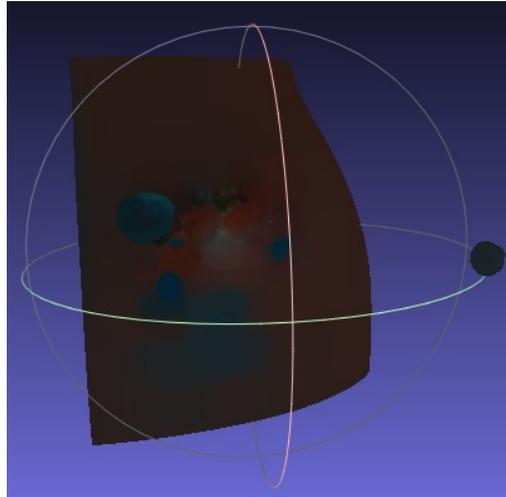


Figura 3.1.1.6

3.1.2 CLOUDCOMPARE

CloudCompare è un software per l'analisi di nuvole di punti tridimensionali che può gestire mesh triangolari e immagini calibrate, attraverso l'utilizzo di diversi algoritmi.

Il programma è disponibile per windows, MasOS e Linux.

A differenza di VisualSFM, si tratta di un OpenSource regolato dalla *GNU General Public Licence (GPL)* dove vi è anche espresso il libero utilizzo a scopo commerciale o educativo.

Non ci si trova assolutamente di fronte ad un progetto morto, al contrario, esso è continuamente supportato. Ci si accorge del continuo supporto accedendo al sito web di CloudCompare all'indirizzo <https://www.danielgm.net/cc/> e notando che, tra le varie versioni disponibili, vi è una versione alpha (versione primitiva e non ancora ultimata di quello che sarà poi un update corposo). Qualora, per questioni di sicurezza, non si volesse fare affidamento sulla versione alpha, è comunque possibile scaricare la versione stabile più recente (al momento, la versione 2.11.3).

La versione presa in esame e che sarà presentata è invece proprio la versione alpha (versione 2.12.alpha).

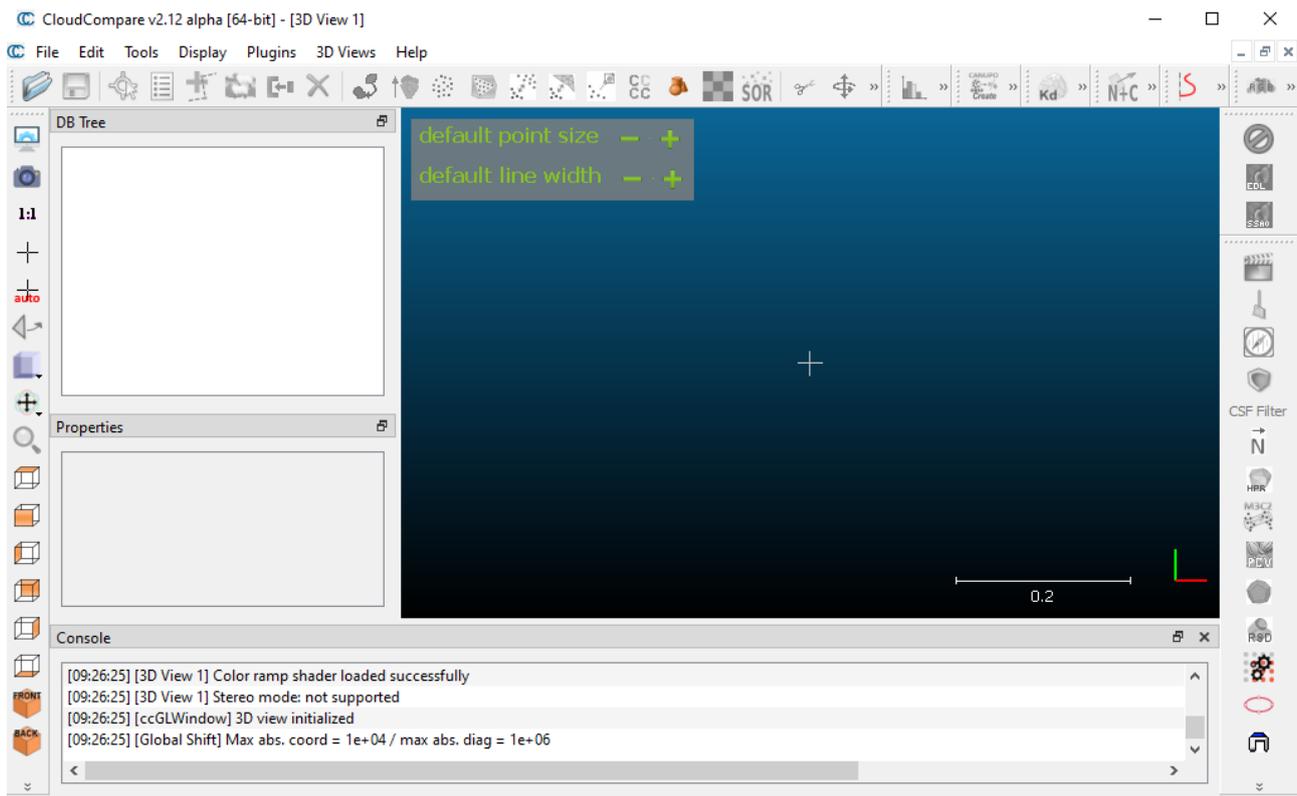


Figura 3.1.1.7

È esattamente così che il programma si presenta una volta avviato.

Tra le varie possibilità vi è quella di aprire un progetto che sia una nuvola di punti o una mesh o anche delle immagini (è ovviamente possibile visualizzare il percorso del file aperto). Subito dopo, andando su “Properties”, si potranno, dunque, visualizzare tutte le caratteristiche del progetto importato (nome, colore, dimensione ecc.). Alle nuvole di punti importate è possibile applicare una mesh. A tale scopo CloudCompare utilizza due algoritmi **Delaunay 2.5D (XY Plane)** e **Delunay 2.5D (Best Fitting Plane)** con la possibilità di impostare dei limiti sulle mesh triangolari di cui tener conto.

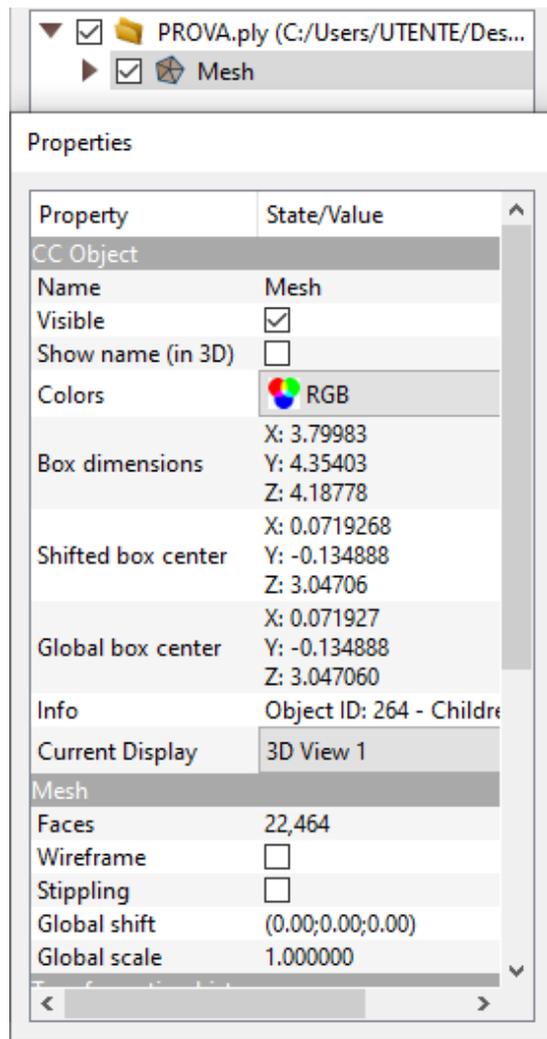


Figura 3.1.1.8

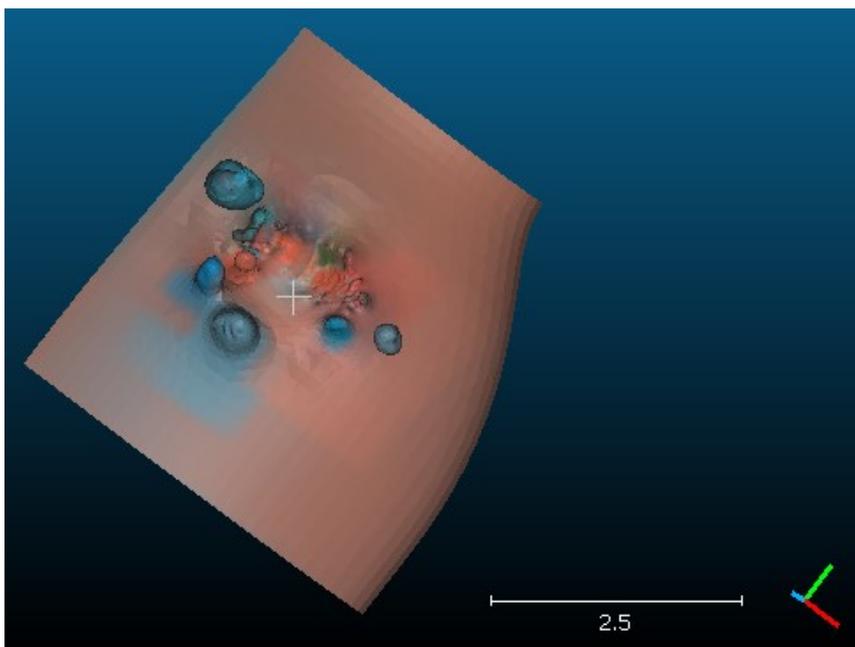


Figura 3.1.1.9

Applicata la mesh, la figura nella precedente immagine è ottenibile in seguito ad una normalizzazione della mesh (la nuvola di punti utilizzata è la stessa che è stata realizzata tramite VisualSFM). E' il programma stesso a suggerire questo tipo di operazione, attraverso il messaggio "mesh has no normal, you can manually compute them".

Per normalizzare la mesh => Edit > Normals > Compute.

Di seguito troviamo una serie di impostazioni attraverso cui è possibile modificare la visuale del nostro progetto, passando da una visuale frontale ad una centrata verso destra o sinistra. E' anche possibile cambiare manualmente le impostazioni della camera, andando a cliccare sul simbolo della macchina fotografica.



Figura 3.1.1.10

Anche qui vi è una console che presenta le stesse funzioni della log window di Visual SFM:

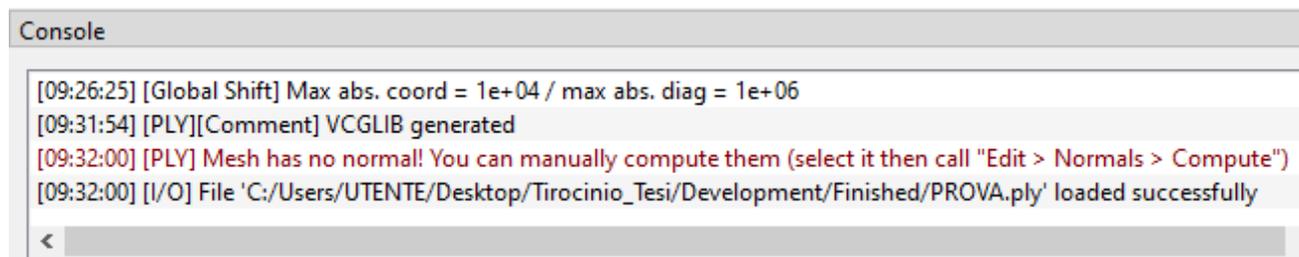


Figura 3.1.1.11

3.1.4 AGISOFT METASHAPE

Agisoft è un'applicazione non OpenSource per i processi di fotogrammetria di immagini digitali e la generazione di dati 3D che possono essere utilizzati in applicazioni GIS (applicazioni che consentono la visualizzazione di dati geografici).

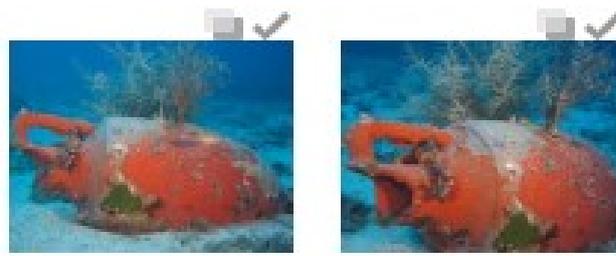
Agisoft è disponibile in due versioni: Standard e Pro. La Standard è sufficiente per le attività multimediali interattive, mentre la versione Pro è progettata per la creazione di contenuti GIS.

La versione analizzata in questa presentazione è la versione Pro. Agisoft Metashape offre un periodo di prova di trenta giorni invitando però gli utenti a contattare l'assistenza, nel caso si volesse utilizzare il programma per un progetto lavorativo, al fine di trovare una licenza opportuna al tipo di utilizzo.

Per chi ha un account Agisoft, viene messo a disposizione anche uno spazio cloud in cui poter eventualmente salvare i propri progetti.

Di seguito sarà mostrato come procedere per la costruzione di un modello tridimensionale a partire da una o più immagini.

- La prima cosa da fare è caricare le foto da cui vogliamo ricavare il modello 3D, quindi creare un nuovo chunk con **Add Chunk toolbar** e dunque aggiungere nuove foto con **Add photos** dal menù.
- Il secondo passo è la calibrazione della camera. È possibile, infatti, dal menu **Tools** selezionare il tipo di calibrazione che desideriamo e calibrare le lenti della camera. Nell' **Adjustment tab** della **Camera Calibration** è possibile visualizzare i risultati.
- Per confrontare le immagini bisogna selezionare **Align photos...** dal menù di **Workflow**. Fatto ciò, sarà possibile passare alla costruzione della nuvola densa di punti.
- Selezionare quindi **Build Dense Cloud...** e impostare i parametri di ricostruzione desiderati. Qui di sotto vi è un esempio di ricostruzione della nuvola densa di punti con impostazioni predefinite (qualità media):



IMG_4147

IMG_4148

Figura 3.1.4.1

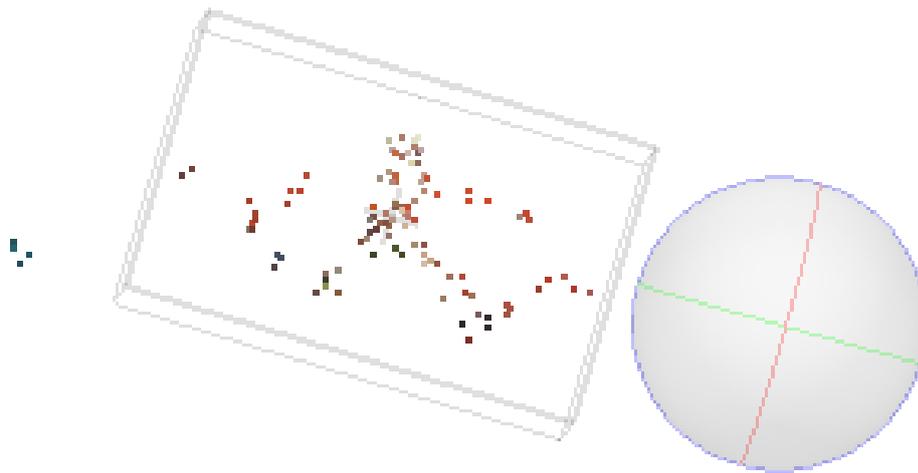


Figura 3.1.4.1

- Workspace (2 chunks, 2 cameras)
- ▼ Chunk 1 (2 cameras, 107 points)
 - ▶ Cameras (2/2 aligned)
 - ▶ Components (1)
 - Tie Points (107 points)
 - Depth Maps (2, Medium quality, Mild filtering)
 - Dense Cloud (211,283 points, Medium quality)
 - ◆ 3D Model

Figura 3.1.4.2

Metashape è in grado di ricostruire il modello poligonale della mesh, basandosi sulle informazioni della nuvola di punti, selezionando **Build Mesh...** dal menù **Workflow**. Risultato di tale procedura:

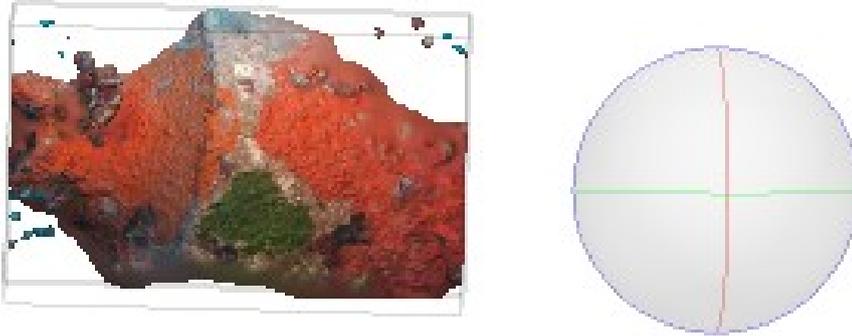


Figura 3.1.4.3

Anche qui, come in VisualSfM e CloudCompare, vi è una console in cui è possibile visualizzare le varie operazioni compiute e il relativo periodo di completamento delle stesse o, nel caso ve ne fossero, eventuali errori.

In conclusione, si potrebbe dire che tra VisualSfM, CloudCompare e Agisoft Metashape, quest'ultimo risulta essere il migliore, tenendo in conto però che si tratta di un servizio a pagamento e non presenta i vantaggi di un software OpenSource al contrario di CloudCompare.

Pur essendo, il modello ottenuto con Agisoft Metashape e mostrato precedentemente, di qualità media per impostazioni predefinite, sembra comunque presentare più dettagli rispetto ai suoi concorrenti.

Agisoft MetaShape è inoltre l'unico ad avere il vantaggio di generare dati 3D che possono essere utilizzati in applicazioni GIS.

3.1.5 MESHROOM

Online è possibile trovare diversi siti che introducono gli utenti nel mondo della fotogrammetria.

Tra i più conosciuti vi è sicuramente Scopus. Esso si presenta come un largo database di informazioni riguardanti la letteratura scientifica (campo tecnologico, scientifico, umanistico e sociale ecc.).

Scopus offre agli utenti la possibilità di iscriversi, ma garantisce il suo servizio anche agli utenti non iscritti. L'obiettivo è quello di fornire un supporto ai vari ricercatori con lo scopo di rendere più fluida la loro ricerca, attraverso chiaramente le varie informazioni che troviamo all'interno del sito.

Esiste una sezione, dedicata ai migliori programmi per la ricostruzione 3D del 2021, in cui vi è una spiegazione generale sul significato di fotogrammetria e in che cosa consiste ed una lista di programmi sia gratuiti, sia a pagamento.

Tra i programmi citati sono presenti anche VisualSfM e Agisoft Metashape, già presentati in queste slide. Un altro programma, presente in lista e che quindi rientra tra i migliori del 2021, è Meshroom.

Meshroom è un programma per la ricostruzione tridimensionale di tipo aerea, corto-raggio e completamente gratuito.

È il risultato di una collaborazione tra partner accademici e industriali provenienti dalla Repubblica Ceca, Francia e Norvegia. Tra questi vi è ad esempio la *Mikros Image* (compagnia di Parigi), un'azienda che si occupa di *VFX* (effetti visivi) nel campo cinematografico.

Per celebrare il lancio di *Meshroom 2019.1* aggiunsero un tutorial introduttivo per principianti sul sito di *sketchfab.com*. Nel tutorial viene spiegato come effettuare una costruzione tridimensionale a partire da un set di immagini. Oltre al tutorial (ne è presente anche uno per l'installazione), vengono mostrati anche i prerequisiti di sistema. Tali requisiti sono molto esigenti, dal momento che sono richiesti una scheda *Nvidia*, per la ricostruzione della densità a qualità alta, e *32GB* di *RAM* per il meshing.

Meshroom è rilasciato come OpenSource sotto i permessi della licenza *MPLv2*.

L'ultima versione ad oggi disponibile è la versione *Meshroom 2021.1*.

La procedura per generare un modello 3D da un set di immagini bidimensionali, è la seguente:

- Dopo l'installazione (è possibile scaricare l'archivio *RAR*, estrarre la cartella ed utilizzare l'applicazione direttamente dall'interno della cartella stessa), Il primo step consiste nell'importare le immagini all'interno del software e, per farlo, ci basterà semplicemente trascinarle nello spazio *Images*.
- *Meshroom* possiede un database di sensori per determinare i parametri intrinseci ed estrinseci della camera. Qualora le immagini importate avessero dei metadati mancanti, ci verrebbe segnalato attraverso il prompt dei comandi. Con alcuni metadati mancanti, il processo andrà avanti comunque ma con un possibile degrado dei risultati.
- A questo punto, è sufficiente cliccare sul tasto **Start** e la ricostruzione avrà inizio. È importante precisare quanto i requisiti di sistema siano importanti (non possedere una scheda *Nvidia* può rappresentare inizialmente un problema). In fondo alla schermata troviamo un **Graph editor**

e un **Task manager**, dai quali otteniamo informazioni riguardo le operazioni da svolgere e quelle già compiute. Non solo possiamo osservare dalla log window quali operazioni si stanno svolgendo, possiamo intuire se tutto procede in maniera corretta osservando i colori (VERDE:= Operazioni compiute; AZZURO := Operazioni da svolgere; ROSSO := Presenza di errori e conseguente arresto del processo).

- Il primo possibile errore possiamo riscontrarlo nella **Structure From Motion**: Il set di immagini presenta la mancanza di alcuni metadati o non si hanno immagini a sufficienza per la ricostruzione (risolvibile facilmente andando a modificare il preset nel nodo **FeatureExtraction** da normale ad alto o ultra).
- Prima si accennava all'importanza dei prerequisiti del sistema. Dalla versione *Meshroom 2019.1* è possibile effettuare una ricostruzione 3D anche senza una scheda *Nvidia*, ma andando a modificare la struttura dei nodi. I nodi si trovano all'interno del **Graph editor** e sono collegati tra loro in funzione di un percorso preimpostato dal software e che presuppone una scheda Nvidia. Se non possediamo tale scheda quello che dobbiamo fare è modificare il collegamento tra i vari nodi e rimuoverne tre in particolare:

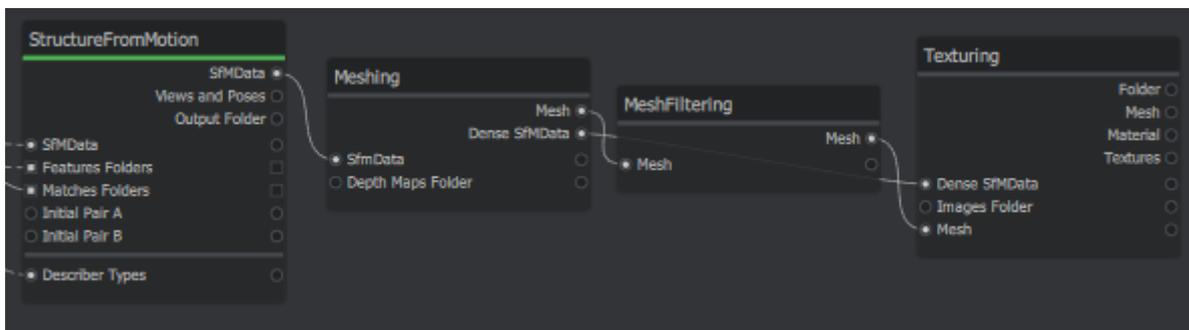


Figura 3.1.5.1

Completata la Structure From Motion, questo sarà il risultato:

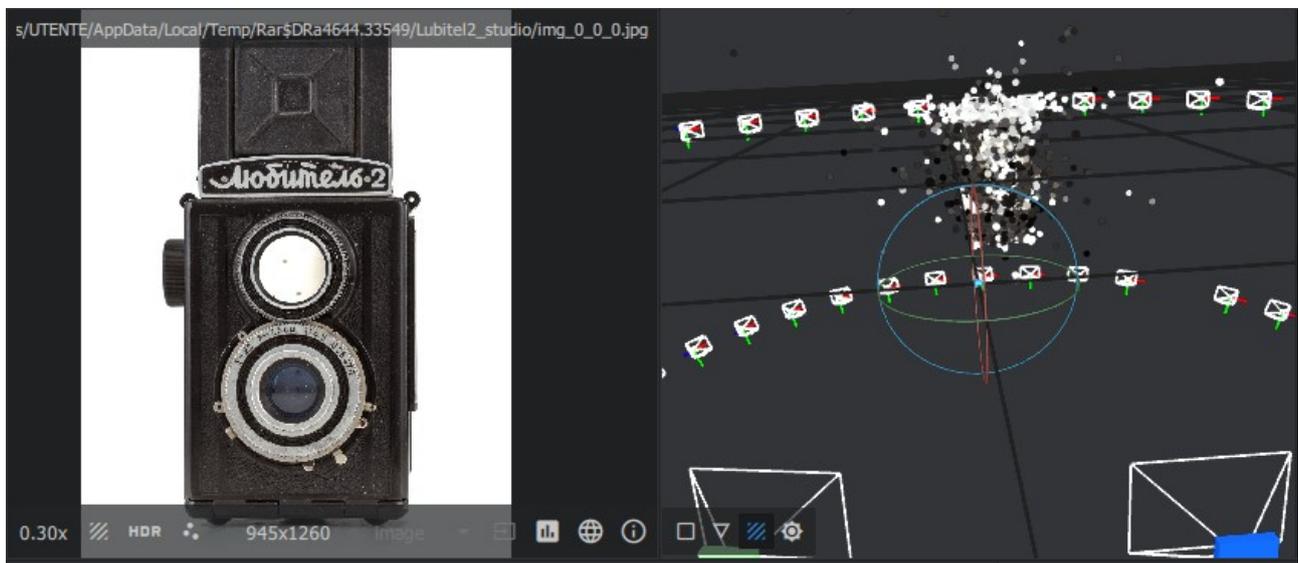


Figura 3.1.5.2

Il prossimo passo è la costruzione della Mesh e in seguito il Texturing. Qui di seguito è presente il risultato finale, non del tutto ottimale, considerati i limiti dell'hardware e il mancato utilizzo di un GPU Nvidia.



Figura 3.1.5.3

3.1.6 TABELLA DI CONFRONTO

Dopo aver introdotto anche Meshroom, sarà presentato un confronto diretto attraverso una tabella che mette in relazione ogni singolo programma rispetto a funzioni utili per la cattura di immagini e l'analisi ambientale.

<u>Tabella di confronto</u>	VisualSfM	CloudCompare	Agisoft Metashape	Meshroom
<i>Avvio da remoto con comandi Batch</i>	Si possono importare ordini dalla linea di comando. Utilizzandolo in file di tipo batch, gli utenti windows devono anteporre ad ogni comando <code>'cmd /c VisualSfM'</code>	Dalla versione 2.3 è possibile importare ordini dalla linea di comando e vi sono differenti modi per utilizzare CloudCompare con un file di tipo batch	Agisoft supporta la possibilità di essere automatizzato utilizzando comandi di tipo batch. Inoltre, può essere utilizzato, attraverso script in Python, dalla linea di comando	Meshroom può essere utilizzato tramite file di tipo batch, inserendolo all'interno di una cartella contenente le immagini, automatizzando così le operazioni
<i>Possibilità di importare i parametri di calibrazione della camera</i>	Salvato il progetto, vi è all'interno un file di testo <code>'cameras_v2'</code> contenente i parametri di calibrazione delle camere. I parametri vengono quindi calcolati automaticamente. La calibrazione opzionale esiste solo se si usa la funzione <code>'set fixed calibration'</code>	<i>Edit > Sensors > Camera sensors > Create Menu.</i> Dalla versione 2.5.6 è possibile, attraverso questa funzione, creare dei sensori per le camere, potendo così creare/modificare i parametri estrinseci e intrinseci delle camere	Metashape supporta una funzione per importare i parametri di calibrazione della camera. È possibile, infatti, selezionare <i>File > Import cameras.</i> È possibile, inoltre, selezionare il formato del file che si vuole importare	Per ottenere risultati migliori, è possibile fare un update del database del sensore di camera, andando ad aggiungere un modello di camera mancante e dunque inserendo dei nuovi parametri di calibrazione

<u>Tabella di confronto</u>	VisualSfM	CloudCompare	Agisoft Metashape	Meshroom
<i>Possibilità di importare le posizioni delle camere prima della ricostruzione</i>	Non vi è alcuna funzione dedicata	Non è possibile importare posizione o orientazione delle camere, ma vi è una funzione apposita per l'impostazione manuale degli aspetti principali	Le posizioni delle camere e le locazioni GCP possono essere importate in diversi formati. Utilizzando ad esempio Agisoft XML, si ottiene un file di tipo <i>.xml</i>	Durante la modellazione, prima della SfM, è possibile inserire le posizioni delle camere andando a modificare il file 'cameras' nella cartella 'StructureFromMotion'
<i>Possibilità di esportare le posizioni delle camere dopo la ricostruzione</i>	Dopo l'utilizzo dei tool CMVS/PMVS, viene generato il file 'camera_v2' contenente anche le posizioni relative alle camere	<i>Display > Save viewport (CTRL+V) permette di salvare i parametri delle camere come entità 'Viewport'. Tali entità possono essere salvate come file di tipo .BIN</i>	Le posizioni delle camere possono essere esportate e le locazioni GCP possono essere stimate. Utilizzando ad esempio Agisoft XML, si ottiene un file di tipo <i>.xml</i>	Terminata la modellazione, possiamo controllare ed esportare le posizioni delle camere dal file 'cameras' nella cartella 'StructureFromMotion'
<i>Output in formato .ply</i>	Salvato il progetto, vi è all'interno la cartella 'Models' contenente un file in formato <i>.ply</i>	<i>Importato un file .ply e apportate varie modifiche alla mesh, si ottiene in uscita un file di tipo .BIN</i>	Salvato il progetto, viene generato un file di tipo <i>.psx</i> ed un archivio RAR contenente un file Mesh di tipo <i>.ply</i>	Alla creazione del modello, viene generato un file <i>.MG</i> contenente i risultati della SfM ed un file <i>.OBJ</i> (nella cartella meshing) contenente la mesh del modello

CAPITOLO 4

4.1 APPLICAZIONE DELLE TEORIE DI RICOSTRUZIONE E STUDIO SU PROBLEMATICHE PRESENTI NELL'AMBIENTE MARINO

4.1.1 RICOSTRUZIONE DI UN CUBO SITUATO SUL FONDALE MARINO

Affrontati i concetti teorici dietro la funzione di un software di fotogrammetria e confrontati vari software fotogrammetrici tra quelli più utilizzati, l'attività di tirocinio prevedeva che si utilizzasse uno di questi software al fine di ricostruire, partendo da un set di immagini bidimensionali e scattate secondo uno schema preciso, le facce di un cubo situato sul fondale marino.

Tra i software presi in considerazione, poiché presenta alcuni vantaggi rispetto agli altri concorrenti, è stato scelto Agisoft Metashape per effettuare la ricostruzione delle facce.

Le immagini sono state scattate partendo da un'estremità della faccia e procedendo in linea retta scattando una foto ogni circa 10cm. Di seguito l'esempio di una sequenza di scatto:

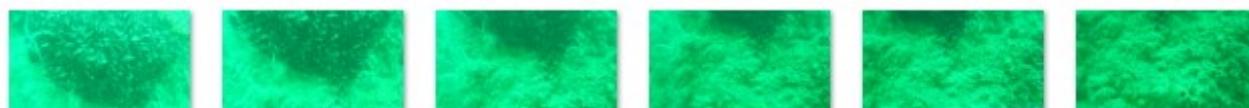


Figura 4.1.1.1

Per catturare l'intera faccia non bastava una sola striscia e, dunque, ogni faccia è divisa in più strisce a loro volta divise in più immagini, catturate seguendo il metodo appena indicato.

Striscia per striscia, le immagini sono state importate all'interno di Agisoft Metashape ed è dunque stato avviato il processo di allineamento delle immagini.

Non esiste alcun software sviluppato appositamente per la fotogrammetria sottomarina. Agisoft Metashape è un programma che funziona benissimo nell'ambito della fotogrammetria aerea, ma ha presentato qualche difficoltà fin da subito nell'allineamento delle immagini scattate sott'acqua.

In alcuni casi l'allineamento presentava delle imprecisioni, mentre in altri il software falliva completamente. Ciò poteva dipendere dal fatto che le foto fossero state scattate a distanza molto ravvicinata l'una dall'altra e dalla visibilità ridotta a causa dell'acqua. Dunque, a causa della nitidezza delle immagini inferiore rispetto, ad esempio, ad una foto scattata in aria da un drone, il software non era in grado di trovare abbastanza punti in comune ed allineare correttamente tutte le immagini.

Fortunatamente, Agisoft Metashape presenta la possibilità di allineare le immagini manualmente.

Gli oggetti utilizzati per l'allineamento sono chiamati Marker. Generalmente, un Marker è utilizzato per marcare, appunto, un preciso punto sull'immagine che poi verrà evidenziato anche sul modello tridimensionale ricostruito, nel caso fosse necessario tenere traccia di alcuni punti d'interesse durante le varie fasi di ricostruzione del modello.

Un secondo utilizzo è quello di identificare vari punti di confronto tra le immagini. Quello che si fa, dunque, è creare un marker, posizionarlo sull'immagine di riferimento, aprire una seconda immagine e posizionare su di essa lo stesso marker sul punto identico a quello di riferimento.

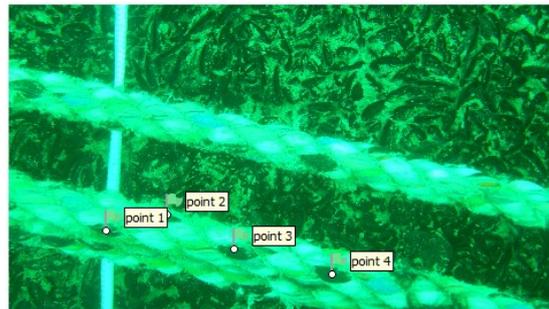


Figura 4.1.1.2

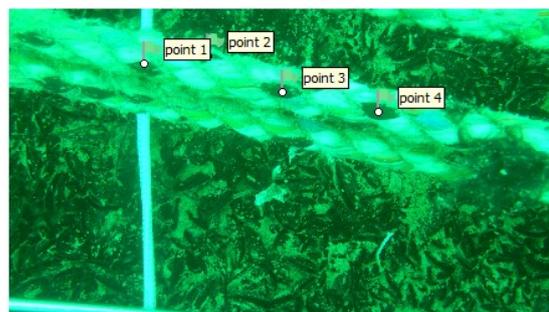


Figura 4.1.1.3

È necessario un minimo di quattro marker per avviare il processo di allineamento manuale delle immagini. Posizionati i marker e avviato il processo, le immagini saranno allineate manualmente e si aggiungeranno al set di immagini già allineate automaticamente da Agisoft Metashape.

È possibile che anche questo metodo fallisca o che non restituisca comunque un risultato ottimale. Inoltre, rischia di diventare un metodo troppo macchinoso e di portare via molto tempo, nel caso si avesse un largo set di immagini che il software non è riuscito ad allineare correttamente.

Esiste un terzo metodo per allineare un set di immagini, andando a lavorare sui metadati. Tale metodo consiste nell'avviare l'allineamento automatico ed esportare i parametri intrinseci ed estrinseci della camera, andando a modificarli come si desidera.

In Agisoft Metashape è possibile esportare i parametri delle camere in formato *XML*.

È capitato più volte, durante l'attività di tirocinio, che Agisoft Metashape non riuscisse ad allineare interi gruppi di immagini ed è, quindi, stato necessario esportare i parametri della camera e modificare il file, per poi importarlo nuovamente su Agisoft Metashape e avviare il processo di allineamento delle immagini. Questa volta però il software vedrà che i parametri sono già impostati e allineerà tutte le immagini di conseguenza, senza restituire alcun tipo di errore o imprecisione.

```
<cameras next_id="88" next_group_id="0">
  <camera id="44" sensor_id="2" label="DSCN3430" enabled="0">
    <orientation>1</orientation>
  </camera>
  <camera id="45" sensor_id="2" label="DSCN3431" enabled="0">
    <transform>9.8697721016912188e-001 -9.4700803194994361e-003 1.6058114517439120e-001 -1.84419
    <orientation>1</orientation>
  </camera>
  <camera id="46" sensor_id="2" label="DSCN3432" enabled="0">
    <transform>9.8697721016912188e-001 -9.4700803194994361e-003 1.6058114517439120e-001 -1.84419
    <orientation>1</orientation>
  </camera>
  <camera id="47" sensor_id="2" label="DSCN3433" enabled="0">
    <transform>9.8697721016912188e-001 -9.4700803194994361e-003 1.6058114517439120e-001 -1.84419
    <orientation>1</orientation>
  </camera>
  <camera id="48" sensor_id="3" label="DSCN3434" enabled="1">
    <transform>9.8697721016912188e-001 -9.4700803194994361e-003 1.6058114517439120e-001 -1.84419
    <orientation>1</orientation>
```

Figura 4.1.1.4

Per lavorare sul file XML è stato utilizzato il software Visual Studio Code, utile per la programmazione in Python.

Per ogni singola immagine si hanno tutti i dati inerenti alle camere, compresa la matrice di calibrazione della camera, già presentata nel capitolo 2 della tesi, che determina la sua posizione

all'interno della scena. In questo caso, poiché le immagini sono state scattate lungo un asse fisso (asse y), è stato necessario modificare un solo valore per ogni camera affinché Agisoft Metashape restituisse un allineamento delle immagini soddisfacente. Per la qualità dell'allineamento delle immagini, Agisoft Metashape propone differenti opzioni: Lowest; Low; Medium; High; Highest. Durante l'attività di tirocinio, tenendo conto dei limiti dell'hardware, è stata utilizzata l'opzione High.

Dopo la fase dell'allineamento si è potuto procedere con la costruzione della nuvola densa di punti. Anche qui Agisoft Metashape propone differenti opzioni di qualità: Lowest; Low; Medium; High; Ultra High. Sempre tenendo in considerazione i limiti dell'hardware, è stata utilizzata l'opzione High.

Il risultato ottenuto dalla costruzione delle nuvole dense di punti delle facce è il seguente:

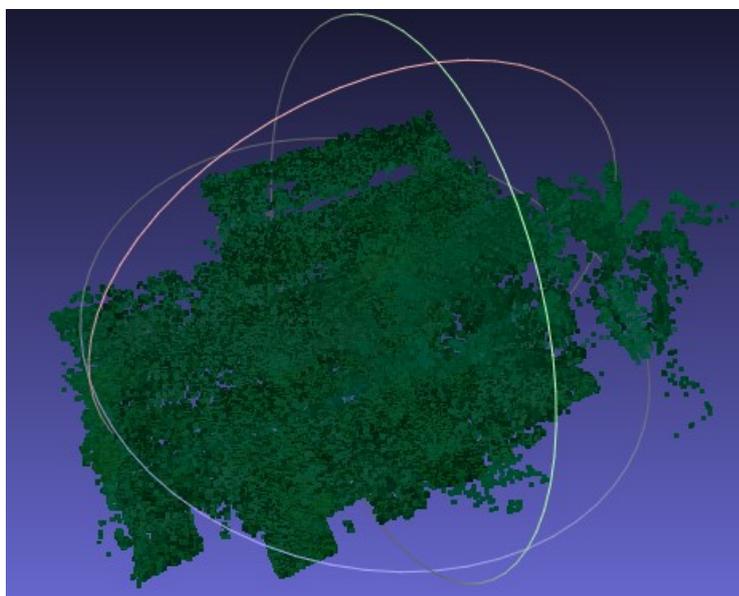


Figura 4.1.1.5

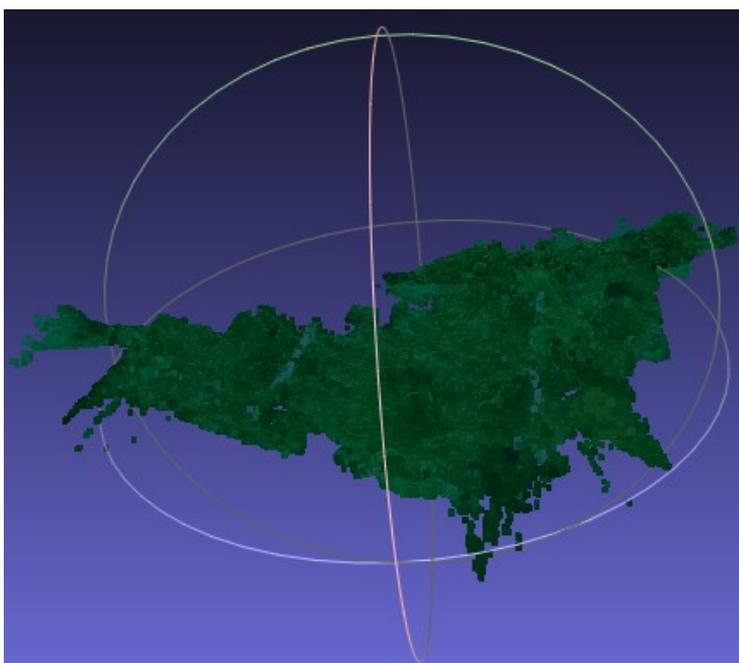


Figura 4.1.1.6

Dopo la costruzione della nuvola densa, è stato possibile costruire la mesh del modello.

La costruzione della mesh è il processo che richiede più tempo. Il tempo di completamento del processo dipende infatti dalla RAM e dal processore del pc che si utilizza e dal numero di punti della nuvola densa.

In questo caso, avendo un set che va all'incirca dalle 400 alle 600 immagini per faccia ed un hardware non molto performante, il tempo di completamento dell'intero processo poteva richiedere anche 3 o 4 ore.

Agisoft Metashape da, comunque, la possibilità di lasciar lavorare il software in background e utilizzare il proprio pc per altre attività, ma non è comunque consigliato, se si vuole ottenere un risultato il più ottimale possibile, poiché sarebbe meglio che la CPU lavori solo ed esclusivamente al modello.

Nella pagina successiva vi è il risultato finale ottenuto dalla ricostruzione delle facce.

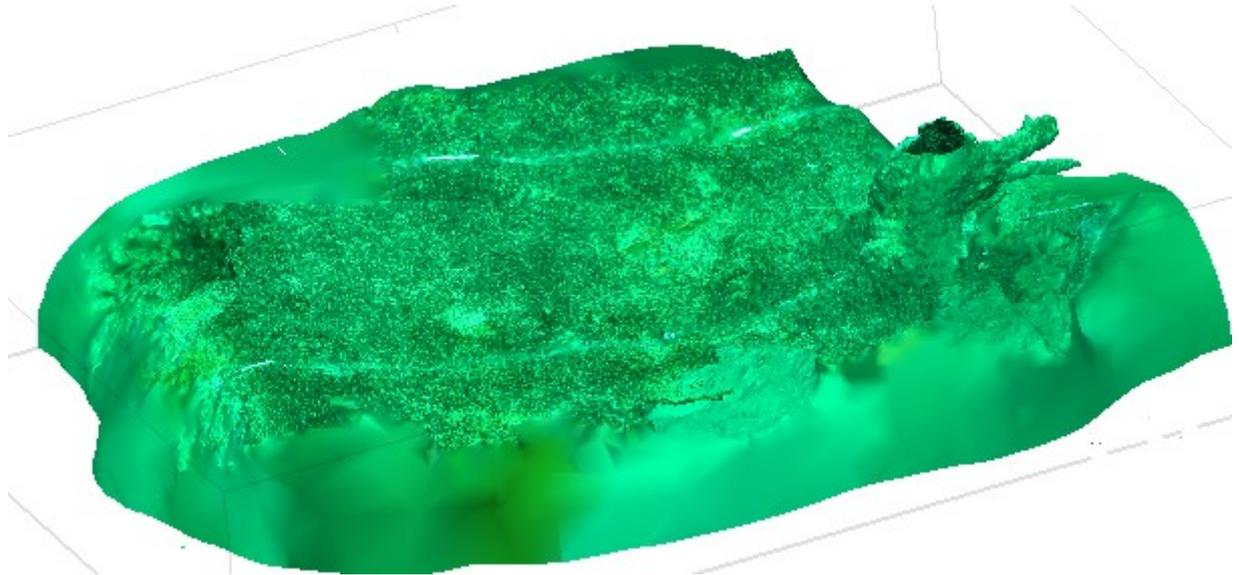


Figura 4.1.1.7

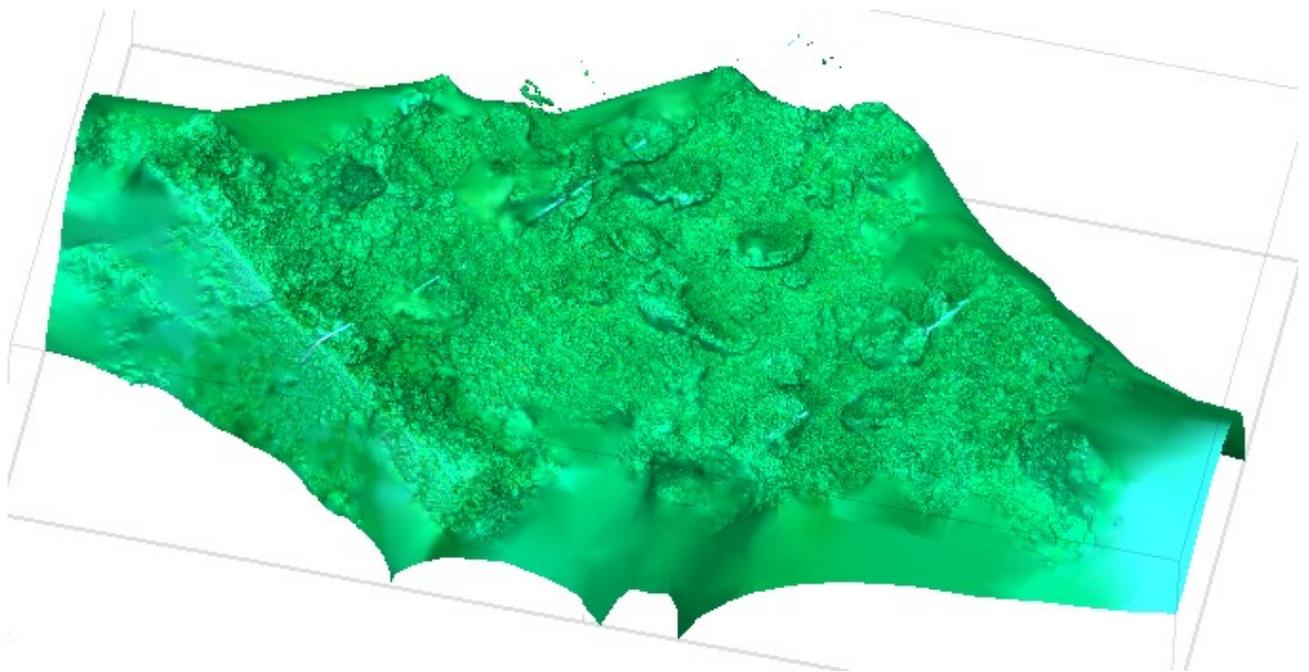


Figura 4.1.1.8

4.1.2 POSSIBILI SVILUPPI FUTURI

Quello che è stato fatto rappresenta in realtà solamente una parte di tutto quello che si potrebbe ancora fare.

Si è visto precedentemente com'è avvenuta la ricostruzione del cubo immerso in acqua, al fine di dimostrare quanto potrebbe essere utile ed efficace la fotogrammetria sottomarina in determinati ambiti.

Tuttavia, i vari procedimenti rendono la costruzione del modello piuttosto macchinosa. Si potrebbe, quindi, cercare di trovare un modo per semplificare ulteriormente il lavoro.

Scattare immagini sott'acqua non è assolutamente semplice come scattarle al di fuori. Supponiamo che un fotografo scatti alcune foto di un oggetto presente sul fondale marino. Solo una volta riemerso, egli potrà controllare la qualità delle immagini e utilizzare un software fotogrammetrico per ricostruire l'oggetto in questione. Tutto ciò è dispendioso a livello di tempo, tralasciando il fatto che alcune foto potrebbero essere venute male e il fotografo dovrà rimmergersi per scattarle nuovamente nella stessa identica posizione, rendendo il tutto non solo dispendioso in termini di tempo, ma anche in quanto a risorse.

Un ingente risparmio di tempo e risorse lo si potrebbe favorire attraverso l'utilizzo di un server e un client.

Scattate le foto, il client invierebbe le immagini al server e, una volta ricevute, si avvierebbe automaticamente la costruzione del modello tridimensionale attraverso il software fotogrammetrico in utilizzo.

Dunque, potrebbe essere proficuo ricorrere all'uso di comandi Batch. In informatica con il termine Batch Processing ci si può riferire a più aspetti. L'aspetto che si vuole mettere in luce in questa tesi è quello dell'interattività dei programmi. In altre parole, per interattività dei programmi, si intende l'esecuzione di un determinato processo da parte di quest'ultimo in tempo reale.

Tornando al discorso di prima, il fotografo potrebbe scattare delle foto, il client invierebbe le immagini al server che, attraverso un software fotogrammetrico e l'utilizzo di comandi batch,

avvierebbe la ricostruzione del modello tridimensionale in tempo reale, semplificando di gran lunga il lavoro e rendendolo in buona parte automatizzato.

In questo modo ci si potrebbe accorgere prima di riemergere di eventuali errori o imprecisioni riguardanti la qualità delle immagini catturate.

Agisoft Metashape offre anche la possibilità di eseguire i processi attraverso l'utilizzo di comandi batch, come già accennato nella tabella di confronto tra i vari software nel capitolo 3, attraverso script realizzati in Python.

Qualora si decidesse di utilizzare Python, è importante indicare il percorso delle immagini che dovranno essere importate all'interno di Agisoft Metashape e la cartella in cui Agisoft Metashape è stato installato.

Quello che si fa attraverso Python è scrivere un algoritmo che automaticamente avvii i processi che desideriamo e, per ogni processo, indicarne le specifiche desiderate.

In realtà Agisoft Metashape semplifica di gran lunga l'aspetto dei comandi Batch, rendendo l'utilizzo di Python non per forza necessario.

Vi è anche la possibilità di utilizzare comandi batch direttamente dal software. Esiste proprio una finestra dedicata ai comandi batch che permette all'utente di selezionare tutti i comandi da mettere in fila e a quali chunk applicarli, lasciando così che il software faccia tutto automaticamente.

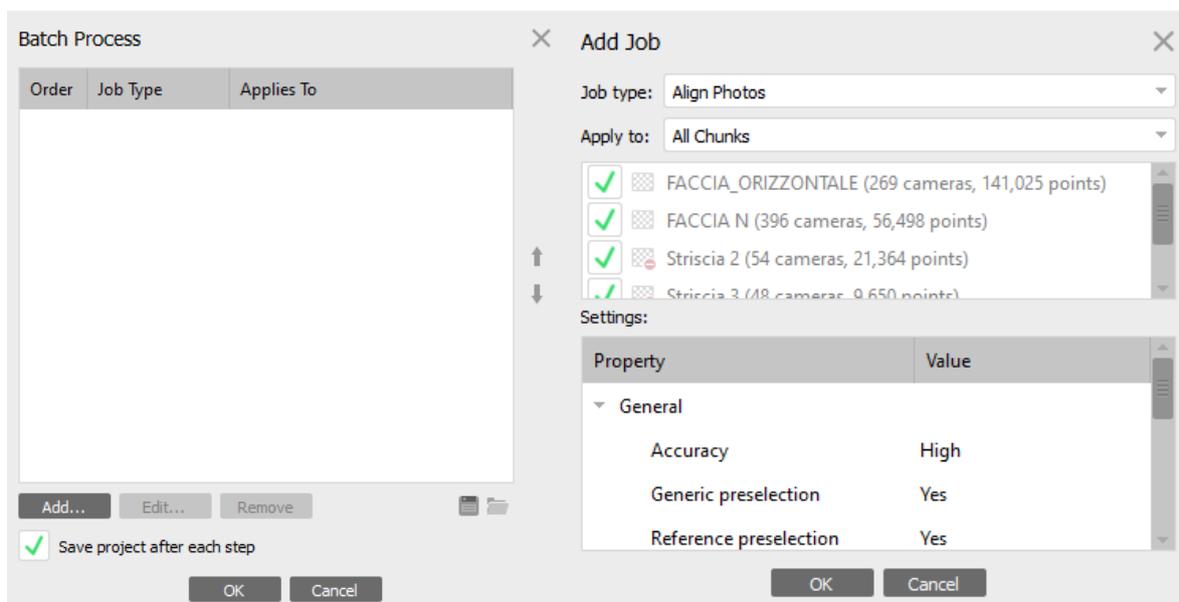


Figura 4.1.1.9

Un secondo interessante sviluppo di questo progetto potrebbe essere la possibilità di sviluppare e, quindi, utilizzare un secondo software per l'allineamento dei modelli 3D realizzati con un software fotogrammetrico.

Dunque, si potrebbe implementare un software in grado di ricevere in input modelli 3D differenti e ricercare passo dopo passo, secondo precise regole matematiche, dei punti di confronto tra essi.

Il software, quindi, esattamente come Agisoft Metashape è in grado di ricercare dei punti di confronto tra varie immagini ed allinearle di conseguenza, sarebbe in grado di ricercare dei punti di contatto tra le figure, affinché i modelli 3D si incastrino perfettamente e risultino, così, allineati.

Potrebbe essere utile ad alleggerire il carico per il software e l'hardware, costruendo piccole parti alla volta ed unendole, infine, in unico modello tridimensionale.

Il software potrebbe anche permettere di operare sul modello, effettuando alcune misurazioni, per facilitare l'analisi del modello stesso o rimediare ad alcune imprecisioni durante la sua costruzione.

Finora si è visto:

- Com'è nata la fotogrammetria e quali sono stati i maggiori contribuenti;
- Di cosa tratta in particolare la fotogrammetria sottomarina e il motivo del suo sviluppo arretrato rispetto alle altre tipologie;
- Quali sono le teorie matematiche dietro la fotogrammetria e in particolare dietro al funzionamento dei software fotogrammetrici;
- Una lista con relativa descrizione dei software fotogrammetrici, tra quelli più utilizzati, e le loro caratteristiche;
- Una tabella di confronto tra i vari software analizzati, riguardante caratteristiche e funzioni relativamente utili nella costruzione di un modello tridimensionale a partire da immagini bidimensionali;
- L'applicazione pratica delle teorie e dei concetti affrontati, tramite la costruzione di alcune facce di un cubo, situato sul fondale marino, mediante il software Agisoft Metashape.

In conclusione, la fotogrammetria sottomarina è ancora argomento di studio e mi aspetto che lo sarà per diverso tempo. Spero di essere stato bravo nel sottolineare al meglio possibile la sua utilità. Sono

tanti, infatti, gli ambiti in cui perfezionare una tecnologia del genere risulterebbe un enorme vantaggio ed alcuni esempi sono stati trattati anche nel corso della tesi.

Sul web si trovano molte informazioni riguardo la fotogrammetria, trattata in maniera generale, e specialmente riguardo la fotogrammetria aerea, inoltre, esistono diversi software dedicati in particolare modo a quest'ultima. Si trovano, invece, davvero pochissime informazioni riguardanti la fotogrammetria sottomarina e questo è un motivo in più per muoversi in questa direzione e mettere in luce qualcosa che merita, di certo, la nostra attenzione.

Investire tempo e risorse in questo tipo di ricerca, non risulterà assolutamente uno spreco, anche se le difficoltà che la materia presenta sono molteplici. Spero, comunque, di aver dimostrato, scrivendo questa tesi, che tali difficoltà non sono di certo insormontabili.

RIFERIMENTI

David Scaradozzi, «Photogrammetry».

M. Aharchi and M. Ait Kbir, «A Review On 3D Reconstruction Techniques From 2D Images».

SITOGRAFIA

<https://www.botteghedellearti.it/>

<https://www.occhioumano.weebly.com/>

<https://www.aeromatic.com/>

<https://www.books.openedition.org/>

<https://www.treccani.it/>

<https://wikipedia.it/>

<https://www.archeomatica.it/>

RINGRAZIAMENTI

Il percorso universitario non è stato per niente semplice.

Tanti, infatti, sono stati i momenti di difficoltà che l'hanno e mi hanno accompagnato fino alla stesura di questa tesi.

Rifarei tutto da capo una seconda volta? La risposta è 'No'.

Consiglierei a qualcuno il mio stesso identico percorso, compresi gli stessi identici errori? 'Assolutamente sì'.

Sebbene non rifarei nulla da capo, non significa che non sia stato bello.

Quello che ho fatto in questi ultimi anni mi ha reso ciò che sono ora, ciò che NON sarei stato, se nulla di tutto questo fosse accaduto (e qui, al solo pensiero, rabbrivisco).

Gli errori commessi alla fine sono la parte più importante, la parte del cammino che ti plasma, che ti mostra davvero com'è la vita, che effettivamente nulla è facile da ottenere (anche quando l'impegno c'è), che l'unica soluzione a volte è sbattere la testa al muro o sulla scrivania ed aspettare il famoso colpo di fortuna (e, fidatevi, anche la fortuna ha avuto una sua parte nella storia).

Se mi dovessi paragonare ad un qualche personaggio reale o immaginario che sia, direi senza dubbio Peter Parker. Una persona sbadata, distratta, a volte ingenua e innocente che però, per ciò che vuole e in cui crede, combatte sempre a denti e pugni stretti senza mai arrendersi e che, nonostante i suoi continui e persistenti errori, non si sa come, ricade sempre in piedi o quasi.

Non sarà forse che la forza di Peter Parker risieda in parte nelle persone che gli sono accanto?

Nessuno riesce a realizzare i propri obiettivi da solo, contando esclusivamente sulle le sue sole forze.

Penso che, se mi mettessi lì a contare il numero di tutte le persone che mi sono state accanto, ci metterei un bel po'.

Il primo grazie va alla mia famiglia. I miei genitori e mia sorella non hanno mai smesso di credere in me e mi hanno sempre spronato ad andare avanti senza mai mollare (importante però è ricordare a papà che non è lui a lurearsi).

Ringrazio in generale i miei parenti. Sono sicuro che i miei nonni non abbiano mai smesso di pregare per me, anche se non sono una persona molto religiosa.

Ringrazio gli amici di una vita sia quelli con cui non parlo più, sia quelli con cui continuo a sentirmi spesso, con cui ho scambiato una marea di emozioni, mi sono divertito molto e che mi sono stati sempre vicini, sperando che tutto ciò non cambi mai (magari vi aspettavate un elenco di nomi o magari no perché mi conoscete e sapete benissimo che non sono il tipo).

Di certo, non mi dimentico di ringraziare tutte le persone e gli amici che ho conosciuto durante il percorso universitario, forse gli unici che hanno visto la mia vera disperazione, mentre mi scervellavo dietro ad algoritmi, sistemi, teoremi e dimostrazioni matematiche. Con loro ho condiviso l'intero percorso e ci siamo sempre aiutati nei momenti di difficoltà. In certi casi, ad alcuni ho letteralmente levato la salute, ma, nonostante ciò, non mi hanno mai negato il loro aiuto. In altri casi, invece, sono stati loro stessi a farsi avanti senza che neanche chiedessi nulla.

Detto ciò, sapete tutti che non sono il tipo dalle mille parole, e forse alcuni di voi all'interno dei ringraziamenti si aspettavano un semplice e solo "grazie a me stesso", ma evidentemente così non è stato.

Di nuovo, **GRAZIE A TUTTI.**

Simone Di Rado