



UNIVERSITÀ DEGLI STUDI DI GENOVA

CORSO DI STUDIO IN INGEGNERIA INFORMATICA

Relazione finale per il conseguimento del titolo di
Dottore in Ingegneria Informatica

**SVILUPPO DI UN SISTEMA PER MISURAZIONI
3D, BASATO SU MICROCONTROLLORE E
SENSORE TOF ED INTEGRAZIONE IN AMBIENTE
DOMOTICO**

Arash Alghasi

Dicembre 2025

Relatore: Prof. Marco Raggio

Correlatore: Dott. Ing. Alberto Cabitto

Sommario

Il presente lavoro di tesi studia l'utilizzo e i casi applicativi di un sensore Time-of-Flight. L'obiettivo è dimostrare che un sensore economico TOF può avere la potenza e la precisione necessarie per scansionare l'ambiente in 3D, gettando le basi per futuri programmi che affronteranno e risolveranno i problemi più comuni relativi alla rilevazione degli ostacoli, alla machine vision, guida autonoma, alla navigazione robotica e alla scansione ambientale.

I programmi sviluppati in questa tesi affrontano quattro problemi: acquisizione dati, trasmissione, elaborazione e visualizzazione dei dati di un sensore time-of-flight.

Il lavoro si è concentrato sull'integrazione del sensore VL53L5CX con un microcontrollore generico, sviluppando un sistema autonomo capace di controllare la posizione del sensore, acquisire dati tridimensionali dell'ambiente, monitorare il sistema e trasmettere informazioni senza l'uso di connessioni cablate.

I risultati ottenuti evidenziano che il sensore VL53L5CX si distingue per la capacità di raccogliere dati ambientali con accuratezza. Questa caratteristica lo rende una soluzione promettente e applicabile in numerosi settori e industrie.

I risultati ottenuti dimostrano che per molte esigenze come prevenzione delle collisioni e rilevazione degli ostacoli non è necessario utilizzare costosi sensori LIDAR e si può fare affidamento sul potenziale di questi sensori poco costosi e a basso consumo.

Si conclude raccomandando ulteriori studi e approfondimenti sul sensore VL53L5CX, al fine di esplorarne le potenzialità in contesti applicativi più ampi e di ottimizzarne ulteriormente le prestazioni.

Ringraziamenti

Innanzitutto, desidero ringraziare mia sorella, Atefeh, per il suo supporto infallibile l'incoraggiamento costante durante i miei anni di studio in Italia. Grazie dal profondo del mio cuore.

Vorrei ringraziare il Professor Marco Raggio, relatore della mia tesi e presidente del Centro di Simulazione e Formazione Avanzata dell'Università degli Studi di Genova, per tutto il suo prezioso aiuto, senza il quale questo lavoro di tesi non sarebbe stato possibile.

Indice

1. Introduzione	8
1.1. Motivazione	10
1.2. Obiettivi della tesi.....	10
1.3. Problemi affrontati	11
1.4. Struttura della tesi.....	12
2. Materiali e Metodologia.....	12
2.1. Componenti hardware.....	12
2.1.1. Sensore TOF VL53L5CX.....	12
2.1.2. Microcontrollore Nucleo-F446RET6.....	19
2.1.3. Bipolar stepper motor driver IHM05A1.....	24
2.1.4. Attuatori	25
2.1.5. Modulo wi-fi ESP8266-01s.....	27
2.1.6. Modulo NRF per comunicazione basso raggio	29
2.1.7. Sistema di controllo degli attuatori e logging locale	32
2.2. Architettura software.....	34
2.2.1. Ambiente di Sviluppo e Strumenti Utilizzati.....	34
2.2.2. Algoritmo di Elaborazione Dati Embedded Nel Sensore TOF.....	38
2.2.3. Protocollo di Comunicazione MQTT	39
2.3. Progettazione Meccanica.....	42
2.3.1. Struttura del Robot e Posizionamento dei Componenti.....	42
2.3.2. Diagrammi Elettrici e Collegamenti	45
3. Implementazione	47
3.1. Integrazione Hardware e Software	47
3.2. Configurazione dei Moduli NRF	50
3.3. Configurazione dei Moduli WiFi	50
3.4. Elaborazione dei Dati del Sensore	51
3.5. Invio Dati via MQTT	54

3.6.	Controllo dei Motori	55
3.7.	Monitoraggio in Tempo Reale	57
4.	Risultati e Analisi	58
4.1.	Visualizzazione dei Dati.....	58
4.2.	Piattaforma Python per la visualizzazione	62
4.3.	Test del Sistema di Misurazione 3D	64
5.	Discussione	71
5.1.	Rilevanza e Applicazioni in Ambito Domotico	71
5.2.	Possibili Miglioramenti del Sistema	73
6.	Conclusioni e Lavori Futuri	74
7.	Bibliografia.....	75

Figure

Figura 1. Dimensioni e tendenze del mercato della machine vision (3)	9
Figura 2. Confronto tra diversi sensori di profondità (4).....	11
Figura 3. Risoluzione multi-zona del VL53L5CX	15
Figura 4. Scheda breakout VL53L5CX-SATEL - Sensore Time-of-Flight 8x8	16
Figura 5. Schemi del VL53L5CX-SATEL	16
Figura 6. Applicazioni ovunque del sensore ToF.....	17
Figura 7. Misurazione del livello dei liquidi in tempo reale (7)	17
Figura 8. Accuratezza della misurazione in funzione della distanza	18
Figura 9. Microcontrollore NUCLEO-F446RE	20
Figura 10. Diagramma del circuito dell'STM32F446xC/E (9).....	21
Figura 11. Diagramma a blocchi hardware STM32F446RET6 (8)	22
Figura 12. Disposizione superiore STM32F446RET6 (8)	23
Figura 13. Connettori di espansione NUCLEO-F446RE (8)	23
Figura 14. Scheda di espansione per il controllo motore X-NUCLEO-IHM05A1 (10)	25
Figura 15. Stepper Motor	26
Figura 16. Servo motor	27
Figura 17. Modulo Wi-Fi ESP-01S.....	28
Figura 18. Schema dei pin dell'ESP8266-01S	29
Figura 19. Modulo NRF24L01	30
Figura 20. Schema dei pin del modulo NRF24L01	31
Figura 21. Sistema di controllo remoto	32
Figura 22. Sistema di logger locale	33
Figura 23. Arduino Uno REV3 Microcontrollore.....	34
Figura 24. Ambiente di sviluppo integrato STM32CubeIDE (14)	36
Figura 25. Ambiente software PuTTY (15)	37
Figura 26. Ambiente software MQTTX.....	38
Figura 27. L5 ha una risoluzione estremamente bassa di 8×8 zone e fornisce una distribuzione della profondità per ciascuna zona.	39
Figura 28. Architettura Publisher-Subscriber di MQTT	41

Figura 29. Foto del Progetto della Tesi completato	44
Figura 30. Schema di configurazione dei pin (2)	46
Figura 31. L'architettura del software X-CUBE-SPN5.....	49
Figura 32. Diagramma di stato del modulo ESP01-S Wi-Fi.....	51
Figura 33. esempio di messaggio del broker MQTT	55
Figura 34. Data logger terminale seriale	57
Figura 35. Esempio di heatmap	59
Figura 36. Esempio di Point Cloud	60
Figura 37. Grafico Wireframe 3D	61
Figura 38. Piattaforma di visualizzazione in Streamlit.....	63
Figura 39. Caso di test piccolo barattolo	64
Figura 40. Caso di test della faccia frontale della statua	65
Figura 41. Caso di test della faccia laterale della statua	66
Figura 42. Caso di test grande barattolo.....	67
Figura 43. Caso di test della teiera	68
Figura 44. Caso di test del libro.....	69
Figura 45. Caso di test del bicchiere da vetro.....	70
Figura 46. Sensore ToF nei sistemi di automazione domestica.....	71
Figura 47. Applicazione del sensore ToF	72
Figura 48. Misurazione di area e volume utilizzando un sensore ToF.....	73

1. Introduzione

I sensori di profondità Time-of-Flight (ToF) leggeri sono piccoli, economici e a basso consumo energetico, e sono stati ampiamente impiegati nei dispositivi mobili per applicazioni come l'autofocus e il rilevamento degli ostacoli. Tuttavia, a causa delle loro misurazioni specifiche (che forniscono una distribuzione della profondità in una regione anziché un valore di profondità per un singolo pixel) e della loro risoluzione estremamente bassa, risultano insufficienti per applicazioni che richiedono una profondità ad alta fedeltà, come la ricostruzione 3D, SLAM.

La visione artificiale tridimensionale è stata discussa per la prima volta negli anni '60' da Larry Roberts, considerato il padre della computer vision, nella sua tesi di dottorato al MIT. In questa ricerca, Roberts introdusse la possibilità di estrarre informazioni geometriche tridimensionali a partire da viste prospettiche bidimensionali di blocchi. Successivamente, nel 1978, un contributo significativo fu fornito da David Marr, un ricercatore del MIT, che propose un approccio di elaborazione bottom-up per lo studio delle scene. Questo metodo prevedeva la costruzione di un'immagine partendo dai più piccoli frammenti di informazioni sensoriali.

Il lavoro di Marr è stato ampiamente apprezzato per anni ed è probabilmente l'opera più influente nel campo della visione artificiale. Tuttavia, alcuni ricercatori hanno cercato di allontanarsi dal paradigma di Marr, adottando un approccio opposto basato sull'elaborazione top-down, in particolare nelle applicazioni per veicoli autonomi. In questi casi, infatti, non è necessario conoscere il modello 3D completo di un oggetto, ma solo alcune informazioni generali e parziali, come ad esempio se un elemento si sta muovendo verso o lontano dal veicolo. (1)

Oggi, la visione artificiale 3D è una tecnologia in continua evoluzione e maturazione, utilizzata in una vasta gamma di applicazioni. Essa nasce dalla necessità di ottenere informazioni più accurate, precise e significative rispetto ai sistemi basati sulla semplice visione 2D. Negli ultimi anni, la visione 3D ha avuto un impatto significativo sul mercato, con una crescita esponenziale nella diffusione dei suoi componenti.

I sistemi robotici per la navigazione utilizzano più comunemente sensori visivi, acustici ed elettromagnetici. Le tecnologie più utilizzate sono LIDAR 2D e 3D (2), lo spettro visibile, lo spettro infrarosso, le telecamere Time-of-Flight e i sensori a ultrasuoni.

Il mercato globale della visione artificiale 3D è stato stimato a 7,16 miliardi di dollari nel 2024 e si prevede che crescerà a un tasso di crescita annuale composto (CAGR) del 14,3% dal 2025 al 2030. Il mercato della visione artificiale dovrebbe essere trainato dalla crescente domanda di automazione e ispezione della qualità in diversi settori industriali. Inoltre, la richiesta di sistemi robotici guidati dalla visione nei settori dell'imballaggio, alimentare e delle bevande, farmaceutico e chimico, nonché in quello automobilistico, è destinata a favorire l'espansione del mercato. L'adozione di questa tecnologia è inoltre prevista in aumento nel periodo di previsione, grazie alla crescente domanda di sistemi di visione artificiale orientati alle applicazioni. (3)

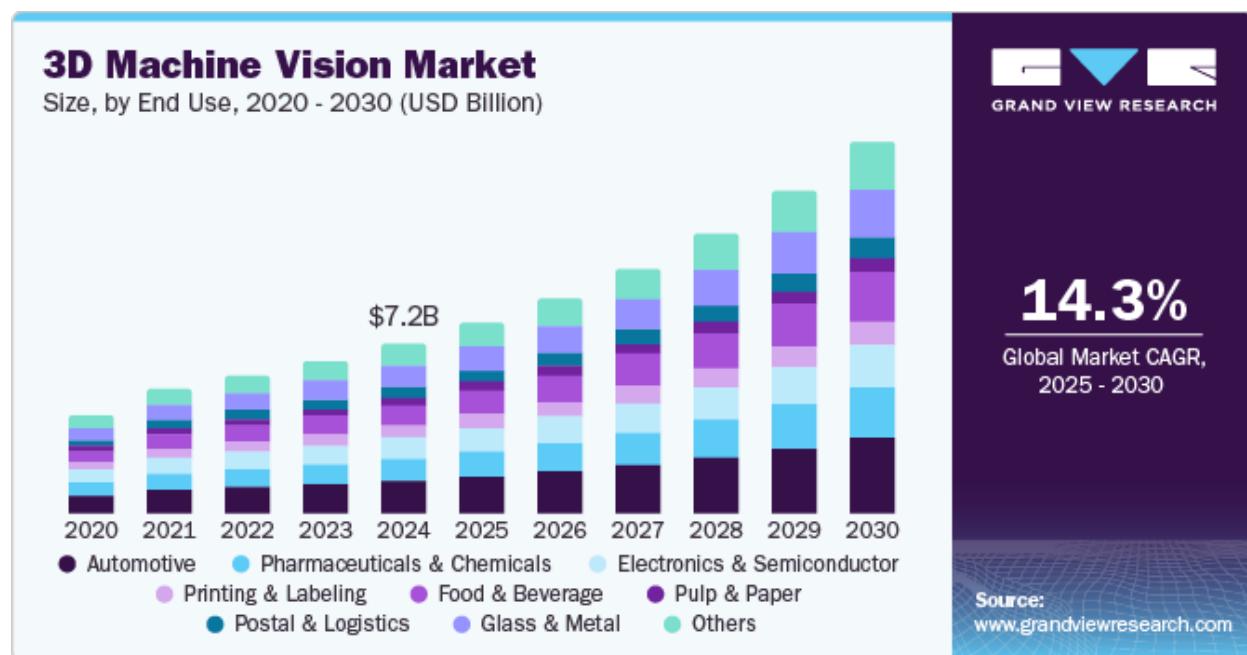


Figura 1. Dimensioni e tendenze del mercato della machine vision (3)

1.1. Motivazione

Le applicazioni possibili della visione artificiale 3D sono molteplici, spaziando dall'automobilistico al gaming, dal settore sanitario e alla robotica. In particolare, questa tecnologia viene sempre più utilizzata nei settori automobilistico e robotico.

Nel settore automobilistico, si punta a sviluppare sensori migliori e a raccogliere dati più accurati per migliorare le prestazioni dei veicoli a guida autonoma. Una delle caratteristiche più importanti che un'auto a guida autonoma deve possedere è l'evitamento delle collisioni, diventato essenziale a seguito dell'elevato numero di incidenti causati da negligenza del conducente.

Per quanto riguarda la robotica, la visione 3D trova impiego nel controllo qualità, nelle linee di assemblaggio e nei robot collaborativi. Nel controllo qualità, la visione 3D è vantaggiosa grazie alla sua capacità di analizzare superfici complesse a una velocità superiore rispetto agli esseri umani. Nelle linee di assemblaggio, offre maggiore flessibilità, indipendenza e affidabilità rispetto ai sistemi 2D, poiché consente di riconoscere parti differenti e di afferrarle su un nastro trasportatore nel modo più corretto, anche quando non sono tutte orientate nello stesso modo. Infine, i robot collaborativi si basano su sistemi di visione per evitare collisioni con gli esseri umani presenti nell'ambiente circostante. La visione artificiale 3D rende questo processo più sicuro, permettendo al robot collaborativo di operare a velocità più elevate e con carichi maggiori.

1.2. Obiettivi della tesi

L'obiettivo di questo lavoro è dimostrare l'elevata capacità di un sensore ToF economico e a basso consumo, oltre a un sistema per controllare dove punta il sensore. Inoltre, è stata sviluppata un'applicazione web per la visualizzazione di questi dati.

Il sistema di controllo remoto permette di gestire gli attuatori e Il sistema di logging permette monitorare il comportamento del minirobot mediante l'invio di dati in tempo reale.

1.3. Problemi affrontati

Uno dei problemi più critici affrontati durante la realizzazione della tesi è stata la mancanza di componenti per costruire una base su cui montare i vari elementi elettronici. È stato quindi necessario progettare un supporto utilizzando software di disegno come AutoCAD o una stampante 3D.

Tuttavia, nonostante il prezzo e il consumo energetico dei sensori ToF siano già significativamente inferiori rispetto ad altre tecnologie, come la luce strutturata (ad esempio Microsoft Kinect V1), rimangono ancora uno o due ordini di grandezza superiori rispetto a una tipica fotocamera RGB quando si cerca di ottenere una risoluzione simile. Questo è dovuto al grande numero di fotoni che devono essere emessi, raccolti e processati.

D'altro canto, i sensori ToF leggeri sono progettati per essere economici, di piccole dimensioni e a basso consumo energetico, motivo per cui sono stati ampiamente implementati nei dispositivi mobili per applicazioni come l'autofocus e il rilevamento degli ostacoli. Tuttavia, a causa del loro design elettronico semplificato, la profondità misurata da questi sensori presenta una maggiore incertezza (ossia, viene fornita come distribuzione anziché come singolo valore di profondità) e una bassa risoluzione spaziale (inferiore a 10×10). Di conseguenza, non possono supportare applicazioni avanzate come la ricostruzione 3D o lo SLAM. (4)

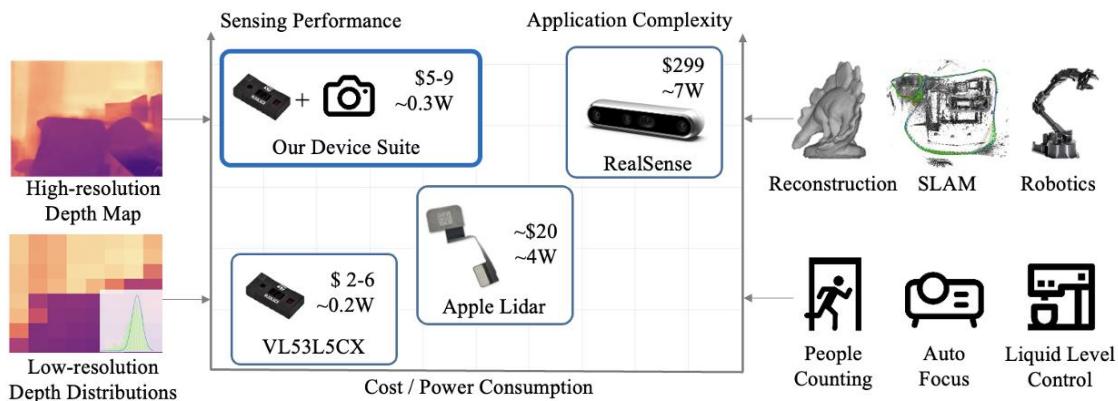


Figura 2. Confronto tra diversi sensori di profondità (4)

1.4. Struttura della tesi

La tesi è suddivisa in quattro sezioni principali.

La prima sezione riguarda l'integrazione dei componenti elettronici per la realizzazione del progetto, che consiste nella combinazione e configurazione di diverse librerie provenienti da varie aziende, oltre al cablaggio e alla configurazione dei pin.

La seconda sezione è dedicata all'acquisizione dei dati dal sensore, alla loro manipolazione e pulizia prima del trasferimento.

La sezione successiva è dedicata al trasferimento dei dati acquisiti dal sensore. Non è previsto il salvataggio dei dati sulla scheda di memoria SD, poiché l'obiettivo è analizzarli successivamente tramite un software di terze parti in grado di funzionare autonomamente.

L'ultima sezione della tesi riguarda la ricezione dei dati tramite il broker MQTT, l'elaborazione dei dati e la loro visualizzazione attraverso grafici basati sulle misurazioni del sensore.

2. Materiali e Metodologia

2.1. Componenti hardware

2.1.1. Sensore TOF VL53L5CX

Scheda breakout ST VL53L5CX – Sensore Time-of-Flight 8x8 (5)

Il sensore ToF VL53L5CX (denominato L5) è una scheda breakout che fornisce 8×8 zone di rilevamento, ciascuna con una distribuzione della profondità, coprendo un campo visivo diagonale (Field of View) di 63° e con un consumo energetico di circa 200 mW (rispetto ai 4W

del Lidar di Apple). È in grado di misurare la distanza effettiva indipendentemente dalla dimensione e dalla riflettanza del bersaglio.

Il VL53L5CX è un sensore laser ToF all'avanguardia che arricchisce la famiglia di prodotti ST FlightSense. Integrato in un modulo miniaturizzato e reflowabile, include una matrice di SPAD (Single Photon Avalanche Diode), filtri infrarossi fisici e ottiche diffrattive (DOE) per garantire le migliori prestazioni di rilevamento in diverse condizioni di illuminazione ambientale e con una varietà di materiali per il vetro di copertura. L'uso di un DOE sopra il laser a cavità verticale a emissione superficiale (VCSEL) consente di proiettare un campo visivo quadrato sulla scena. La riflessione di questa luce viene focalizzata dalla lente del ricevitore su una matrice di SPAD.

A differenza dei sensori IR convenzionali, il VL53L5CX utilizza l'ultima generazione della tecnologia ToF diretta di ST, che permette la misurazione assoluta della distanza indipendentemente dal colore e dalla riflettanza del bersaglio. Fornisce misurazioni accurate fino a 400 cm e può operare a velocità elevate (60 Hz), rendendolo il sensore ToF multi-zona più veloce e compatto sul mercato. Grazie agli algoritmi brevettati ST histogram, il VL53L5CX è in grado di rilevare più oggetti all'interno del FoV, con informazioni sulla distanza fino a 60 Hz. L'algoritmo Istogramma fornisce inoltre immunità al cross talk del vetro di copertura oltre i 60 cm. Le misurazioni di distanza multi-zona sono possibili fino a una risoluzione di 8x8 zone con un ampio FoV diagonale di 63°, riducibile via software.

Applicazioni

- Riconoscimento della scena: il rilevamento multizonale e multi-oggetto consente la mappatura 3D degli ambienti e il rilevamento ostacoli per la robotica
- Ampio FoV e scansione multizonale per la gestione del contenuto (carico in camion, serbatoi, contenitori di rifiuti)
- Riconoscimento gestuale
- Controllo del livello dei liquidi
- Correzione della distorsione trapezoidale per proiettori video
- Autofocus assistito da laser (LAF): migliora la velocità e l'affidabilità del sistema AF della fotocamera, soprattutto in condizioni di scarsa illuminazione o basso contrasto

- Miglioramento della realtà aumentata/virtuale (AR/VR): stereoscopia a doppia fotocamera e assistenza alla profondità 3D grazie alla misurazione della distanza multizonale
- Edifici e illuminazione smart (rilevamento dell'utente per attivazione dei dispositivi)
- IoT (rilevamento di utenti e oggetti)
- Tracking video focus: il rilevamento a 60 Hz ottimizza l'algoritmo di messa a fuoco continua

Caratteristiche

- Sensore di distanza multizonale veloce e accurato
 - Uscita di misurazione della distanza con 4x4 o 8x8 zone separate
 - Modalità a basso consumo autonomo con soglia di interruzione programmabile per riattivare l'host
 - Misurazione della distanza fino a 400 cm
 - Rilevamento multi-target e misurazione della distanza in ogni zona
 - Frequenza di aggiornamento fino a 60 Hz
 - Elaborazione con algoritmo Istogramma per ridurre o eliminare l'impatto del cross talk del vetro di copertura
 - Indicatore di movimento per ogni zona per rilevare il movimento degli oggetti e la loro direzione
- Modulo miniaturizzato completamente integrato con ampio campo visivo (FoV)
 - Emettitore: laser VCSEL a luce invisibile da 940 nm con driver analogico integrato
 - FoV diagonale quadrato di 63° con elementi ottici diffrattivi (DOE) su trasmettitore e ricevitore
 - Array di foto rilevatori SPAD
 - Microcontrollore a basso consumo con firmware integrato
 - Dimensioni: 6,4 × 3,0 × 1,5 mm
- Facile integrazione

- Singolo componente reflowable
- Opzioni di alimentazione flessibili: 3,3 V o 2,8 V singoli, oppure combinazione di 3,3 V o 2,8 V AVDD con 1,8 V IOVDD
- Compatibile con una vasta gamma di materiali per vetro di copertura (6)

Risoluzione multi-zona

L'uso di un DOE sopra il laser VCSEL consente di proiettare un campo visivo quadrato di $45^\circ \times 45^\circ$ (63° diagonale) sulla scena. La riflessione della luce viene focalizzata dalla lente del ricevitore su una matrice SPAD. Il VL53L5CX può eseguire misurazioni su 8x8 zone a 15 Hz per una maggiore risoluzione o su 4x4 zone a 60 Hz per una misurazione più veloce. Inoltre, grazie agli algoritmi brevettati ST Histogram, il VL53L5CX è in grado di rilevare più oggetti all'interno del FoV e di fornire informazioni sulla distanza fino a 60 Hz. (7)

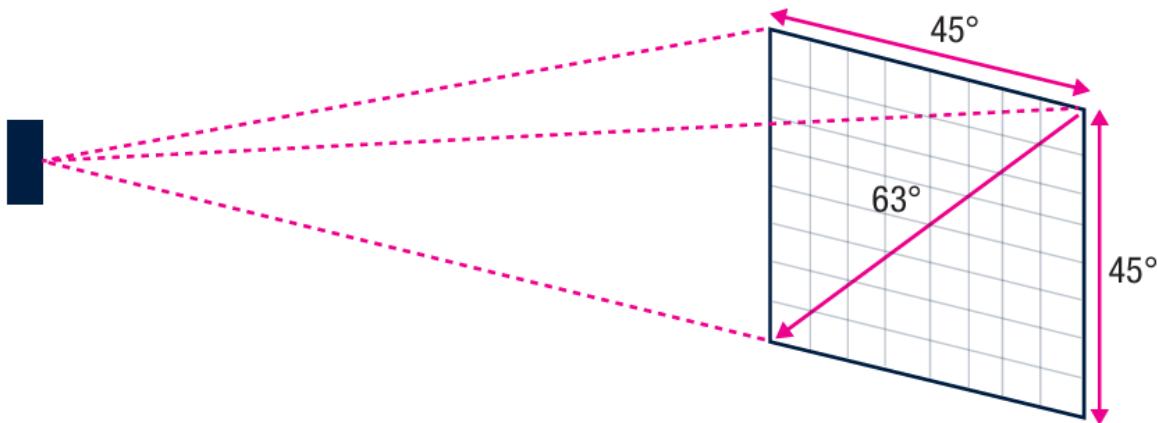


Figura 3. Risoluzione multi-zona del VL53L5CX

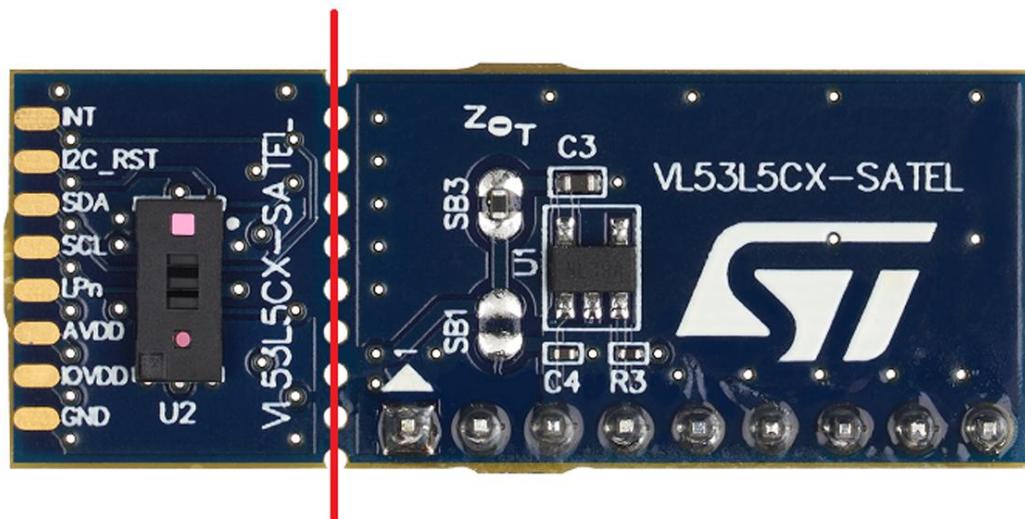


Figura 4. Scheda breakout VL53L5CX-SATEL - Sensore Time-of-Flight 8x8

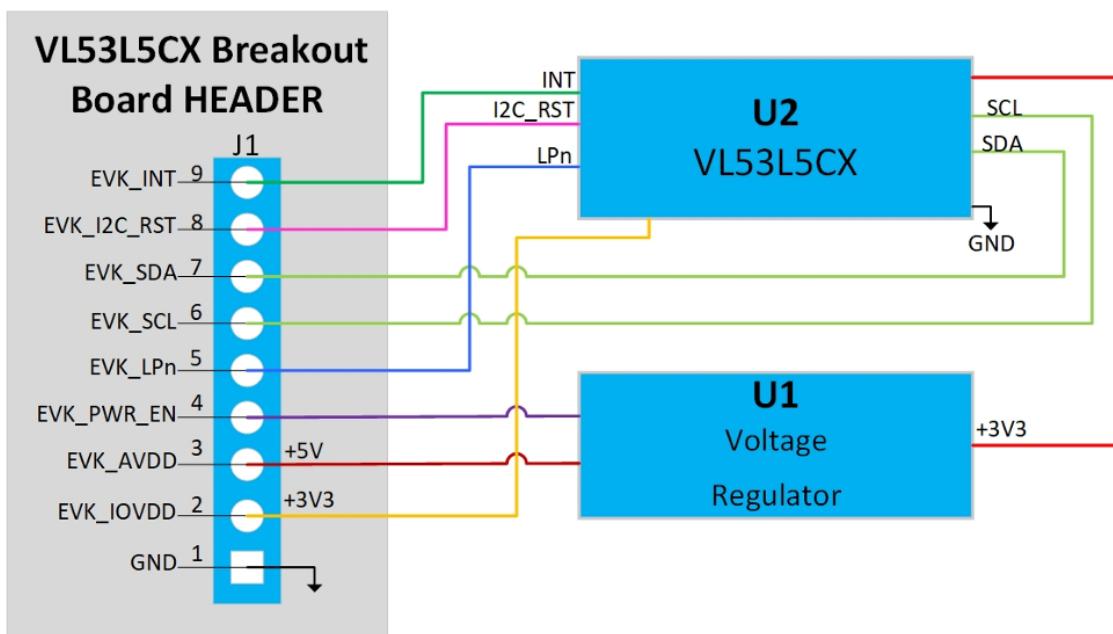


Figura 5. Schemi del VL53L5CX-SATEL

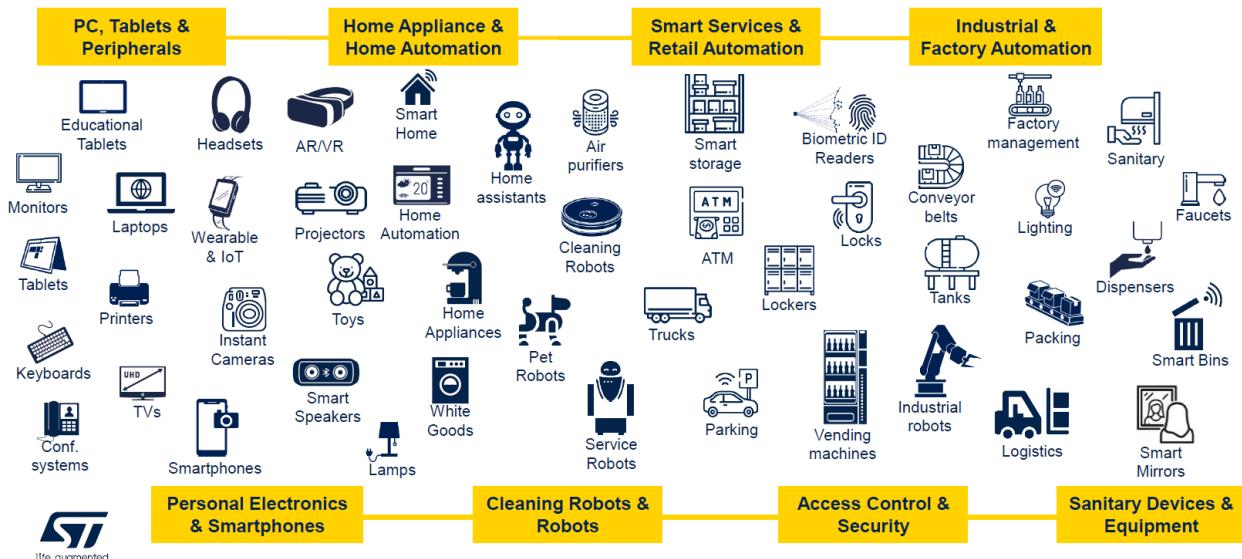


Figura 6. Applicazioni ovunque del sensore ToF

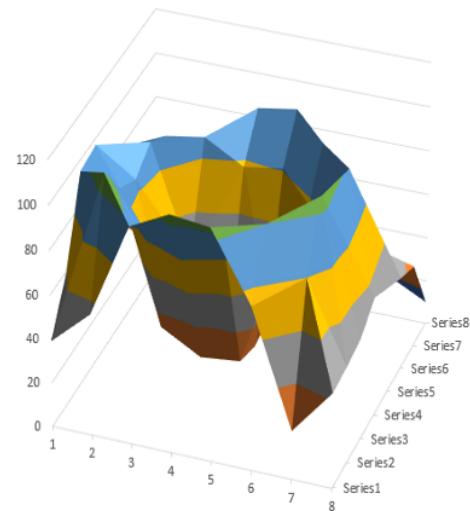


Figura 7. Misurazione del livello dei liquidi in tempo reale (7)

I fattori che influenzano il volume di rilevamento del sensore ToF

- Tutto ciò che influisce sul rapporto segnale-rumore può modificare il volume di rilevamento. I principali fattori sono:

- La riflettanza del bersaglio: una maggiore riflettanza aumenta il volume di rilevamento, poiché il bersaglio viene rilevato anche con una piccola sovrapposizione tra il bersaglio e il campo visivo del sistema ToF.
- Il livello di luce ambientale: un livello elevato di luce ambientale riduce il volume di rilevamento, poiché il rumore ambientale interferisce con il sensore.
- Il tempo di integrazione o timing budget: le impostazioni del sensore definiscono il tempo utilizzato per integrare la luce. Un valore più alto comporta un aumento del volume di rilevamento.
- La distanza del bersaglio: il volume di rilevamento varia in base alla quantità di segnale ricevuto. (7)

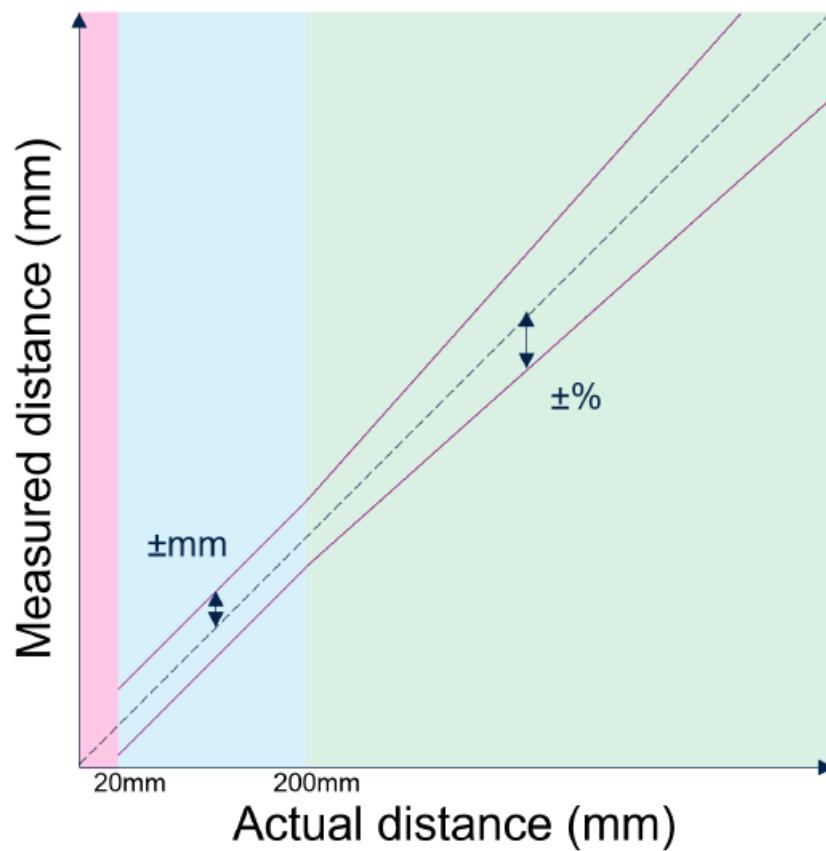


Figura 8. Accuratezza della misurazione in funzione della distanza

2.1.2. Microcontrollore Nucleo-F446RET6

La scheda STM32 Nucleo-64 offre un modo economico e flessibile per sperimentare nuovi concetti e sviluppare prototipi, consentendo di scegliere tra diverse combinazioni di prestazioni e consumo energetico fornite dal microcontrollore STM32.

Il supporto alla connettività ARDUINO Uno V3 e i connettori ST Morpho permettono di espandere facilmente le funzionalità della piattaforma di sviluppo STM32 Nucleo con un'ampia gamma di Shield specializzati.

La scheda STM32 Nucleo-64 non richiede debugger esterni, poiché integra il debugger/programmatore ST-LINK. (8)

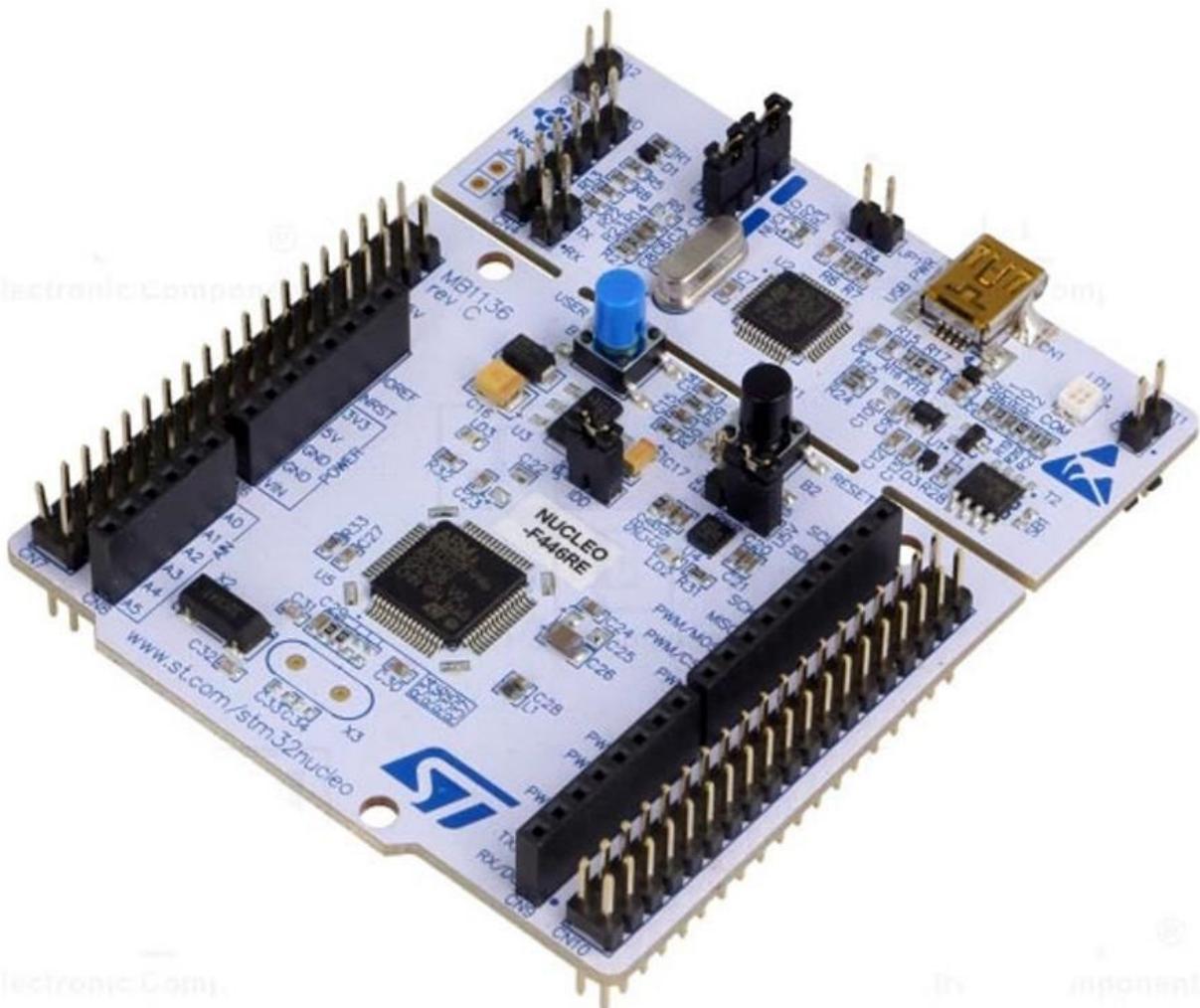


Figura 9. Microcontrollore NUCLEO-F446RE

Il chipset STM32F446

I dispositivi STM32F446xC/E sono basati sul core ad alte prestazioni Arm Cortex-M4 a 32 bit RISC, che opera a una frequenza fino a 180 MHz. Il core Cortex-M4 è dotato di un'unità a virgola mobile (FPU) a precisione singola, che supporta tutte le istruzioni e i tipi di dati a precisione singola di Arm. Inoltre, implementa un set completo di istruzioni DSP e un'unità di protezione della memoria (MPU) per migliorare la sicurezza delle applicazioni.

I dispositivi STM32F446xC/E integrano memorie ad alta velocità, tra cui una memoria Flash fino a 512 Kbyte, fino a 128 Kbyte di SRAM e 4 Kbyte di backup SRAM. Dispongono inoltre di un'ampia gamma di I/O avanzati e periferiche connesse a due bus APB, due bus AHB e una matrice multi-AHB a 32 bit.

Tutti i dispositivi offrono tre ADC a 12 bit, due DAC, un RTC a basso consumo, dodici timer generici a 16 bit (inclusi due timer PWM per il controllo motore) e due timer generici a 32 bit.

(9)

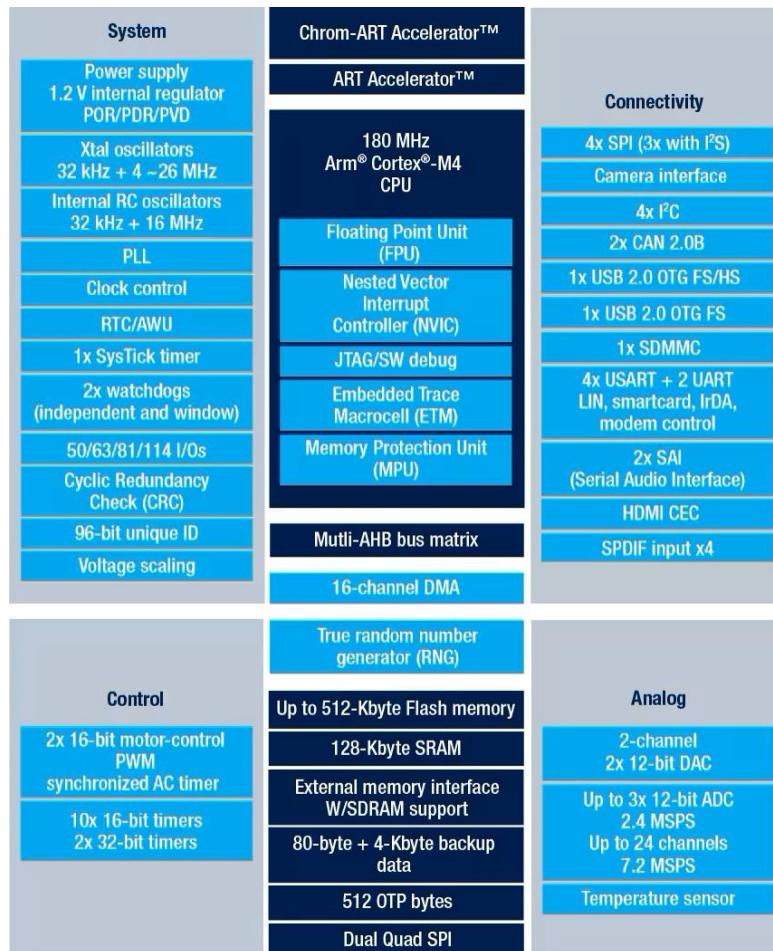


Figura 10. Diagramma del circuito dell'STM32F446xC/E (9)

La scheda STM32 Nucleo è suddivisa in due parti: la sezione ST-LINK e la sezione del microcontrollore STM32. La parte ST-LINK del PCB può essere rimossa per ridurre le dimensioni della scheda. In questo caso, la sezione rimanente del microcontrollore STM32

può essere alimentata solo tramite VIN, E5V e +3.3V sul connettore ST Morpho CN7, oppure tramite VIN e +3.3V sul connettore ARDUINO CN6.

È comunque possibile utilizzare la sezione ST-LINK per programmare il microcontrollore STM32 principale collegando i pin del connettore CN4 ai segnali SWD disponibili sul connettore ST Morpho (SWCLK pin 15 di CN7 e SWDIO pin 13 di CN7). (8)

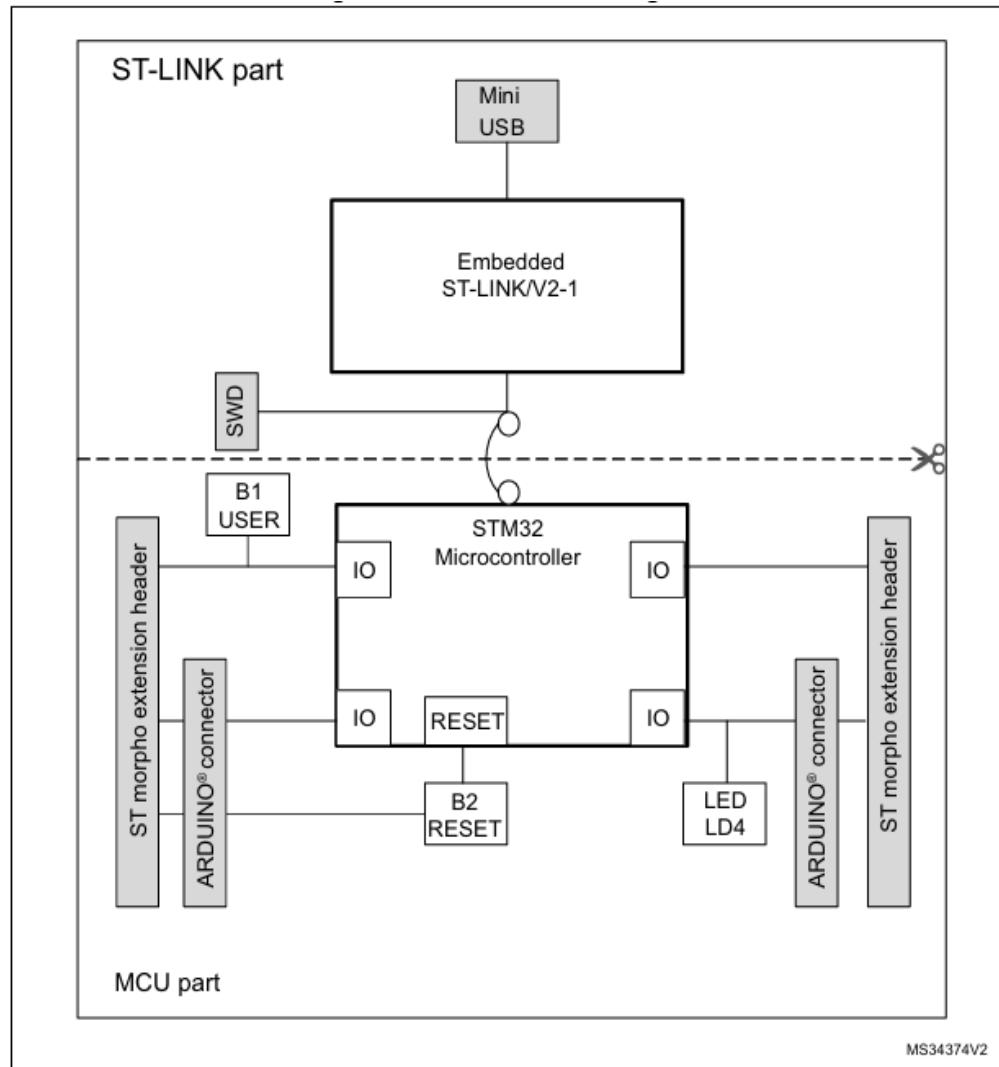


Figura 11. Diagramma a blocchi hardware STM32F446RET6 (8)

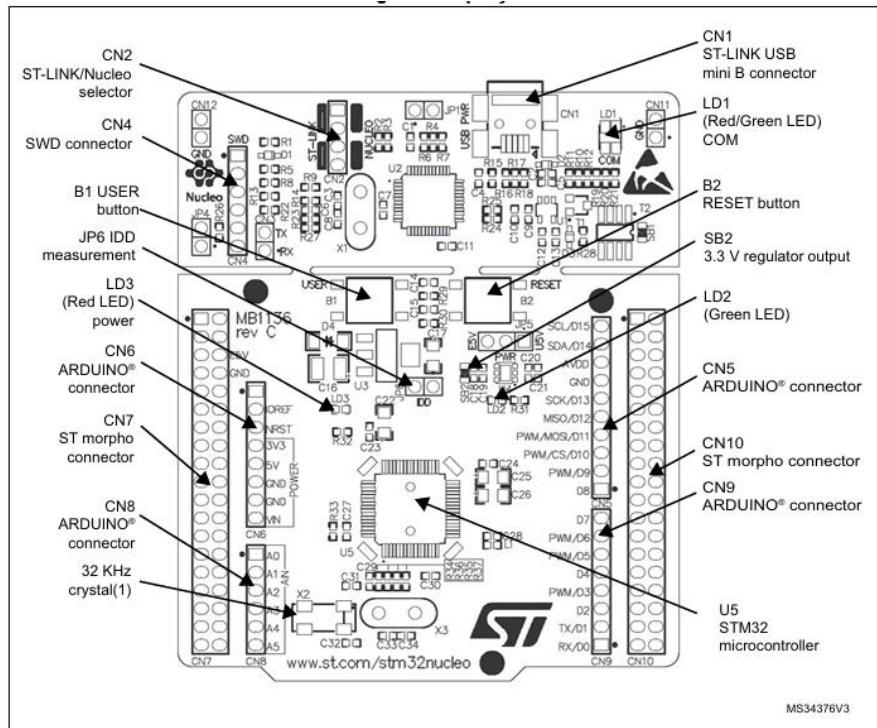


Figura 12. Disposizione superiore STM32F446RET6 (8)

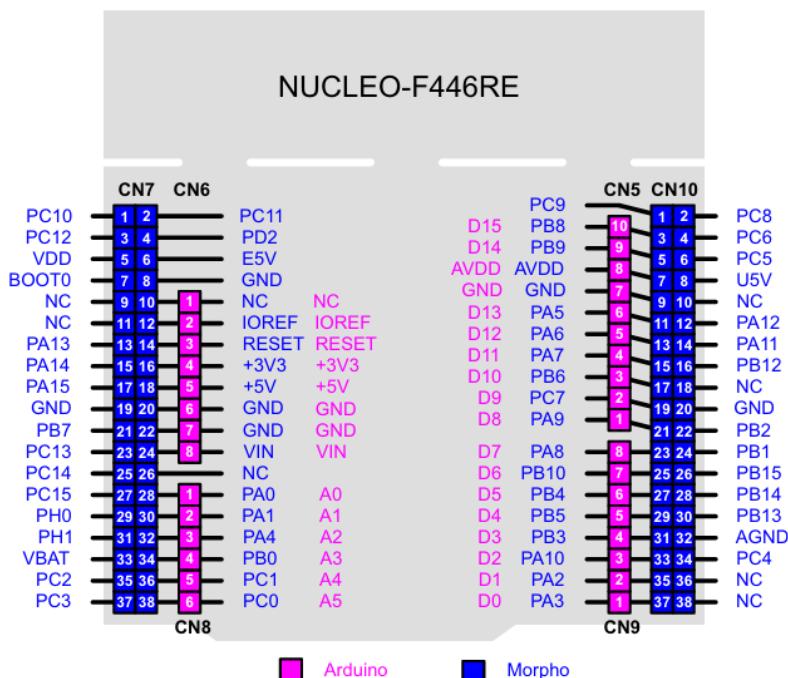


Figura 13. Connettori di espansione NUCLEO-F446RE (8)

2.1.3. Bipolar stepper motor driver IHM05A1

L'X-NUCLEO-IHM05A1 è una scheda di espansione per il controllo di motori Stepper bipolarì basata sul driver L6208. Offre una soluzione economica e facile da utilizzare per la gestione di motori Stepper bipolarì nei progetti con STM32 Nucleo.

L'X-NUCLEO-IHM05A1 è compatibile con il connettore Arduino UNO R3 e supporta l'aggiunta di altre schede di espansione STM32 utilizzando una singola scheda STM32 Nucleo. L'utente può inoltre montare il connettore ST Morpho. (10)

Caratteristiche:

- Intervallo di tensione: da 8 V a 50 V
- Corrente di fase: fino a 2,8 A r.m.s.
- Doppi controllori di corrente PWM indipendenti
- Selezione della modalità di decadimento rapido e lento
- Logica di decodifica per motori Stepper, con supporto per modalità full-step e half-step
- Protezione da sovraccorrente
- Spegnimento termico di protezione
- Compatibile con il connettore Arduino UNO R3
- Compatibile con le schede STM32 Nucleo
- Conforme alla normativa RoHS

Ambiti di applicazione:

- Automazione industriale
- Per motori Stepper bipolarì
- Micro-stepping gestito dal microcontrollore

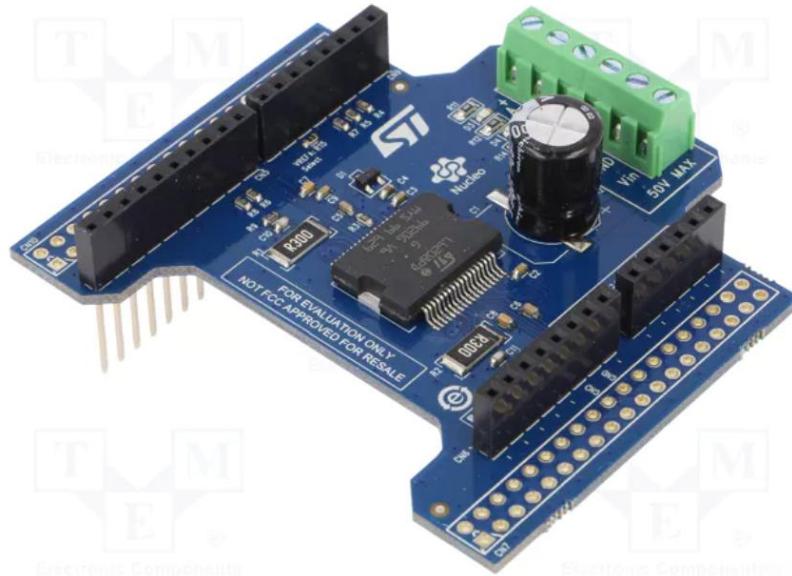


Figura 14. Scheda di espansione per il controllo motore X-NUCLEO-IHM05A1 (10)

L'offerta di driver per motori Stepper STSPIN di ST varia da circuiti integrati relativamente semplici, con controllo di corrente e generazione di fase, a soluzioni più complesse. Questi driver integrano in un singolo chip tutto il necessario per pilotare autonomamente un motore Stepper, utilizzando comandi di movimento di alto livello provenienti dal sistema di controllo del motore o del movimento, come un microcontrollore, un DSP o un FPGA.

Disponibili in un'ampia gamma di package compatti e con migliorata dissipazione termica, i driver per motori Stepper STSPIN offrono una soluzione pronta all'uso e ottimizzata per i sistemi di controllo motore e movimento, con un'ampia gamma di tensioni e correnti supportate.

2.1.4. Attuatori

I motori Stepper sono ampiamente utilizzati in applicazioni di mantenimento e posizionamento nei settori dell'informatica, della sicurezza, dell'automazione industriale e

dell'industria automobilistica. Il loro funzionamento dipende dal numero di fasi, dalla configurazione degli avvolgimenti e dal livello di fluidità del movimento richiesto.

Nei motori Stepper bipolarari, la corrente può fluire in entrambe le direzioni, rendendo necessario l'uso di un convertitore a ponte intero per pilotare ciascuno dei due avvolgimenti di un motore a due fasi. Durante il movimento, il tipo di controllo elettronico adottato (full step, half step, micro-stepping) e la forma d'onda della corrente di fase influenzano diversi parametri operativi, tra cui il livello di vibrazioni, il rumore acustico, la fluidità del movimento e la sensibilità alle risonanze. (10)

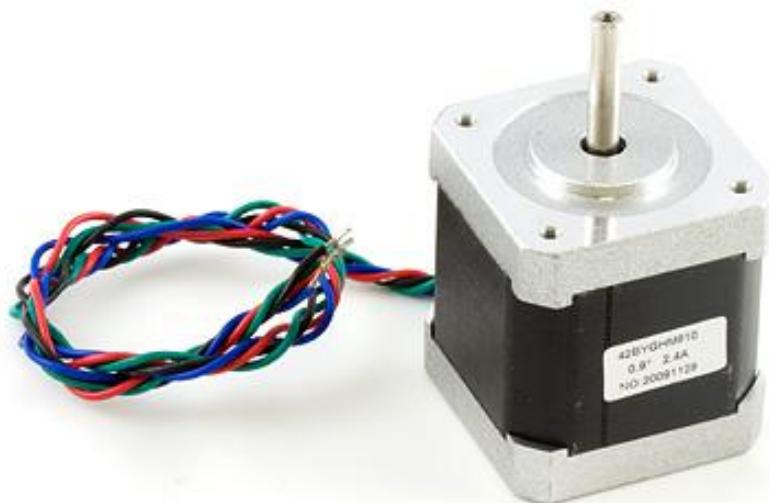


Figura 15. Stepper Motor

Un servomotore è un motore altamente specializzato progettato per il controllo preciso del movimento rotatorio o lineare. Si tratta di un dispositivo che utilizza un meccanismo di retroazione per garantire un posizionamento accurato, basandosi su un segnale di controllo che guida il motore verso una posizione desiderata. Questo sistema consente una regolazione precisa del movimento, rendendo i servomotori essenziali nelle applicazioni che richiedono alta precisione e fluidità nel posizionamento.



Figura 16. Servo motor

2.1.5. Modulo wi-fi ESP8266-01s

Il modulo ESP-01S, basato sul microcontrollore ESP8266, è un ricetrasmettitore wireless Wi-Fi progettato per applicazioni nell'ambito dell'internet of things (IoT). Grazie alle sue caratteristiche di compattezza, basso consumo energetico e capacità di connessione diretta a microcontrollori, trova impiego in diversi contesti, tra cui l'automazione domestica, il monitoraggio remoto e i sistemi di controllo wireless.

Il modulo utilizza il protocollo Wi-Fi IEEE 802.11 b/g/n ed è dotato di una porta seriale TTL a 3,3V per la comunicazione con microcontrollori esterni. La memoria Flash integrata da 1 MB consente l'archiviazione del firmware e dei dati necessari per l'elaborazione locale. Le dimensioni ridotte (25 mm × 14 mm × 4 mm) lo rendono adatto a sistemi compatti e portatili.

Dal punto di vista hardware, il modulo ESP-01S è basato sul processore ESP8266, il quale integra un'unità di calcolo Tensilica L106 a 32 bit a bassissimo consumo energetico, con supporto alla modalità a 16 bit. La frequenza operativa può raggiungere gli 80 MHz o i 160 MHz, consentendo l'esecuzione di sistemi operativi in tempo reale (RTOS). Il modulo incorpora inoltre un'interfaccia Wi-Fi completa, comprensiva di unità MAC, RF, PA, LNA e un'antenna integrata, riducendo la necessità di componenti esterni per la gestione del segnale wireless.

L'ESP8266 è un sistema altamente integrato (SoC) che offre la possibilità di incorporare funzionalità Wi-Fi in altri dispositivi o di operare come modulo indipendente con costi contenuti e minime esigenze di spazio. Il microcontrollore può eseguire applicazioni in modo autonomo, avviandosi direttamente dalla memoria Flash esterna, o può essere utilizzato come co-processore per la gestione delle funzionalità di rete Wi-Fi in sistemi più complessi. Inoltre, la cache integrata migliora le prestazioni nelle applicazioni che richiedono una risposta rapida.

Grazie alla sua versatilità, il modulo ESP-01S trova applicazione in diversi settori, tra cui l'automazione domestica, il monitoraggio ambientale, l'interfacciamento con dispositivi mobili per il controllo remoto e i sistemi di notifica in tempo reale. Il suo basso consumo energetico e la compatibilità con numerose librerie software ne facilitano l'integrazione in progetti di prototipazione e sviluppo di sistemi IoT. (11)

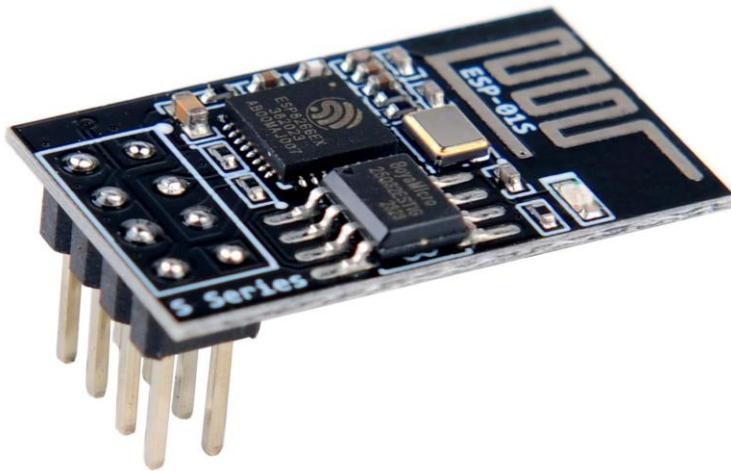


Figura 17. Modulo Wi-Fi ESP-01S

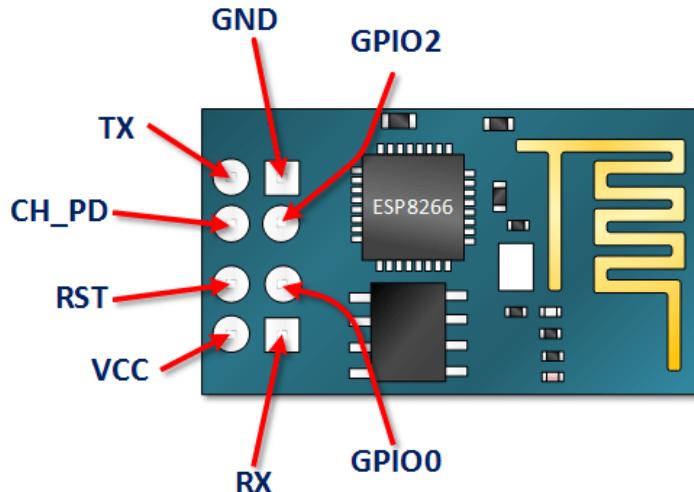


Figura 18. Schema dei pin dell'ESP8266-01S

2.1.6. Modulo NRF per comunicazione basso raggio

Il NRF24L01 è un chip di comunicazione wireless sviluppato da Nordic Semiconductor. Questo modulo opera nella banda di frequenza compresa tra 2,4 GHz e 2,5 GHz, che è una banda di libero utilizzo. Al contrario di altre bande, che richiedono una licenza per la trasmissione, questa può essere utilizzata senza restrizioni. Il chip integra sia un trasmettitore che un ricevitore, consentendo una comunicazione bidirezionale.

La gamma di frequenza disponibile va da 2.400 MHz a 2.525 MHz, con la possibilità di selezionare tra 125 canali distinti, distanziati di 1 MHz l'uno dall'altro. Tuttavia, l'utilizzo della banda a 2,4 GHz può causare interferenze con altri dispositivi che operano sulla stessa frequenza, come reti Wi-Fi o droni. Per ridurre il rischio di interferenze, è consigliabile selezionare frequenze superiori a 2.501 MHz.

Dal punto di vista tecnico, il modulo NRF24L01 funziona con una tensione di alimentazione compresa tra 1,9 V e 3,6 V, rendendolo facilmente integrabile con microcontrollori come Arduino, sfruttando la connessione a 3,3 V. Può essere alimentato anche tramite batterie o

alimentatori compatibili. Inoltre, la velocità di trasmissione è configurabile tra 250 Kbps, 1 Mbps e fino a 2 Mbps, offrendo flessibilità in base alle esigenze di comunicazione e alla distanza operativa richiesta.

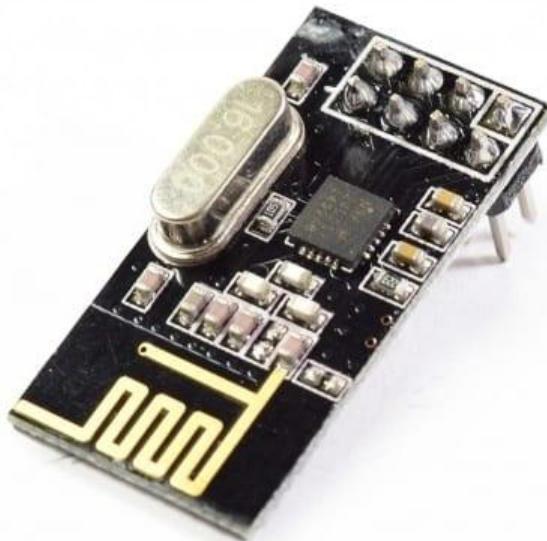


Figura 19. Modulo NRF24L01

Il chip NRF24L01 può gestire contemporaneamente fino a sei connessioni con diversi dispositivi, consentendo la trasmissione e la ricezione dei dati da più punti senza difficoltà. Inoltre, il modulo è dotato di circuiti logici per la correzione degli errori e il ritrasferimento dei dati in caso di necessità, alleggerendo così il carico di elaborazione del microcontrollore.

Il controllo del modulo avviene tramite il bus SPI, rendendolo facilmente compatibile con microcontrollori come Arduino. I pin dati dell'NRF24L01 supportano tensioni fino a 5V senza problemi. Il consumo energetico in modalità standby è molto basso, mentre in fase operativa il modulo richiede solo 15mA per l'invio e la ricezione dei dati, rendendolo un'opzione efficiente dal punto di vista energetico.

L'NRF24L01 è un chip economico, adatto a una vasta gamma di progetti. Il modello senza antenna esterna è acquistabile a un prezzo molto contenuto, mentre la versione con antenna

esterna, pur avendo un costo leggermente superiore, rimane comunque accessibile e solitamente non supera 1,7 €.

Per stabilire una comunicazione wireless è necessario acquistare due moduli NRF24L01, uno per il lato trasmettitore e l'altro per il lato ricevente. Entrambi i moduli possono essere configurati per operare sia come trasmettitori che come ricevitori, a seconda delle esigenze del progetto. (12)

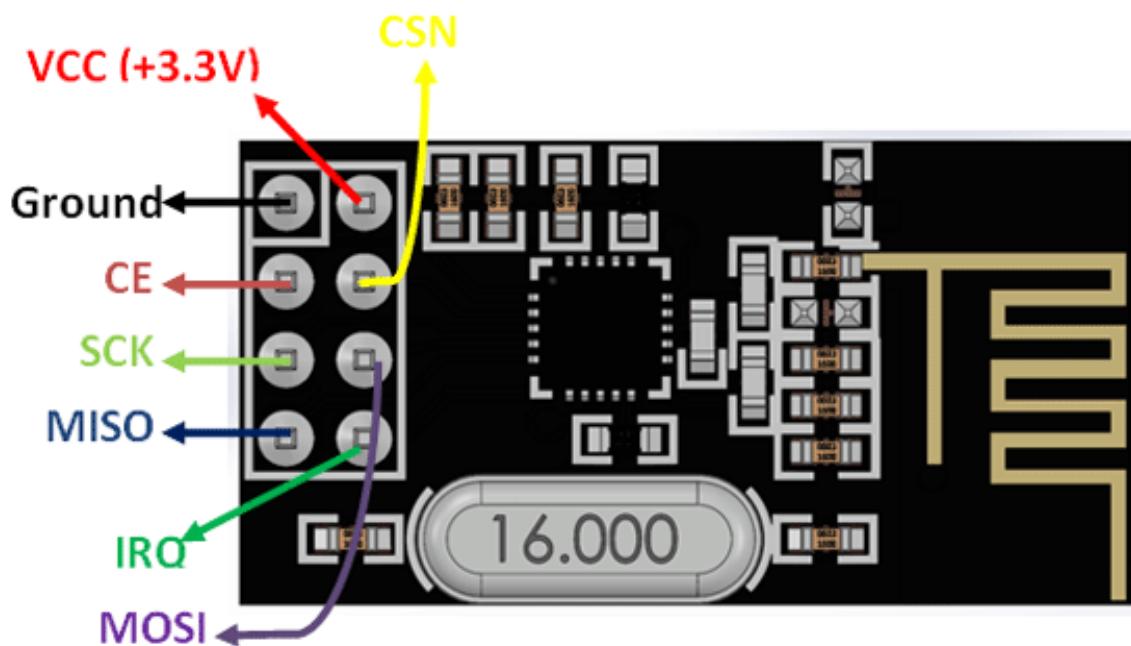


Figura 20. Schema dei pin del modulo NRF24L01

2.1.7. Sistema di controllo degli attuatori e logging locale

Il sistema di controllo remoto è composto da alcuni potenziometri, pulsanti e LED, che consentono di controllare entrambi i motori modificandone lo stato.

L'intensità della luce del LED posizionato vicino ai potenziometri indica il valore del potenziometro, mentre il LED situato vicino al modulo NRF segnala una connessione stabile con il microcontrollore STM32, verificata la connessione tramite la ricezione dell'ACK dopo l'invio del pacchetto di dati.

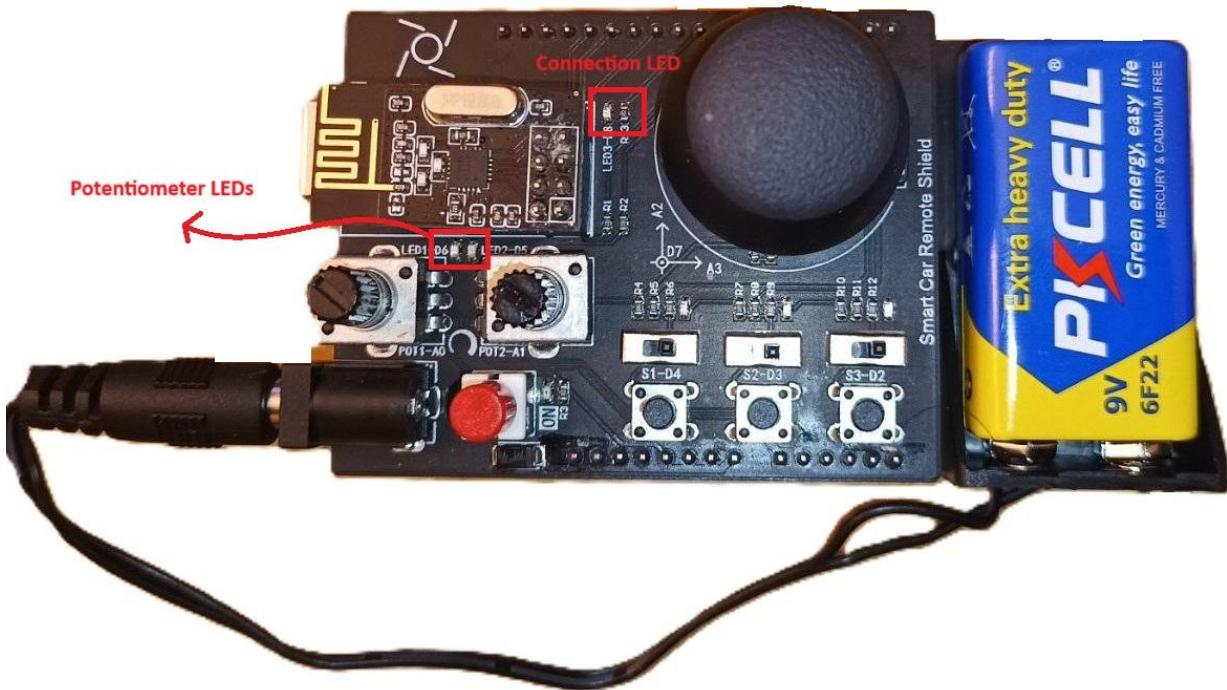


Figura 21. Sistema di controllo remoto

Attraverso il logger locale è possibile ricevere i log del programma tramite UART. Collegando il logger alla porta USB di un dispositivo, i log possono essere visualizzati in un terminale seriale.

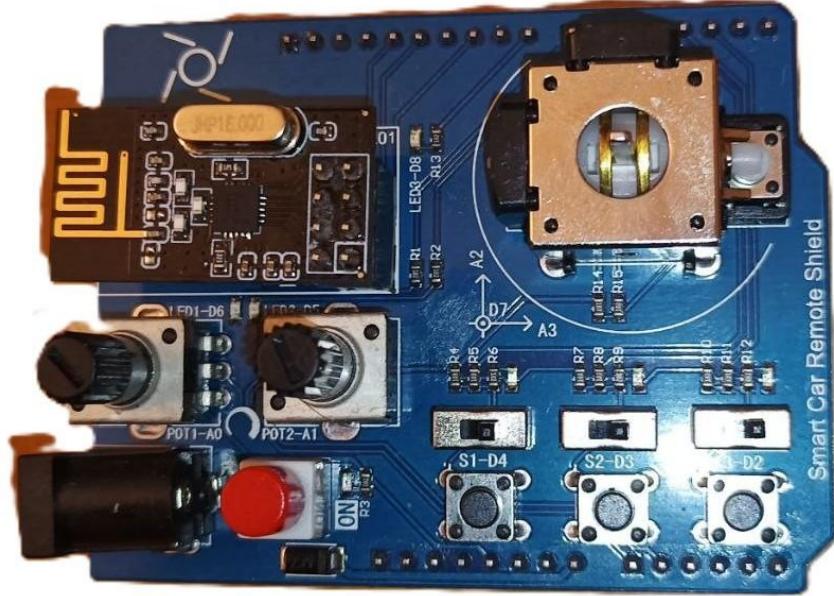


Figura 22. Sistema di logger locale

L'Arduino Uno è un microcontrollore utilizzato sia nel sistema di controllo remoto che nel logger locale. Si tratta di una scheda basata sul microcontrollore ATmega328P, dotata di 14 pin di input/output digitali (di cui 6 utilizzabili come uscite PWM), 6 ingressi analogici, un risonatore ceramico da 16 MHz (CSTCE16M0V53-R0), una porta USB, un connettore di alimentazione, un'intestazione ICSP e un pulsante di reset.

La scheda contiene tutto il necessario per il funzionamento del microcontrollore ed è sufficiente collegarla a un computer tramite cavo USB o alimentarla con un adattatore AC-DC o una batteria per iniziare a utilizzarla. Inoltre, Arduino Uno è progettato per essere facilmente riparabile: in caso di malfunzionamento, è possibile sostituire il chip ATmega328P con un costo contenuto e riprendere il lavoro senza difficoltà.

Il nome "Uno", che in italiano significa "uno", è stato scelto per marcare il rilascio della versione 1.0 del software Arduino (IDE), che ha rappresentato un punto di riferimento per l'intera piattaforma. Arduino Uno è la prima scheda di una serie di schede USB Arduino e rappresenta il modello di riferimento per la piattaforma Arduino. Per un elenco completo delle schede attuali, passate o obsolete, è possibile consultare l'indice delle schede Arduino.
(13)

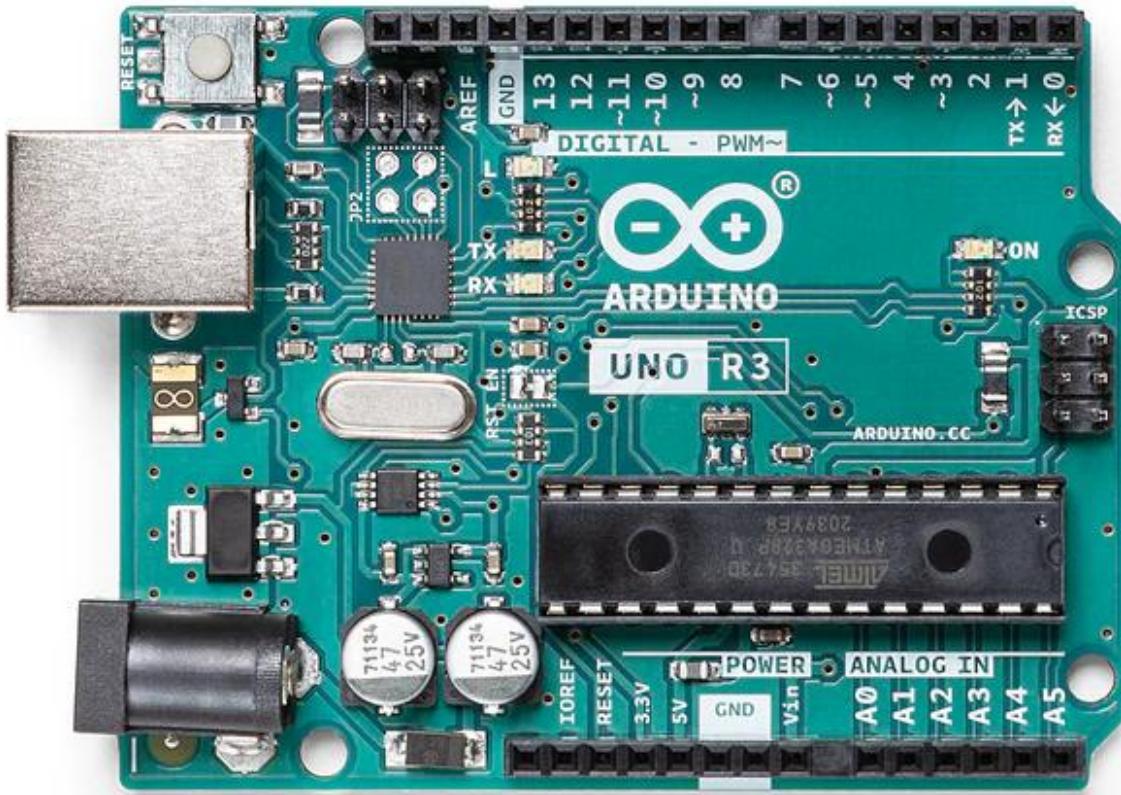


Figura 23. Arduino Uno REV3 Microcontrollore

2.2. Architettura software

2.2.1. Ambiente di Sviluppo e Strumenti Utilizzati

STM32CubeIDE è uno strumento di sviluppo multi-OS, all-in-one, parte dell'ecosistema software STM32Cube.

STM32CubeIDE è una piattaforma avanzata per lo sviluppo in C/C++ che include configurazione delle periferiche, generazione del codice, compilazione e funzionalità di debug per microcontrollori e microprocessori STM32. È basato sul framework Eclipse®/CDT™.

e sulla toolchain GCC per lo sviluppo, mentre per il debugging utilizza GDB. Inoltre, supporta l'integrazione di numerosi plugin già esistenti per arricchire le funzionalità dell'IDE Eclipse®.

STM32CubeIDE integra le funzionalità di configurazione STM32 e creazione di progetti offerte da STM32CubeMX, consentendo un'esperienza di sviluppo unificata e riducendo i tempi di installazione e sviluppo. Dopo la selezione di un microcontrollore o microprocessore STM32 vuoto, preconfigurato o basato su una scheda o su un progetto di esempio, il progetto viene creato e il codice di inizializzazione generato. Durante lo sviluppo, l'utente può tornare alla configurazione delle periferiche o dei middleware e rigenerare il codice di inizializzazione senza influenzare il codice utente.

STM32CubeIDE include strumenti di analisi della compilazione e della memoria che forniscono informazioni utili sullo stato del progetto e sui requisiti di memoria.

Inoltre, STM32CubeIDE offre funzionalità di debug standard e avanzate, tra cui la visualizzazione dei registri della CPU, delle memorie e dei registri delle periferiche, il monitoraggio in tempo reale delle variabili, l'interfaccia Serial Wire Viewer e l'analisi degli errori di sistema.

Funzionalità principali:

- Integrazione con STM32CubeMX: per la selezione di microcontrollori, microprocessori, piattaforme di sviluppo e progetti di esempio, configurazione di pinout, clock, periferiche e middleware, creazione del progetto e generazione del codice di inizializzazione.
- Basato su Eclipse®/CDT™: con supporto per add-on di Eclipse®, toolchain GNU C/C++ per Arm® e debugger GDB.
- Supporto alla serie STM32MP1: per progetti su OpenST Linux e compatibilità con il sistema operativo Linux.
- Strumenti avanzati di debug: inclusi la visualizzazione dei registri della CPU, delle periferiche e della memoria, il monitoraggio delle variabili in tempo reale, il tracing del sistema (SWV) e l'analisi degli errori della CPU.
- Supporto al debugging RTOS-aware, compatibile con Azure RTOS.

- Compatibilità con strumenti di debug: ST-LINK (STMicroelectronics) e J-Link (SEGGER).
- Importazione di progetti: da Atollic® TrueSTUDIO® e AC6 System Workbench for STM32 (SW4STM32).
- Supporto multi-OS: compatibile con Windows®, Linux® e MacOS®, esclusivamente nelle versioni a 64 bit. (14)

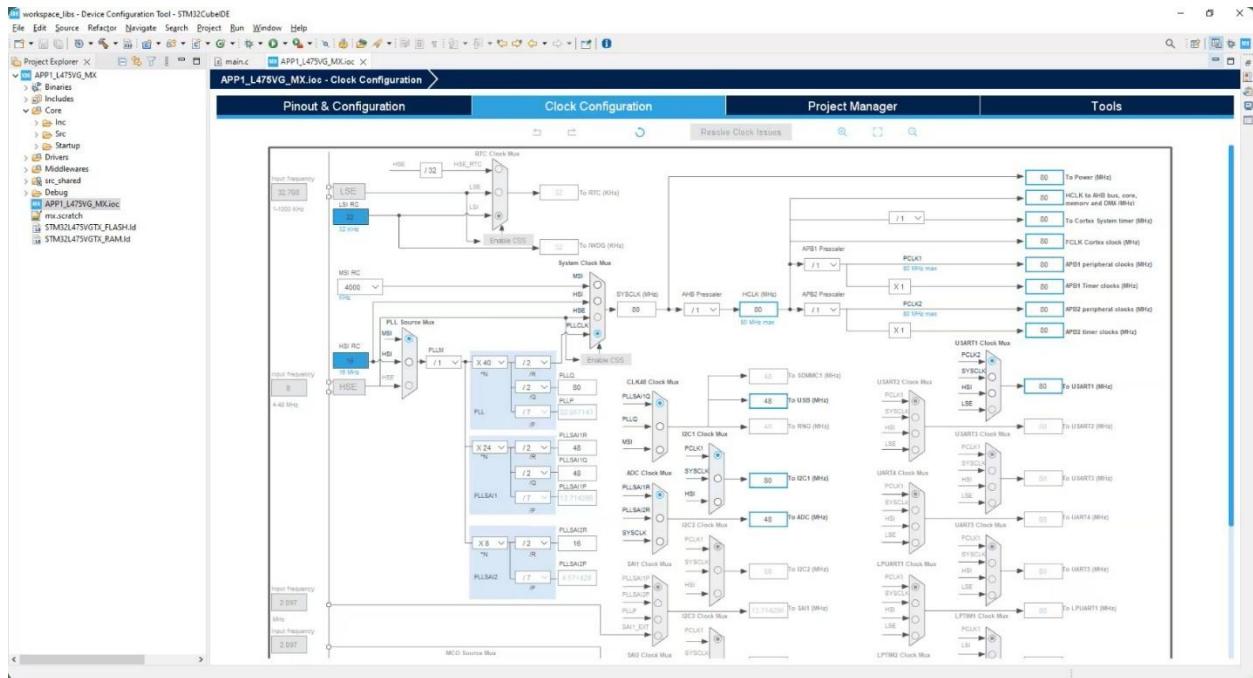


Figura 24. Ambiente di sviluppo integrato STM32CubeIDE (14)

PuTTY è un emulatore di terminale, una console seriale e un'applicazione per il trasferimento di file di rete gratuito e open-source. Supporta diversi protocolli di rete, tra cui SCP, SSH, Telnet, login e connessioni raw socket. Inoltre, consente la connessione a una porta seriale. Il nome "PuTTY" non ha un significato ufficiale.

Originariamente sviluppato per Microsoft Windows, PuTTY è stato portato su diversi altri sistemi operativi. Sono disponibili versioni ufficiali per alcune piattaforme Unix-like, mentre sono in corso lavori per il porting su Classic Mac OS e MacOS. Inoltre, esistono versioni non ufficiali per piattaforme come Symbian, Windows Mobile e Windows Phone.

PUTTY è stato scritto ed è principalmente mantenuto da Simon Tatham, un programmatore britannico. (15)

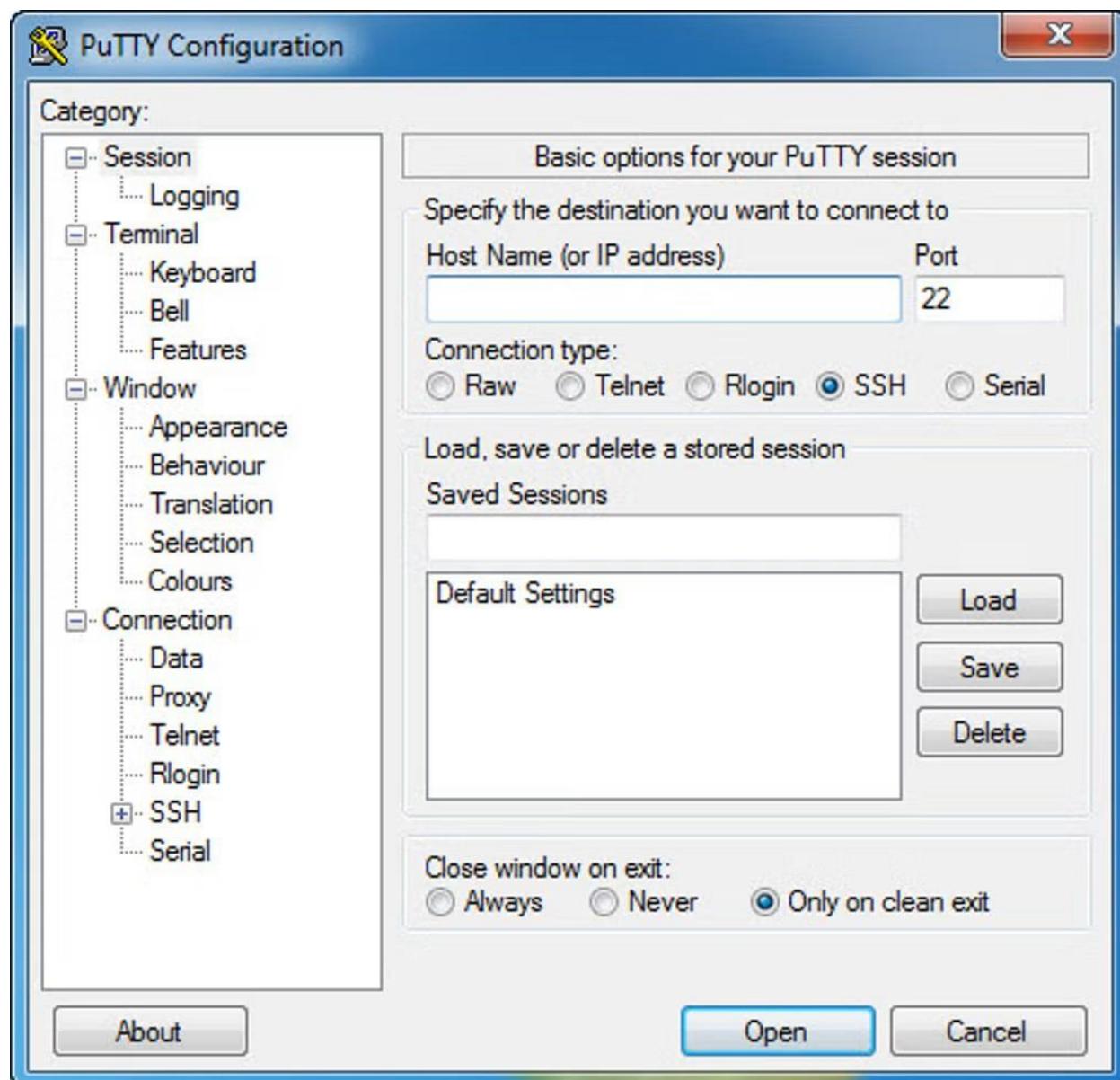


Figura 25. Ambiente software PuTTY (15)

MQTTX semplifica e velocizza lo sviluppo e il test di applicazioni MQTT.

Offre tre versioni principali: Desktop, CLI e Web, progettate per adattarsi a diversi ambienti di lavoro e garantire compatibilità multiplattaforma su MacOS, Linux e Windows.

Oltre alle operazioni MQTT di base, MQTTX include funzionalità avanzate come scripting, benchmarking, simulazione di dati IoT e MQTTX Copilot basato su intelligenza artificiale, fornendo una soluzione completa per MQTT.

MQTTX è completamente conforme agli standard MQTT 5.0, 3.1.1 e 3.1, mantenendosi costantemente aggiornato con le ultime specifiche del protocollo per garantire la piena compatibilità delle funzionalità di test. (16)

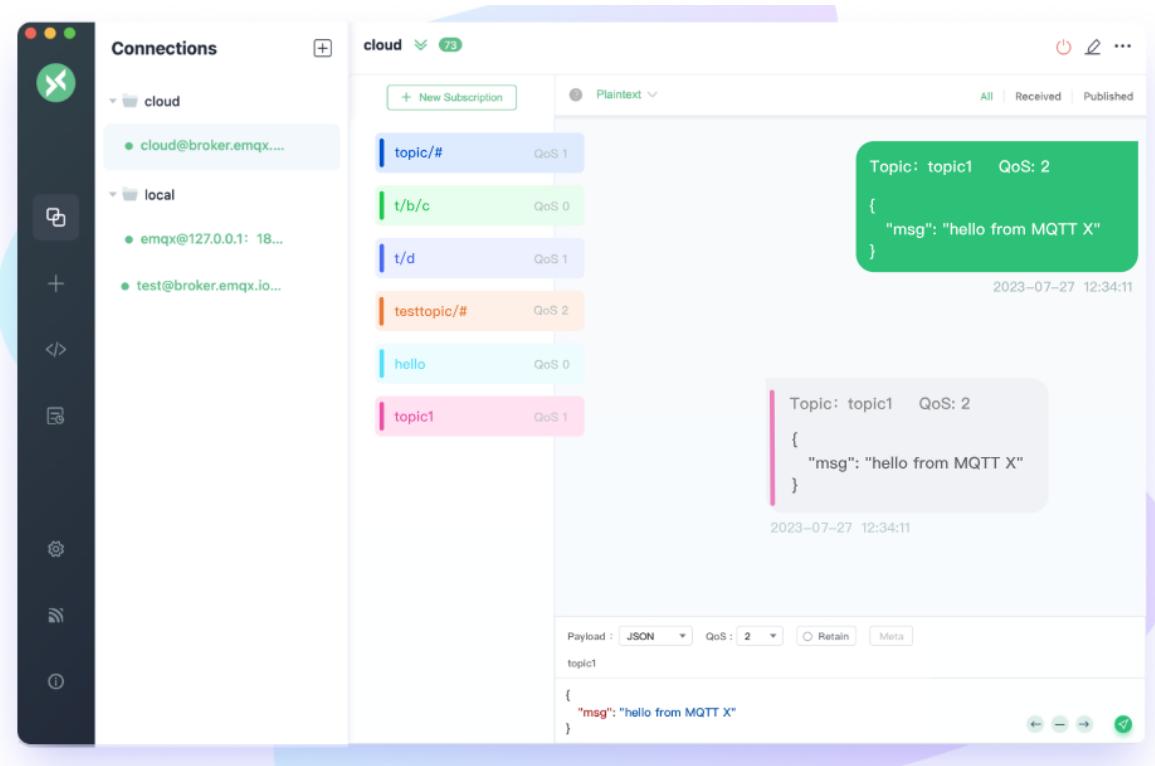


Figura 26. Ambiente software MQTTX

2.2.2. Algoritmo di Elaborazione Dati Embedded Nel Sensore TOF

L5 è un sensore di profondità leggero basato sulla tecnologia Time-of-Flight (ToF). Nei sensori ToF convenzionali, l'output ha tipicamente una risoluzione superiore a dieci mila

pixel e misura la distanza per pixel lungo il raggio che va dal centro ottico alle superfici osservate.

Al contrario, L5 fornisce più distribuzioni di profondità con una risoluzione estremamente bassa di 8×8 zone, coprendo un campo visivo diagonale totale di 63° . La distribuzione viene inizialmente misurata contando il numero di fotoni restituiti in ciascun intervallo di tempo discretizzato e successivamente adattata a una distribuzione gaussiana. Questo approccio consente di ridurre il carico di banda e il consumo energetico, poiché vengono trasmessi solo il valore medio e la varianza.

A causa della sua bassa risoluzione e dell'elevata incertezza nelle misurazioni, L5 non può essere utilizzato direttamente per la stima densa della profondità in ambienti interni.

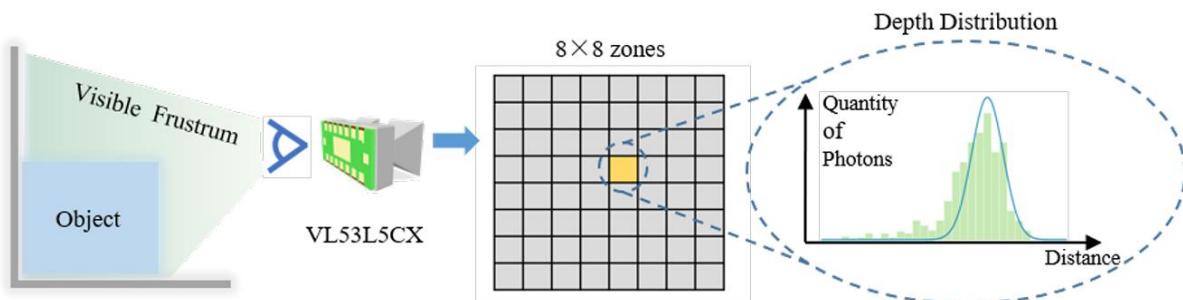


Figura 27. L5 ha una risoluzione estremamente bassa di 8×8 zone e fornisce una distribuzione della profondità per ciascuna zona.

2.2.3. Protocollo di Comunicazione MQTT

MQTT è un protocollo di messaggistica basato su standard, utilizzato per la comunicazione tra dispositivi. I sensori intelligenti, i dispositivi indossabili e altri dispositivi dell'Internet delle Cose (IoT) devono trasmettere e ricevere dati su reti con risorse e larghezza di banda limitate. Questi dispositivi utilizzano MQTT per la trasmissione dei dati, poiché è facile da implementare e consente una comunicazione efficiente. MQTT supporta la messaggistica tra dispositivo e cloud e viceversa.

Il protocollo MQTT è diventato uno standard per la trasmissione dei dati IoT grazie ai seguenti **vantaggi**:

- Leggerezza ed efficienza

L'implementazione di MQTT su un dispositivo IoT richiede risorse minime, permettendone l'uso anche su microcontrollori di piccole dimensioni. Ad esempio, un messaggio di controllo MQTT minimo può occupare solo due byte di dati. Inoltre, le intestazioni dei messaggi MQTT sono compatte, consentendo un'ottimizzazione della larghezza di banda della rete.

- Scalabilità

L'implementazione di MQTT richiede poco codice e un basso consumo energetico. Il protocollo include funzionalità integrate per supportare la comunicazione con un grande numero di dispositivi IoT, rendendolo adatto alla connessione di milioni di dispositivi.

- Affidabilità

Molti dispositivi IoT operano su reti cellulari instabili, con larghezza di banda ridotta e alta latenza. MQTT dispone di meccanismi integrati per ridurre il tempo di riconnessione al cloud. Inoltre, definisce tre livelli di qualità del servizio per garantire l'affidabilità della comunicazione: al massimo una volta (0), almeno una volta (1) ed esattamente una volta (2).

- Sicurezza

MQTT semplifica la crittografia dei messaggi e l'autenticazione dei dispositivi e degli utenti, supportando protocolli di sicurezza moderni come OAuth, TLS 1.3 e certificati gestiti dal cliente.

- Supporto

L'implementazione di MQTT è ampiamente supportata in diversi linguaggi di programmazione, come Python. Ciò consente agli sviluppatori di adottarlo rapidamente con una quantità minima di codice in qualsiasi tipo di applicazione.

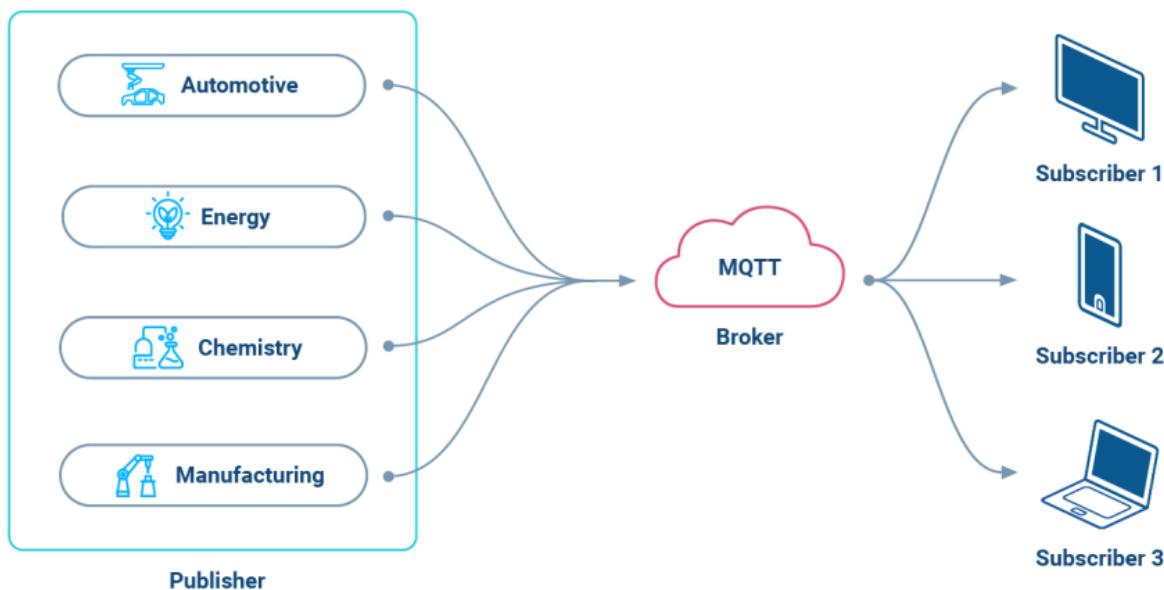


Figura 28. Architettura Publisher-Subscriber di MQTT

MQTT implementa il modello pubblica/sottoscrivi definendo i client e il broker come segue.

- Client MQTT

Un client MQTT è qualsiasi dispositivo, da un server a un microcontrollore, che esegue una libreria MQTT. Se il client invia messaggi, assume il ruolo di publisher, mentre se riceve messaggi, agisce come subscriber. In generale, qualsiasi dispositivo che utilizza MQTT per la comunicazione su una rete può essere considerato un client MQTT.

- Broker MQTT

Il broker MQTT è il sistema backend responsabile della gestione e del coordinamento dei messaggi tra i vari client. Le sue principali funzioni includono:

- Ricezione e filtraggio dei messaggi

- Identificazione dei client sottoscritti a ciascun messaggio
- Invio dei messaggi ai client corrispondenti
- Autorizzazione e autenticazione dei client MQTT
- Inoltro dei messaggi ad altri sistemi per ulteriori analisi
- Gestione dei messaggi persi e delle sessioni dei client
- Connessione MQTT

La comunicazione tra client e broker avviene tramite una connessione MQTT. Il processo inizia quando un client invia un messaggio CONNECT al broker. Il broker risponde con un messaggio CONNACK, confermando l'avvenuta connessione. Per poter comunicare, sia il client MQTT che il broker necessitano di uno stack TCP/IP. È importante notare che i client non si connettono mai direttamente tra loro, ma solo al broker.

2.3. Progettazione Meccanica

2.3.1. Struttura del Robot e Posizionamento dei Componenti

Il robot è stato creato su un telaio in legno, su cui sono stati montati il microcontrollore STM32 e la scheda di espansione. Successivamente, sono stati collegati il modulo NRF, il modulo Wi-Fi, il motore Stepper e il servo motore. La loro posizione è stata scelta molto vicino al microcontrollore per evitare problemi legati alla distanza dei cablaggi.

Il ruolo principale dei componenti è il seguente:

il microcontrollore STM32 è il componente principale di questo sistema, da cui dipende l'intero funzionamento del progetto

la scheda di espansione è responsabile del movimento accurato del motore Stepper, oltre alla regolazione della tensione e alla gestione dell'alimentazione necessaria per il motore Stepper

il modulo NRF è utilizzato sia per il controllo remoto che per il data logger locale. È il componente chiave per la realizzazione della soluzione wireless, che consente di controllare sia il motore Stepper che il servo tramite il controllo remoto e di collegare il data logger locale al computer per visualizzare i log del programma principale su un terminale seriale

il modulo ESP-01S ha il compito di estendere la comunicazione oltre l'ambito locale e connettersi a Internet. Grazie alla modalità stazionaria, il modulo si comporta come un router in una rete domestica, permettendo di inviare dati a un broker MQTT e di riceverli su un altro dispositivo MQTT subscriber

Il sistema sviluppato ha due gradi di libertà, permettendo la rotazione orizzontale e l'inclinazione verticale del sensore ToF. Il primo motore, uno Stepper, è responsabile della rotazione dell'intero sistema lungo l'asse orizzontale, consentendo una scansione a 180 gradi dell'ambiente circostante. Il secondo motore, un servo, è montato sopra il primo e controlla l'inclinazione verticale del sensore, permettendo di regolare l'angolo di puntamento rispetto all'orizzonte. Questa configurazione consente un'acquisizione dati versatile e precisa, adattabile a diverse applicazioni. Il sistema può essere definito come sistema di scansione pan-tilt, evidenziando il movimento panoramico sul piano orizzontale e l'inclinazione verticale del sensore. Altri termini adeguati a descrivere la struttura includono sistema di movimentazione biassiale o piattaforma di scansione rotazionale-verticale, in quanto il meccanismo consente la raccolta di informazioni tridimensionali regolando in modo indipendente i due assi di movimento.

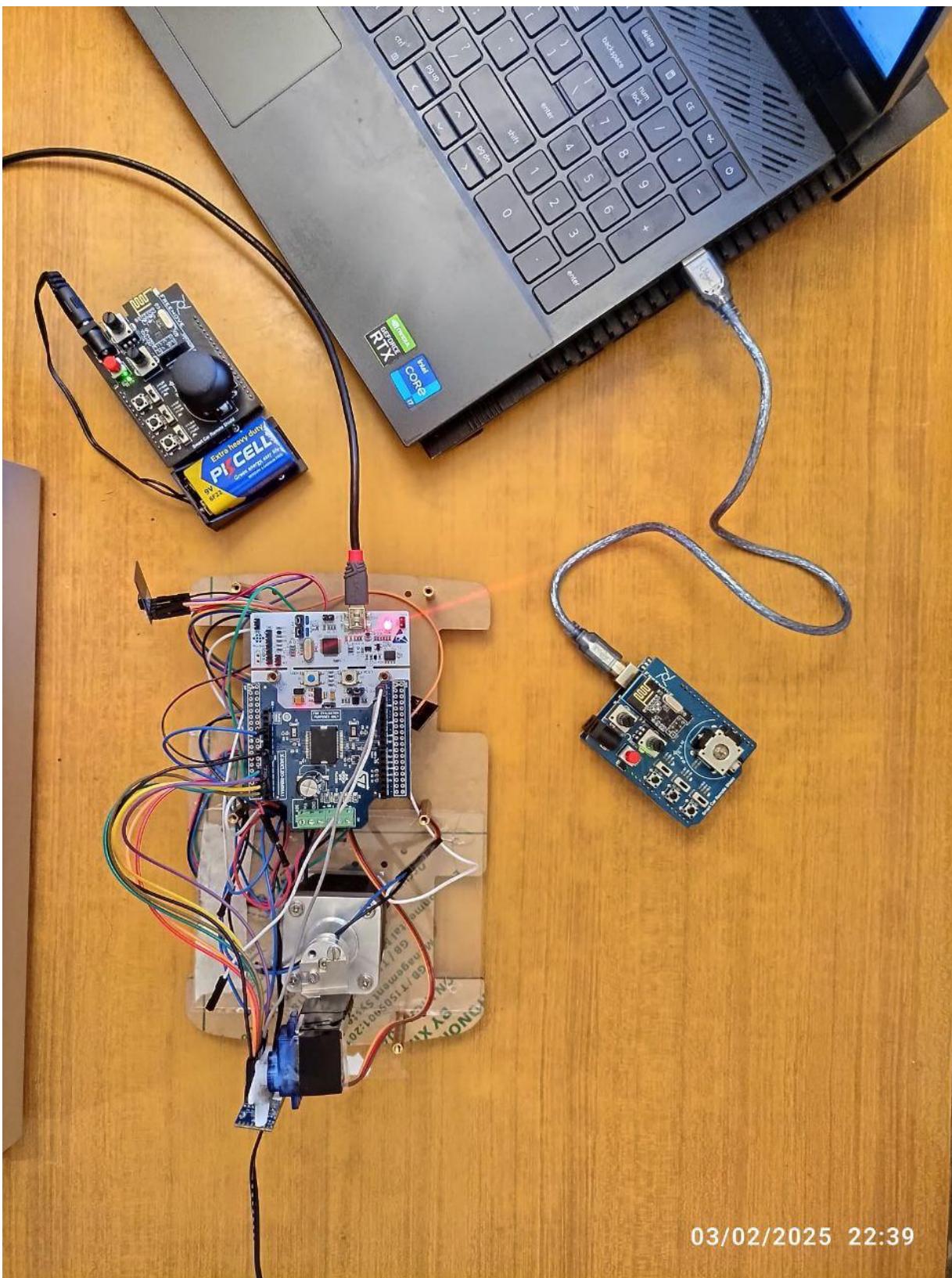


Figura 29. Foto del Progetto della Tesi completato

2.3.2. Diagrammi Elettrici e Collegamenti

La configurazione dei pin del progetto che deve essere configurata per il microcontrollore STM32 è disponibile nella seguente tabella:

Table 1. Configurazione dei pin

STM32 Pin	Componente	Componente pin
PA10	IHM05A1	ENBLE
PB3	IHM05A1	VREF A
PB5	IHM05A1	HALF/FULL
PB4	IHM05A1	CONTROL
PB10	IHM05A1	CLOCK
PA8	IHM05A1	CW/CCW
PA9	IHM05A1	Reset
PC7	IHM05A1	VREF B
PB9	VL53L5CX	SDA
PB8	VL53L5CX	SCL
PA0	VL53L5CX	LPn
PA1	VL53L5CX	I2C_RST
PA4	VL53L5CX	INT
PB0	VL53L5CX	PWR_EN
GND	VL53L5CX	GND
3V3	VL53L5CX	IOVDD
3V3	VL53L5CX	AVDD
PC1	NRF24L01	MOSI
PC12	NRF24L01	CE
PC10	NRF24L01	CLK
PC11	NRF24L01	MISO
PD2	NRF24L01	CSN
GND	NRF24L01	GND
3V3	NRF24L01	VCC
GND	SG90	GND
5V	SG90	5V
PA11	SG90	CONTROL
3V3	ESP-01S	3V3
GND	ESP-01S	GND
3V3	ESP-01S	EN
PB7	ESP-01S	TX
PB6	ESP-01S	RX

Schema di cablaggio dell'intero progetto:

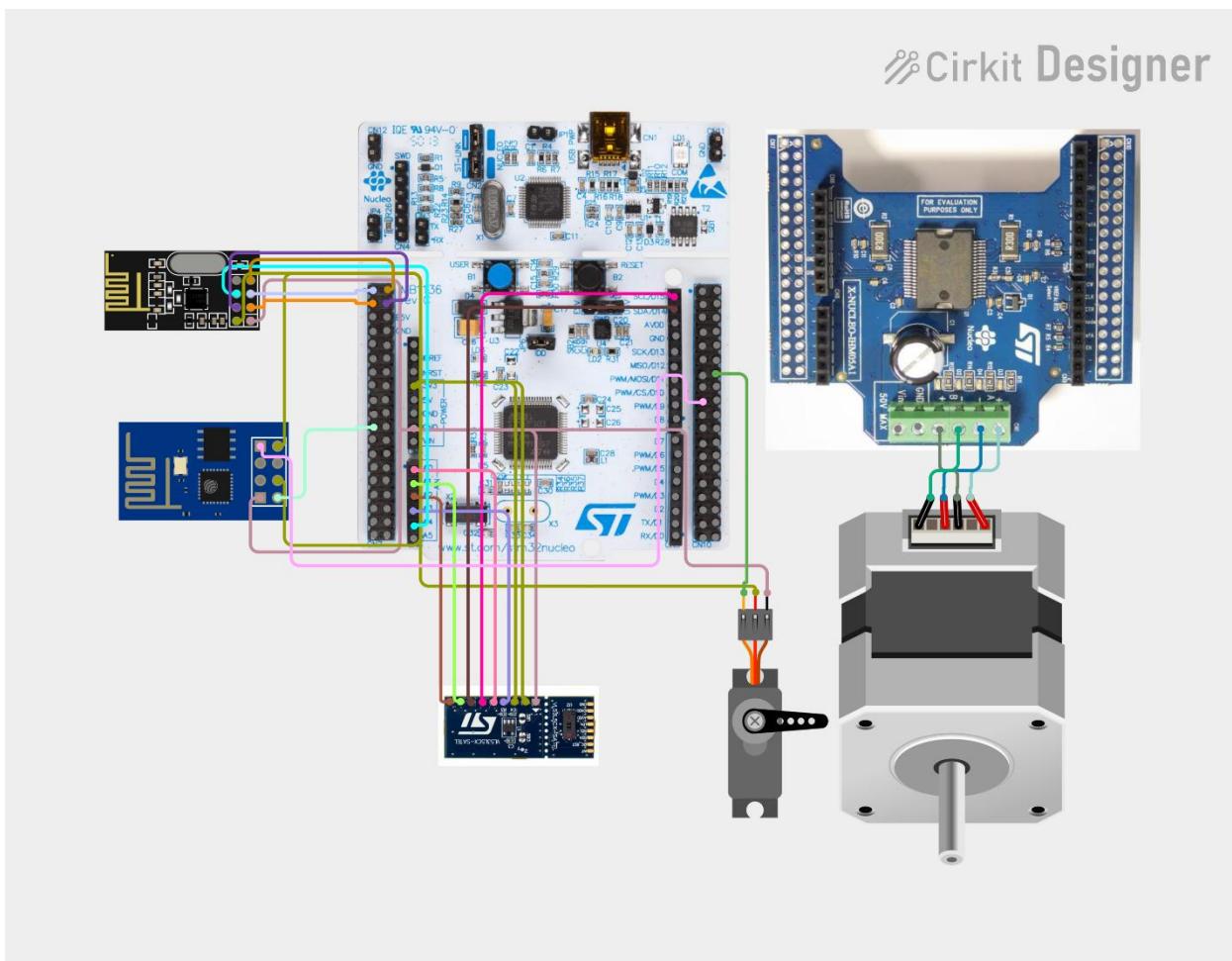


Figura 30. Schema di configurazione dei pin (2)

3. Implementazione

3.1. Integrazione Hardware e Software

L'intero progetto di base è la combinazione di sei singoli progetti, alcuni sviluppati da STMicroelectronics e altri da diverse aziende.

Per sviluppare un'applicazione sul microcontrollore, è necessario utilizzare una libreria HAL, che semplifica notevolmente la programmazione rispetto all'implementazione bare-metal, la quale risulterebbe più complessa rispetto alle conoscenze attuali di ingegneria di cui dispongo. Pertanto, ho cercato le soluzioni migliori e più semplici per soddisfare le esigenze del progetto.

STM32Cube HAL è un layer software embedded che garantisce la massima portabilità tra i microcontrollori STM32, mentre le API LL costituiscono un livello più leggero, veloce e orientato agli esperti, più vicino all'hardware rispetto alla HAL. Entrambe le API, HAL e LL, possono essere utilizzate simultaneamente con alcune limitazioni.

Le API HAL e LL sono pronte per la produzione e sono sviluppate in conformità con le linee guida MISRA-C®:2004, con alcune eccezioni documentate, e con lo standard ISO/TS 16949. Inoltre, STMicroelectronics applica processi di validazione specifici per garantire una qualificazione più approfondita.

STM32CubeF4 raccoglie in un unico pacchetto tutti i componenti software embedded generici necessari per sviluppare un'applicazione sui microcontrollori STM32F4. Seguendo l'iniziativa STM32Cube, questi componenti software sono altamente portabili, non solo all'interno della serie STM32F4 ma anche verso altre serie STM32. Inoltre, le API low-layer forniscono un'alternativa ad alte prestazioni e a basso consumo di memoria rispetto alla HAL di STM32CubeF4, sacrificando però portabilità e semplicità.

Le API HAL e LL sono disponibili con licenza BSD open-source per facilitare l'uso da parte degli sviluppatori.

Il software utilizzato per lo sviluppo sul microcontrollore comprende la libreria di espansione per sensori Time-of-Flight, denominata X-CUBE-TOF1, che viene eseguita sugli STM32 e include driver per il riconoscimento dei sensori e il rilevamento della distanza su uno o più dispositivi. Questa libreria è basata sulla tecnologia software STM32Cube per garantire la portabilità tra diversi microcontrollori STM32.

Il software fornisce un'implementazione di esempio dei driver compatibili con le schede di sviluppo VL53L1, VL53L3CX, VL53L4CD, VL53L4CX, VL53L4ED, VL53L5CX, VL53L7CX e VL53L8CX, collegate a una scheda di sviluppo STM32 Nucleo.

Sono disponibili applicazioni di esempio, tra cui:

- Rilevamento della distanza per schede di espansione e schede breakout
- Rilevamento multi-sensore
- Calibrazione dei sensori

Per il controllo del motore stepper bipolare, il software utilizzato è X-CUBE-SPN5, sviluppato da STMicroelectronics. (17)

X-CUBE-SPN5 è un pacchetto software di espansione per STM32Cube. Questo software è eseguito sui microcontrollori STM32 e fornisce una gestione completa del driver L6208 per il controllo micro-stepping dei motori Stepper. L'espansione è basata sulla tecnologia software STM32Cube per facilitare la portabilità tra diversi microcontrollori STM32. Il software include un'implementazione di esempio dei driver per il controllo di un motore Stepper bipolare.

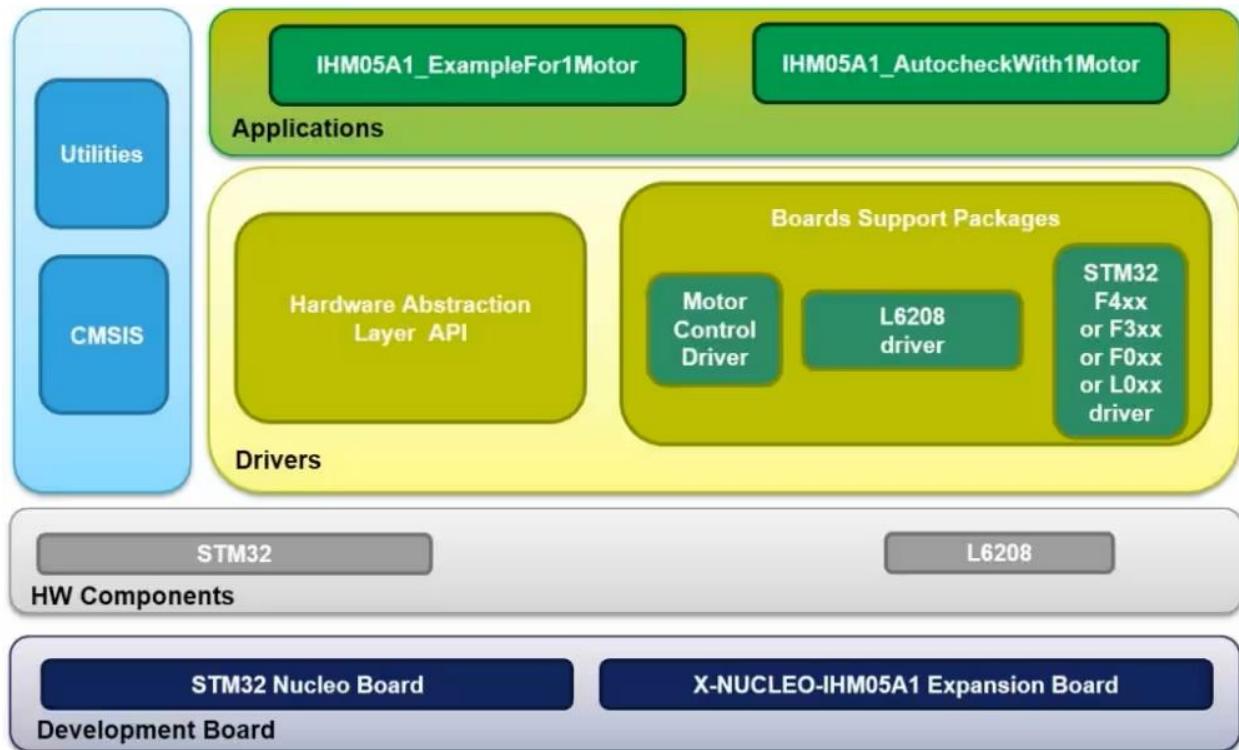


Figura 31. L'architettura del software X-CUBE-SPN5

L'integrazione del sistema è stata completata utilizzando diverse librerie reperite online. In alcuni casi, è stato necessario sviluppare il firmware in modo autonomo, poiché non esistevano librerie adeguate a specifiche funzionalità, come ad esempio per il protocollo MQTT e per il modulo ESP01-S.

Come è noto, l'integrazione di singoli moduli o progetti non garantisce il corretto funzionamento complessivo del sistema. Come previsto, sono stati riscontrati numerosi errori di programmazione, conflitti nei pin di collegamento e incompatibilità tra le librerie e il modello specifico del microcontrollore utilizzato. Attraverso un accurato processo di debugging, eseguito riga per riga, sono stati identificati e risolti tutti i problemi. Una volta superate queste difficoltà, è stato possibile procedere con l'implementazione dell'obiettivo principale della tesi, utilizzando tutte le risorse disponibili.

3.2. Configurazione dei Moduli NRF

Per avere una unità di controllo locale e sicura e un data logger, è necessario un modulo di radiofrequenza a bassa portata. Possiamo suddividere la parte fisica del progetto in tre sezioni:

La parte principale, che ospita il microcontrollore STM32, ha un modulo NRF configurato su 2 canali separati di comunicazione, uno con il controllo remoto e uno con il data logger.

Il primo canale di comunicazione è dedicato al controllo remoto, ricevendo i valori dei potenziometri e dei pulsanti per controllare il motore stepper e il servo motore, e successivamente invia una risposta ACK. D'altra parte, se il controllo remoto riceve l'ACK, si accende il LED vicino al modulo NRF, che indica una connessione stabile sia con il controllo remoto che con il microcontrollore STM32 principale.

Il secondo canale di comunicazione è dedicato al data logger, che, collegando il logger al computer tramite USB, consente di visualizzare i log del sistema in tempo reale su un terminale seriale.

Poiché nel microcontrollore STM32 entrambe le comunicazioni sono gestite tramite un singolo modulo NRF, ogni volta che si desidera inviare o ricevere un messaggio, è necessario modificare la configurazione iniziale del modulo, cambiando lo stato del modulo NRF tra trasmettitore e ricevitore.

3.3. Configurazione dei Moduli WiFi

Per pubblicare i dati del sensore al broker MQTT per ulteriori sviluppi e analisi, è stato necessario configurare il modulo ESP01-S Wi-Fi. La configurazione del modulo è stata effettuata tramite comandi AT, iniziando con l'aggiornamento del firmware del modulo alla versione più recente. È stato utilizzato un hotspot cellulare per fornire la connessione a Internet globale.

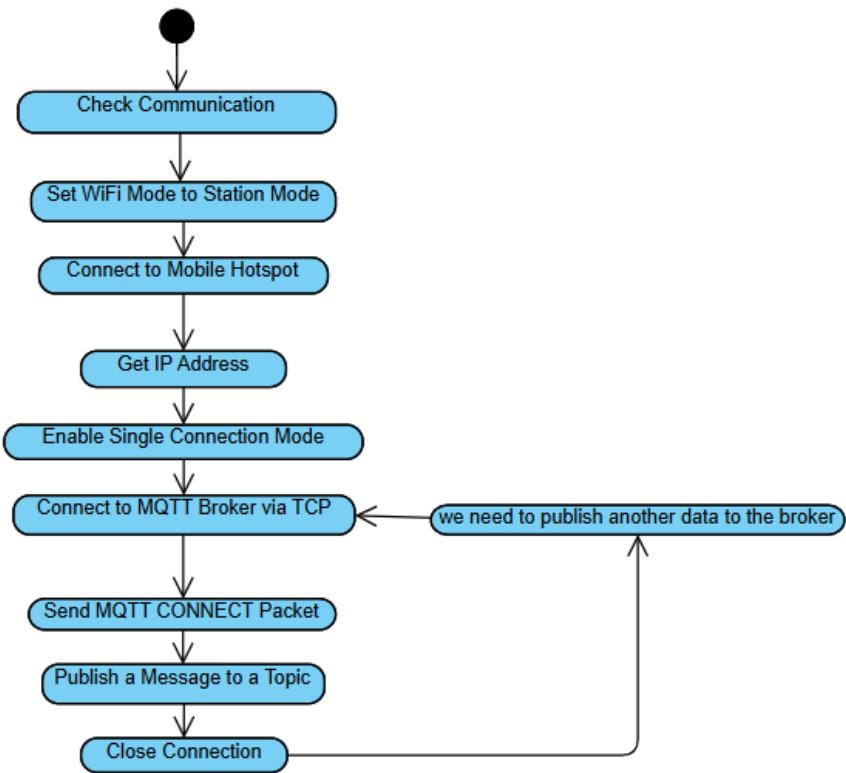


Figura 32. Diagramma di stato del modulo ESP01-S Wi-Fi

3.4. Elaborazione dei Dati del Sensore

Il rumore è qualsiasi variazione o distorsione indesiderata nei dati del sensore che può ridurne l'accuratezza e l'affidabilità. Il rumore può derivare da diverse fonti, come fattori ambientali, malfunzionamenti del sensore, interferenze o errori di misurazione. Per identificare e rimuovere il rumore dai dati del sensore, è possibile utilizzare metodi come il filtraggio, l'appianamento, il rilevamento di valori anomali o il rilevamento di anomalie. Questi metodi possono aiutare a ridurre il livello di rumore e migliorare la qualità del segnale nei dati del sensore. (18)

Un outlier è un punto di osservazione che si trova distante rispetto agli altri dati.

I dati mancanti e gli outliers possono avere un impatto significativo sull'analisi dei dati e sui modelli di machine learning. Ignorare i dati mancanti può portare a risultati distorti o imprecisi, poiché i valori mancanti potrebbero contenere informazioni importanti. D'altro canto, gli outliers possono distorcere le misure statistiche e influire sulle performance dei modelli predittivi introducendo rumore. Pertanto, è fondamentale gestire adeguatamente i dati mancanti e gli outliers per garantire un'analisi dei dati affidabile e robusta. (19)

Adottiamo un approccio di elaborazione in quattro fasi che filtra sistematicamente il rumore, rimuove gli outliers e preserva i dettagli strutturali importanti.

- 1- Ignorando il valore -1 relativo a ciascuna cella dei dati misurati dal sensore.
- 2- Applicando il filtro Hampel per la rimozione degli outliers.
- 3- Utilizzando la media trimmata per un'aggregazione robusta.
- 4- Applicando il filtro bilaterale per la riduzione del rumore consapevole dei bordi.

Il primo passo consiste nell'ignorare i valori -1, che rappresentano misurazioni di profondità mancanti o non valide. Questi si verificano a causa della bassa riflettività, dell'assorbimento del segnale o delle limitazioni del sensore. Se i valori -1 venissero interpolati o considerati nell'aggregazione statistica, potrebbero introdurre distorsioni nella mappa di profondità. Pertanto, vengono completamente esclusi da tutti i calcoli statistici. Data una serie di valori di distanza per una determinata zona, le misurazioni valide sono rappresentate come:

$$D_{\text{valid}} = \{d_i \in D \mid d_i \neq -1\}$$

Dove D è il set originale di misurazioni, e D_{valid} contiene solo i valori di profondità validi.

Il secondo passo consiste nell'applicare il filtro Hampel per rimuovere gli outliers. I sensori ToF spesso producono valori estremi a causa di interferenze da percorso multiplo o rumore del sensore. Il filtro Hampel è robusto a tali anomalie e funziona calcolando la mediana e la deviazione assoluta dalla mediana (MAD).

La MAD è definita come:

$$\text{MAD} = \text{median}(|d_i - \text{median}(D_{\text{valid}})|)$$

Un valore di soglia per gli outliers viene quindi determinato come:

$$T = 3 \times \text{MAD}$$

Qualsiasi misurazione d_i che soddisfi:

$$|d_i - \text{median}(D_{\text{valid}})| > T$$

è considerata un outlier e viene sostituita con la mediana.

Il terzo passo prevede l'utilizzo della media trimmata per un'aggregazione robusta su più fotogrammi. Poiché il sensore acquisisce più misurazioni al secondo, una media semplice sarebbe molto sensibile agli outliers residui. Invece, la media trimmata rimuove una certa percentuale dei valori più bassi e più alti prima di calcolare la media. Dato un set ordinato di valori di profondità validi D_{valid} , il set trizzato D_{trimmed} è definito come:

$$D_{\text{trimmed}} = \{d_i \in D_{\text{valid}} \mid p \leq i \leq (n - p)\}$$

dove p è il numero di valori rimossi da ciascun estremo, e n è il numero totale di misurazioni valide. Il valore di profondità aggregato finale è quindi dato da:

$$\bar{D} = \frac{1}{|D_{\text{trimmed}}|} \sum_{d_i \in D_{\text{trimmed}}} d_i$$

L'ultimo passo consiste nell'applicare il filtro bilaterale per levigare i valori di profondità preservando i bordi. I filtri tradizionali, come la levigatura gaussiana, sfocano i bordi, rendendo meno distinti le discontinuità di profondità. Il filtro bilaterale, invece, considera sia la distanza spaziale che la somiglianza di profondità. Il valore di profondità filtrato alla posizione (x, y) viene calcolato come:

$$D_{\text{filtered}}(x, y) = \frac{1}{W} \sum_{i,j} D(i, j) \cdot G_s(i, j) \cdot G_r(D(i, j), D(x, y))$$

dove $G_s(i, j)$ è il peso spaziale:

$$G_s(i, j) = \text{exp}\left(-\frac{(i - x)^2 + (j - y)^2}{2\sigma_s^2}\right)$$

e $G_r(D(i, j), D(x, y))$ è il peso di intervallo:

$$G_r(D(i,j), D(x,y)) = \exp\left(-\frac{(D(i,j) - D(x,y))^2}{2\sigma_r^2}\right)$$

Dove σ_r controlla l'influenza spaziale e σ_d controlla l'influenza della somiglianza di intensità.

Questo approccio garantisce che la matrice di profondità finale rappresenti accuratamente l'ambiente, mantenendo i confini degli oggetti ben definiti. La combinazione di filtro Hampel, media trimmata e filtro bilaterale riduce il rumore del sensore e migliora l'accuratezza della profondità senza distorcere i dettagli strutturali importanti.

3.5. Invio Dati via MQTT

Dopo la manipolazione e la correzione dei dati acquisiti dal sensore, in questa fase è necessario inviarli a un broker MQTT. Il broker MQTT scelto per questa parte del progetto è ThingSpeak.

ThingSpeak è un servizio della piattaforma IoT Analytics che consente di aggregare, visualizzare e analizzare flussi di dati in tempo reale nel cloud. Attraverso ThingSpeak, è possibile inviare dati dai dispositivi, creare visualizzazioni istantanee in tempo reale e inviare avvisi utilizzando servizi web come Twitter e Twilio. Inoltre, grazie all'integrazione con MATLAB, ThingSpeak permette di scrivere ed eseguire codice MATLAB per la preelaborazione, la visualizzazione e l'analisi dei dati. Questo strumento offre a ingegneri e scienziati la possibilità di prototipare e sviluppare sistemi IoT senza la necessità di configurare server o sviluppare software web.

Dopo aver creato un account sulla piattaforma ThingSpeak, è stato configurato un canale. Quest'ultimo dispone di diversi campi, e il primo campo è stato utilizzato per contenere l'ultima matrice originale di dati 8x8 acquisita dal sensore, mentre il secondo campo è stato utilizzato per contenere la matrice filtrata in base alla manipolazione discussa nel capitolo precedente.

```

Topic: channels/2808764/subscribe QoS: 0

{
  "channel_id": 2808764,
  "created_at": "2025-02-02T13:26:49Z",
  "entry_id": 78,
  "field1": "[286, 236, 254, -1, -1, -1, -1, 332, 288, 253, -1, -1, -1, -1, 827, 277, -1, -1, -1, -1, -1, 877, 832, -1, -1, -1, -1, 927, 921, 869, 850, -1, -1, -1, -1, 877, 854, -1, -1, -1, -1, 945, 910, 885, 863, -1, 507, 930, 963, 940, 916, 898, 874, 413, 525, 960, 973, 951, 930, 914, 895]", "field2": "[274, 253, 247, -1, 315, 865, 852, 531, 277, 261, 1149, -1, -1, 888, 860, 837, 279, -1, -1, 787, 664, 517, 842, -1, -1, 915, 1034, 951, 919, 861, 851, -1, -1, -1, 942, 956, 854, 871, 862, -1, -1, 954, 950, 942, 911, 881, 873, -1, 500, 943, 951, 942, 918, 897, 885, 413, 511, 939, 952, 947, 925, 906, 895]", "field3": null, "field4": null, "field5": null, "field6": null, "field7": null, "field8": null, "latitude": null, "longitude": null, "elevation": null, "status": "MQTT_PUBLISH"
}

```

Figura 33. esempio di messaggio del broker MQTT

3.6. Controllo dei Motori

I motori saranno controllati dal telecomando, con i valori dei potenziometri e dei pulsanti sulla scheda del telecomando.

Il telecomando legge i valori di input dai potenziometri e dai pulsanti, li organizza in un pacchetto dati strutturato e li trasmette tramite NRF24L01. Il pacchetto include:

- Potenziometro 1: Valore del primo potenziometro (controllo avanzamento motore).
- Potenziometro 2: Valore del secondo potenziometro (controllo arretramento motore).
- Switch 1: Pulsante per il ritorno alla posizione di "Home".
- Switch 2: Pulsante per cambiare la posizione del servomotore.
- Switch 3: Pulsante per una posizione alternativa del servomotore.

Il valore letto dal potenziometro varia tra 0 e 1023 e deve essere convertito in un numero di passi per il motore passo-passo. Il sistema utilizza micro-step da 1/16 per garantire precisione nel movimento.

Per una rotazione di 360°, il motore deve compiere 6400 passi.

Per una rotazione di 180°, servono 3200 passi.

Per una rotazione di 60°, servono STEPS_FOR_60_DEGREES = 6400 / 6 = 1066 passi.

Per una rotazione di 6°, servono STEPS_FOR_6_DEGREES = 6400 / 60 = 106.66 passi.

La conversione da valore del potenziometro a passi viene effettuata utilizzando la relazione:

$$\text{STEPS_FOR_60_DEGREES} = \frac{\text{STEPS_FOR_360_DEGREES}}{6} = \frac{6400}{6} = 1066$$

Il motore viene controllato in base all'incremento del valore del potenziometro rispetto alla lettura precedente:

Con il valore del potenziometro 1, il sensore si sposta di 90 gradi in senso antiorario rispetto alla posizione di home.

Con il valore del potenziometro 2, il sensore si sposta di 90 gradi in senso orario rispetto alla stessa posizione di home.

Per evitare spostamenti indesiderati, viene introdotta una tolleranza.

Premendo il pulsante switch1, il motore torna alla posizione iniziale, impostata come "Home". La posizione "Home" è quella in cui il motore inizia a lavorare e corrisponde al momento in cui viene inizializzato.

Il servomotore viene controllato tramite i pulsanti switch2 e switch3, che regolano la posizione del motore variando il duty cycle del segnale PWM.

switch2 → Imposta il servo a 1100 µs, corrispondenti a circa 75 gradi rispetto all'asse di riferimento orizzontale del servo motore (posizione predefinita).

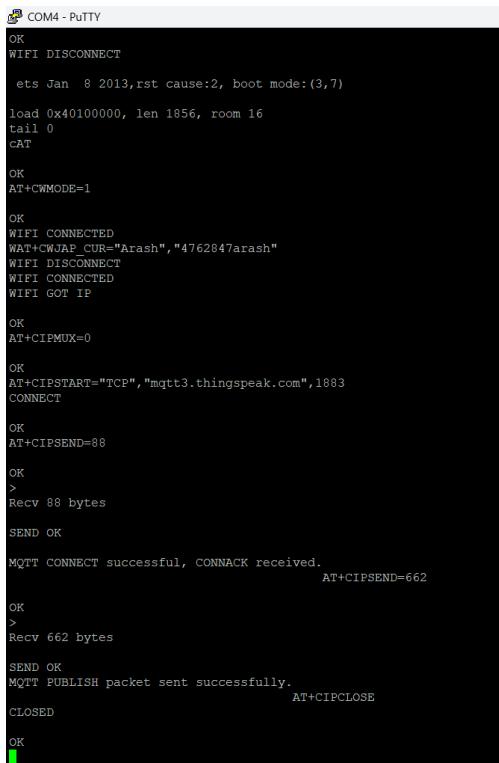
switch3 → Imposta il servo a 400 µs, corrispondenti a circa 30 gradi rispetto all'asse di riferimento orizzontale del servo motore (la superficie terrestre).

3.7. Monitoraggio in Tempo Reale

Il monitoraggio in tempo reale del sistema viene effettuato attraverso la stampa di log in diversi punti critici del progetto. Questi log vengono inviati a un terminale seriale su un computer personale, permettendo di osservare lo stato del sistema in ogni momento.

Per trasmettere i log in modalità wireless, viene utilizzato un modulo NRF24L01 collegato al microcontrollore STM32. Il microcontrollore invia i dati di monitoraggio in tempo reale al modulo NRF, che li trasmette al dispositivo data logger. Quest'ultimo è connesso al computer tramite un cavo USB e riceve i dati per la visualizzazione su un terminale seriale.

Grazie a questo metodo, è possibile diagnosticare il funzionamento del sistema senza la necessità di un'interfaccia grafica avanzata, garantendo un monitoraggio affidabile e immediato.



```
COM4 - PuTTY
OK
WIFI DISCONNECT
ets Jan  8 2013,rst cause:2, boot mode:(3,7)
load 0x40100000, len 1856, room 16
tail 0
cAT

OK
AT+CWMODE=1

OK
WIFI CONNECTED
WIFI+CWAP_CUR="Arash","4762847arash"
WIFI DISCONNECT
WIFI CONNECTED
WIFI GOT IP

OK
AT+CIPMUX=0

OK
AT+CIPSTART="TCP","mqqt3.thingspeak.com",1883
CONNECT

OK
AT+CIPSEND=88

OK
>
Recv 88 bytes

SEND OK

MQTT CONNECT successful, CONNACK received.          AT+CIPSEND=662

OK
>
Recv 662 bytes

SEND OK
MQTT PUBLISH packet sent successfully.           AT+CIPCLOSE

CLOSED

OK
```

Figura 34. Data logger terminale seriale

4. Risultati e Analisi

4.1. Visualizzazione dei Dati

La visualizzazione dei dati è utile per la pulizia dei dati, l'esplorazione della struttura dei dati, il rilevamento degli outlier e dei gruppi insoliti, l'identificazione di tendenze e cluster, l'individuazione di schemi locali, la valutazione dei risultati dei modelli e la presentazione dei risultati.

È fondamentale per l'analisi esplorativa dei dati e per il data mining, poiché permette di verificare la qualità dei dati e aiuta gli analisti a familiarizzare con la struttura e le caratteristiche dei dati a loro disposizione. Questo aspetto dell'analisi dei dati è spesso sottovalutato nei libri di testo, ma è sempre presente nelle indagini effettive.

Per il tipo di dati acquisiti, sono state adottate diverse tecniche di visualizzazione per rappresentare in modo efficace le informazioni. Le principali metodologie utilizzate sono:

Opzioni di visualizzazione per una matrice 8x8 di dati di distanza:

Heatmap:

Rappresenta i valori della matrice con una mappa di colori, dove ogni cella viene colorata in base alla distanza misurata. È utile per identificare rapidamente aree con valori anomali o gradienti di profondità. Quando non esistono dati di distanza, la cella non viene colorata e ha il colore fisso bianco.

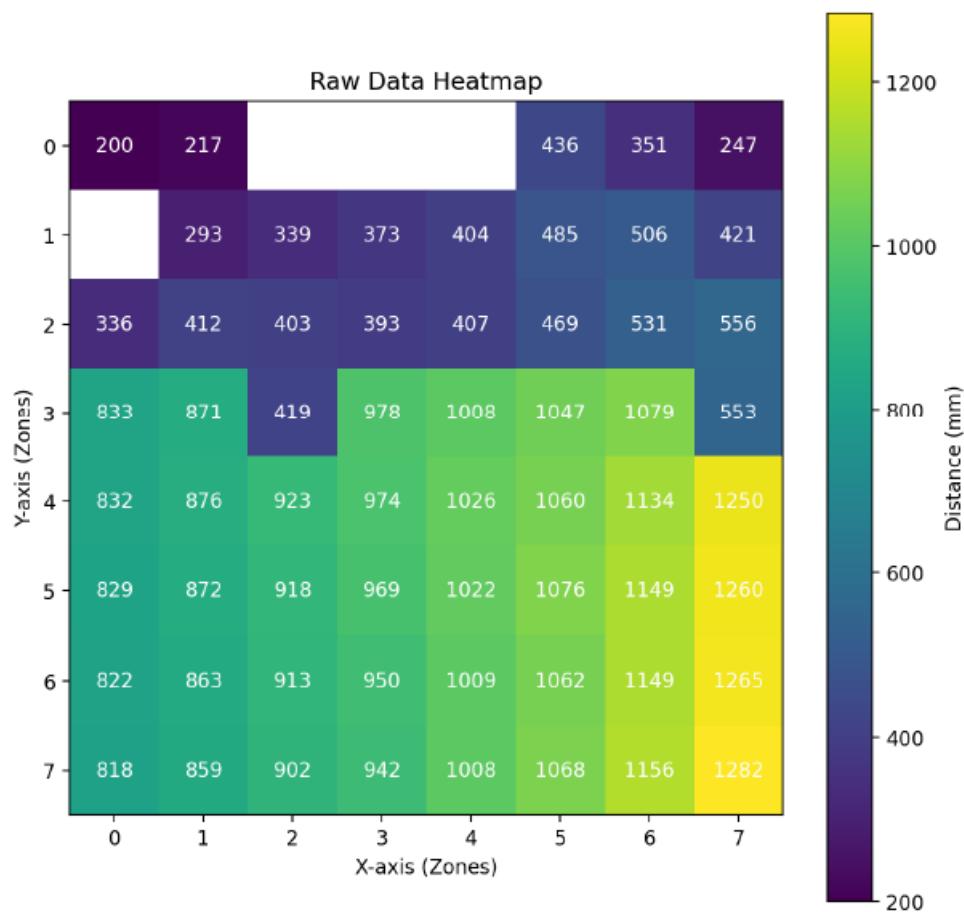


Figura 35. Esempio di heatmap

Nuvola di Punti (Point Cloud 3D):

Visualizza ogni punto della matrice in uno spazio tridimensionale, utile per integrazioni con software di analisi più avanzati, come quelli utilizzati nelle scansioni 3D.

3D Scatter Filtered Data

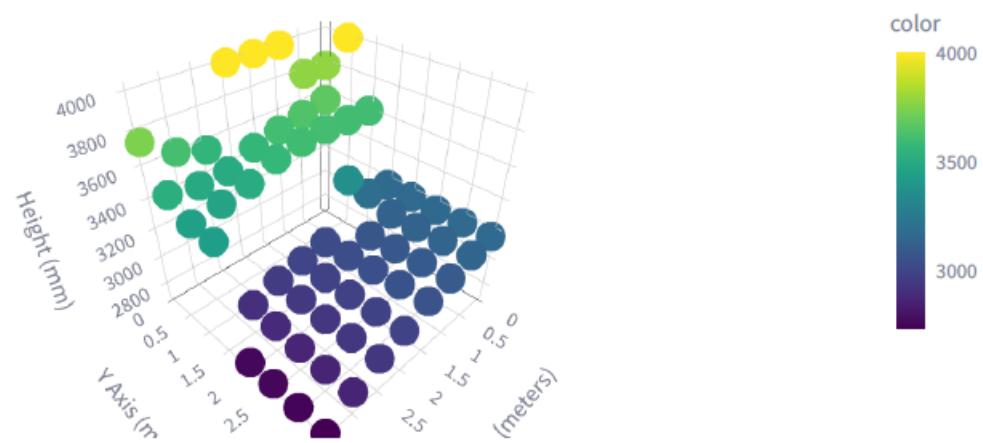


Figura 36. Esempio di Point Cloud

Wireframe 3D:

Simile al grafico a mesh, ma con una rappresentazione più schematica della distribuzione dei dati senza colorazione. È utile per comprendere l'andamento generale delle misure.

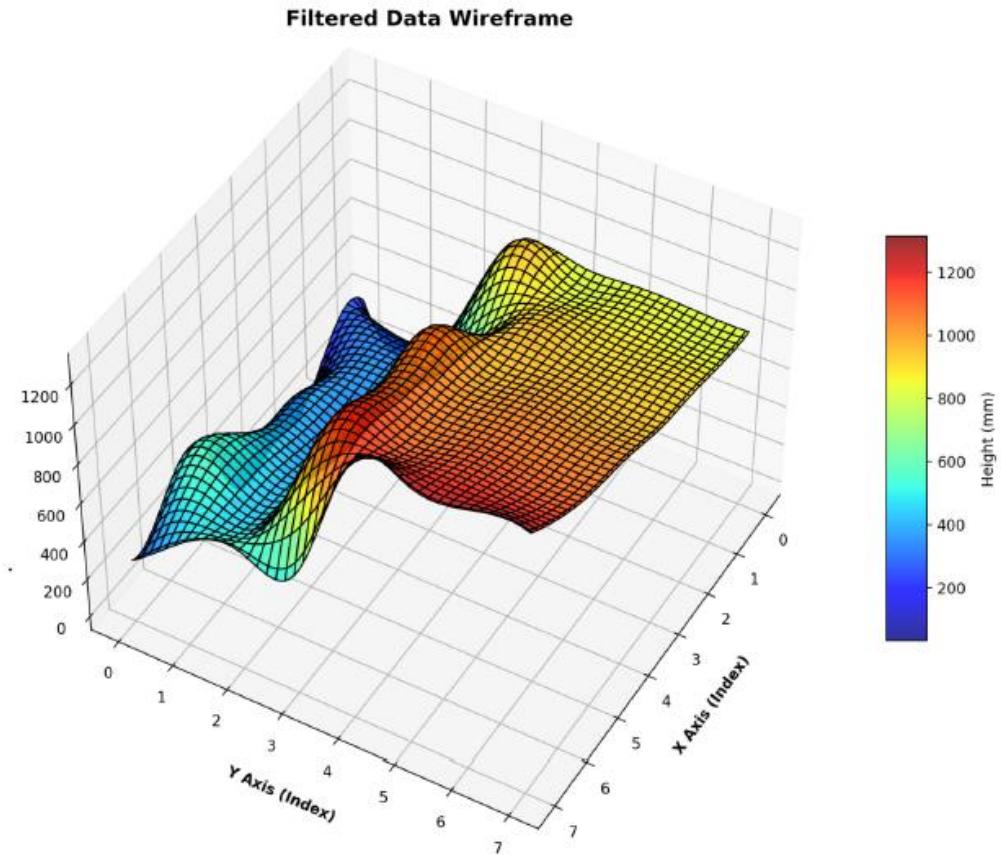


Figura 37. Grafico Wireframe 3D

L'obiettivo è ottenere una rappresentazione chiara e dettagliata della matrice di dati, in modo da facilitare l'analisi e l'interpretazione delle misurazioni.

4.2. Piattaforma Python per la visualizzazione

La piattaforma software sviluppata in Python per l'elaborazione e la visualizzazione dei dati ottenuti dal sensore VL53L5CX consente di segmentare e analizzare i dati tridimensionali acquisiti utilizzando diversi algoritmi di clustering e tecniche di filtraggio. L'interfaccia grafica è implementata con Streamlit, che permette una gestione interattiva dell'analisi dei dati.

Un modo più veloce per costruire e condividere applicazioni di dati. Streamlit trasforma gli script di dati in applicazioni web condivisibili in pochi minuti. Tutto in puro Python. Non è necessaria esperienza nel front-end. (20)

Visualizzare i dati originali, la segmentazione del piano e i risultati del clustering in grafici 3D interattivi.

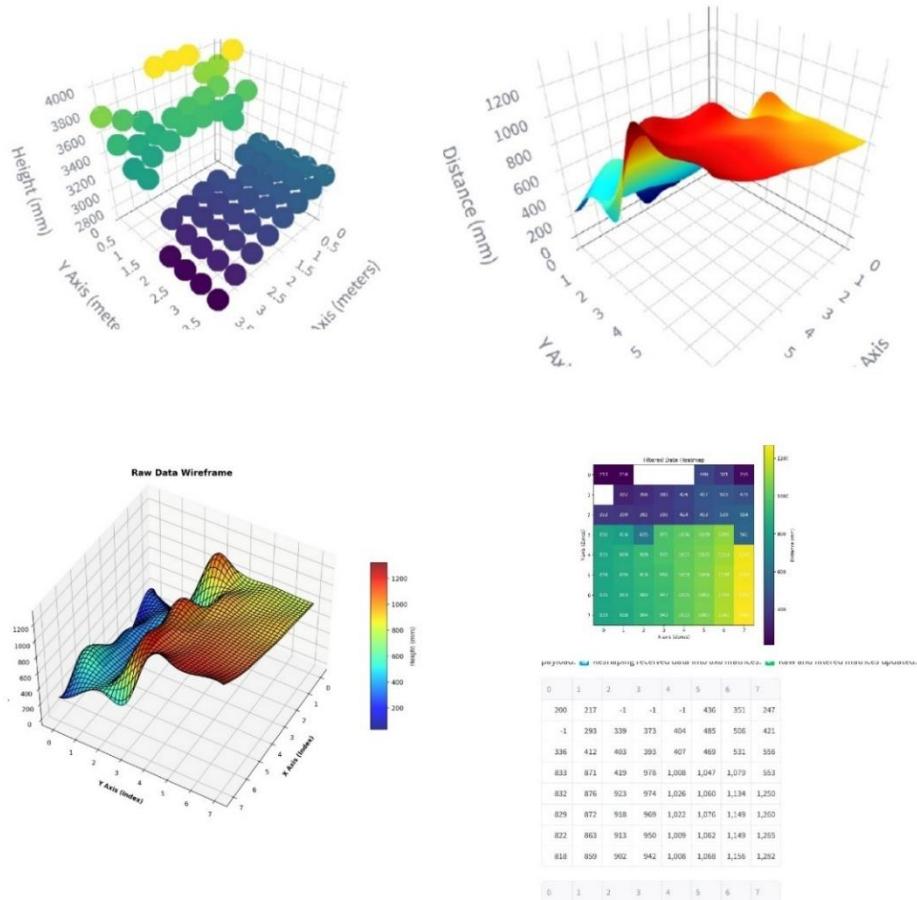


Figura 38. Piattaforma di visualizzazione in Streamlit

4.3. Test del Sistema di Misurazione 3D

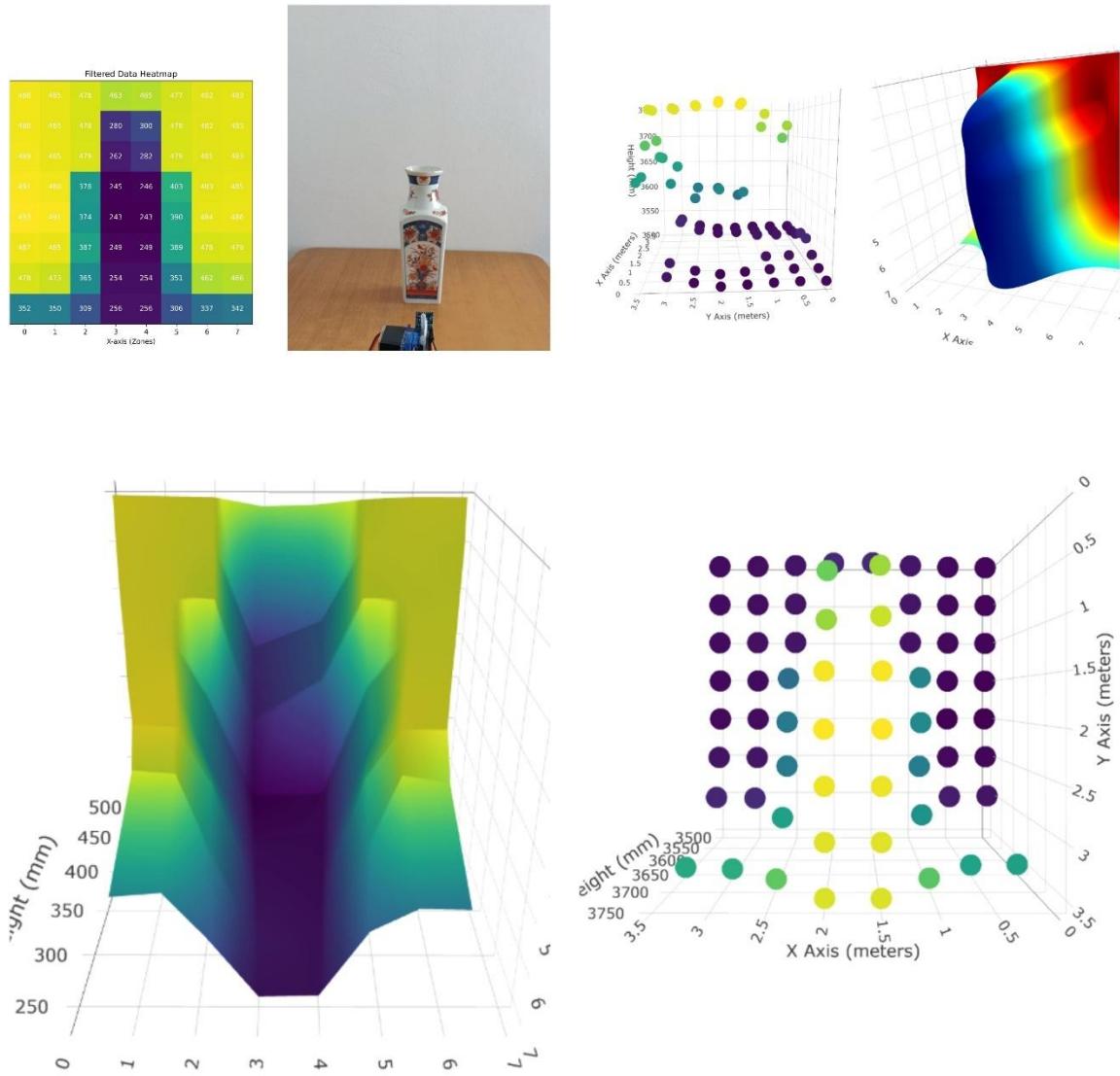


Figura 39. Caso di test piccolo barattolo

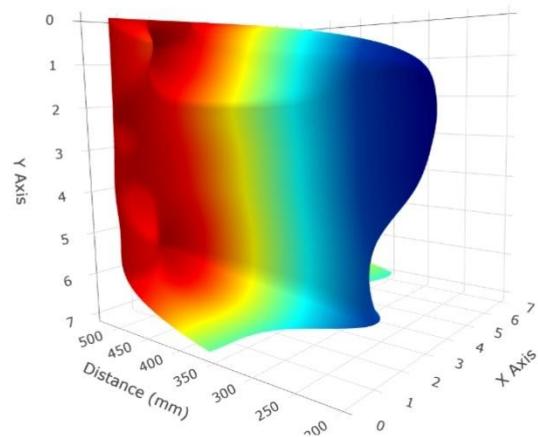
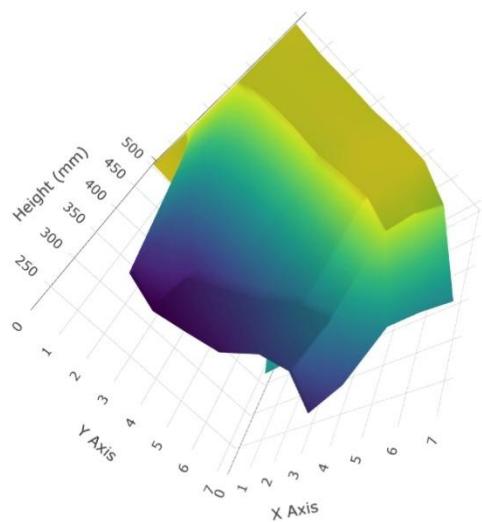
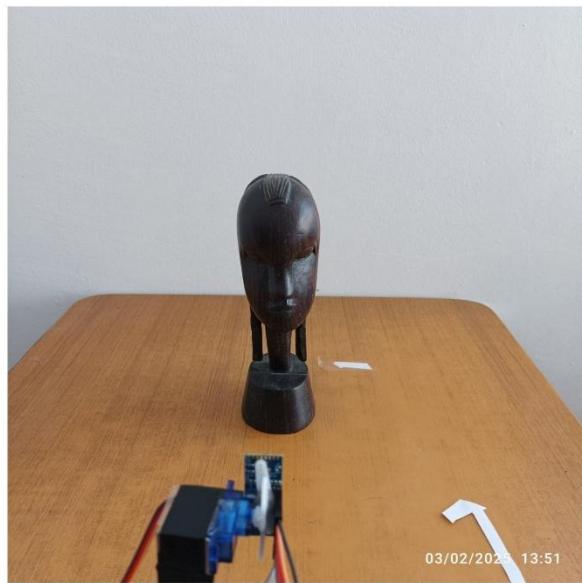
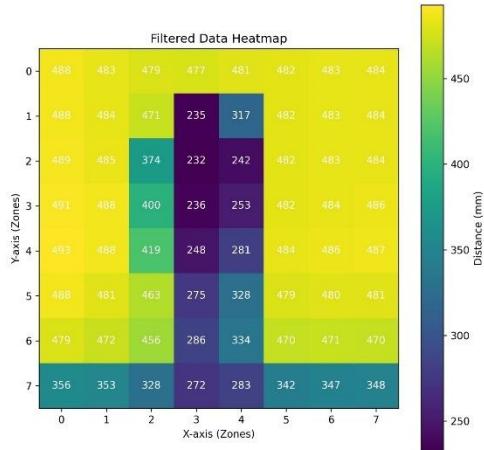


Figura 40. Caso di test della faccia frontale della statua

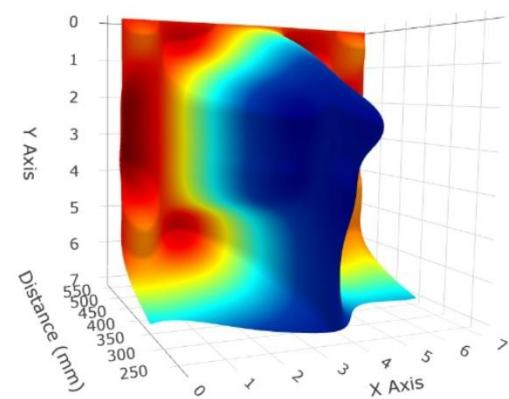
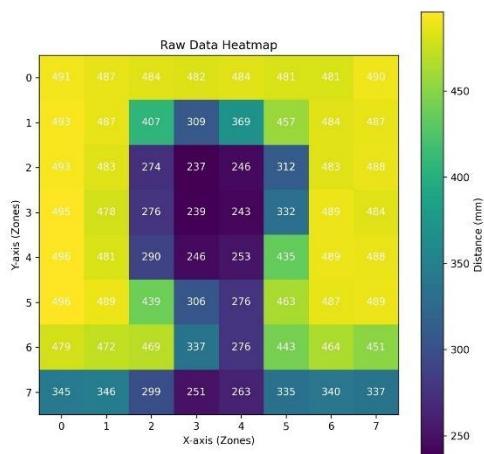
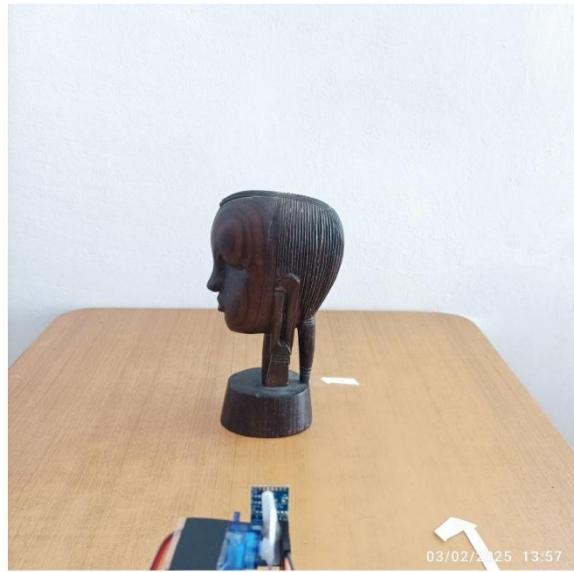
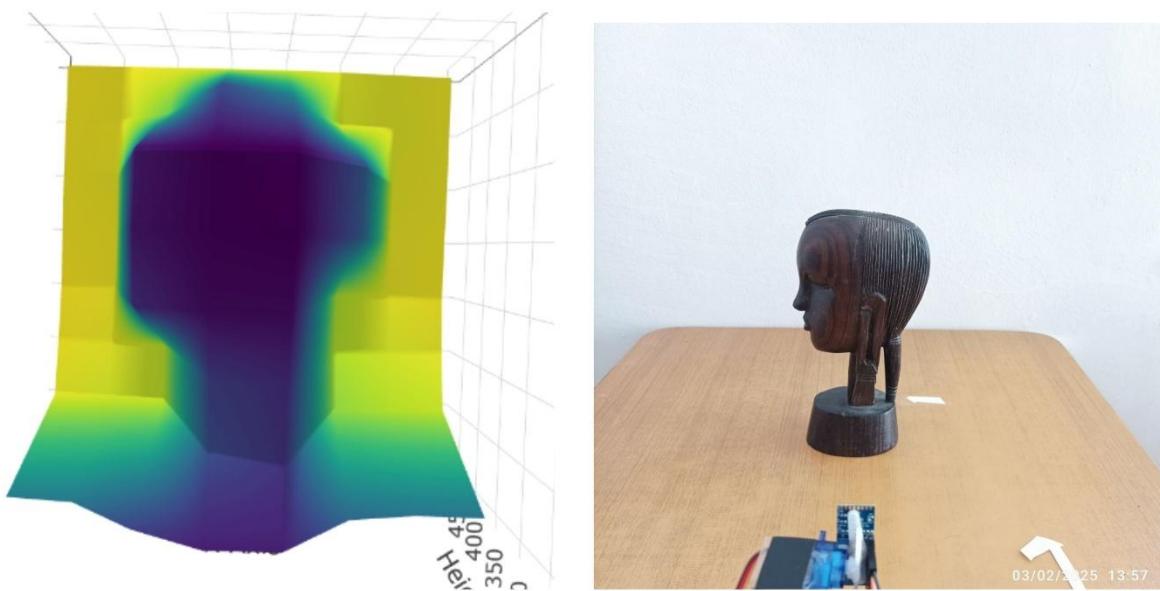


Figura 41. Caso di test della faccia laterale della statua

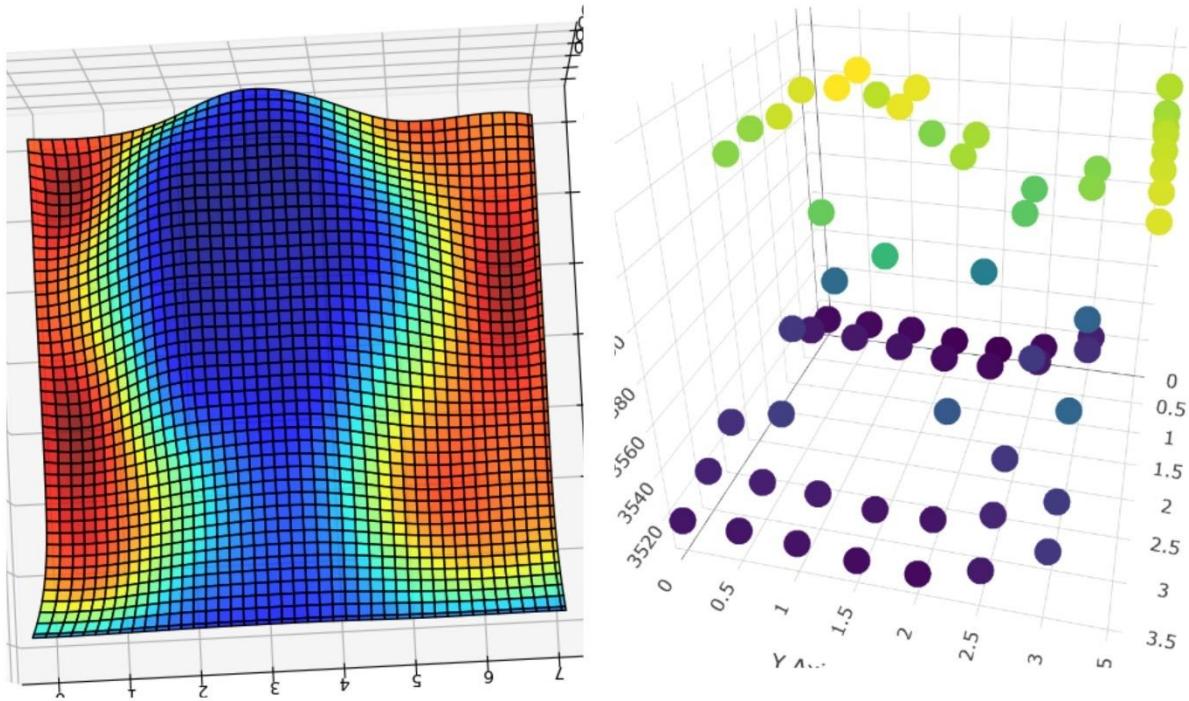


Figura 42. Caso di test grande barattolo

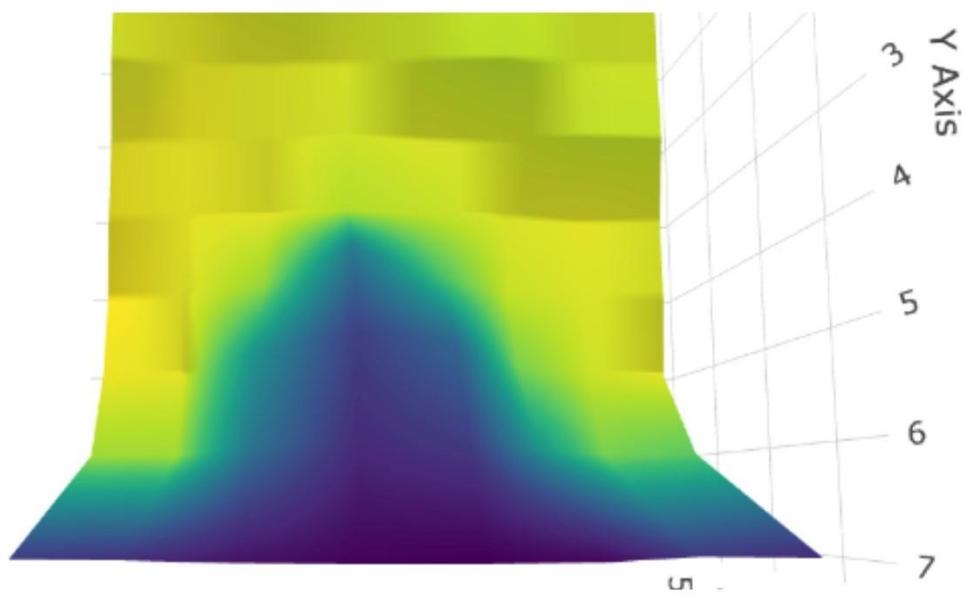


Figura 43. Caso di test della teiera

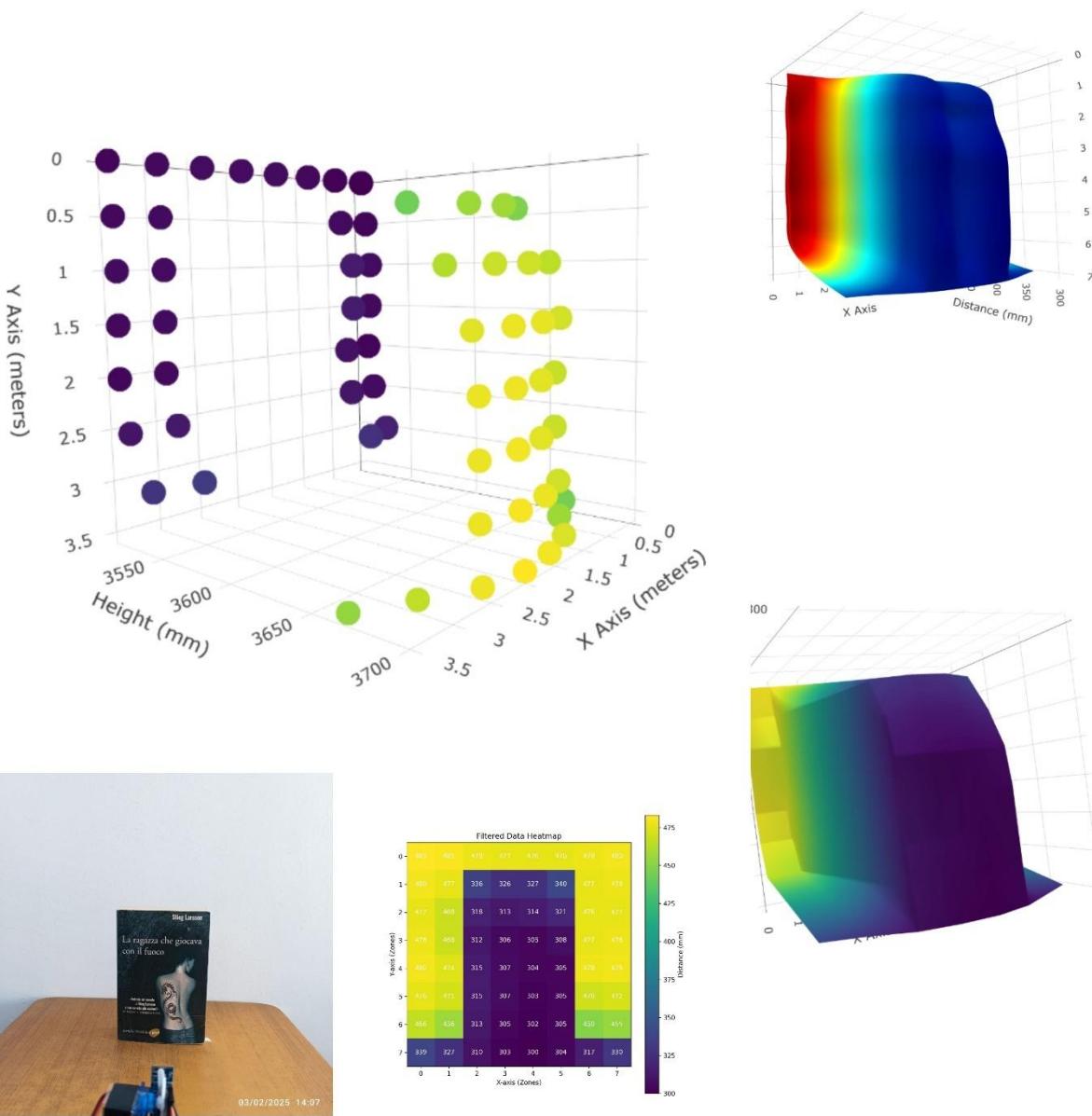


Figura 44. Caso di test del libro

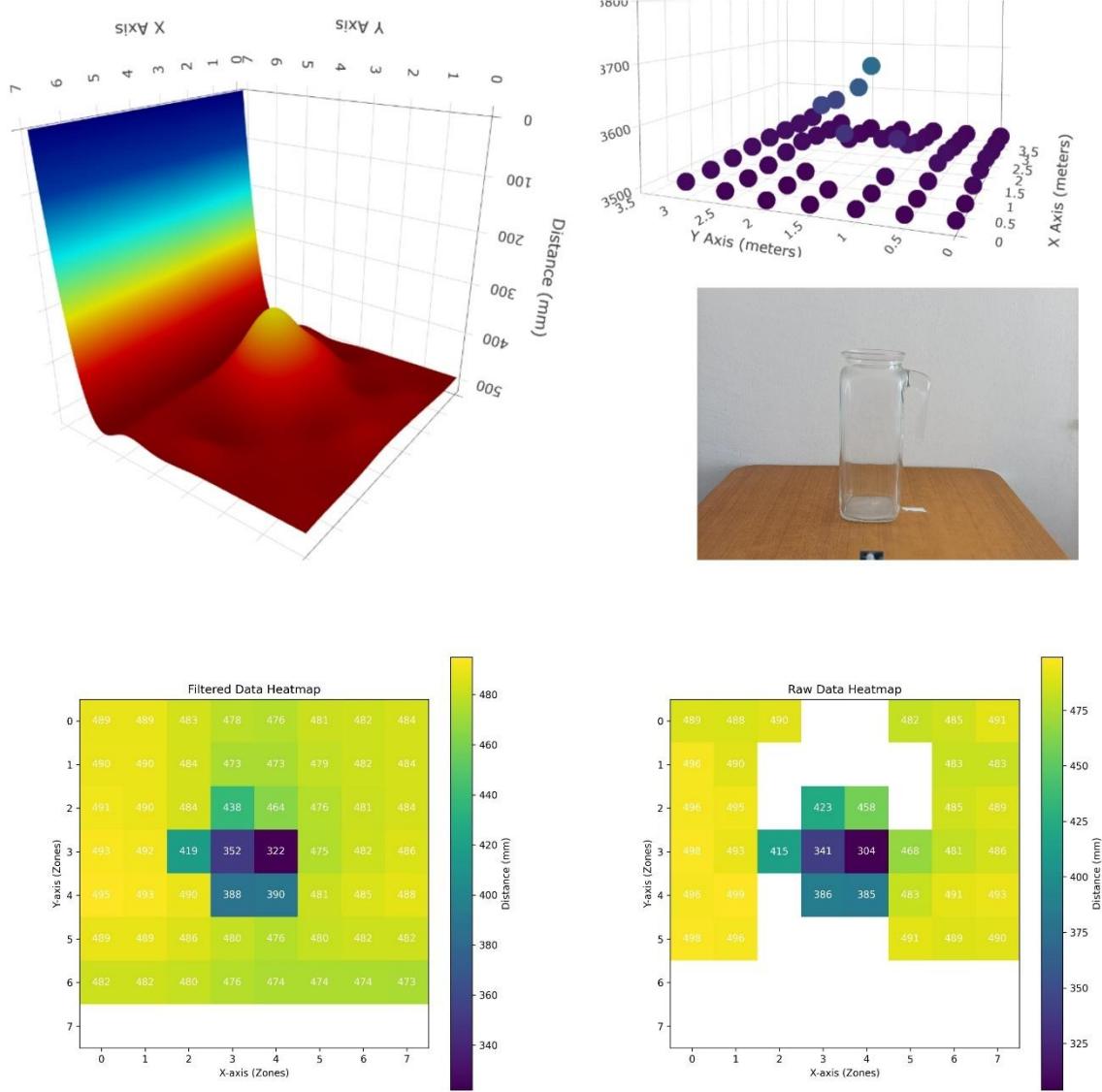


Figura 45. Caso di test del bicchiere da vetro

5. Discussione

5.1. Rilevanza e Applicazioni in Ambito Domotico

I sensori ToF possono essere utilizzati per migliorare la sicurezza di una casa. Possono rilevare la presenza di intrusi e attivare un allarme. Possono essere utilizzati per ottimizzare il consumo di energia, rilevando la presenza delle persone in uno spazio. Inoltre, sono efficaci anche in assenza di luce o al buio. Possono essere utilizzati in dispositivi alimentati a batteria, poiché consumano poca energia.

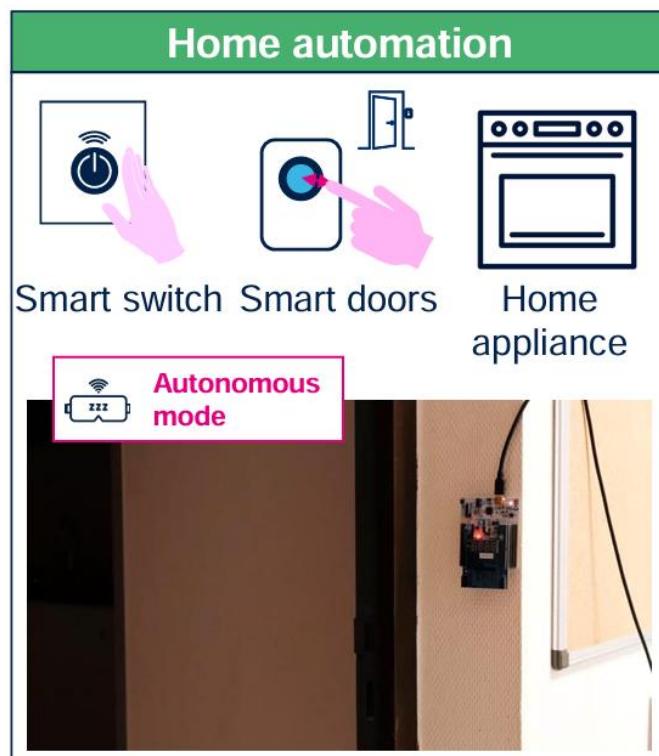


Figura 46. Sensore ToF nei sistemi di automazione domestica

L'integrazione di sensori TOF (Time-of-Flight) nei sistemi domotici rappresenta un'importante evoluzione nell'automazione e nella gestione intelligente degli ambienti. Il sistema sviluppato in questo progetto, basato su microcontrollore e sensore VL53L5CX, offre diverse possibilità di applicazione nel contesto della smart home, grazie alla sua capacità di acquisire misurazioni tridimensionali dell'ambiente in tempo reale.

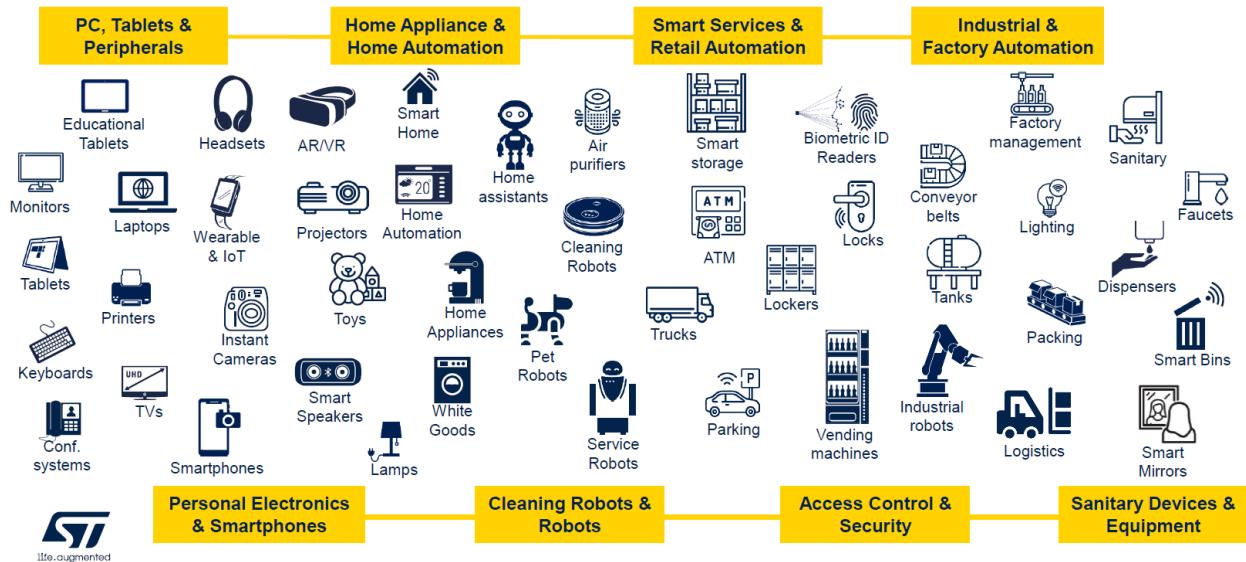


Figura 47. Applicazione del sensore ToF

Uno degli utilizzi più significativi è la misurazione dell'area e del volume degli oggetti posti di fronte al sensore, purché l'oggetto sia posizionato vicino a una parete e su un tavolo, creando così una base stabile per la misurazione. Questo può essere molto utile nel contesto industriale.

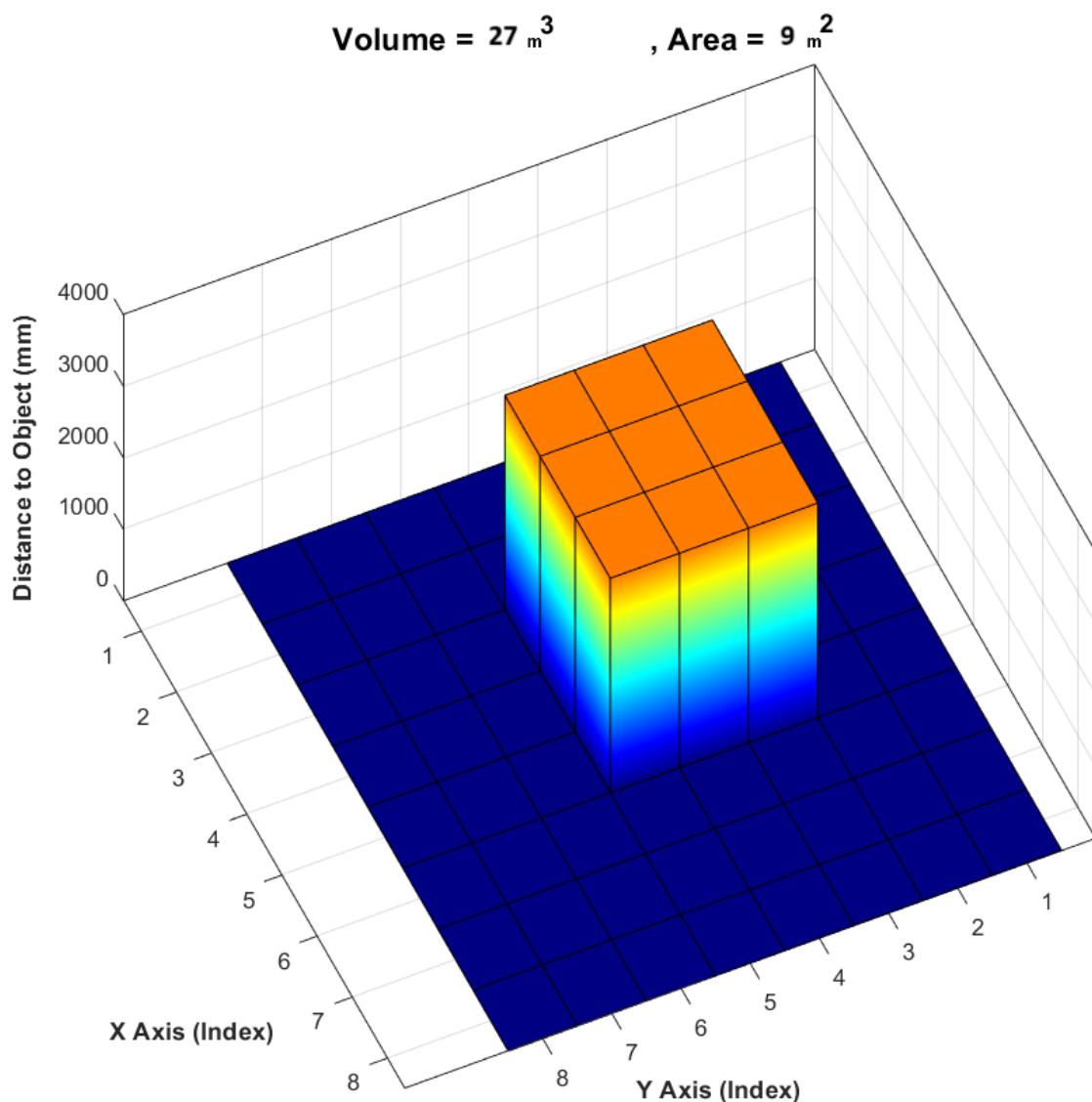


Figura 48. Misurazione di area e volume utilizzando un sensore ToF

5.2. Possibili Miglioramenti del Sistema

Dopo l'acquisizione dei dati dal sensore, possiamo applicare metodi matematici e statistici o utilizzare tecniche di machine learning per migliorare ulteriormente la qualità dei dati acquisiti. Successivamente, comunicando i dati di alta qualità, possiamo implementare un

algoritmo per mappare l'area tramite il movimento del sensore, sfruttando il movimento di due motori. In questo modo, possiamo facilmente accedere a tutti i punti davanti al sensore, coprendo un angolo di 180 gradi, come una semisfera.

La parte meccanica del progetto e il cablaggio richiedono un lavoro significativo, combinando conoscenze di meccanica e design per renderlo più robusto.

6. Conclusioni e Lavori Futuri

L'obiettivo di questa tesi è dimostrare che, per molte esigenze operative, come la prevenzione delle collisioni e il rilevamento degli ostacoli, non è strettamente necessario ricorrere a costosi sensori LIDAR. È infatti possibile ottenere risultati soddisfacenti facendo affidamento su sensori a basso costo e a basso consumo energetico. Questo lavoro si propone di esplorare il potenziale di tali sensori alternativi, valutandone l'efficacia in scenari reali e confrontandone le prestazioni con quelle offerte da soluzioni tecnologiche.

La soluzione è stata testata in vari scenari, con applicazioni nell'automazione domestica e nella scansione ambientale. I risultati dimostrano che gli obiettivi sono stati raggiunti con successo, con il sistema che funziona efficacemente nei casi d'uso previsti.

Le ricerche future si concentreranno sul miglioramento del dataset e sull'ulteriore sperimentazione di diverse architetture. Inoltre, è prevista l'applicazione di tecniche di machine learning. Un altro obiettivo sarà l'integrazione di componenti elettronici aggiuntivi al fine di sviluppare un sistema embedded in grado di muoversi a distanza, simile a una mini-macchina robotica telecomandata.

Questo prototipo sarà pensato per operare in ambienti stretti o difficilmente accessibili, con lo scopo di rilevare ostacoli che bloccano il passaggio o compromettono la navigabilità dello spazio. Tali sviluppi potranno essere fondamentali per applicazioni in ambito esplorativo, di soccorso o manutentivo, dove l'autonomia e la compattezza del sistema rappresentano requisiti chiave.

7. Bibliografia

1. **Systems, EPIC.** *https://epicsysinc.com/*. [Online] <https://epicsysinc.com/blog/machine-vision-history-3/>.
2. **Alghasi, Arash.** Tesi. *cirkitdesigner*. [Online] cirkitdesigner. [Riportato: 01 02 2025.]
<https://app.cirkitdesigner.com/project/a39a4cca-3c85-4704-bdaa-b40c989628d8>.
3. **research, grand view.** *https://www.grandviewresearch.com/*. [Online]
<https://www.grandviewresearch.com/industry-analysis/3d-machine-vision-market>.
4. . Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. 2011.
5. **https://www.st.com/**. *https://www.st.com/*. [Online] STMicroelectronics . [Riportato:]
<https://www.st.com/en/evaluation-tools/vl53l5cx-satell.html>.
6. **STMicroelectronics.** Time-of-Flight 8x8 multizone ranging sensor with wide field of view.
STMicroelectronics. [Online] [Riportato: 30 01 2025.]
https://www.st.com/resource/en/data_brief/vl53l5cx.pdf.
7. —. Time-of-Flight (ToF) 8x8 multizone ranging sensor with wide field of view.
STMicroelectronics. [Online] [Riportato: 30 01 2025.] <https://www.st.com/en/imaging-and-photonics-solutions/vl53l5cx.html#documentation>.
8. —. STM32 Nucleo-64 development board with STM32F446RE MCU, supports Arduino and ST morpho connectivity. *STMicroelectronics*. [Online] [Riportato: 30 01 2025.]
<https://www.st.com/en/evaluation-tools/nucleo-f446re.html#documentation>.
9. —. High-performance foundation line, Arm Cortex-M4 core with DSP and FPU, 512 Kbytes of Flash memory, 180 MHz CPU, ART Accelerator, Dual QSPI. *STMicroelectronics*. [Online]
[Riportato: 30 01 2025.] <https://www.st.com/en/microcontrollers-microprocessors/stm32f446re.html#documentation>.
10. —. Bipolar stepper motor driver expansion board based on L6208 for STM32 Nucleo.
STMicroelectronics . [Online] [Riportato: 30 01 2025.] <https://www.st.com/en/ecosystems/x-nucleo-ihm05a1.html#documentation>.
11. **Modulo esp-01s esp8266 ricetrasmettitore wireless wifi remoto per Arduino.** *Mitomo informatica*. [Online] [Riportato: 30 01 2025.]

- <https://www.informaticait.it/elettronica/trasmettitore-rf-wireless/modulo-esp-01s-ricetrasmettitore-wireless-wifi.html>.
12. NRF24L01: modulo per comunicazione wireless per Arduino. *Hardwarelibre*. [Online] [Riportato: 31 01 2025.] <https://it.hwlible.com/nrf24l01/>.
13. company, Arduino. Arduino Uno Rev3. *Arduino*. [Online] [Riportato: 31 01 2025.] <https://store.arduino.cc/products/arduino-uno-rev3?srsltid=AfmBOoq1c5KfaP8X--YgCMMkIRU4dl1qYfNexCLzmS6EKqoOCPenQmgM>.
14. STMicroelectronics. Integrated Development Environment for STM32. *STMicroelectronics*. [Online] STMicroelectronics . [Riportato: 31 01 2025.] https://www.st.com/en/development-tools/stm32cubeide.html#st_all-features_sec-nav-tab.
15. wikipedia. PuTTY. *wikipedia*. [Online] [Riportato: 31 01 2025.] <https://en.wikipedia.org/wiki/PuTTY>.
16. MQTTX. *MQTTX*. [Online] MQTTX. [Riportato: 31 01 2025.] <https://mqtx.app/>.
17. STMicroelectronics. Bipolar stepper motor driver software expansion for STM32Cube. <https://www.st.com>. [Online] STMicroelectronics . [Riportato: 31 01 2025.] <https://www.st.com/en/embedded-software/x-cube-spn5.html>.
18. How can you overcome common challenges when cleaning sensor data? <https://www.linkedin.com/>. [Online] [Riportato: 02 02 2025.] <https://www.linkedin.com/advice/1/how-can-you-overcome-common-challenges-when-cleaning-igxvc>.
19. Hong, Zhong. How to Handle Missing Data and Outliers in Data Science. <https://medium.com/>. [Online] [Riportato: 02 02 2025.] <https://medium.com/@zhonghong9998/how-to-handle-missing-data-and-outliers-in-data-science-23af1150eac7>.
20. streamlit. *streamlit*. [Online] [Riportato: 02 02 2025.] <https://streamlit.io/>.
21. Electronics, STM. <https://www.st.com>. *STMicroelectronics*. [Online] STMicroelectronics . [Riportato:] https://www.st.com/content/st_com/en/about/media-center/press-item.html/t4210.html.
22. STMicroelectronics amplia l'ecosistema STM32Cube con l'IDE gratuito STM32CubeIDE. *Elettronica In*. [Online] [Riportato: 31 01 2025.] <https://ei.futuranet.it/2019/04/27/stmicroelectronics-amplia-ecosistema-stm32cube-con-ide-gratuito-stm32cubeide/>.

23. STMicroelectronics. Time-of-Flight sensors software expansion for STM32Cube.
<https://www.st.com/>. [Online] STMicroelectronics NV. [Riportato: 31 01 2025.]
<https://www.st.com/en/ecosystems/x-cube-tof1.html>.
24. —. STM32Cube MCU Package for STM32F4 series (HAL, Low-Layer APIs and CMSIS, USB, TCP/IP, File system, RTOS, Graphic - and examples running on ST boards). *STMicroelectronics* . [Online] STMicroelectronics . [Riportato: 31 01 2025.] <https://www.st.com/en/embedded-software/stm32cubef4.html#overview>.
25. *Why Is Data Visualization Important? What Is Important in Data Visualization?* Unwin, Antony. <https://doi.org/10.1162/99608f92.8ae4d525>,
26. Wei, W. Luo and W. *A low-cost high-resolution lidar system ...* s.l. : IEEE Transactions on Instrumentation and, 2022.