

DESIGN AND IMPLEMENTATION OF AN EXPERIMENTAL MODEL OF COMPUTERIZED LIDAR MEASUREMENT SYSTEM FOR OBSTACLE DETECTION

ENACHE Andrei-George

Facultatea: Inginerie Industrială și robotică , Specializarea: Informatică Aplicată în Inginerie Industrială,
Anul de studii: IV, e-mail: enache.andrei.george@gmail.com

Conducător științific: Prof. Dr. Ing. **Savu TOM**

ABSTRACT: In today's activities, more and more tasks usually performed by humans are being replaced by automated or even autonomous counterparts that are specialized on that specific domain and so can do it better, faster and without delay. A common usage of such an automation is the usage of AGVs in industrial enterprises. In this paper will be presented the first steps of an experimental model that can read the data received by a LIDAR sensor preponderantly used in automated or autonomous vehicles. The data received will then be processed and various computerized simulations of the real scanning will be made in order to estimate the current position of the real sensor in a premade map, considering the X/Y coordinates and the angles of each point from the real scan.

KEY WORDS: LIDAR, obstacle detection and avoidance

1. Introducere

LIDAR-ul este un senzor utilizat pentru măsurarea distanței până la elementele mediului înconjurător, utilizând lumina infraroșie cu lungimea de undă variabilă între 600 mm și 1500 mm. Principalele metode de emisie a luminii pentru scanarea perimetrului sunt scanarea cu undă laser și citire cu fotodiodă receptoare, flash luminos și panouri de arseniură de galiu, sau rețea de fază și citire cu ajutorul unui chip.

Din cauza costului crescut al ultimelor două variante constructive, cei mai mulți utilizatori ai sistemelor LIDAR aleg varianta cu raze laser, aceasta fiind și cea mai răspândită pe piață. Pentru măsurarea distanțelor, aceștia utilizează “Timpul de zbor” (“Time Of Flight”), ce constă în determinarea timpului ca fasciculul de lumină emis să se întoarcă la senzorul infraroșu al LIDAR-ului, distanța fiind calculată în funcție de viteza luminii.

În funcție de modul în care acești senzori citesc și afișează informațiile, ei se împart în sisteme 1D (un punct), 2D (linii și perimetre formate din puncte, Fig.1.) sau 3D (mai multe nivele de straturi de linii sau perimetre formate din puncte, Fig.2.).

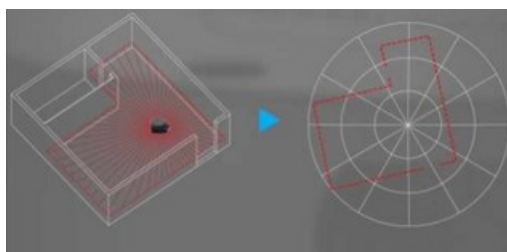


Fig. 1. Model grafic de citire LIDAR 2D la 360°



Fig. 2. Model grafic de citire LIDAR 3D la 360°

În cadrul acestei lucrări s-a urmărit dezvoltarea unui sistem software capabil de a citi și interpreta datele primite de la senzorul LIDAR prin intermediul mediului de programare LabVIEW, aceste date fiind utilizate ulterior pentru diverse prelucrări.

Pentru ca sistemul nostru să poată recunoaște elementele mediului înconjurător, acestuia i-a fost determinată o hartă prestabilită în cadrul căreia se vor face simulări de scanare ce vor fi comparate cu sistemul real pentru a putea localiza poziția actuală a acestuia pentru un ghidaj ulterior. Punctele simulate vor fi preluate la același unghi ca și al punctelor reale scanate, putând fi schimbat chiar și punctul frontal (punctul 0) al LIDAR-ului simulat pentru o acuratețe crescută. Pentru scanarea perimetrului a fost utilizat senzorul LIDAR YDLIDAR X4.

2. Stadiul actual

Una dintre cele mai mari zone de utilizare ale sistemelor tip LIDAR este reprezentată de sistemele de transport automate. În domeniul industrial, astfel de sisteme sunt constituite din Vehiculele Ghidate Automate (AGV), acestea fiind niste roboți mobili ce pot transporta bunuri în diverse locații, fără a necesita intervenția omului pentru efectuarea sarcinilor de deplasare. În acest mod se pot elimina erorile umane și timpii morți prin intermediul a diferite software-uri de management al flotelor de roboți, aceștia putând de asemenea lucra o perioadă mai lungă de timp, chiar și noaptea, având nevoie doar de o reîncărcare periodică.

În momentul actual, AGV-urile industriale sunt dotate cu un senzor LIDAR pentru detectarea obstacolelor la distanțe critice (putând astfel declanșa frâna de urgență pentru a nu cauza un accident), sau doi senzori în cazul sistemelor AMR (Autonomous Mobile Robots), acestea utilizând senzorii nu doar pentru detectarea coliziunii, ci și pentru scanarea mediului înconjurător în scopul localizării poziției proprii pe hartă. Spre deosebire de AGV-urile clasice, AMR-urile nu au nevoie de evidențiatori externi tip benzi sau etichete pentru detectarea poziției proprii în spațiu, acestea folosind senzori de distanță, iar comparând datele primite de la scanarea perimetrului cu harta reală își pot estima locația și direcția prin intermediul algoritmilor și diferitelor programe software.

3. Algoritmul de comandă al senzorului

În prima etapă a procesului a fost necesară crearea unui program central de comandă al senzorului LIDAR, acesta efectuând diverse funcții, depinzând de datele în format hexadecimale primite de la utilizator

Principalele funcții ale sistemului sunt [1]:

- A5 60 – Comandă de scanare;
- A5 65 – Comandă de oprire;
- A5 90 – Informații ale dispozitivului;
- A5 40 – Comandă de restart;

4. Algoritmul de procesare a datelor

În a doua etapă a procesului, pentru minimizarea timpului necesar procesării, a fost necesară crearea algoritmului de recepționare și citire a datelor de la LIDAR prin intermediul unui singur mediu de programare (LabVIEW). În mod uzual, pentru acest tip de senzor poate fi descărcată o aplicație proprie creată de producător, în cadrul acesteia putându-se vedea scanările în timp real, dar acest proces ar necesita timp și putere de calcul suplimentare la legarea programului central.

Datele citite de sistem ajung către utilizator în format hexadecimal, acestea venind sub forma a doi octeți consecutivi citiți cu sistemul little-endian (cel mai nesemnificativ byte pe poziția cea mai mică). În acest sistem de citire, un element de tipul 0x55AA va fi recepționat inițial ca AA 55.

Pentru crearea algoritmului a fost necesară crearea protocolului de citire conform fișei tehnice, aceasta fiind făcută în funcție de comanda de scanare a LIDAR-ului.



Fig. 3. Protocolul de primire a datelor după comanda de scanare [A5 60]

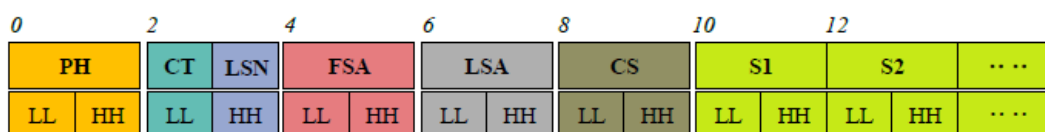


Fig. 4. Conținutul structurii "Content" din comanda de scanare de la Fig. 3.

Tabel 1. Elementele pachetelor de date obținute cu comanda de scanare A5 60

Continut	Nume	Descriere
PH(2B)	Capăt pachet	Lungimea este 2B, fixată 0x55AA, valorile "Low" în față, valorile "High" în spate
CT(1B)	Tip pachet	Indică pachetul curent, 0x00: pachet de nor de puncte, 0x01: Pachetul zero
LSN(1B)	Cantitate de date scanate	Indica numărul de puncte scanate conținute de pachetul curent; Există doar un punct zero în pachetul zero, valoarea lui fiind 1
FSA(2B)	Unghi de început	Unghiul corespunzător primei date citite din pachet
LSA(2B)	Unghi de sfârșit	Unghiul corespunzător ultimei date citite din pachet
CS(2B)	Cod de verificare	Codul de verificare al pachetului curent
Si(2B)	Date scanate	Datele obținute de sistem sunt date de distanță până la punctul scanat

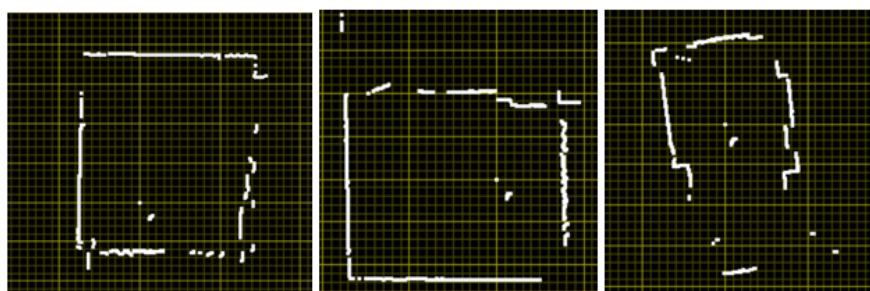


Fig. 5. Date reprezentate grafic, obținute din scanarea a două camere ale clădirii

Pentru realizarea acestui algoritm au fost necesare datele inițiale primite de la LIDAR sub forma unor pachete de diferite dimensiuni. Pentru realizarea hărții reale scanate a fost necesară stocarea și afișarea datelor după fiecare pachet inițial tip 0x01, după fiecare scanare pachetul anterior fiind șters din program pentru eliberarea memoriei.

Datele necesare acestei procesări au fost cele prezentate în Tabelul 1, datele de ieșire ale programului fiind un mănunchi (cluster) de puncte în coordonate X și Y, dar și unghiurile acestor puncte.

5. Algoritm de simulare a scanării

A treia etapă în realizarea sistemului a fost crearea programului pentru simularea unei scanări reale a unei hărți prestabilite, aceasta fiind creată cu ajutorul unor puncte în coordonate relative. Datele de intrare ale programului sunt unghiurile ce urmează a fi scanate (acestea sunt luate în raport cu unghiurile scanării reale pentru a facilita etapele următoare ale programului), punctele X și Y ale poziției simulate a senzorului pe hartă, lungimea razei de scanare și unghiul Fi ce reprezintă rotația poziției frontale a LIDAR-ului (0° reprezintă traiectoria verticală cu direcția în sus, acest unghi mărindu-se în sens trigonometric pana ce ajunge din nou în punctul initial la 360°). În acest mod, putem observa cum robotul poate obține datele din mediul simulat la aceleași unghiuri ca punctele mediului real, putând modifica orientarea razei inițiale de scanare pentru a obține o rotire a datelor senzorului.

Pentru realizarea simulării a fost necesară crearea unui alt subprogram ce calculează punctul de intersecție a două drepte[2]. Datele de intrare a acestui subprogram sunt punctele de capăt ale razei de scanare simulate și fiecare latură a perimetrului obținut din coordonatele relative. Astfel, la fiecare unghi initial scanat, raza se va intersecta cu diferite laturi ale perimetrului, alegând-o pe cea mai apropiată, asemeni unei scanări reale.

Datele de ieșire ale algoritmului de simulare a scanării sunt reprezentate de un cluster de puncte în coordonate X și Y ce pot fi comparate cu punctele reale. Punctul alb reprezinta poziția simulată a senzorului pe hartă.

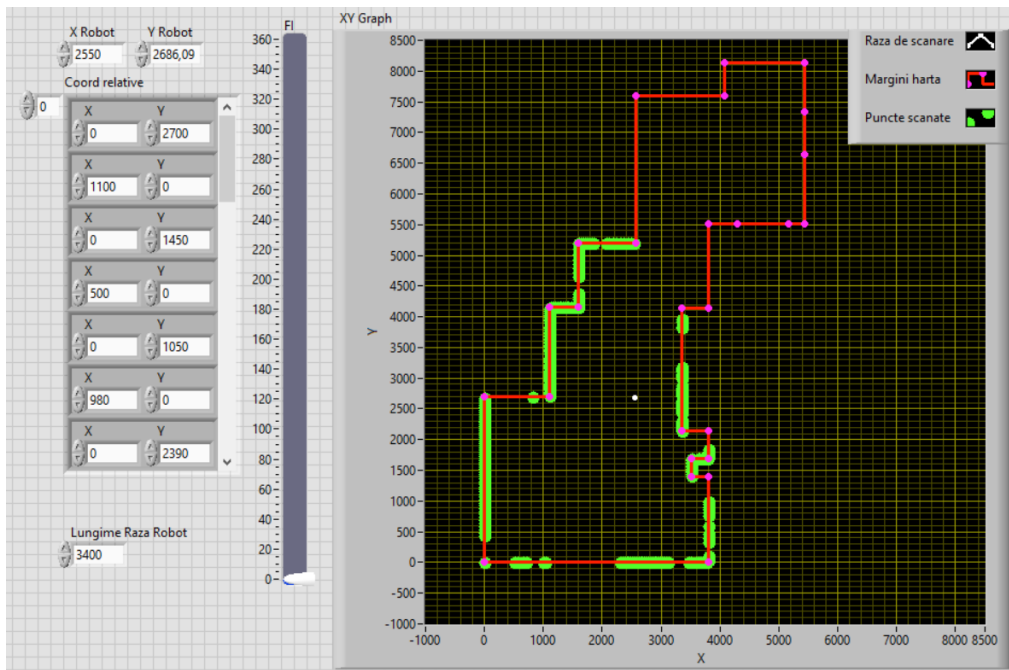


Fig. 6. Simulare de scanare în funcție de unghiurile reale de scanare

6. Algoritm de comparare a graficelor

În următoarea etapă a procesului a fost nevoie de prelucrarea datelor obținute de la programele anterioare. Având ca date de intrare coordonatele X și Y a punctelor reale scanate, acest program va obține unghiurile necesare ce urmează a fi scanate. Punctele de coordonate 0

sau -0 au fost eliminate, acestea reprezentând erori aberante ce ar fi crescut șansa erorilor ulterioare în timpul rulării.

În prima variantă a programului s-a încercat rotirea punctelor reale pentru a se potrivi cu cele simulate, acest lucru îngreunând însă calculele ulterioare necesare, astfel unghiul de înclinație al acestor puncte va fi lăsat la zero.

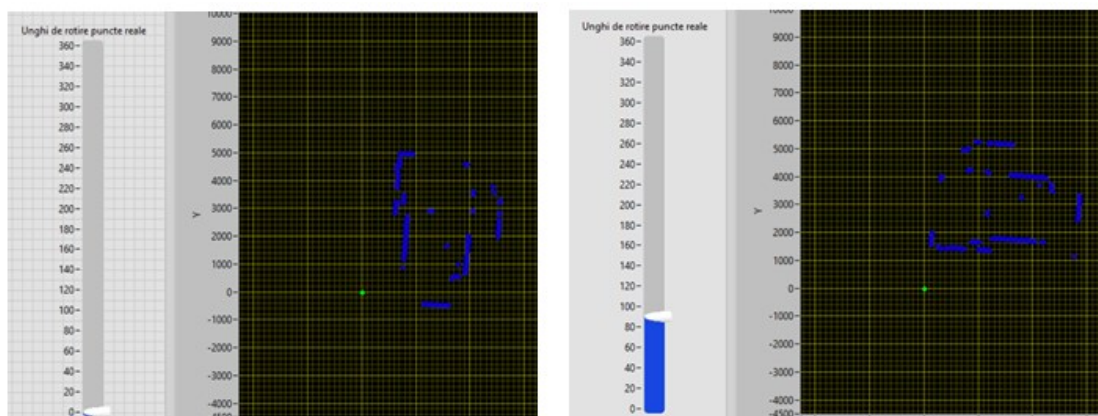


Fig. 7. Vedere a punctelor scanate cu unghiul de rotație 0°, respective 90°

Pentru a putea estima poziția zonei senzorului simulat pe hartă în funcție de scanările reale a fost utilizată metoda sumei pătratelor abaterilor dintre punctele simulate și punctele reale. Suma astfel obținută va fi împărțită la numărul total de puncte existente în grafic.

Pentru a putea obține rezultate cât mai coerente în urma analizei acestei sume este necesară realizarea comparației punctelor aflate la același unghi.

În cadrul programului, prin creșterea sau scăderea unghiului θ_i al razei de scanare se obține o rotire teoretică a zonei frontale a LIDAR-ului. Pentru ca realizarea diferenței punctelor menționată în paragraful anterior să fie posibilă, unghiul θ_i va fi adăugat unghiurilor de scanare inițiale pentru a obține o “vedere față” pe o poziție diferită, acest unghi fiind scăzut ulterior pentru a se reveni la unghiurile inițiale scanate.

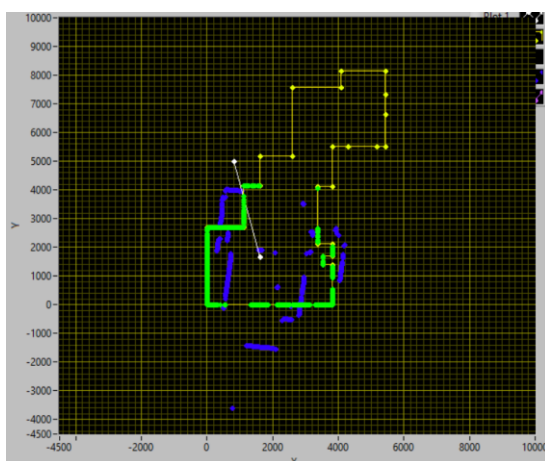


Fig. 8. Scanarea simulată și cea reală pe harta înregistrată

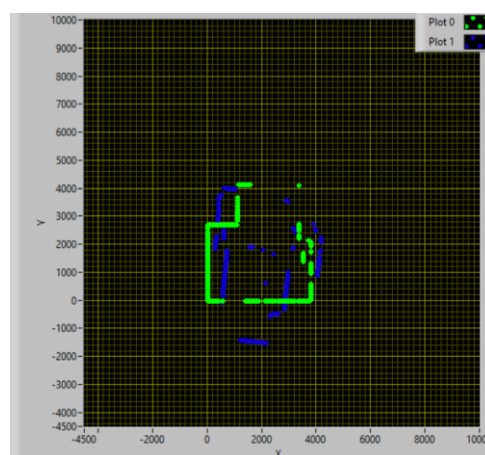


Fig. 9. Suprapunerea scanării simulate (verde) pe cea reală (albastru)

În cadrul Fig.8, raza albă ce reprezintă raza de scanare nu este paralelă cu direcția orizontală datorită unei deviații ale punctelor de scanare reale, acestea începând de la 1,74 radiani (aproximativ 100°).

Utilizând cele două grafice suprapuse se poate calcula suma pătratelor abaterilor obținute din diferența distanței dintre punctul senzorului și punctul scanat real la un anumit grad și distanța până la un punct simulat aflat la același grad. Pentru evitarea erorilor grosolane s-a evitat adunarea diferenței în cazul în care raza de scanare simulată la un anumit punct nu ar fi întâlnit niciun perete (în acest caz dând valoarea zero ca și coordonate XY ale punctului de intersecție).

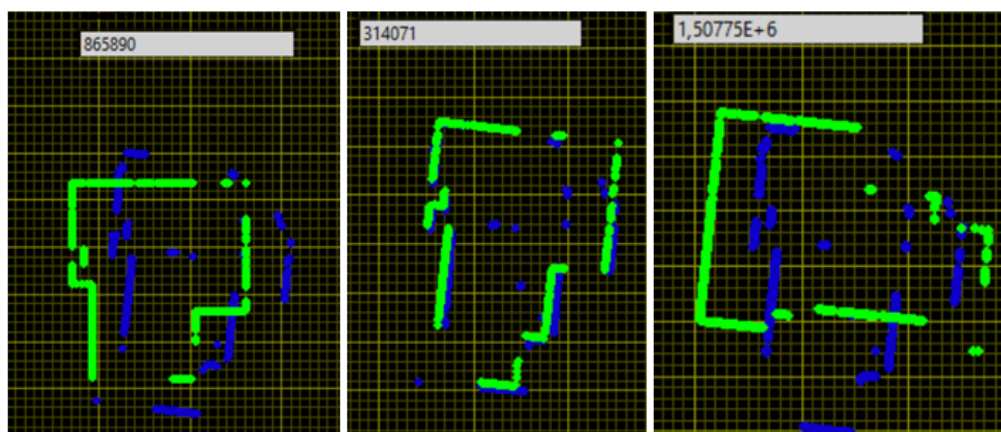


Fig. 10. Valori ale sumei pătratelor abaterilor la diferite unghiuri și poziții ale senzorului

7. Concluzii și îmbunătățiri

În concluzie se poate observa o valoare foarte mare a sumei pătratelor abaterilor, acest lucru fiind datorat faptului că măsurătorile se fac în milimetri, fiecare deviație crescând exponențial suma totală. Se poate observa de asemenea cum LIDAR-ul oferă rezultate satisfăcătoare în raport cu prețul de achiziționare relativ mic, având o claritate bună a datelor scanate.

În următoarele etape ale proiectului vor fi realizate părțile sistemului ce deservește la stabilirea estimativă a locației și unghiului de orientare al senzorului, acest lucru va fi făcut prin implementarea algoritmului Hooke-Jeeves de optimizare a căutării de modele, după aceea trebuind de asemenea să se realizeze o comparație între algoritmi deja existenți pentru stabilirea traiectoriei (RRT, RRT*, A*, D*).

Odată aceste programe realizate, programul va putea fi inserat pe o placă de procesare MyRIO capabilă să controleze motoarele aferente roților unui vehicul ce va fi trimis într-un punct oarecare X, ocolind obstacolele din drum.

8. Bibliografie

- [1]. YDLIDAR X4 Development Manual
- [2]. "Weisstein, Eric W. "Line-Line Intersection." From MathWorld". A Wolfram Web Resource. Retrieved 2008-01-10