УДК 004.896
МРНТИ 28.23.27

https://doi.org/10.55452/1998-6688-2023-20-4-27-39

**[1]MUHAMMAD ILYAS, [1]ABDULLAH AAMIR HAYAT,
[1]PRABAKARAN VEERAJAGADHESWAR, [1]MOHAN RAJESH ELARA**
[1]ROAR Lab, Engineering Product Development Pillar,
Singapore University of Technology and Design (SUTD), Singapore 487372
*E-mail: milyasmeo2005@gmail.com

3D ABSOLUTE POSE ESTIMATION OF A STAIRCASE CLEANING ROBOT UTILIZING STAIRCASE GEOMETRY AND MINIMAL LOW-COST SENSORS

Abstract

Thanks to its shape-shifting abilities, the sTetris robot can climb stairs while performing the cleaning task. The sTetris robot's overall system operation depends on localization and positioning data, which are essential for its goal of autonomously navigating multi-floor environments. The mobile robots designed to work indoors generally rely on external systems for localization information. Regretfully, this frequently requires additional hardware fixing or changes to the indoor working environment in order to achieve accurate three-dimensional (3D) position and orientation (pose) for successful operation of the mobile platform. Nonetheless, the robot can be localized on the staircase by utilizing the information of the staircase's geometry measurements, which are known ahead of time. This article demonstrates how the known geometry of staircases and measurements from minimal number of sensors can be used to accomplish 3D pose of the robot. Experiments carried out on a real robot in an authentic indoors setting successfully demonstrate the effectiveness of the suggested approach.

Key words: Indoor environment; three-dimensional localization; staircase geometry; reconfigurable robot; low-cost sensors.

Introduction

While many systems and subsystems are necessary for any mobile robotic system to function properly, three essential modules–the path planner, and the localization and positioning and the control system modules–are required to enable the autonomous functionality of the mobile robots [1, 2]. Of them, the localization module is responsible for giving the mobile platform's precise position and orientation information [3]. The accuracy of the localization system has a big impact on how well the other important modules of a robotic system work.

Absolute positioning systems and incremental/relative positioning systems are the two main categories of localization/positioning systems [4, 5]. When a system is used for absolute positioning, such as GPS, the position data is accessible in relation to a global reference coordinate frame and often is not dependent on space or time. On the other hand, the position update in an incremental or relative positioning system comes from the steps that came before it. The disadvantage of the latter is that pose drifts with time, such as inertial measurement unit and wheel encoder-based position calculation, and sensor error is also aggregated, resulting in mistakes in the pose estimation [8, 9].

Literature Review

When it comes to outdoor mobile robots, GPS is the primary source of localization data, either by itself or in conjunction with other sensors or systems like an IMU, camera, Lidar, vision sensors, etc. [10, 11]. Since GPS is not always available for interior mobile robotic platforms, alternative methods of absolute pose estimation are applied, such as estimating the absolute position of mobile devices using WiFi signals, UWB, vision, RFID signals, or indoor Global Positioning System, for instance [12]. To determine a mobile platform's exact location inside a building, various beaconbased absolute positioning methods are used [3, 6, 13–18]. Nevertheless, this necessitates a few more hardware adjustments in the workspace, which raises the system's overall cost and/or complexity.

In this study, we introduce a novel yet straightforward technique for estimating the three-dimensional absolute pose (position and orientation) information of an indoor mobile robot–dubbed sTetris–developed at SUTD's RoAR Laboratory using a limited number of low-cost sensors [19]. The location of the sensors and the robot's typical working environment are depicted in Figure 1. Using low-cost onboard sensor measurements, the previously known staircase geometry is utilized to estimate localization information because the sTetris robot operates in a staircase with fixed and known tread and riser dimensions for each step.

Figure 2 (p. 29) provides a list of some terminology pertaining to staircase geometry. The robot has two time-of-flight (TF) sensors mounted on its left and right sides. The y-axis position on the tread is determined using the distance measurements collected from these two TF sensors. The x- and z-axis positions are determined by the known staircase geometry as the robot ascends the stair steps because the tread and riser dimensions are known a prior. Furthermore, the robot has two TF sensors positioned in front of it, oriented toward the front riser. The heading angle of the sTetris robot is determined by utilizing the range/distance measurements from frontal sensors as robot moves left and right on a step of stair.

The remaining contents of this article are arranged as follows: An overview of the sTetris robot is given in the second section. The sTetris 3D localization computation method is shown in the third section, and the orientation estimation scheme is introduced in the fourth. The experimental setup and findings are covered in the fifth section. The final section wraps up this work and suggests some future lines of inquiry for additional study and advancement.

Materials and Methods

sTetris: Introduction to Staircase Cleaning Robot

The sTetris is a re-configurable robot that climbs stairs using a vertical conveyor belt mechanism, as shown in Figure 1. This robot is designed to clean staircases and inspired from the famous Tetris game [20] and the transformation design ideas described in [21] are the sources of sTetris operating principle. The three cuboids that make up the sTetris robot's body are joined by two sliders that are fastened to either side of the cuboid in the center. The sweeping mechanism, vacuum/suction tools, and electronic units are enclosed in the hollow spaces inside each parallelepiped block. Because of the modular design of the blocks, many system components can be reused. The robot's movement can be customized by the user with the help of a graphical user interface (GUI). With the help of the reconfiguration mechanism, the sTetris robot can move quickly across the multi-floor environment which is impossible with existing commercial home cleaning robots.

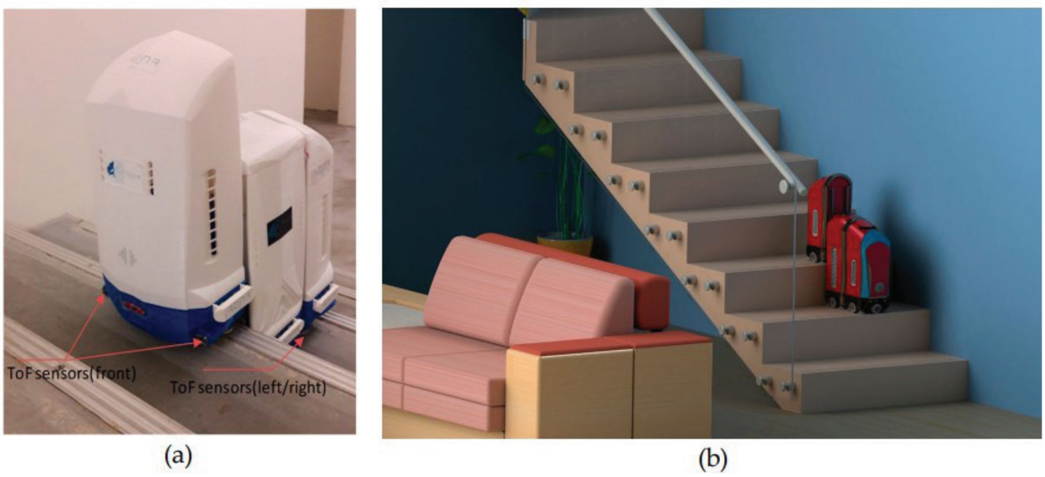

Figure 1 – sTetris Robot: a) equipped with sensors specifically designed for this task;
b) robot's operating environment.

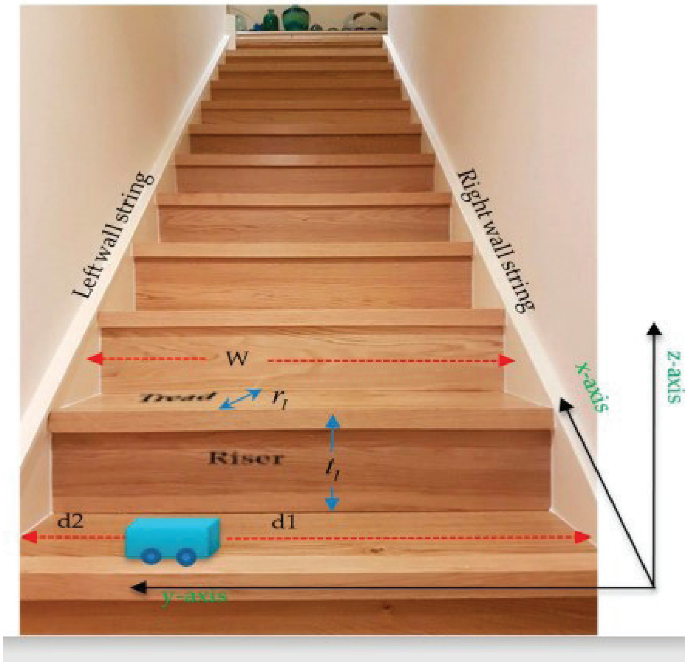

Figure 2 – Geometry of staircase to determine the robot's 3D position and orientation.

Absolute 3D Position of sTetris

In this study, TF sensors are used to measure the y-axis (lateral) position in order to obtain 3D absolute position information. The staircase's known geometry is used to calculate the position along the other two axes, i.e. x-axis and z-axis. For example, the riser height and the stair step's tread width are fixed and known of a staircase and can measurable manually or from 3D drawing. If the step number on a staircase is known, it is possible to calculate the distance traveled along the x- and z-axes as the robot climbs the staircase. Equation (1) is used to calculate the sTetris 3D absolute position,

$$\begin{aligned} x_i &= t_l \times \text{ step }_i - \left(\frac{t_l}{2}\right) \\ z_i &= r_l \times \text{ step }_i \quad ; i = 1,2, \ldots., N. \\ y_i &= \begin{cases} d_j & ; \text{ if sensor output } < 120 \text{ cm} \\ W - d_j & ; \text{ otherwise} \end{cases} \end{aligned} \tag{1}$$

where W is the staircase's width, i is the step number counter, t_l and r_l are the staircase's tread and riser lengths, x_i and z_i are their respective positions in the x- and z-axes, and d_j is the y-axis's TF sensor measurement.

The robot's position along the y-axis is calculated using the data from distance sensors positioned at its two sides. Two Time-of-Flight sensors are positioned on the left and right sides of sTetris because their limited range (120 cm) makes it impossible for the sensors to reliably measure distance when the robot is on the extreme left or right side of the stair. In this instance, the position on the stair along the y-axis is determined using the output from the sensor on the opposite side of the robot. Algorithm I below provide the full pseudo-code for estimating 3D pose i.e. position and orientation (heading) of the robot.

Algorithm I: Pseudo code for 3D localization and heading estimation of *sTetro.*

Initializ: $x_0 \leftarrow 0, y_0 \leftarrow Y_0, z_0 \leftarrow 0$

$i = 1$ *//step number*

Loop : while $(i \leq N)$

$x_k \leftarrow t_l \times (i) - 0.5 t_l$

$z_k \leftarrow r_l \times (i)$

if $(sensor\ output < limitMax)$ $//limitMax = 120cm$

$y_k \leftarrow d_k$

else

$y_k \leftarrow W - d_k$

$\theta_k \leftarrow \tan^{-1}\left(\frac{d_2 - d_1}{D}\right)$

if $(\theta_k > 0)$

$rotateCW()$

if $(\theta_k < 0)$

$rotateCCW()$

$i \leftarrow readStep()$

end loop

Main provision

Orientation Calculation and Misalignment Correction

Since the staircase in which sTetris operates has nearly flat treads, during normal robot operation, sTetris roll and pitch angles are almost zero. When going left or right on the treads, only the heading angle (rotation about the z-axis) can vary. The sTetris robot's front end is equipped with two TF sensors to monitor the heading angle. These two sensors' job is to use their respective range measurements to determine the robot's heading angle. Direct distance measurements are provided by time-of-flight sensors in relation to the step's front riser. Equation (2) can be used to calculate the heading angle.

$$\theta_k = \tan^{-1}\left(\frac{d_2 - d_1}{D}\right) \quad (2)$$

where D is the sTetris robot's width and d_1 and d_2 are distance readings from TF sensors positioned in front of the robot. The robot is aligned with the front riser and the heading angle is almost zero if the two distances are almost equal. A positive or negative heading angle indicates the robot's misalignment with the front riser if the two range measurements are not equal. Then, in order to maintain sTetris' alignment with the front riser for smooth motion, the control system tells actuators to rotate it either clockwise (CW) or counterclockwise (CCW). Figure 3 below depicts the heading angle estimation and correction scheme used in this work.

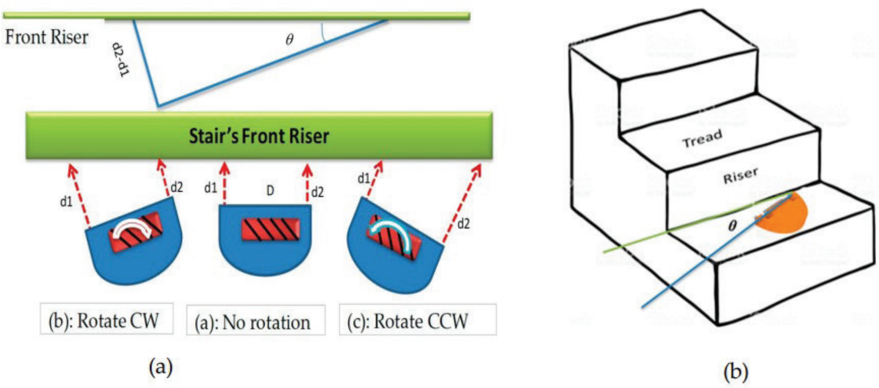

Figure 3 – a) Computation of heading angle based on Time-of-Flight sensor reading;
b) show how robot may go misaligned on a stair tread.

Table 1 – Staircase geometry and sTetris main parameters

Parameters	Value (cm)	Remarks
Riser height (rl)	14	
Tread length (tl)	32	
ToF's Reliable range (max)	120	
sTetro size (1, w, h)	45 ×20 × 40	
No. of steps (N)	16	
Staircase size	480 ×205 × 165	(L ×W × H)

We have fully detailed the design and operation of the sTetris robot in our earlier work [19]. The staircase cleaning robot is outfitted with four extra TF sensors for the experiments carried out in this work. Up to two meters can be measured absolutely by the distance sensor (VL53L0X). Figure 4 depicts an experimental setup used for this work.

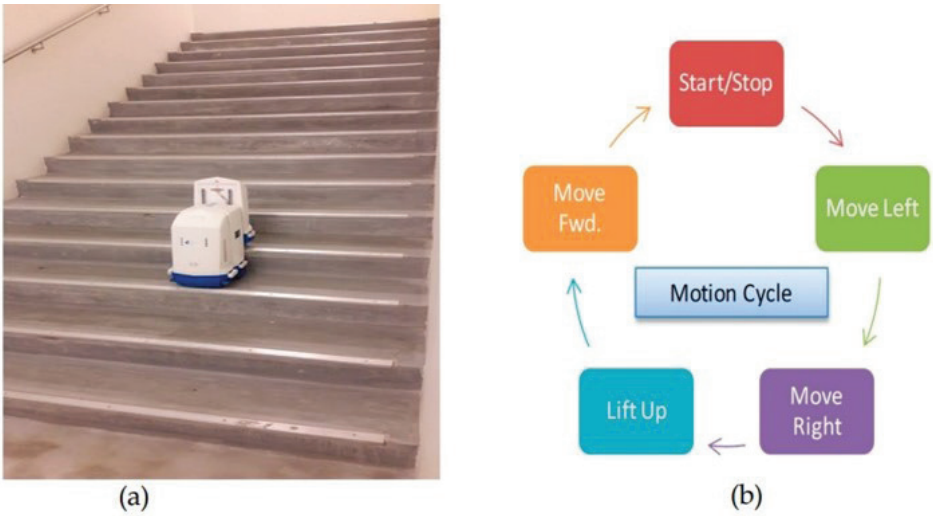

Figure 4 – Experimental setup: a) sTetris robot moving across the staircase;
b) A single motion cycle of sTetris.

The sTetris robot is propelled to move across the stairwell, as seen in Figure 4(a). The motion cycle for finishing a single stair step is depicted in Figure 4(b). The left and right wall strings limit the robot's range of motion. The first block of sTetris has two TF sensors in front of it, and the central block has two TF sensors on either side of it. The robot started to follow the tread of the step from right to left.

The robot moves toward the right on the tread after reaching the far left of the tread, where its left bumper makes contact with the left wall string. The robot's right bumper touches the right wall string when it reaches the farthest right side. It stops when it reaches the halfway point of the tread and raises the final and central blocks to ascend to the next level. Every time the robot moves up to the next step, the software increments a counter, counting the steps. The first step of the stairs is followed by a repetition of this motion cycle on the second step and so on. All four sensors measure the distances to the front riser of the staircase and the left and right wall strings during the left and right motion. These measurements are used to determine the robot's y-position and heading angle, respectively.

Results and Discussion

Static Test: Heading Angle:

The heading angle of the robot in the first test is determined using the range data (d_1, d_2) from the two front sensors. A static test is conducted initially to evaluate the suggested method's correctness. The robot rotates continuously from $+45^0$ to -45^0 about z-axis with its back away from the stair's front riser. Using an angle protector, straight lines are indicated at 0^0, $+45^0$, and -45^0 for the specified angles.

Beginning with the robot facing the riser (i.e., heading angle 0°), it is rotated leftwards i.e. CCW, increasing the left sensor's d_1 distance in relation to the right sensor's d_2 distance. From 0^0 to -45^0, the heading angle begins to increase. The robot is held motionless for some time at around -45^0 before rotating back in a CW direction. Now that d_1 is starting to decrease, the heading angle measures roughly 0° when both distances (d_1, d_2) are almost equal once more at zero degrees. After then, the robot is spun in a CW direction. As a result, the heading angle increases to -45^0 because distance d_1 is now increasing faster than d_2.

A rotation cycle is completed when the robot rotates leftward towards zero degrees after being maintained still for a moment at -45^0. Figures 5 present the findings from this static experiment. The heading estimation results obtained with TF sensors are compared with a commercially available Attitude Heading Reference System (AHRS) sensor from VectorNav Inc. This sensor has an absolute heading angle precision of 2^0 RMS [22]. A zoom-in view of the heading angle calculation at various rotation stages is provided in Figure 5-ii (a-d).

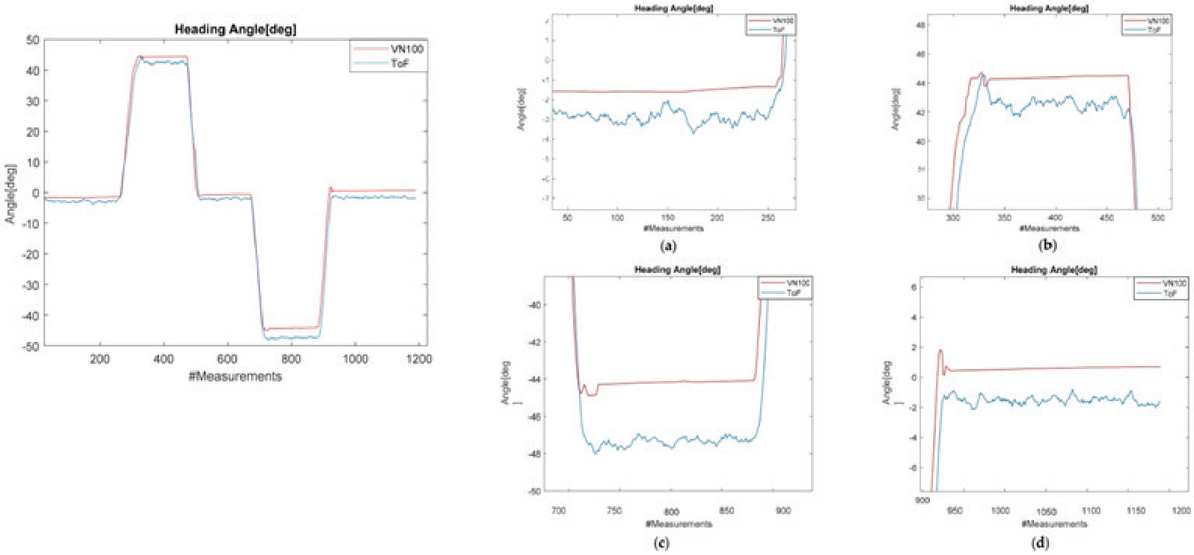

Figure 5 – i) Heading angle comparison with reference sensor: AHRS VN100: 0^0 to $+45^0$, back to 0^0 to -45^0, ii) Zoom-in view of heading angle approximately at 0^0, $+45^0$, and -45^0

Comparing the angle measurements from the reference AHRS with those from the suggested approach yields the heading angle error, as Table 2 below illustrates. A little rotation about the z-axis is observed to improve the accuracy of the heading angle calculation using the proposed method. When employing TF sensors, the heading angle error is more noticeable at greater rotation degrees (about +45°). The cause of the increased heading calculation errors for large rotation degrees along the vertical axis is due to the fact that at oblique angles, the inaccuracy in range readings from TF sensors grows.

Dynamic Test: 3D Position and Orientation:

Through the use of the staircase geometry and only onboard TF range sensors, the 3D localization of sTetris robot is established. Certain parameters of the TF sensors and the sTetris robot, as well as the shape of the staircase, are given in Table 1. A reliable range of only approximately 120 cm is measured between the left and right TF sensors. The range measurement data in Figure 6(a) confirms that the range data is unreliable for calculating location above this point in the experiment.

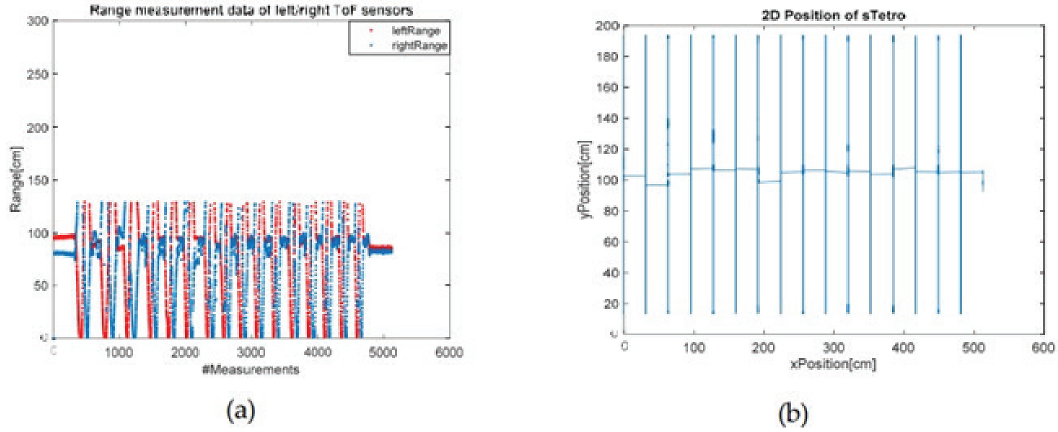

Figure 6 – a) Range measurements of the left and right TF sensors. b) sTetris 2D position (x, y).

The robot is initially placed at ($x_0 = 0, y_0 = Y_0, z_0 = 0$), which is roughly halfway up the first step of the staircase. The robot moves rightwards until right bumper hits the right string of staircase, maintaining a constant x- and z-position ($x = const., y = d_j, z = const.$ throughout rightward mobility) based on distance readings from the right-side TF sensor. The robot first touches the wall on the right before advancing leftward till its left bumper makes contact with the staircase's left wall string. Subsequently, the robot returns to the center of the staircase. With this, robot's one traversal on the stair step is finished.

In order to reach the next level, the robot then begins to raise its second and third blocks. When it is lifted up, its z-position shifts up but its y- and x-positions stay the same (i.e., ($x = const., y = const., z =$:). Once the lift-up motion is finished, the robot advances to the tread's center to begin the subsequent stair step traverse. Its x-position changes as it moves forward, but its y- and z-positions stay the same ($x = x_i, y = const., z = cons$).

Figure 6 (a) displays the range measurements of the left and right TF sensors (d_3, d_4). After roughly 120 cm, these range measurands lose their usefulness. This is due to maximum range of the sensor used. The robot's y-position on the staircase is determined using measurements from the left and right range sensors. Figure 6 (b) displays the sTetris 2D position (x, y). When excluding robot width from the y-axis distance calculation, the robot travels approximately 480 cm in the x-axis and 180 cm in the y-axis. In Figure 7, the 3D position plot is displayed.

The left and right TF sensors' raw range readings (d_3, d_4) are displayed in Figure 6(a). Calculating the robot's y-position on the stairs requires range measurements from the both sensors installed on right and left of sTetris. See Figure 6(b) for the 2D position (x, y) of sTetris. When y-axis distance is calculated without accounting for robot width, the robot moves approximately 180 cm in the y-axis and 480 cm in the x-axis direction. As seen in 3D position plot given in Figure 7 (p. 34).

When comparing the 3-axis position data with the staircase's real geometry (dimensions), it is clear that the data is extremely smooth and accurate. As seen in this picture, the robot travels around 165 cm along the z-axis. Figure 8 (p. 34) displays the estimated heading angle. The observation is made that the absolute orientation stays within +1° for the whole staircase. The heading error's computed standard deviation is roughly 0.9 degrees. In this experiment, the misalignment correction (control) is not used.

Error analysis

Error in the sensor's own range measurements is the cause of heading angle error. It is seen that the heading angle inaccuracy is around +1°, with a range error of approximately +3cm. In this work, we conducted studies to compare heading angle accuracy with the commercially available AHRS sensor. Figure 8 shows the absolute heading angle estimation using the suggested scheme and its error with respect to the reference AHRS sensor.

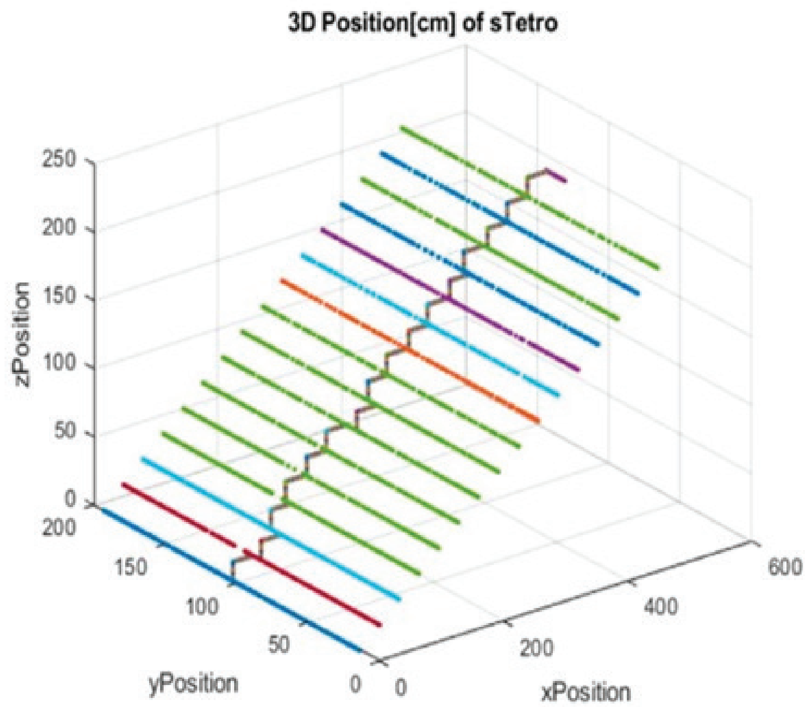

Figure 7 – Three dimensional absolute position (x,y,z) plot
of sTetris on a 16-step staircase

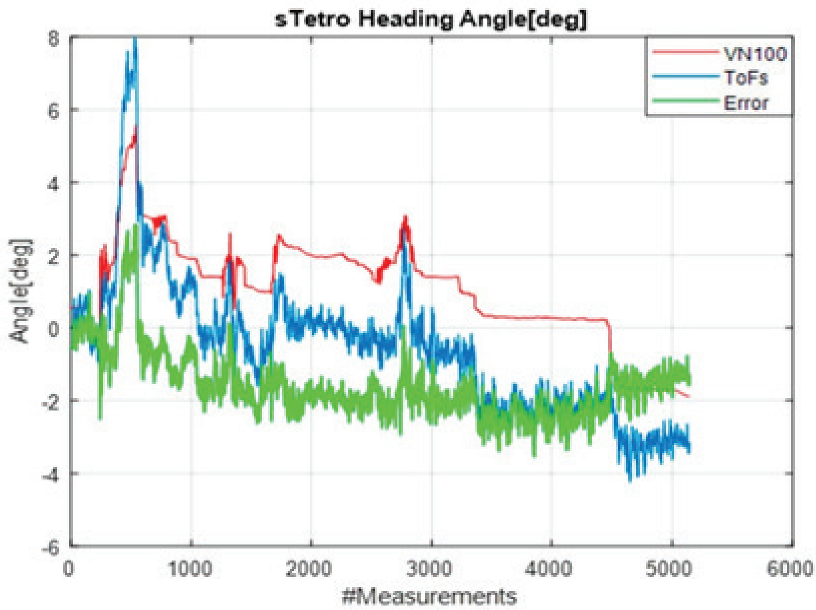

Figure 8 – Absolute heading angle estimation of sTetris

Equation (3) is used to calculate the y-position error. In this case, we calculated the error in the total distance estimated for each step in the y-axis relative to the width of the staircase. Table 3 displays the results.

$$Error = W - \left[\left(y_{m_{a_x}} - y_{m_{i_n}}\right) + R_w\right] \quad (3)$$

In this case, $y_{m_{ax}}$ and $y_{m_{in}}$ are the maximum and minimum range measurements of the left and right TF sensors on a specific step, and W is the width of the stair step and R_w is the width of the robot. The distance covered on a tread is determined by subtracting the robot width from the maximum and minimum range measurements for each step. By deducting this variance from the stair's manually measured width, the error is computed.

Table 2 – Accuracy of proposed scheme: Orientation error.

Rotation angle	VN100 (deg)	ToF (deg)	Error (deg)
0 degree	-1.80	-2.80	1.0
+45 degree	-44.2	-46.1	1.90
0 degree	1.60	0.40	1.20
-45 degree	44.4	42.60	1.80
0 degree	-1.83	-2.94	1.11

Table 3 – Accuracy of proposed scheme: Positional error

Staircase Step #	y_{max}(cm)	y_{min}(cm)	Error [cm]	Heading angle error
1	193.1	14.0	2.9	
2	193.0	14.1	3.1	Mean =0.99°
3	193.2	14.3	3.1	StdDev. =0.93°
4	193.1	14.1	3.0	
5	192.9	14.2	3.3	
6	193.1	14.3	3.2	
7	193.0	14.3	3.3	
8	193.1	13.9	2.8	
9	193.1	14.2	3.1	
10	193.2	14.1	3.1	
11	193.3	14.3	3.0	
12	193.1	14.2	3.1	
13	192.9	14.0	3.2	
14	193.1	14.3	3.1	
15	193.2	14.3		
16	193.0	14.0		

Conclusion

Here, we report a new and simple technique that takes advantage of the staircase geometry to compute the staircase cleaning robot's 3D absolute position and orientation with the least number of onboard sensors. By assuming that the robot moves through a known environment, we can precisely determine its position and orientation by using the staircase's known geometry. The position calculation algorithm receives manual measurements of the stair's tread and riser, which increases its accuracy. A moving average of the TF sensor measurements is used to reduce the amount of uncertainty in the y-axis that results from random noise in the range sensors. The primary cause of the y-axis error is the intrinsic measurement error in the sensor (3 cm according to the datasheet). The mean position error is only 3.1 cm with a standard deviation of 0.13 cm only. In a similar vein, the heading angle has a mean error of 0.99 degrees and a standard deviation of 0.93 degrees. To maintain alignment with the front step riser, the sTetris robot's heading angle should ideally stay within +5 degrees. The suggested approach faithfully satisfies this requirement. To improve the overall redundancy and reliability of the system, this technique could be integrated with other indoor positioning systems in the future.

REFERENCES

1 Ilyas M., Cho K., Baeg S.H., Park S. (2015) Drift reduction in IMU-only pedestrian navigation system in unstructured environment. 2015 10th Asian Control Conference: Emerging Control Techniques for a Sustainable World, ASCC 2015.

2 George M.B.K.W. (1988) A Navigation Control System for an Autonomous Mobile Robot. Robotics and Factories of the Future ’87. Springer, pp. 678–686.

3 Da Mota F.A.X., Rocha M.X., Rodrigues J.J.P.C., De Albuquerque V.H.C., De Alexandria A.R. (2018) Localization and navigation for autonomous mobile robots using petri nets in indoor environments. IEEE Access, 6, 31665–31676.

4 Burgard W., Fox D., Hennig D. (1996) Estimating the absolute position of a mobile robot using position probability grids. Proceedings of the Thirteenth National Conference on Artificial Intelligence, pp. 1–15.

5 Goel P., Roumeliotis S.I., Sukhatme G.S. (1999) Robust localization using relative and absolute position estimates. 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 1134–1140.

6 Gorostiza E.M., Galilea J.L.L., Meca F.J.M., Monzú D.S., Zapata F.E., Puerto L.P. (2011) Infrared sensor system for mobile-robot positioning in intelligent spaces. Sensors, 11, 5416–5438.

7 Borenstein J., Everett H.R., Feng L., Wehe D. (1997) Mobile robot positioning: Sensors and techniques. Journal of Robotics and Systems, 14, 231–249.

8 Alatise M.B., Hancke G.P. (2017) Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter. Sensors (Switzerland), 17.

9 Cho B.S., Moon W., Seo W.J., Baek K.R. (2011) A dead reckoning localization system for mobile robots using inertial sensors and wheel revolution encoding. Journal of Mechanical Science and Technology, no. 25, 2907–2917.

10 North E., Georgy J., Tarbouchi M., Iqbal U., Noureldin A. (2009) Enhanced mobile robot outdoor localization using INS/GPS integration. Proceedings of the 2009 International Conference on Computer Engineering and Systems, pp. 127–132.

11 Hammouche M., Sakhi S., Belhocine M., Elourdi A., Bouaziz S. (2016) A Fuzzy Controller for GPS/INS/Odom Integrated Navigation System. Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics, pp. 390–397.

12 Biswas J., Veloso M. (2010) WiFi localization and navigation for autonomous indoor mobile robots. 2010 IEEE International Conference on Robotics and Automation, pp. 4379–4384.

13 Lim J., Lee S., Tewolde G., Kwon J. (2015) Indoor localization and navigation for a mobile robot equipped with rotating ultrasonic sensors using a smartphone as the robot’s brain. IEEE International Conference on Electro Information Technology, vol. 2015–June, pp. 621–625.

14 Xu D., Han L., Tan M., Li Y.F. Ceiling-based visual positioning for an indoor mobile robot with monocular vision. IEEE Transactions on Industrial Electronics 2009, 56, 1617–1628.

15 Lee T.J., Kim C.H., Cho D.I.D. A Monocular Vision Sensor-Based Efficient SLAM Method for Indoor Service Robots. IEEE Transactions on Industrial Electronics 2019, 66, pp. 318–328.

16 Park S., Hashimoto S. Autonomous navigation system for mobile robot using randomly distributed passive RFID tags. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences 2010, E93–A, pp. 711–719.

17 Falsi C., Dardari D., Mucchi L., Win M.Z. (2006) Time of arrival estimation for UWB localizers in realistic environments. EURASIP Journal on Applied Signal Processing 2006, pp. 1–13.

18 Tedeschi A., Calcaterra S., Benedetto F. Ultrasonic Radar system (URAS): Arduino and virtual reality for a light-free mapping of indoor environments. IEEE Sensors Journal 2017, 17, 4595–4604.

19 Ilyas M., Yuyao S., Mohan R.E., Devarassu M., Kalimuthu M. (2018) Design of sTetro: A Modular, Reconfigurable, and Autonomous Staircase Cleaning Robot.

20 Tetris game. https://en.wikipedia.org/ (Accessed on 2 Feb. 2023).

21 Hayat A.A., Yi L., Kalimuthu M., Elara M., Wood K.L. (2022) Reconfigurable robotic system design with application to cleaning and maintenance. Journal of Mechanical Design, 144, 063305.

[1]МУХАММАД ИЛЬЯС, [1]АБДУЛЛА ААМИР ХАЯТ,
[1]ПРАБАКАРАН ВИРАДЖАГАДЕШВАР, [1]МОХАН РАДЖЕШ ЭЛАРА
[1]ROAR Lab, Инженерлік өнімді әзірлеу бөлімі, Сингапур технология және дизайн университеті (SUTD), Сингапур, 487372
*E-mail: milyasmeo2005@gmail.com

БАСПАЛДАҚ ГЕОМЕТРИЯСЫН ЖӘНЕ МИНИМАЛДЫ АРЗАН ДАТЧИКТЕРДІ ҚОЛДАНА ОТЫРЫП, БАСПАЛДАҚ ТАЗАЛАУШЫ РОБОТТЫҢ АБСОЛЮТТІ ЖАҒДАЙЫН 3D БАҒАЛАУ

Аңдатпа

Пішінін өзгерту қабілетінің арқасында sTetris роботы тазалау кезінде баспалдақпен көтеріле алады. STetris робот жүйесінің жалпы жұмысы көп қабатты кеңістіктердегі офлайн навигация үшін қажетті локализация мен орналасу деректеріне байланысты. Ғимарат ішінде жұмыс істеуге арналған мобильді роботтар әдетте локализация туралы ақпаратты алу үшін сыртқы жүйелерге сүйенеді. Өкінішке орай мобильді платформаның сәтті жұмыс істеуі дәл үш өлшемді (3D) позицияға және бағдарға (позаға) қол жеткізу үшін жиі қосымша жабдықты бекіту немесе ішкі жұмыс ортасын өзгертуді қажет етеді. Дегенмен роботты алдын ала белгілі баспалдақтың геометриялық өлшемдері туралы ақпаратты пайдалана отырып, баспалдақта локализациялауға болады. Бұл мақалада роботтың үш өлшемді позициясын құру үшін белгілі баспалдақ геометриясы мен датчиктердің ең аз санын өлшеуді қалай қолдануға болатындығы көрсетілген. Шынайы кеңістіктегі нақты роботта жүргізілген тәжірибелер ұсынылған тәсілдің тиімділігін дәлелдеді.

Тірек сөздер: ішкі орта, үш өлшемді локализация, баспалдақ геометриясы, қайта конфигурацияланатын робот, арзан датчиктер.

[1]МУХАММАД ИЛЬЯС, [1]АБДУЛЛА ААМИР ХАЯТ,
[1]ПРАБАКАРАН ВИРАДЖАГАДЕШВАР, [1]МОХАН РАДЖЕШ ЭЛАРА
[1]ROAR Lab, Отдел разработки инженерной продукции, Сингапурский университет технологий и дизайна (SUTD), Сингапур, 487372
*E-mail: milyasmeo2005@gmail.com

3D-ОЦЕНКА АБСОЛЮТНОГО ПОЛОЖЕНИЯ РОБОТА–УБОРЩИКА ЛЕСТНИЦ С ИСПОЛЬЗОВАНИЕМ ГЕОМЕТРИИ ЛЕСТНИЦЫ И МИНИМАЛЬНЫХ НЕДОРОГИХ ДАТЧИКОВ

Аннотация

Благодаря своим способностям менять форму робот sTetris может подниматься по лестнице, выполняя уборку. Общая работа системы робота sTetris зависит от данных локализации и позиционирования, которые необходимы для автономной навигации по многоэтажным помещениям. Мобильные роботы, предназначенные для работы внутри помещений, обычно полагаются на внешние системы для получения информации о локализации. К сожалению, часто требуется дополнительная аппаратная фиксация или изменения рабочей среды в помещении для достижения точного трехмерного (3D) положения и ориентации (позы) для успешной работы мобильной платформы. Тем не менее робота можно локализовать на лестнице, используя информацию о геометрических размерах лестницы, которая известна заранее. В данной статье показано, как известная геометрия лестниц и измерения минимального количества датчиков могут быть использованы для создания трехмерного позиционирования робота. Эксперименты, проведенные на реальном роботе в аутентичном помещении, успешно демонстрируют эффективность предложенного подхода.

Ключевые слова: внутренняя среда, трехмерная локализация, геометрия лестницы, реконфигурируемый робот, недорогие датчики.

Information about the authors

Mohammad Ilyas (Corresponding author)
Researcher in the ROAR lab, Engineering Product Development Unit,
Singapore University of Technology and Design (SUTD), Singapore 487372
ORCID ID: 0000-0001-7555-3839
E-mail: milyasmeo@gmail.com
Abdullah Aamir Hayat
Researcher in ROAR Lab, Engineering Product Development Unit, Singapore University
of Technology and Design (SUTD), Singapore 487372
ORCID ID: 0000-0001-6141-4600
E-mail: abdullahaamir@sutd.edu.sg
Prabhakaran Veerazhagadheswar
Researcher in ROAR lab, Engineering Product Development Unit, Singapore University
of Technology and Design (SUTD), Singapore 487372
ORCID ID: 0000-0002-5612-2619
Email: prabakaran@sutd.edu.sg
Mohan Rajesh Elara
Professor, Vice-Chancellor's Chair and Associate Professor, Engineering Product Development, Engineering Product Development Division, Singapore University of Technology and Design (SUTD), Singapore 487372
ORCID ID: 0000-0001-6504-1530
E-mail: rajeshelara@sutd.edu.sg

Авторлар туралы мәліметтер

Мохаммад Ильяс (корреспонденция авторы)
ROAR зертханасының зерттеушісі, Инженерлік өнімді әзірлеу бөлімі,
Сингапур технология және дизайн университеті (SUTD), Сингапур 487372
ORCID ID: 0000-0001-7555-3839
Email: milyasmeo@gmail.com
Абдуллах Аамир Хаят
ROAR зертханасының зерттеушісі, Инженерлік өнімді әзірлеу бөлімі,
Сингапур технология және дизайн университеті (SUTD), Сингапур 487372
ORCID ID: 0000-0001-6141-4600
E-mail: abdullahaamir@sutd.edu.sg
Прабакаран Вееражагадхесвар
ROAR зертханасының зерттеушісі, Инженерлік өнімді әзірлеу бөлімі,
Сингапур технология және дизайн университеті (SUTD), Сингапур 487372
ORCID ID: 0000-0002-5612-2619
Email: prabakaran@sutd.edu.sg
Мохан Ражеш Елара
Инженерлік өнімді әзірлеу кафедрасының проректоры және доценті, профессор,
Инженерлік өнімді әзірлеу бөлімі, Сингапур технология және дизайн
университеті (SUTD), Сингапур 487372
ORCID ID: 0000-0001-6504-1530
Email: rajeshelara@sutd.edu.sg

Информация об авторах

Мохаммад Ильяс (автор для корреспонденции)
Исследователь в лаборатории ROAR, Отдел разработки инженерной продукции,
Сингапурский университет технологий и дизайна (SUTD), Сингапур 487372
ORCID ID: 0000-0001-7555-3839
E-mail: milyasmeo@gmail.com
Абдуллах Аамир Хаят
Исследователь в лаборатории ROAR, Отдел разработки инженерной продукции,
Сингапурский университет технологий и дизайна (SUTD), Сингапур 487372
ORCID ID: 0000-0001-6141-4600
E-mail: abdullahaamir@sutd.edu.sg
Прабакаран Вееражагадхесвар
Исследователь в лаборатории ROAR, Отдел разработки инженерной продукции,
Сингапурский университет технологий и дизайна (SUTD), Сингапур 487372
ORCID ID: 0000-0002-5612-2619
E-mail: prabakaran@sutd.edu.sg
Мохан Ражеш Елара
Профессор кафедры проректора и доцент кафедры разработки инженерной продукции,
Отдел разработки инженерной продукции, Сингапурский университет технологий
и дизайна (SUTD), Сингапур 487372
ORCID ID: 0000-0001-6504-1530
E-mail: rajeshelara@sutd.edu.sg