Turnitin Originality Report

ICCED by Putra Wisnu

From publikasi (Publikasi)

Processed on 16-Jun-2023 16:05 WIB
ID: 2117206863
Word Count: 3723

Similarity Index	Similarity by Source	
3%	Internet Sources:	1%
	Publications:	3%
	Student Papers:	1%

sources:

1 1% match (Putra Wisnu Agung Sucipto, Khusnul Yaqin, Muhammad Amin Bakri, Setyo Supratno, Annisa Firasanti, Eki Ahmad Zaki Hamidi. "Statistical and Spectral Feature Extraction of Oryzias Celebensis Heart Rate", 2022 16th International Conference on Telecommunication Systems, Services, and Applications (TSSA), 2022)

Putra Wisnu Agung Sucipto, Khusnul Yaqin, Muhammad Amin Bakri, Setyo Supratno, Annisa Firasanti, Eki Ahmad Zaki Hamidi. "Statistical and Spectral Feature Extraction of Oryzias Celebensis Heart Rate", 2022 16th International Conference on Telecommunication Systems, Services, and Applications (TSSA), 2022

2 1% match (Xinda Wang, Xiao Luo, Baoling Han, Yuhan Chen, Guanhao Liang, Kailin Zheng. "Collision-Free Path Planning Method for Robots Based on an Improved Rapidly-Exploring Random Tree Algorithm", Applied Sciences, 2020)

Xinda Wang, Xiao Luo, Baoling Han, Yuhan Chen, Guanhao Liang, Kailin Zheng. "Collision-Free Path Planning Method for Robots Based on an Improved Rapidly-Exploring Random Tree Algorithm", Applied Sciences, 2020

3 1% match (Internet from 22-Dec-2022)

https://manuscriptlink-society-file.s3.ap-northeast-1.amazonaws.com/kics/conference/apcc2022/abs/C7-2.pdf

4 1% match (Hai Wang, Zhi Sun, Dazi Li, Qibing Jin. "An Improved RRT Based 3-D Path Planning Algorithm for UAV", 2019 Chinese Control And Decision Conference (CCDC), 2019)

Hai Wang, Zhi Sun, Dazi Li, Qibing Jin. "An Improved RRT Based 3-D Path Planning Algorithm for UAV", 2019 Chinese Control And Decision Conference (CCDC), 2019

paper text:

New Node Location Setting Using Random Sample Ratio In Occupancy Area On RRT Algorithm Putra Wisnu Agung Sucipto Annisa Firasanti Electronics Engineering Department Electrical Engineering Department Universitas Islam “45” Universitas Islam “45” Bekasi, Indonesia Bekasi, Indonesia wisnu@unismabekasi.ac.id

1**annisa_firasanti@unismabekasi.ac.id** Seta Samsiana **Electrical Engineering Department**

1**Eki Ahmad Zaki Hamidi** Universitas Islam “45” **Electrical Engineering Department** Bekasi, Indonesia **UIN Sunan Gunung Djati**

setasamsiana@unismabekasi.ac.id

1**Bandung, Indonesia ekiahmadzaki@uinsgd.ac.id**

Abstract— The Rapidly Random Tree (RRT) algorithm works to grow trees gradually based on a random sampling process. Each growing branch needs to be arranged so that it can reach the target node. This arrangement is done by maintaining the distance between the edge knot and the parent knot so that they are always in the optimal direction and magnitude. This paper offers an alternative approach to adjusting the position of the edge vertices in a branch based on the modeling occupancy area that can be occupied by random knots and the distance function which is formulated using the growth ratio of random samples that inhabit the target area and the area outside it. With the same number of iterations, the proposed algorithm can make the tree reach a 0.5 cm distance from the target node, while the basic RRT is 27 cm. Furthermore,

the path of the tree created by the new algorithm is 29 cm longer and has 30 fewer branches without leaving any ends of the twigs that grow against the direction of the target node. Keywords: Occupancy area, Sample Rate Ratio, Distance function I. INTRODUCTION RRT has gained a lot of the attention of the research community over the past few decades. Most researchers are interested in RRT because it can work in a high-dimensional state space[1][2][3][4][5]. This algorithm provides the opportunity to achieve a viable optimal solution based on the growth of randomness of the sampling process. Since it is sampling-based, RRT has a probability content in the solution[6][7][8][9]. Spatially, the solution came from the movement of twigs within a residential area. Twigs that are capable to develop well are going to leave seedlings a decent path to reach the final node. Conversely, twigs that don't grow healthily will lead the PRC to create a path that strays from the final node, making it unfit to construct an optimal solution. Therefore, this paper offers an alternative approach to the layout of new nodes against parental nodes. The main contribution made in this paper is the arrangement of the position of the new node in the twig, considering the residential area of the target node side and the root node side. In addition, we propose the function of the distance

4**of the parent node** to **the** edge **node** by paying attention **to the**

number of random samples in a residential area. II. RELATED WORK Setting the direction of growth of the twig from the root towards the target needs to pay attention to the vector of its growth rate. This growth will be better when a random node approaches the target node and vice versa. Some researchers have done work in the search for techniques for growing new twig nodes. This study stems from the basic concept of edge node growth in the PRC based on the safe distance of new edge growth or the so- called threshold limit of the distance of the edge node to the parent node[10][11][12][13]. The growth direction of this edge node corresponds to the direction of the distance line of the parent node and the random node generated each iteration, so there is a possibility of developing away from the target node. Furthermore, the spacing of the edge node and its parents is set using the minimum cost provisions of the neighboring parent distance pool in a hypersphere centered on the position of random nodes[10][14][15][16]. The edge node that is successfully laid, will have

3**the lowest cost** against **the** root **node of the** tree. However, **the** direction **of** twig growth **is**

also likely to grow away from the target node if there is the minimal cost between the parent node pools within the hypersphere that are far from the target node. Another way that can be used is to use the provisions of virtual forces. The distance of the edge to the parent node is set based on the content of the repulsive force and attractive forces in the virtual forces[17][18][10]. The growth direction of the edge branch can reach the target node because of the attractive forces around the target node. However, the distance that is too far due to the resultant virtual style needs further attention. The location determination of the new edge value in the twig growth direction is often also reviewed based on the distance function setting. For example, setting the bias coefficient is targeted based on the probability of transition between branches. This coefficient is configured with some variables: the target's gravity; the ratio

4**between the** target **node** distance **and the** edge **node**; and **the** ratio between **the** random **node**

to the edge node [19] [20] as well as the complexity of the environment are determined subjectively[21]. This provision does not rule out the possibility of using quadratic cost functions in Euclidean terminology and Generalized distance as traditional traversal methods[22] [23] [24] [25]. Based on the reference, research on the directing function of twig growth mostly selects the proximity between adjacent vertices and their derivative aspects as a reference for their formalization. There has been no formulation of restrictions on the growth and development of twigs based on the growth records of random nodes that inhabit an area around the target and the area outside it, so that it can be known which edge knots must be kept growing and which one to be restricted. This paper contributed to providing a new method for the arrangement of the position of nodes in twigs. It is located on the modeling of the residential area of the side of the target node and the root node that can be occupied by a random sample. In addition, we provide the distance function of the parent node towards the edge node based on the ratio of random samples inhabiting the inside and outside the target area. III. RESEARCH METHOD A. Algoritma RRT Basic Trees grown algorithms are developed by several operating processes. First, the algorithm ran the initialization of the root node qstart and the target node, qgoal in the T tree. Then, the branch is grown by placing the edge knot, qnew, at the end of the twig. This operation is preceded by the generation of a random sample, qrand, by the RandomSample() function. The laying of qnew is calculated based on the proximity of qrand to the parent knot, qparrent adjacent to it.

Using the NearestNeighborfunction () is generated a node qparrent that is located from the new q with a certain value. After successfully obtaining qparrent then the new qis is added to the tree branch through the AddNewNode() function. The sequence of this process repeats continuously until the distance qnew and qgoalreaches a certain minimum value, ρmin. The details of this algorithm are organized from [14][16] and described in the following algorithm. Algoritma 1 Algoritma RRT Basic

> 2**1: T ← InitTree**(qstart, qgoal); **2: for i = 1 to n do 3**: qrand ← **RandomSample(i); 4**

ρmin then

> 2**return T; end if end for**

$qf\ oqhdqwiqd$ (1) The number of random samples inhabiting both areas of origin is explained by expression (2) $Oa\ (d(qp;\ qp;\ qf))$ = ∑ $d(qp;\ qp;\ qf)$ (2) $f(pr;pr;pg)$=1 where : Oa = Number of samples in origin area. While those who inhabit the target area are expressed in (3) $Ta\ (d(qp;\ qp;\ qf))$ = ∑ $d(qp;\ qp;\ qf)$ (3) $f(pr;pr;pg)$=0 where : Ta = Number of samples in the target area. The growth of the number of random samples in each area needs to be noted as one of the considerations in the mechanism of extension of the new branch path. The more random samples are in the area of origin as a consequence of the area of occupancy, the slower arrival of random samples in the target area. The occupancy area around the target which is much narrower than the original area will cause less habitable empty space for random samples around the target. This condition brings benefits On the other hand, this condition can be an advantage. This difference in conditions can be used as a reference in making the reciprocation function to extend or suppress the growth rate of branches. Fewer and fewer random samples in the target area can be used to trigger a slowdown in branch extension in the area of origin. Conversely, the number of random samples in the target area that is less than the area of origin can also trigger the extension of branches in the area of origin. Given that convergence towards the target node is a top priority and the fact that the growth of the number of samples in both areas can be adjusted in ratio, this condition provides room to implement the reciprocity paradigm in calculating the distance of the parent node to the edge node. With this paradigm, the difference in the length of the edge node distance in both areas gets the right coefficient. In other words, if the growth ratio of the area of origin is greater than the target area, this ratio can be fed as a coefficient of distance function for the target area that can trigger the branch based on the probability of randomness of the sample in the occupancy space. Conversely, because the area on the target side is much narrower than the area of origin, the extension has the potential to reduce the growth ratio of random samples in the target area, it can be fed to the area of origin as a reference for shortening of twigs around the edge knot. The growth ratio of random samples in the area of origin is defined by equation (4). (4) where : ΔOa = $Oa0{+}aTa$ ΔOa = Random sample ratio in the origin area while random sample ratio in the target area is defined by (5) $T\ a$ (5) where : ΔTa = $Oa{+}Ta$ ΔTa = Random sample ratio in the target area Thus, the distance between the parent and edge node in each occupancy area is defined by dpn = { ΔTa , $id\ d(qp;\ qp;\ qf)$ = 1 $\Delta Oa\ w\ dOR$, $id\ d(qp;\ qp;\ qf)$ = 0 (6) where : dpn = Distance between parent and edge node dOR = Distance between qp and $qp\ q\ p$ = Parent node. Thus, the location of the edge node in the tree branch is determined with (7) and (8). where : $qnx\ qnx\ \theta OR\ qnx$=$(qp-\ dpn)*$cos(θOR) qnx=$(qp-\ dpn)*$sin(θOR) = Edge node coordinate in X axis = Edge node coordinate in Y axis =Angle between parent and random node (7) (8) IV. RESULTS

> 3**AND DISCUSSION In this section, the performance of**

our proposed method is examined against the RRT Basic algorithm which uses the euclidean distance-based distance function as a reference. An evaluation was carried out on several indicators of performance criteria; path length (PJ), number of iterations (JI), number of twigs (JR) formed, ability to approach the target and twigs that turn direction. The comparison of these two algorithms is implemented using Matlab which runs on a Laptop PC with an Intel i7-9750 @ 2.6 GHz (12 CPUs) and 9 MB of RAM. The experiment shows that our algorithm is vastly superior. First, the robot is placed at the initial coordinates, qstarts (1,5). The robot must move towards the final coordinate, qgoal (5,6). Then, the RRT must work to create this robotic trajectory path. During the process of creating the robotic trajectory path, the RRT must pay attention to the stop condition to exit the iteration which is when the position of the RRT's edge node has approached the target at the pre-determined minimal distance, ρmin = 0.5 cm. Fig. 2. Trees with RRT Algorithms using Random Sample Growth Ratios as a Reference for Distance between Parent and Edge node. Our proposed algorithm has succeeded in creating a tree that illustrates the trajectory of the robot as in figure 2. In this experiment, to be able to reach the goal node, qp was given the treatment of limiting the iteration number. The result is that our algorithm is able to approach qp in very close proximity at the 265th iteration. In contrast to the basic RRT algorithm, up to the same number of iterations, the twigs created have not been able to approach qp. An illustration of the distance of the furthest twigs which was successfully evoked by the RRT basic algorithm in the 265th iteration is shown in figure 3. Fig. 3. Trees with RRT Algorithms using Threshold distance between Parental And Edge node In terms of the number of tree branches that are successfully created, the algorithm that we propose is able to form a tree with the least number of branches. From the experiments that have been carried out, we managed to restrain the growth and development of twigs into 43 branches. These branches are allowed to grow if the new node is in an inhabited area around the target node and not the other way around in an inhabited area outside the target area. In contrast, in the basic RRT algorithm, in the span of 265 rounds of iterations, it has produced 73 branches. This branch will continue to grow given the absence of the end of the tree branch approaching the target node. Considering the length of the formed path, the algorithm we created is able to produce a longer trajectory path compared to the basic RRT algorithm. In 265th iterations, we managed to make a path line 43 cm long. Meanwhile, the basic RRT algorithm is only capable of making 14 cm. The size of the length of this path pays attention to the dimensions of the RectangulerArea simulator Webots which measures one-meter square. Based on the experiment result, we can say that the proposed algorithm has a faster ability to approach the target. Although formed by irregularity or randomness, the algorithm we propose is able to control this randomness so that it tends to be able to be made regular. The trajectory resulting from the random generation of nodes tends to be set to grow and move towards the target. This randomness setting ability is seen in the achievement of tree branches that are able to approach the value of pmin. In the 265th iteration, our proposed algorithm was able to make the twigs of the end of the trajectory tree approach the target point within a radius of 0.5 cm, while the basic RRT algorithm was still 27 cm from the target node, without any certainty whether the tip of the twig would move away from or approach the target node. These results show that the performance of the algorithms we compiled is vastly superior to the basic PRC algorithms. Another advantage in the proposed algorithm that cannot be found in the basic RRT algorithm is the absence of tree branches that turn away against the direction of the target. Based on figures 2 and 3, the entire tip of the twig tends to grow straight. There are no twigs that grow curved in opposition to the attraction force of the target node. V. CONCLUSION This study succeeded in making the function coefficient of the distance function of the parent's node to the edge node, in order to achieve convergence of the motion direction of tree branches in the RRT. The distance function can trigger the acceleration of branch growth towards the target node. In addition, the setting of occupancy areas can also be a strategy for the extension and restriction of branch growth. As for future works, it is necessary to make additional settings related to the potential change in the branch direction movement that approaches the target node but has not implemented an optimal restriction of branch length yet.

REFERENCES [1] R. C. Luo and C. Huang, “Anytime dynamic exploring rapid random tree approach in higher dimension search space for non-holonomic robotics,” 2016 Int. Conf. Adv. Robot. Intell. Syst. ARIS 2016, 2017. [2] H. B. Li, W. Wang, H. W. Ding, and J. Dong, “Trees Weighting Random Forest method for classifying high-dimensional noisy data,” Proc. - IEEE Int. Conf. E-bus. Eng. ICEBE 2010, pp. 160–163, 2010. [3] O. Mechali, L. Xu, M. Wei, I. Benkhaddra, F. Guo, and A. Senouci, “A Rectified RRT∗ with Efficient Obstacles Avoidance Method for UAV in 3D Environment,” 9th IEEE Int. Conf. Cyber Technol. Autom. Control Intell. Syst. CYBER 2019, pp. 480–485, 2019. [4] L. Meng, S. Qing, and Z. Q. Jun, “UAV path re-planning based on improved bidirectional RRT algorithm in dynamic environment,” 2017 3rd Int. Conf. Control. Autom. Robot. ICCAR 2017, pp. 658–661, 2017. [5] Y. Song, J. Zuo, J. Wu, Z. Liu, and Z. Li, “Robot Perceptual Classification Method Based on Mixed Features of Decision Tree and Random Forest,” 2021 IEEE 2nd Int. Conf. Big Data, Artif. Intell. Internet Things Eng. ICBAIE 2021, no. Icbaie, pp. 919–922, 2021. [6] M. Kleinbort, K. Solovey, Z. Littlefield, K. E. Bekris, and D. Halperin, “Probabilistic Completeness of RRT for Geometric and Kinodynamic Planning with Forward Propagation,” IEEE Robot. Autom. Lett., vol. 4, no. 2, pp. 277–283, 2019. [7] H. Zhang, Y. Wang, J. Zheng, and J. Yu, “Path planning of industrial robot based on improved RRT algorithm in complex environments,” IEEE Access, vol. 6, pp. 53296–53306, 2018. [8] M. C. Santos, L. Molina, E. A. N. Carvalho, E. O. Freire, J. G. N. Carvalho, and P. C. Santos, “MB-RRT: An inverse kinematics solver of reduced dimension,” IEEE Access, vol. 9, pp. 148558–148573, 2021. [9] F. Meng, L. Chen, H. Ma, J. Wang, and M. Q.-H. Meng, “NR-RRT: Neural Risk-Aware Near-Optimal Path Planning in Uncertain Nonconvex Environments,” vol. 6, no. 1, pp. 1–12, 2022. [10] Y. Li, W. Wei, Y. Gao, D. Wang, and Z. Fan, “PQ-RRT*: An improved path planning algorithm for mobile robots,” Expert Syst. Appl., vol. 152, p. 113425, 2020. [11] F. Baberg and P. Ogren, “Formation obstacle avoidance using RRT and constraint based programming,” SSRR 2017 - 15th IEEE Int. Symp. Safety, Secur. Rescue Robot. Conf., pp. 1–6, 2017. [12] Y. J. Kim, J. H. Wang, S. Y. Park, J. Y. Lee, J. J. Kim, and J. J. Lee, “A RRT-based collision-free and occlusion-free path planning method for a 7DOF manipulator,” 2014 IEEE Int. Conf. Mechatronics Autom. IEEE ICMA 2014, pp. 1017–1021, 2014. [13] J. Janos, V. Vonasek, and R. Penicka, “Multi-Goal Path Planning Using Multiple Random Trees,” IEEE Robot. Autom. Lett., vol. 6, no. 2, pp. 4201–4208, 2021. [14] H. Lee, H. Kim, and H. J. Kim, “Planning and Control for Collision-Free Cooperative Aerial Transportation,” IEEE Trans. Autom. Sci. Eng., vol. 15, no. 1, pp. 189–201, 2018. [15] X. Wang, J. Wei, X. Zhou, Z. Xia, and X. Gu, “Dual-Objective Collision-Free Path Optimization of,” vol. 6, no. 4, pp. 6353–6360, 2021. [16] D. Connell and H. M. La, “Dynamic path planning and replanning for mobile robots using RRT,” 2017 IEEE Int. Conf. Syst. Man, Cybern. SMC 2017, vol. 2017-Janua, pp. 1429–1434, 2017. [17] A. H. Qureshi and Y. Ayaz, “Potential functions based sampling heuristic for optimal path planning," Auton. Robots, vol. 40, no. 6, pp. 1079–1093, 2016. [18] J. W. Woo, J. Y. An, M. G. Cho, and C. J. Kim, "Integration of path planning, trajectory generation and trajectory tracking control for aircraft mission autonomy," Aerosp. Sci. Technol., vol. 118, p. 107014, 2021. [19] W. Gong, "Probabilistic model based path planning," Phys. A Stat. Mech. its Appl., vol. 568, p. 125718, 2021. [20] L. Ye, J. Duan, Z. Yang, X. Zou, M. Chen, and S. Zhang, "Collision-free motion planning for the litchi-picking robot," Comput. Electron. Agric., vol. 185, no. 483, p. 106151, 2021. [21] C. Yuan, G. Liu, W. Zhang, and X. Pan, "An efficient RRT cache method in dynamic environments for path planning," Rob. Auton. Syst., vol. 131, p. 103595, 2020. [22] Y. Zhou, E. Zhang, H. Guo, Y. Fang, and H. Li, "Lifting path planning of mobile cranes based on an improved RRT algorithm," Adv. Eng. Informatics, vol. 50, no. June, p. 101376, 2021. [23] D. Gao, J. Luo, W. Ma, and B. Englot, "Online feedback motion planning for spacecraft obstacle avoidance using positively invariant sets," Adv. Sp. Res., vol. 65, no. 10, pp. 2424–2434, 2020. [24] X. Wang, X. Luo, B. Han, Y. Chen, G. Liang, and K. Zheng, "Collision- free path planning method for robots based on an improved rapidly- exploring random tree algorithm," Appl. Sci., vol. 10, no. 4, 2020. [25] J. Qi, H. Yang, and H. Sun, "MOD-RRT*: A Sampling-Based Algorithm for Robot Path Planning in Dynamic Environment," IEEE Trans. Ind. Electron., vol. 68, no. 8, pp. 7244–7251, 2021.