Recurrent Neural Networks for Range-only SLAM

Hyungtae Lim[1] , Junseok Lee[1], Changgyu Park[1], Ye Eun Kim[1],

***Abstract*— Range-only SLAM is a method for localizing a mobile robot and beacons by mainly utilizing distance measurements. Unlike the traditional probability-based range-only SLAM method, we present a novel approach using a recurrent neural network architecture that directly learns the end-to-end mapping between distance data and robot position.**

I. INTRODUCTION

Trilateration is a conventional algorithm for locating a vehicle in the metropolitan area by range measurements between the vehicle and fixed beacon sensors. [1]. Due to the convenience of trilateration that estimates the position of a receiver of range sensors if one only knows range measurement, trilateration algorithm has been widely incorporated into robotics fields, especially utilized in the indoor environment to estimate the position of an object by distance measurements obtained from range sensors such as UWB, ultrasonic, laser-based beacon sensors [2]–[4]. Specifically, range-only Simultaneous Localization and Mapping(RO-SLAM) methods are utilized popularly, which not only estimate the position of the receiver of range sensors, but also localize the position of range sensors regarded as features on a map, and studies have been conducted continuously in terms of probability-based approach [5]–[8].

In the meantime, as deep learning age has come [9], various kinds of deep neural architectures have been proposed for many tasks related to robotics field, such as detection [10]–[12], navigation [13], [14], pose estimation [15], and so on. Especially, recurrent neural networks (RNNs), originated from Natural Language Process(NLP) area [16], have been shown to achieve better performance in case of dealing with time variant information, thereby RNNs are widely utilized such as not only speech recognition, but also pose estimation and localization [15], [17]–[20].

In this paper, we propose a deep learning-based localization method by stacked bidirectional Long Short-Term Memory(stacked Bi-LSTM) for more accurate localization of the robot. Using deep learning, our structure directly learns the end-to-end mapping between range measurements and robot position. This operation non-linearly maps the relationship not only considering the long-range dependence of sequential distance data by the LSTM, but also using the correlation of the backward information and the forward information of the sequence of each time step by virtue of its bidirectional architecture.

[1]Hyungtae Lim, [1]Junseok Lee, [1]Changgyu Park, and [1]Ye Eun Kim are with the Urban Robotics Laboratory, Korea Advanced Institute of Science and Technology (KAIST) Daejeon, 34141, South Korea. {shapelim, ljs630, cpark, yeeunk}@kaist.ac.kr

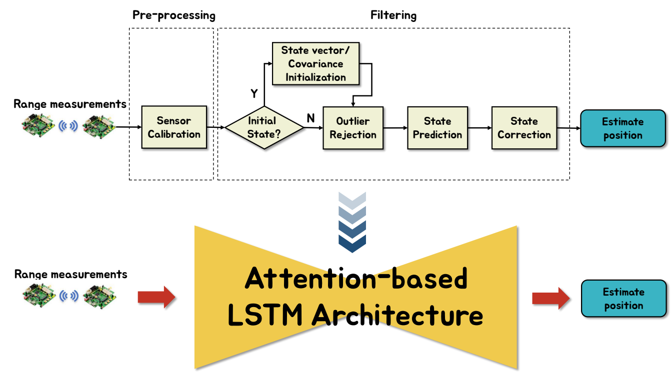

Fig. 1. System overview. A robot localizes its own pose through distance data and the derivative of distance data.

II. RELATED WORKS

In this section, we briefly survey previous researches closely focused on Long Short-Term Memory(LSTM) model and applications of LSTMs to solve domain problems.

1) LSTM: LSTM is a type of Recurrent Neural Networks(RNNs) that has loops so that infer output based on not only the input data, but also the internal state formed by previous information. In other words, while the RNN deals with sequential data, the network has remembered the previous state generated by past inputs and might be able to output the present time step via internal state and input, which is very similar to filtering algorithms.

However, RNNs often have a *vanishing gradient problem*,i.e., RNNs fail to propagate the previous matter into present tasks as time step gap grows by. In other words, RNNs are not able to learn to store appropriate internal states and operate on long-term trends. That is the reason why the Long Short-Term Memory (LSTM) architecture was introduced to solve this long-term dependency problem and make the networks possible to learn longer-term contextual understandings [21]. By virtue of the LSTM architecture that has memory gates and units that enable learning of long-term dependencies [22], LSTM are widely used in most of the deep learning research areas and numerous variations of LSTM architecutres have been studied.

2) Localization with Deep Learning: There have been many approaches combining Simultaneous Localization and Mapping (SLAM) with deep learning, aiming to overcome the limitations on SLAM only technique such as difficulty on tuning the proper parameters in different environments and recovering an exact scale. Actually, those researches are showing the superior performance to the traditional SLAM approaches. One of the popular SLAM techniques with deep learning is CNN-SLAM [23] which takes Convolutional Neural Networks (CNNs) to precisely predict the depth from a single image without any scene-based assumptions or geometric constraints, allowing them to recover the absolute scale of the reconstruction. Another approach using deep learning for localization is Deep VO [24] In this method, Recurrent Convolutional Neural Networks (RCNNs) is utilized. Specifically, feature representation is learned by Convolutional Neural Networks and Sequential information and motion dynamics are obtained by deep Recurrent Neural Networks without using any module in the classic VO pipeline.

3) Applications of LSTMs: There are many variations of LSTM architecture. As studies of deep learning are getting popular, various modified architectures of LSTM have been proposed for many tasks in a wide area of science and engineering. Because LSTM is powerful when dealing with sequential data and infering output by using previous inputs, LSTM is utilized to estimate pose by being attached to the end part of deep learning architecture [18]–[20] as a stacked form of LSTM. In addition, LSTM takes many various data as input; LSTM is exploited for sequential modeling using LiDAR scan data [17], images [15], [18], IMU [25], a fusion of IMU and images [24].

III. Our approaches

A. LSTM

B. Bidirecional LSTM

C. Stacked Architecture

D. Multimodal Architecture

E. Training Loss

We set the experiment on the virtual situation and generate distance data set which corresponds to the position with 10% noise error and let RNN be trained using these distance data. Train data are just zigzag paths and test data is an arbitrary path, so we also check if RNN can estimate the position despite the variation of distance data as input.

Let Θ be the parameters of our RNN model, then our final goal is to find optimal parameters Θ^* for localization by minimizing Mean Square Error (MSE) of Euclidean distance between ground truth position Y_k and estimated position $\hat{Y_k}$.

$$\Theta^* = \underset{\Theta}{\operatorname{argmin}} \sum_{k=1}^{N} \parallel Y_k - \hat{Y_k} \parallel^2 \tag{1}$$

IV. Experiment

A. Training/Test Dataset

B. RNN Architectures

Various kinds of RNN models are tested for robot localization through range measurements. We trained 4 kinds of RNNs; LSTM [21], GRU [26], Bi-LSTM [27], stacked Bi-LSTM, which are basic architecture units in many RNNs researches. Using deep learning, these structures directly learn the end-to-end mapping between range data and robot position.

V. Results

To verify our proposal that RNNs can estimate the robot's position through varying range data, we set the experiment on the simulated environment and generate range data set which corresponds to the position with 10% noise error and let RNN be trained using these range data. Train data are just zigzag paths that generate Δx or Δy changes respectively. Test data are two types; a diagonal oval path and an arbitrary path, which are shown as Fig. **??** Fig. **??**. There's no same path between train data set and test data set. In addition, unlike train data set, x and y directions change at the same time. Thus, we also check the capability of RNNs to estimate the position in the regions not included in the train data set by variation of range data over time as input.

Results of RMSE[m]		
Model	Test1	Test2
Particle filter based	1	2
Bidirectional Multimodal	3	**4**

The results of trajectory prediction are shown in Fig. **??** and Fig. **??** and Root-Mean-Squared Error (RMSE) are shown in Table **??**. Performance is better in order of stacked Bi-LSTM, Bi-LSTM, LSTM and GRU. In case of GRU, it has only two gates which is less complex structure than LSTM [28]. However, due to GRU's less complexity, GRU has less the number of neurons than LSTM so their non-linear mapping achieves less performance. Likewise, Bi-LSTM consists of two LSTMs to process sequence in two directions so that infer output using the correlation of the backward information and the forward information of the sequences of each time step with its two separate hidden layers. Thus, Bi-LSTM has better nonlinear mapping capability than LSTM. For similar reasons, stacked Bi-LSTM is the architecture that stacks two Bi-LSTMs, so inference performance is better than Bi-LSTM. As a result, the stacked Bi-LSTM showed the best performance among unit RNN architectures. Therefore, we can conclude that the performance improves as the non-linearity of the architecture increases.

VI. Conclusion

In this paper, we proposed a novel approach to range-only measurements localization using recurrent neural network models and tested various types of LSTM models for more accurate localization of the mobile robot.

Using deep learning, our structure directly learns the end-to-end mapping between distance data and robot position. The stacked bidirectional LSTM structure exhibits the best estimates of robot positions than other RNN structure units. Therefore, we conclude that the LSTM-based structure improves performance as non-linearity of structures increase and even if the robot position is not included in the ground truth dataset, our method is able to predict robot positions with small errors through sequential distance data.

As a future work, because train/test dataset are generated on simulated environment, the proposed method needs to be tested in the real-world to check whether RNNs can deal

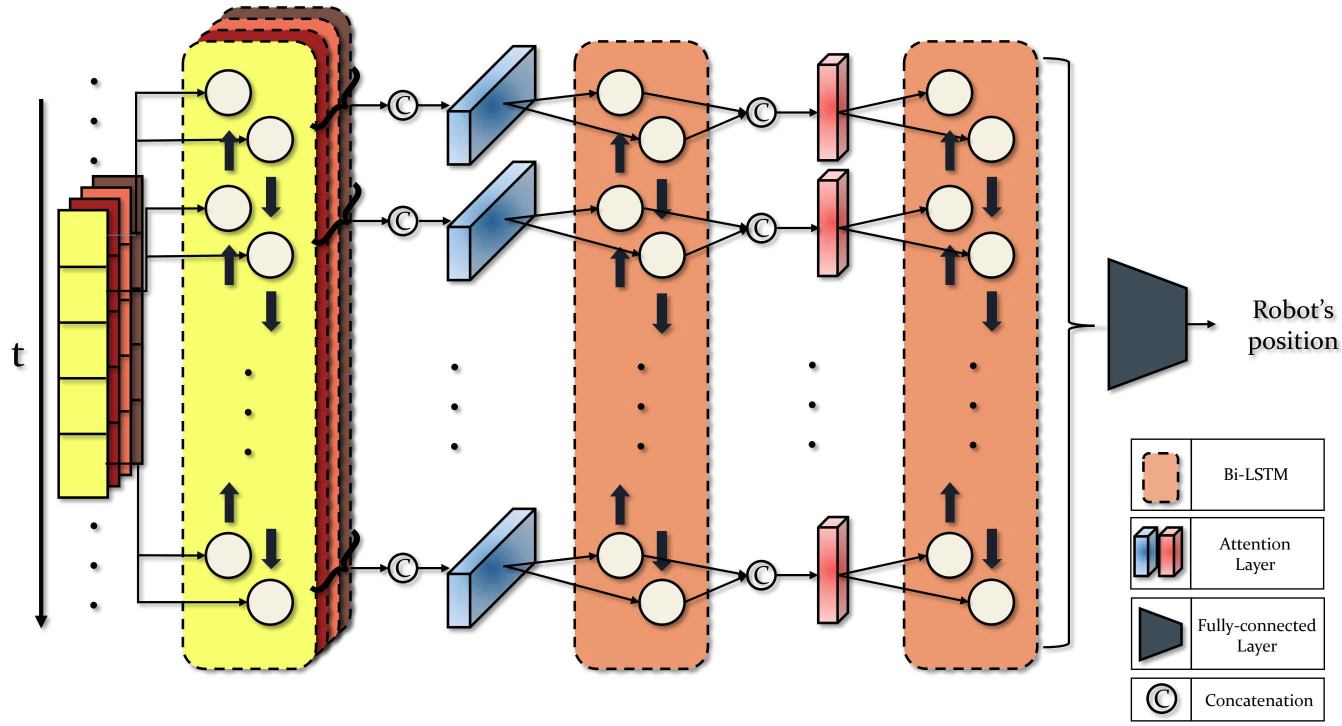

Fig. 2. Blabla.

with multipath problems and line of non-line of sight(NLOS) issues.

REFERENCES

[1] H. Staras and S. Honickman, "The accuracy of vehicle location by trilateration in a dense urban environment," *IEEE Transactions on Vehicular Technology*, vol. 21, no. 1, pp. 38–43, 1972.

[2] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on robotics*, vol. 21, no. 1, pp. 93–101, 2005.

[3] H. Cho and S. W. Kim, "Mobile robot localization using biased chirp-spread-spectrum ranging," *IEEE transactions on industrial electronics*, vol. 57, no. 8, pp. 2826–2835, 2010.

[4] A. N. Raghavan, H. Ananthapadmanaban, M. S. Sivamurugan, and B. Ravindran, "Accurate mobile robot localization in indoor environments using bluetooth," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4391–4396.

[5] J.-L. Blanco, J. González, and J.-A. Fernández-Madrigal, "A pure probabilistic approach to range-only slam." in *ICRA*. Citeseer, 2008, pp. 1436–1441.

[6] J.-L. Blanco, J.-A. Fernández-Madrigal, and J. González, "Efficient probabilistic range-only slam," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 1017–1022.

[7] F. R. Fabresse, F. Caballero, I. Maza, and A. Ollero, "Undelayed 3d ro-slam based on gaussian-mixture and reduced spherical parametrization," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. Citeseer, 2013, pp. 1555–1561.

[8] N. S. Shetty, "Particle filter approach to overcome multipath propagation error in slam indoor applications," Ph.D. dissertation, The University of North Carolina at Charlotte, 2018.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[10] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[11] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *European Conference on Computer Vision*. Springer, 2016, pp. 354–370.

[12] H. H. Smith, "Object detection and distance estimation using deep learning algorithms for autonomous robotic navigation," 2018.

[13] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 3357–3364.

[14] M. Hamandi, M. D'Arcy, and P. Fazli, "Deepmotion: Learning to navigate like humans," *arXiv preprint arXiv:1803.03719*, 2018.

[15] F. Walch, C. Hazirbas, L. Leal-Taixe, T. Sattler, S. Hilsenbeck, and D. Cremers, "Image-based localization using lstms for structured feature correlation," in *Int. Conf. Comput. Vis.(ICCV)*, 2017, pp. 627–637.

[16] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

[17] S. Gladh, M. Danelljan, F. S. Khan, and M. Felsberg, "Deep motion features for visual tracking," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE, 2016, pp. 1243–1248.

[18] S. Wang, R. Clark, H. Wen, and N. Trigoni, "Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2043–2050.

[19] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[20] M. Turan, Y. Almalioglu, H. Araujo, E. Konukoglu, and M. Sitti, "Deep endovo: A recurrent convolutional neural network (rcnn) based visual odometry approach for endoscopic capsule robots," *Neurocomputing*, vol. 275, pp. 1861–1870, 2018.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] W. Zaremba and I. Sutskever, "Learning to execute," *arXiv preprint arXiv:1410.4615*, 2014.

[23] K. Tateno, F. Tombari, I. Laina, and N. Navab, "Cnn-slam: Real-time

dense monocular slam with learned depth prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2017.
[24] R. Clark, S. Wang, H. Wen, A. Markham, and N. Trigoni, "Vinet: Visual-inertial odometry as a sequence-to-sequence learning problem." in *AAAI*, 2017, pp. 3995–4001.
[25] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.
[26] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
[27] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
[28] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.