Improving resolution in deep learning-based estimation of drone position and direction using 3D maps

Masatoshi Hamanaka[1]

***Abstract*— We propose a method to improve the resolution of drone position and direction estimation on the basis of deep learning using three-dimensional (3D) topographic maps in non-global positioning system (GPS) environments. GPS is typically used to estimate the position of drones flying outdoors. However, it becomes difficult to estimate the position if the signal from GPS satellites is blocked by tall mountains or buildings, or if there are interference signals. To avoid this loss of GPS, we previously developed a learning-based flight area estimation method using 3D topographic maps. With this method, the flight area could be estimated with an accuracy of 98.4% in experiments conducted in 25 areas, each 40 meters square. However, a resolution of 40 meters square is difficult to use for drone control. Therefore, in this study, we will verify whether it is possible to improve the resolution by multiplexing the area division and the data acquisition direction. We also investigated whether the flight direction of the drone can be detected using a 3D map. Experimental results show that the position estimation was 96.8% accurate at a resolution of 25 meters square, and the direction estimation was 92.6% accurate for 12-direction estimation.**

I. INTRODUCTION

Our goal is to use drones for delivery in mountainous areas. Drones require a lot of energy to climb, so in the mountains, it is more energy efficient to fly them in the valleys rather than along the peaks. When flying in a valley, signals from global positioning system (GPS) satellites can become blocked, which reduces navigational accuracy and makes flight difficult. Furthermore, GPS signals are very weak and subject to a variety of disturbances in general, and such problems increase the chances of a crash when flying a GPS-controlled drone [1], [2].

Assisted-GPS (A-GPS) [3], [4] and quasi-zenith satellite systems (QZSS) [5] are two approaches that have aimed to solve these problems. A-GPS augments regular GPS with cell tower data and Wi-Fi positioning to enhance the quality and precision when operating in poor satellite signal conditions. However, there are no cell towers or Wi-Fi on mountainsides or high up in the air. As for QZSS, while it provides highly precise and stable positioning services that are compatible with GPS, the services are only available in the Asia-Oceania region. Moreover, QZSS cannot be used if there are jamming waves, as is often the case with GPS.

As a drone technology in a non-GPS environment, a method of finding an emergency landing site using SLAM has been proposed [6]. Another method using artificial intelligence (AI) to determine the safety of landing sites using ultrasonic sensors has been proposed [7]. These assume that the drone will move slowly when GPS signals cannot be acquired [6], [7]. Our premise is that even if the GPS cannot be captured during high-speed movement, the proposed method will correctly estimate the position and continue high-speed movement.

To address these non-GPS issues, we previously proposed a method to estimate the flight area using deep learning techniques [8] to match the ground surface vectors obtained by a two-dimensional (2D) LiDAR [9], [10] mounted on a drone with a three-dimensional (3D) map acquired by a land observation satellite (Fig. 1)[11]. Although deep learning has been used to estimate the positions of persons and objects from camera images, at the time of writing, no such method has been proposed for estimating the position of an unmanned aircraft system (UAS) [12], [13].

The main advantage of our method is that we consider terrain undulations as unique information similar to fingerprints and solve localization as a machine-learning problem. Our flight experiment results showed that this method could estimate the flight area with an accuracy of 98.4%. However, we encountered the following two problems when it came to controlling the drone's flight.

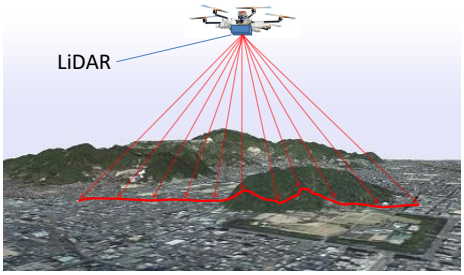

Fig. 1. Acquisition of surface vectors by LiDAR.

Low resolution: In the flight experiment, we estimated which area was 40 meters square from the five vertical and five horizontal areas. The resolution of 40 meters square is insufficient when flying at a low altitude (e.g., when landing), so it is necessary to detect the position with a higher resolution. At this time, if the area is reduced to increase the resolution, the learning data for each area is also reduced, and the accuracy of area estimation by deep learning is consequently lowered as well.

Area estimation methods based on deep learning suffer from low estimation accuracy near area boundaries [11] because it is difficult to learn samples that have very similar surface vectors but different areas. When the area is sub-

[1]Masatoshi Hamanaka is the team leader of the Music Information Intelligence Team at RIKEN Center for Advanced Intelligence Project, Tokyo, Japan masatoshi.hamanaka@riken.jp

divided, the average estimation accuracy decreases because the number of samples near the area boundary increases relatively.

Therefore, in our improved method, we do not change the size of the area division but perform more than one estimation instead. In these multiple estimations, we construct four area estimators with different area boundaries and form a common area by superimposing all four estimation areas, which is then set as the final estimation area.

Compass error: Assuming that the nose of the drone is facing north, we used the surface vectors located 25 meters to the left and right of the drone as data for learning and estimating the deep neural networks. However, there are many places where compass errors occur (e.g., near power lines), and it is difficult to always fix the nose direction to the north. If the scanning direction of LiDAR greatly differs from that at the time of deep learning, it becomes difficult to estimate the area. Furthermore, to implement drone autopilot in places where GPS is difficult to use, it is necessary to correctly estimate the direction in addition to the position of the drone.

At present, it is possible to estimate the direction by generating learning data when the drone faces various directions. However, a ground surface vector is similar to a surface vector whose LiDAR acquisition direction has slightly changed, and it is difficult to distinguish them even by using deep learning. Similar to area estimation, it is difficult to learn samples with similar surface vectors but different direction information.

To sufficiently distinguish the directions in which the drone is facing, our improved method uses a four-direction estimator with 90-degree increments. The common direction formed by superimposing the estimation directions of the three estimators is the final estimated direction.

The rest of this paper is organized as follows. In Section 2, we present the learning and evaluation data and the basic algorithm of position estimation based on the ground surface shape. Section 3 describes the multiplexing of area divisions, and Section 4 describes the direction estimation. We discuss the experiments and results in Section 5 and conclude in Section 6 with a summary and mention of future work.

II. POSITION ESTIMATION USING SURFACE VECTOR

The flight area of a drone can be estimated by using pattern matching between the 3D topographical map and the ground surface shape acquired by the drone. In this case, it is more efficient to perform pattern matching between 3D shapes acquired using 3D LiDAR. However, 3D LiDAR is heavy and difficult to mount on small drones. Lightweight 3D LiDAR has also become available [14]. However, 3D LiDAR that can be measured over long distances has a narrow vertical field of view, and the data that can be obtained is not much different from that of 2D LiDAR. In addition, the sensor or mirror rotation type used in many small 3D LiDARs has the problem that the scanning point shifts by the distance that the drone has flown during the scanning time when it flies at high speed.

Therefore, in this work, we equipped the drone with a compact, lightweight 2D LiDAR that was originally developed for automated driving and used it to acquire a 2D surface cross-section (Fig. 1).

The estimation of the flight area of a drone using deep learning is performed in two steps. First, the 2D surface vector is acquired by the 2D LiDAR mounted on the drone. Next, the area is estimated from the 2D surface vector by using a network learned in advance. In this section, we describe the construction of the learning and evaluation data.

A 2D LiDAR (LD-MRS, SICK AG) is mounted on the drone to acquire the ground cross-sectional shape (Fig. 2). The altitude of the ground surface can be measured to an accuracy of about 20 centimeters if a LiDAR with an angular resolution of 0.125 degrees is mounted facing vertically downward and the UAS flies 30 meters above the ground [16]. In our experiments, we set 100 measurement points (one every 50 centimeters on the left and right of the flight path) and recorded the differences in height between these points as a 100-dimensional 2D surface vector. The height of the drone was fixed at 30 meters from an average altitude (Fig. 3). The method itself is capable of handling various altitudes, enabling the area in the height direction to be estimated simultaneously [11]. However, we chose to set the altitude to a constant level to prevent difficulty in verifying the basic performance of the method due to the large amount of training data. Another reason is hardware limitations. LD-MRS can measure up to 150 meters, but as the altitude increases, it becomes more difficult for the laser to reach the ground, which increases the number of points that are difficult to measure. Thus, the altitude was set at 30 meters.

Fig. 2. Matrice 600 [15] equipped with LD-MRS.

We divided the flight area into a lattice and assigned different labels to its elements. The initial position of the drone was randomly determined, and a set of the ground shape vectors calculated from the 3D map and the labels of the area were used as learning data.

We used an AW3D global high-resolution 3D map acquired by Maxer (formerly DigitalGlobe) satellites [17]. Maxer's imagery provides high-precision elevation models

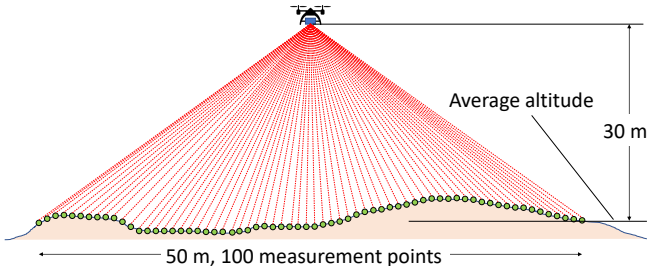

Fig. 3. 100 measurement points.

at 0.5-m to 2-m resolution.

Pattern matching between the 3D map and the ground surface vector acquired by 2D LiDAR typically requires numerous matchings, which makes it difficult to perform in real time. Thus, we estimated the flight area from the ground surface shape in real time by using a multilayer perceptron (MLP) [18]. It is possible to use a convolutional neural network [19] or support vector machine [20] as a discriminator, but since real-time performance is important for drone position detection, we decided to use the MLP, which has a simple structure and fast processing speed.

The MLP for estimating the flight area is shown in Fig. 4. The input for the experiment was 100 units corresponding to the 100-dimensional ground surface vector. Each dimension of the surface vector was normalized to zero mean and variance one. The output units of the MLP were area labels. The MLP had n hidden layers; i.e., we varied n and the number of units in the experiments. The input layer, intermediate layers, and output layer were all fully connected.

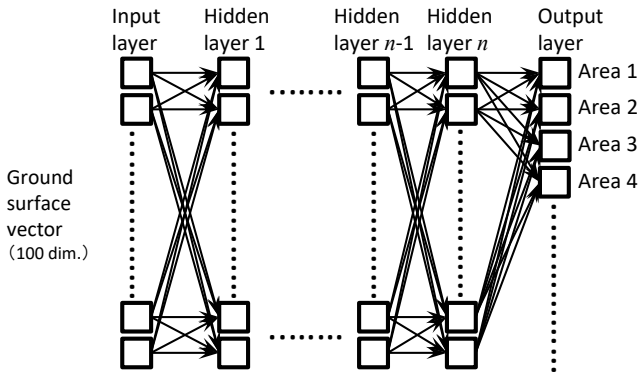

Fig. 4. MLP for estimating area labels.

Supervised learning with back-propagation was used to determine all of the network parameters, including the bias of the units and the weights of the connections between the units.

III. MULTIPLE ESTIMATIONS OF DRONE POSITION

The lattice that divides the area is shifted by a quarter of the size of one area, and the four lattices overlap. Then, overlaying the lattices creates a sub-area that is one-fourth the size of the original area. For example, if we stack four lattices with five 100-meter square areas vertically and horizontally, we will have a 25-meter square sub-area. The range of the identifiable sub-area is 425 meters square in 289 sub-areas (= 17 × 17), where all four lattices overlap (Fig. 5).

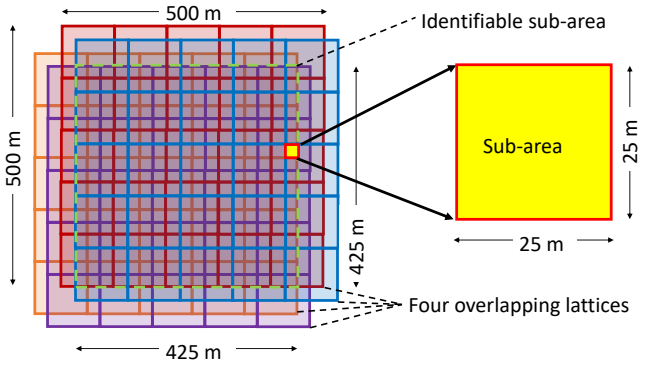

Fig. 5. Four overlapping lattices and sub-area.

When the area label is attached as shown on the left of Fig. 6, the sub-area indicated in black on the right of Fig. 6 with lattice 1 has the area label 6, lattice 2 is labeled with 1, lattice 3 is labeled with 2, and lattice 4 is labeled with 1. That is, the combinations of labels on the four grids in each sub-area are all different and identifiable.

In the figure, it appears as though one area is estimated by each of the four lattices, but in fact, the probability of being in each of the 25 areas is calculated. Then, the probabilities are added to estimate which sub-area is the highest.

IV. DIRECTION ESTIMATION USING SURFACE VECTOR

It is possible to estimate the direction of the drone by dividing the directions into twelve. However, if the 12-direction estimator is made directly, the accuracy will be very low because many learning and estimating data are near the boundary. Therefore, we make three 4-direction estimators that are offset by 30 degrees. Then, we superimpose them and identify the overlapping part to construct the 12-direction estimator.

There are 100 LiDAR scan points across 50 meters perpendicular to the drone (Fig. 7). The center point of the drone is the center of the LiDAR scan. We rotate the LiDAR scan around the center point of the drone with a resolution of 1 degree to create surface vectors in all directions. Labels are added to the surface vector in accordance with the direction of the drone.

We set three of the aforementioned four-direction estimators with an angle of 30 degrees and assign a different area label to each. At the time of estimation, the direction in which the estimation results of the three estimators overlap is set as the final estimation direction (Fig. 8). Similar to the sub-area estimation, the figure appears to show that one of the directions is estimated by all three direction estimators, but in fact, the existence probabilities of each of the four directions are calculated. Then, we estimate the direction in

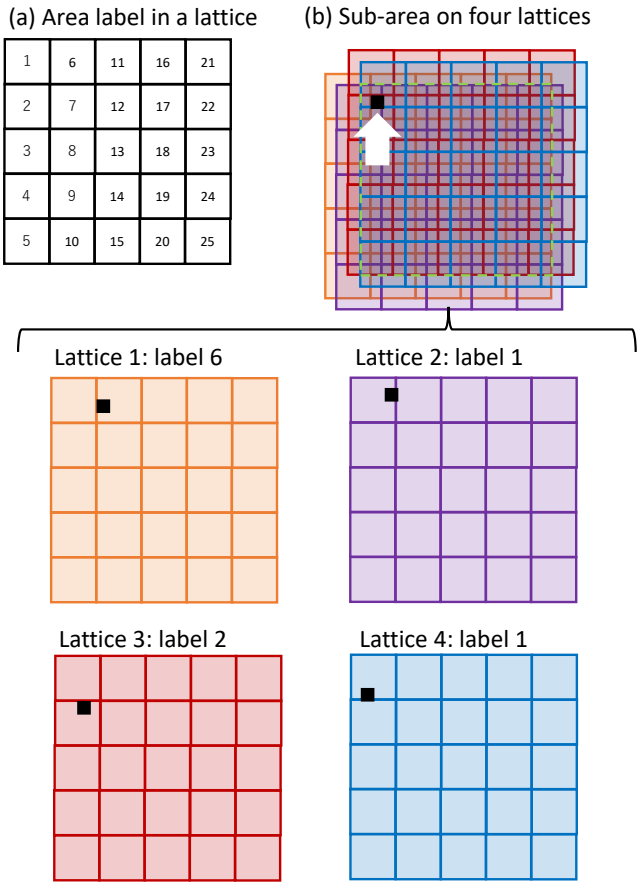

Fig. 6. Area label of sub-area.

which the highest probability is calculated by adding the probabilities.

V. IMPLEMENTATION

We constructed a deep neural network by inputting a 100-dimensional surface vector and outputting position labels for each of the four lattices and direction labels for each of the three estimators. The first to fifth layers are fully connected layers, and ReLU was used as the activation function (Fig. 9).

The sixth to eighth layers are also fully connected layers but are branched for each output, and ReLU was used as the activation function. Softmax was used as the activation function from the eighth layer to multiple outputs. As for the number of units, it is known that the accuracy increases due to avoiding overfitting when the number of units decreases by 0.6 times from the first layer until one layer before the output layer, so we utilized this design here as well [11]. The number of units in the first layer was set to 1800, which was highly accurate due to random sampling.

It is possible to calculate sub-area and direction probabilities from the output of the trained network shown in Fig. 9. In addition, the network can be fine-tuned by separating the output shown in Fig. 9 and combining the 286 sub-areas and 12 directions as outputs (Fig. 10).

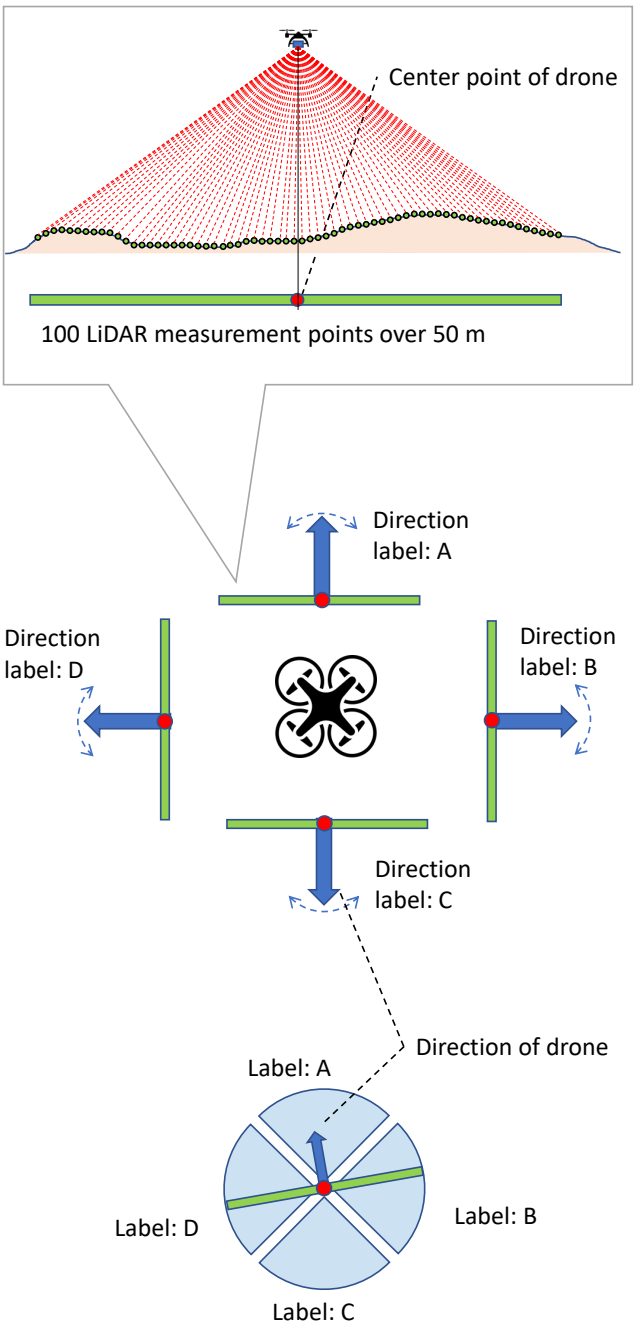

Fig. 7. Four direction labels.

VI. EXPERIMENTAL RESULTS

The network described in the previous section was learned and the performance was then evaluated. We used an NVIDIA RTX 4090 for learning. The experimental data is 600 meters square of a W3D global high-resolution 3D map acquired by Maxer satellites, and its center is 139.108 east longitude and 36.624 north latitude. Four 500-meter square areas were taken in a 600-meter square area with a 25-meter shift (Fig. 11). Each 500-meter square was divided into 25 and labeled. We randomly sampled 80% of all the data from the 289 (= 17 × 17) sub-areas as training data and the

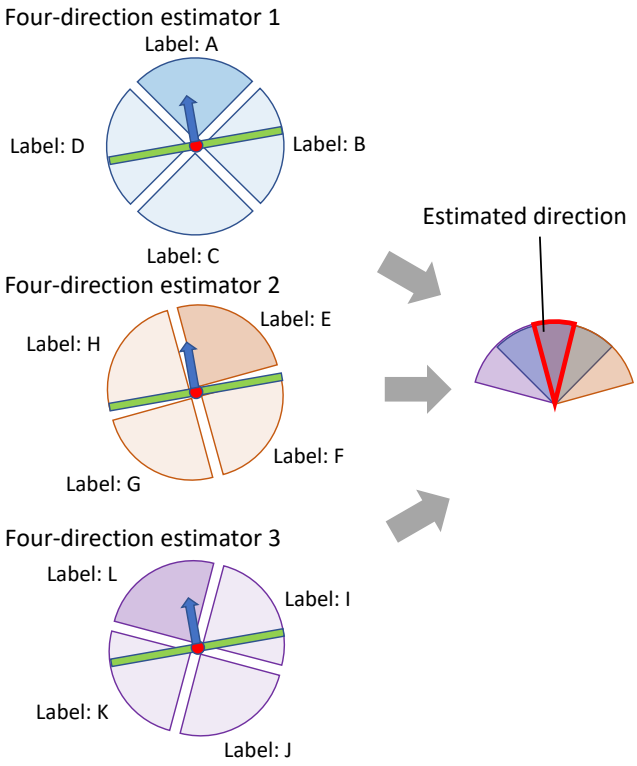

Fig. 8. Estimation of 12 directions.

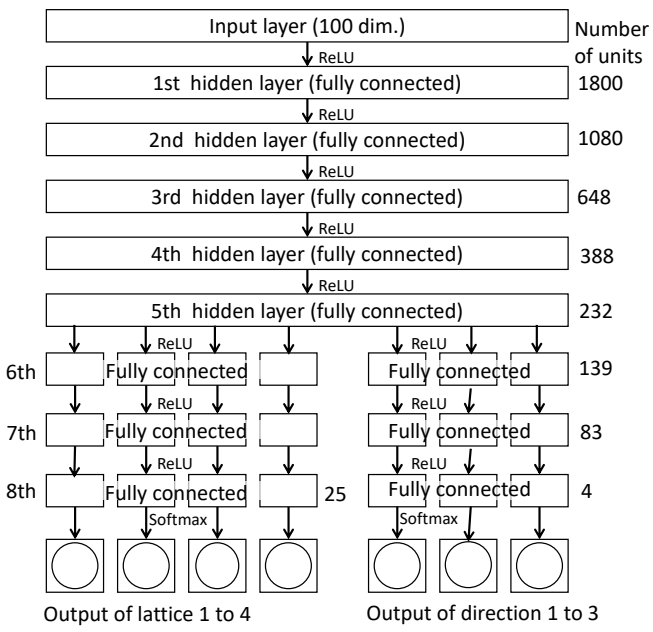

Fig. 9. Implementation of deep neural network.

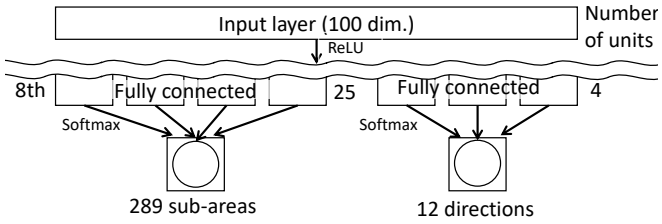

Fig. 10. Fine-tuning network.

remaining 20% was validation data. We initially used Adam as the optimizer, but the validation accuracy decreased in the middle of learning, so we used RMSprop instead.

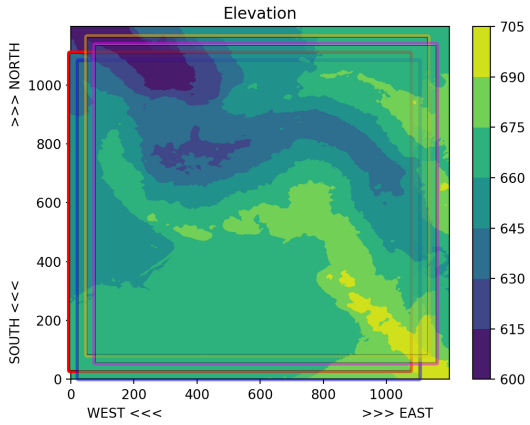

Fig. 11. 3D map acquired by Maxer satellites.

A. Positon Estimation Results

After learning for 1000 epochs, the performance of 25 areas of each lattice was estimated as shown in Table I. Figure 12 shows the learning curve of lattice 1.

TABLE I
ESTIMATION ACCURACY OF EACH LATTICE

Lattice 1	Lattice 2	Lattice 3	Lattice 4
85.6%	86.1%	86.2%	85.7%

By superimposing the four grids, 289 sub-areas were created. When the sub-areas were estimated by adding the probabilities that drones exist in each area, the accuracy was 67.4% at a chance rate of 1/289.

We created a heat map to clarify what kind of position estimation errors occurred. As shown in Fig. 11, we found that 32.6% of the data was incorrect. The blue square indicates the estimated area, and the red square indicates the correct area. The numbers in the heat map are the sum of the correct answer probabilities of the four lattices, where the maximum and minimum values are 4 and 0, respectively. Even if the estimation was incorrect, as shown in Figs. 13(a)–(c), the adjacent position was often inferred. However, in a number of cases, another position was estimated, as shown in (d).

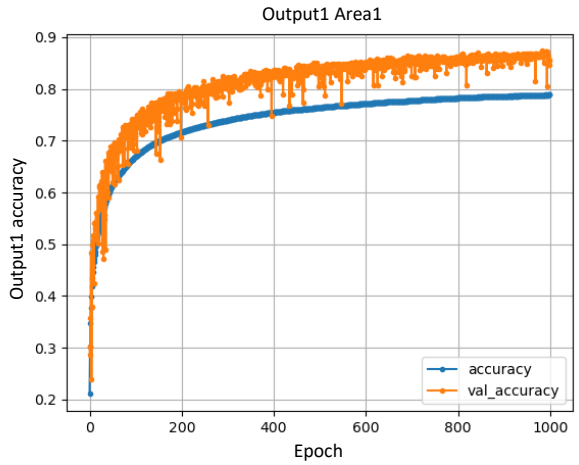

Fig. 12. Learning curve of lattice 1.

B. Direction Estimation Results

The accuracy of each estimator for direction estimation is listed in Table II. Figure 14 shows the learning curve of direction estimator 1.

TABLE II
ESTIMATION ACCURACY OF EACH DIRECTION ESTIMATOR

Estimator 1	Estimator 2	Estimator 3
79.0%	78.5%	78.9%

When the directions were estimated by adding the results of the three direction estimates, the accuracy was 59.4% at a chance rate of 1/12.

The graph shown in Fig. 15 was created to clarify the errors that occurred in direction estimation. We found that 40.9% of the data was incorrect. The estimated and correct directions are shown in orange and brown, respectively. The polar coordinate numbers are the sum of the correct answer probabilities of the three estimators, with a maximum and minimum value of 3 and 0, respectively. As shown in Figs. 15(a)–(c), there were many cases where the adjacent direction was estimated even if the estimation was incorrect. However, in a number of cases, another direction was estimated, as shown in (d).

C. Fine-Tuning Results

We improved the resolution by summing the output of four lattice probabilities for position estimation and the outputs of three estimators for direction estimation. In this method, if even one of the lattices or estimators is misestimated, the final position or direction will be wrong. Therefore, the estimation accuracy was low compared with one lattice or one estimator.

Even when an estimation error occurred, there were many instances in which the estimated area was adjacent to the correct area due to a slight difference in probability. If this probability can be adjusted, there is a possibility that the accuracy can be improved. Therefore, we performed fine-tuning using the network in Fig. 10, in which 289 sub-area units and 12 directional units with softmax functions are combined with the trained network from the input to the eighth layer.

After fine-tuning, the accuracy of position estimation with the 25-meter square resolution was 96.8%, and the accuracy of 12-direction estimation was 92.6%. Learning did not progress when the network in the initial state was used instead of fine-tuning.

VII. CONCLUSION

We proposed a method for estimating the position and direction of drones by learning the ground surface vector acquired by LiDAR with a multi-output deep neural network. The main contributions of this paper can be summarized as follows.

Improved resolution of area estimation: While the size of the area to be estimated was 100 meters square, the area estimation multiplexing made it possible to estimate a sub-area of 25 meters square. The sub-area could be estimated with an accuracy of 67.4% at a chance rate of 1/289. Even for the incorrect estimation, which was 32.6%, most of the errors were related to the adjacent sub-area.

Direction estimation: Previously, it was assumed that the drone would always fly northward, but by learning the surface vectors in all directions (i.e., 360 degrees), we were able to estimate the direction of the drone. It was possible to estimate the sub-area with an accuracy of 59.4% at a chance rate of 1/12. Most of the incorrect answers (40.9%) were estimation errors in adjacent directions.

Fine-tuning: After fine-tuning using a network that combines 289 sub-areas, 12 directional units, and the learned network from input to the eighth layer with softmax, the position estimation accuracy with a resolution of 25 meters square was 96.8%, and the 12-direction estimation accuracy was 92.6%.

Improved resolution in estimating drone position and orientation: Our previous deep learning-based position detection method suffered from a drop in accuracy when we attempted to improve the resolution. The experimental results showed that multiple lattices and direction estimators can improve resolution without a significant loss of accuracy.

Our next steps are as follows.

Further improvement in resolution: Our proposed method can further improve the resolution by setting the shift of the lattices and direction estimators very small. We will verify the extent to which the resolution can be improved without a significant drop in estimation accuracy.

Network construction corresponding to changes in altitude: In the experiment, it was assumed that the drone would fly at a fixed altitude of 30 meters, but in actual flight, altitude changes occur. Therefore, we plan to divide the area in the altitude direction and perform learning and estimation.

Learning in a large area: A problem with current learning networks is that once the drone leaves the training area, it loses its position. To prevent this, we need to prepare

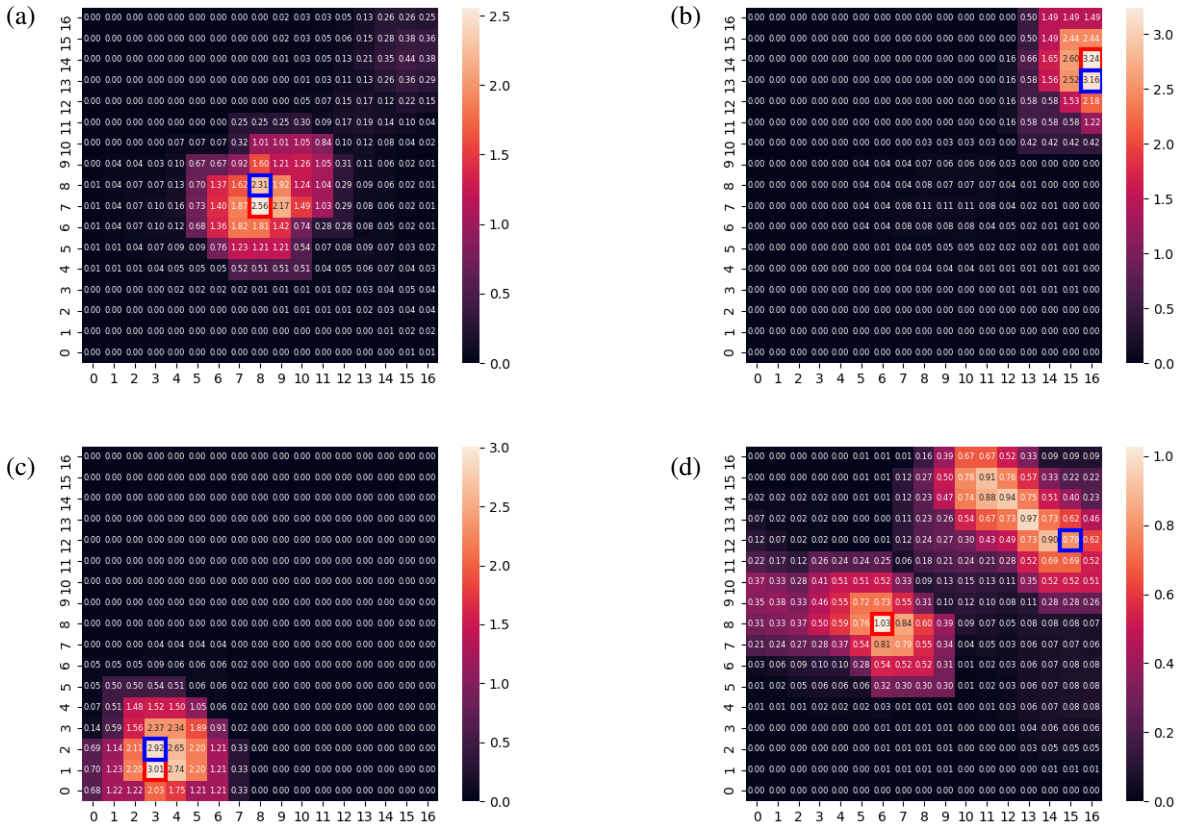

Fig. 13. Heat map of position estimation error.

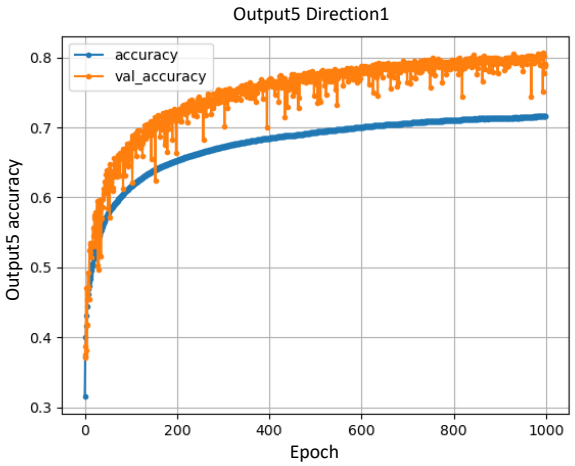

Fig. 14. Learning curve of direction estimator 1.

a large labeled area around the area where the flight is planned. We are planning to evaluate the performance in a large area.

Non-repetitive scanning 3D LiDAR: Although 2D LiDAR was used in the experiments in this paper, a number of 3D LiDARs perform non-repetitive scanning [21]. They produce a scanning pattern in which the laser resembles the shape of a flower. We plan to enable faster estimation by inputting one petal pattern or one curve pattern into the network.

Field flight experiments: A trained network can estimate position and direction in real time with a small computer. We plan to carry out a field flight experiment by mounting the computer on an unmanned aircraft operated by the Japan Aerospace Exploration Agency (JAXA).

ACKNOWLEDGMENT

This work was supported by the Japan Society for the Promotion of Science (JSPS KAKENHI Grant Number 17K19972).

REFERENCES

[1] M. Ng, F. Vanegas, K. Morton, J. Sandino, and F. Gonzalez, "Design and flight test of an aerial manipulator for applications in gps-denied environments," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, July 2022, pp. 20–29. [Online]. Available: https://ieeexplore.ieee.org/document/9836060

[2] B. Gong, S. Wang, M. Hao, X. Guan, and S. Li, "Range-based collaborative relative navigation for multiple unmanned aerial vehicles using consensus extended kalman filter," *Aerospace Science and Technology*, vol. 112, p. 106647, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1270963821001577

[3] M. Karunanayake, E. Cannon, and G. Lachapelle, "Evaluation of assisted GPS (AGPS) in weak signal environments using a hardware simulator," in *Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, Septmber 2004, pp. 2416–2426. [Online]. Available: https://www.ion.org/publications/abstract.cfm?articleID=5926

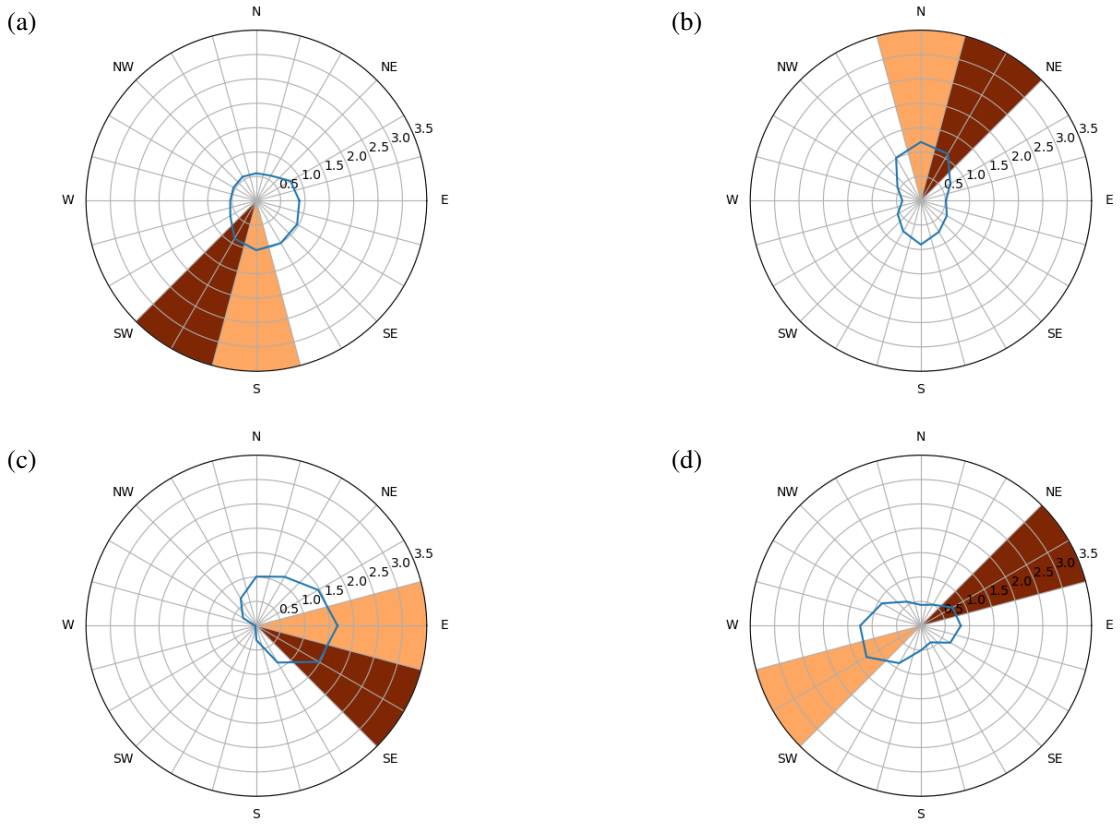

Fig. 15. Graph of direction estimation error.

[4] K. Nur, S. Feng, C. Ling, and W. Ochieng, "Integration of GPS with a WiFi high accuracy ranging functionality," *Geo-spatial Information Science*, vol. 16, no. 3, pp. 155–168, 2013. [Online]. Available: https://doi.org/10.1080/10095020.2013.817106

[5] T. Iwata, Y. Kawasaki, M. Imae, T. Suzuyama, T. Matsuzawa, S. Fukushima, Y. Hashibe, N. Takasaki, K. Kokubu, A. Iwasaki, F. Tappero, A. Dempster, and Y. Takahashi, "Remote synchronization system of quasi-zenith satellites using multiple positioning signals for feedback control," *NAVIGATION*, vol. 54, no. 2, pp. 99–108, 2007. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2161-4296.2007.tb00397.x

[6] T. Yang, P. Li, H. Zhang, J. Li, and Z. Li, "Monocular vision slam-based uav autonomous landing in emergencies and unknown environments," *Electronics*, vol. 7, no. 5, 2018. [Online]. Available: https://www.mdpi.com/2079-9292/7/5/73

[7] M. Hamanaka and F. Nakano, "Surface-condition detection system of drone-landing space using ultrasonic waves and deep learning," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 1452–1459.

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

[9] N. H. Malle, F. F. Nyboe, and E. Ebeid, "Survey and evaluation of sensors for overhead cable detection using UAVs," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, June 2021, pp. 361–370. [Online]. Available: https://ieeexplore.ieee.org/document/9476724

[10] A. Moffatt, E. Platt, B. Mondragon, A. Kwok, D. Uryeu, and S. Bhandari, "Obstacle detection and avoidance system for small UAVs using a LiDAR," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, September 2020, pp. 633–640. [Online]. Available: https://ieeexplore.ieee.org/document/9213897

[11] M. Hamnanaka, "Deep learning-based area estimation for unmanned aircraft systems using 3d map," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2018, pp. 416–423. [Online]. Available: https://ieeexplore.ieee.org/document/8453463

[12] J. C. Caicedo and S. Lazebnik, "Active object localization with deep reinforcement learning," in *2015 IEEE International Conference on Computer Vision (ICCV)*. Los Alamitos, CA, USA: IEEE Computer Society, dec 2015, pp. 2488–2496. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICCV.2015.286

[13] A. Kendall and R. Cipolla, "Geometric loss functions for camera pose regression with deep learning," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, jul 2017, pp. 6555–6564. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR.2017.694

[14] Ouster, "Os2 long-range lidar sensor for autonomous vehicles, trucking, and drones," 2023, https://ouster.com/products/scanning-lidar/os2-sensor/.

[15] Dji, "Matrice 600," 2023, https://www.dji.com/matrice600.

[16] S. AG, "The ld-mrs family – compact laser scaners for outdoor use," 2009, https://www.cimtecautomation.com/industrial-automation-resources/resources/SICK/SICKLitFolder/LMS/SICKLD-MRSBrochure.pdf.

[17] J. Takaku, T. Tadono, K. Tsutsui, and M. Ichikawa, "Validation of "aw3d" global dsm generated from alos prism," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. III-4, pp. 25–31, 2016. [Online]. Available: https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/III-4/25/2016/

[18] S.-i. Amari, T. Ozeki, R. Karakida, Y. Yoshida, and M. Okada, "Dynamics of Learning in MLP: Natural Gradient and Singularity Revisited," *Neural Computation*, vol. 30, no. 1, pp. 1–33, 01 2018. [Online]. Available: https://doi.org/10.1162/neco_a_01029

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, 1998, pp. 2278–2324. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.42.7665

[20] V. N. Vapnik, *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995.

[21] Livox, "Avia lidar sensor," 2023, http://www.livoxtech.com/avia/.