A UAV Exploration Method by Detecting Multiple Directions with Deep Learning

Duc Viet Bui, Tomohiro Shirakawa, Hiroshi Sato.
National Defense Academy of Japan, Department of Computer Science, Japan
Email: vietviet2411@gmail.com

***Abstract*—Recently, autonomous exploration using robots has been researched and developed for different objectives and requirements. The advancement in image processing using deep learning has made some remarkable results in controlling UAV autonomously in the situation without GPS information. However, there are few types of research on autonomous image-based exploration, especially in the situation that requires the ability to recognize and predict multiple directions from images, which is an important key to perform pathfinding in an exploration mission correctly. In this paper, we propose an approach for this problem by applying a supervised-learning method to predict possible directions from images. We introduce a deep learning architecture using the transfer learning technique to evaluate our dataset. The experiment results show the promising capability of the model for handling situations with multiple directions.**

***Index Terms*—UAV, exploration, GPS, multiple directions, monocular camera, deep learning, transfer learning**

I. Introduction

In recent years, there has been significant progress in aerial robotics, driven by the rapid development of inexpensive drones and their practical applications. Unmanned Aerial Vehicles (UAV) have been deployed in various fields, including aerial surveillance [1], precision agriculture [2], intelligent transportation, military, search and rescue operations, and more.

Recently, although various approaches have been developed to navigate UAV [3] autonomously, the Global Positioning System (GPS) has emerged as the most prevalent one [4]. However, GPS technology is not ideal for real-time applications as GPS would be either inactive or not powerful enough in both indoor and outdoor environments [5] in which the GPS signals could be blocked by high buildings and trees. In addition, GPS systems of civilian use are much less accurate than the military GPS system, leading to a difficulty for countries lacking GPS military systems [5]. Therefore, autonomous AUV navigation in civilian GPS has been a challenging study so far.

In the last few years, many solutions for autonomous navigation in aerial robotics have been put in practice. For example, quite a few methods of autonomous navigation in ground robotics have been proposed in the literature. In [6]-[9], 3D environmental models are produced from using sensors such as laser range finders (LIDARs), RGB-D sensors, stereo vision and recent complex algorithms such as Simultaneous Localization and Mapping (SLAM), which results in the relative position between UAVs at any instant. This also helps in finding the relative position of the device at any instant of time. However, complicated algorithms demand computationally heavy processes as well as heavy-weight expensive sensors that limit the applications for light-weight UAV. Furthermore, because the laser pulses depend on the principle of reflection, LIDARs may not work well in areas or situations where there are high sun angles or huge reflections, and RGB-D also may not have good accuracy in an outdoor environment. The drawback of both sensors usages and algorithms could lead to a critical error when operating in real-world environments or confronting texture-less surfaces, which are often present in the indoor scenes.

Recently, vision-based navigation has attracted attention in the field of aerial robotics due to its applicability to commercial quadcopters which are commonly equipped with a forward-looking camera. This leads to various developments of research mainly focusing on using only monocular cameras on UAV, such as the adaption of vision-based SLAM technique (Visual SLAM) [10], [11]. Simultaneously, the advancement of machine learning and deep learning in fields of computer vision have shown the capabilities of applying the vision-based technique on UAV. Among these researches, obstacle detection and avoidance methods are mainly focused, as they are important steps toward safe autonomous UAV exploration and navigation [12], [13]. However, when it comes to exploration missions, the two methods raise a challenge. The exploration task requires the robot's capability to correctly detect all the available spaces in the environment around it, then the next movement is determined upon the situation. Particularly, in situations with multiple directions, recent deep learning approaches can only predict the next available direction for autonomous UAV but cannot recognize other directions that are also available in such situations, which is an important key for pathfinding task in exploration missions.

In this paper, we address the problem of predicting multiple directions in the UAV exploration mission using a monocular camera's input by employing a deep

Manuscript received November 6, 2019; revised March 4, 2020.

doi: 10.18178/ijmerr.9.10.1419-1426

learning approach. We firstly do our tests on an indoor environment with our custom dataset. The contributions of the paper can be summarized by the following:
(1) We propose a human's perspective based method using deep learning architecture for predicting multiple directions in the UAV exploration mission.
(2) We provide our custom dataset for the proposed method. The dataset contains our processed images at different positions from corridors in buildings.

The remaining part of this paper is organized as follows. In section 2, we present an overview of the related researches done in this field, focusing on the used method. Section 3 explains our proposed method on UAV exploration tasks, the custom dataset creation, and the deep learning architecture used to accomplish the task. Section 4 demonstrates our experiment's results and analysis. Finally, in section 5, we offer the conclusions and our next plans on the research.

II. Related Works

A UAV is usually provided with GPS and sensors to estimate its position on the global map to detect obstacles. Several researches have obtained the localizations and path planning of UAVs based on GPS. However, those works are unsuccessful in the places where either GPS access is denied (urban environment with the presence of high buildings, trees; indoor scenes) or GPS positioning information is temporary incorrect due to geographical changes or natural disasters (earthquake, typhoon), making them hard for UAV autonomous explorations.

Due to the disadvantage of GPS in such situations, the SLAM algorithm is a good candidate to produce a 3D map of the surrounding environment using non-visible data sources (such as radar, LIDARs) or visual data (such as cameras). By using these sensors, data received from the surrounding environment is huge, which makes it easier for UAV to simultaneously build a 3D map of the environment and self-localizes in it, as well as computing the next available directions for driving based on the created map. H. Michael Tulldahl et al. [14] performed experiments to demonstrate 3D mapping capabilities from a small multirotor UAV with the Velodyne HDL-32E LIDAR. Bachrach et al. [15] generated a 3D map using an RGB-D camera with the help of the SLAM algorithm, which is later used for localization and path planning in an unknown corridor environment. Bry et al. [16] combined an inertial measurement unit (IMU) with an RGB-D camera to localize MAV and enable reliable flight for localization task in indoor scenes. Nevertheless, the major drawback with SLAM is that the 3D map regeneration is very complex, which requires significantly high computational cost and power consumption. Moreover, to create a detailed map for navigation with SLAM, besides the sensors mentioned above, it also requires additional metric sensors, which is difficult for applications on light-weight UAV. Additionally, SLAM is a feature-based method, so it may not give desirable results with the indoor surface (walls, floors) [17] as the intensity gradient on these conditions is very poor. This makes the SLAM technique may not be suitable for autonomous exploration in indoor environments.

To deal with these problems of SLAM, researchers have paid attention to vision-only approaches. They only use monocular cameras as input for localization and mapping, with the implementation of machine learning / deep learning (ML/DL) techniques, which obtained good results in fields of image processing [13], [18]-[22]. Most of the recently introduced works which involved these approaches can be divided into two types. One is the trial-and-error learning strategy, which is known as Reinforcement Learning (RL) [18], [19], and the other is a supervised-learning method that enables the development of end-to-end learning strategies. In supervised learning, the feature extraction and learning are performed by using a huge set of learnable parameters from the researcher's handcrafted selected features [13], [21], [22].

RL approaches often focus on correlating raw camera's inputs with UAV's control command and combine with the RL algorithm to make model learn by demonstration. In [18], Lillicrap et al. proposed a system that applied RL to end-to-end policies for many classic continuous physics problems. Also, Ross et al. [19] describe an RL model that learned to avoid obstacles in the context of a UAV flying in the forest. However, RL models usually require a huge amount of experience, so lacks training conditions lead to limitations in generalizing the model's capabilities, which then raises safety concerns on controlling UAV correctly and handling crashes in the real-world environment's experiments. Learning RL control policies can also be implemented through simulators (AirSim, Gazebo ROS), which is described in several researches [20], [23]. However, the gap between empirical and simulation models still exists and thus, makes these policies hard to carried out in the physical world.

Supervised-learning based approaches offer a more viable way to learn control policies and apply them to real-world conditions. These approaches often based on imitation learning, in which a human expert controls UAV in a real-world environment to collect input images/ pilot's choices upon situations. Collected pilot's choices later are used as ground truth labels for images in training an ML/DL model, to make models imitate human's behavior in different situations. Previous works in [21] and [22] developed a system in which the DL model was trained from video collected only by GoPro cameras, and later successfully flew an autonomous UAV that can follow trails in the forest. Following these works, state-of-the-art research has been introduced by Loquercio et al. [13], in which a DL model was trained from data collected by cars and bicycles in the urban environment, which later demonstrated that this approach could also be implemented in cities.

However, the works mentioned above mainly focus on the obstacle avoidance task of an exploration mission, but not the pathfinding task. The researches only provide the capability of predicting the current condition and estimating the next available control command for the

situation. This is necessary for UAV to autonomous safely avoid obstacle when traveling in one direction, but not for the pathfinding task, which requires the capability to recognize other directions that are also available in the situation. This is even difficult for a normal person to predict from a single image. Models in previous works [21], [22] may fail to predict the directions and the commands correctly. For example, in Fig. 1, most of the previous ML/DL works [21], [22] may be prone to estimate only the "Moving forward" command, as there aren't any obstacles in front of the UAV. However, it can be seen from the normal perspective that we can turn the UAV to the left or right to explore the map instead of only moving forward.

Figure 1. A common situation that requires UAV's ability to recognize possible directions.

This problem requires well-annotated data with an appropriate learning strategy, as it's even difficult for a normal person to discriminate in different situations and estimate all available directions only from an image's information. In this paper, to predict all the possible directions that a UAV can go in various situations, we proposed a DL model with a supervised-learning strategy that can run inferences depends on various situations from each camera input's image, which is similar to the works mentioned above. The difference of this paper from the previous one is that in each position, instead of classifying every input image to the pilot's current command or angular velocity, we introduce a method that builds robot perception based on a human's normal perspective. Even though the human imitation approach may have limitations in generalizing UAV's capability of driving safely, our goal is to prove that we can deploy this concept of human's normal perspective on the exploration and navigation task of UAV.

III. Proposed Method

A. Problem Formulation

In this paper, our approach aims at estimating multiple possible driving directions in encountered situations from single image input. When ordinary people are controlling a UAV in a situation like Fig. 1, it's easy to acknowledge from the image that this situation has three choices of moving direction: Moving Forward, Turning Left, or Turning Right, as the road and building form the available directions clearly. From that point of view, we assume that for every situation, if UAV can estimate all the available commands correctly similar to a normal person, then it may offer more driving options for UAV and enable the development of map exploration algorithms without creating a 3D map.

To analyze the capability of detecting multiple directions on every situation, we conduct our experiments on indoor scenes, specifically the corridor. The corridor is usually covered by walls, roofs, and floors, which makes the available directions not very difficult for a single person to recognize. To our knowledge, many public datasets are covering indoor scenarios [24]. However, they are not useful in our work as none of them provide ground truth values related to directions in each image. We also consider the fact that manually labeling these datasets with our perspective may increase the chance of receiving errors in ground-truth value, as we're not fully aware of those dataset's conditions. The inaccuracies in ground truth may cause a faulty training process, which leads to undesirable results in terms of detecting available directions. Therefore, the creation of a custom dataset for our approach is needed to achieve the objective of the research.

Following the image processing related works [13], [21], [22], our dataset consisted of images that are captured with a camera from different positions of the corridor. However, in the dataset made by previous works, only one specified command is assigned as a label for each situation. On the other hand, in our custom dataset, we assigned multiple commands to each situation as a label because we want to leave the space to UAV to choose the possible alternative behaviors in the situation. We assign the situations one by one to class labels and use these labeled data to develop a supervised-learning DL model that can predict UAV's encountered situations with their possible commands. The common situations with selected driving options in our custom dataset are illustrated from Fig. 2. To Fig. 8.

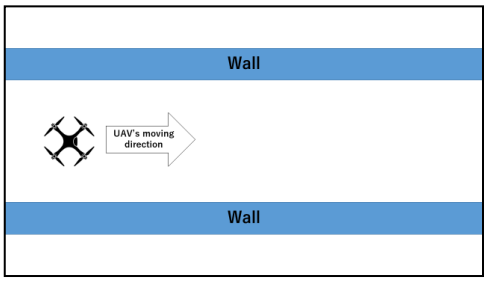

Figure 2. Situation 1: Moving Forward

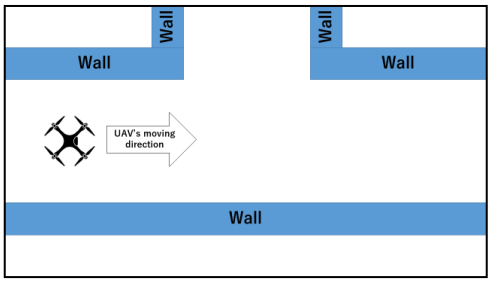

Figure 3. Situation 2: Moving Forward or Turning Left

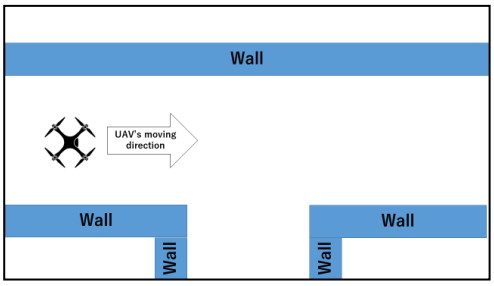

Figure 4. Situation 3: Moving Forward or Turning Right

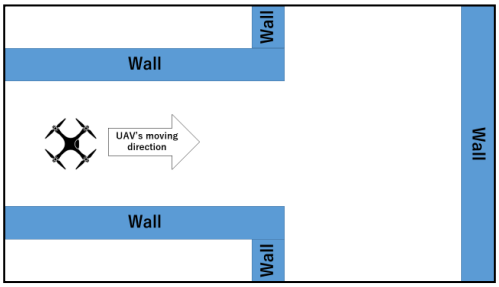

Figure 5. Situation 4: Turning Left or Turning Right

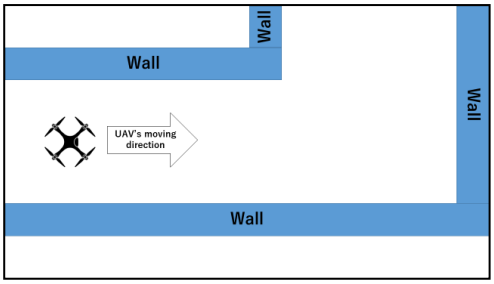

Figure 6. Situation 5: Turning Left

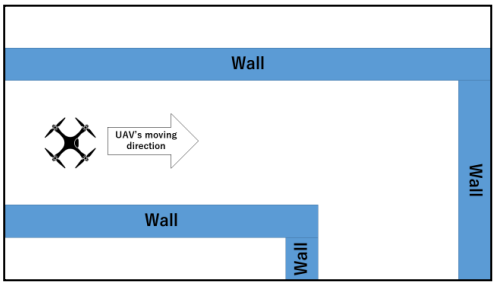

Figure 7. Situation 6: Turning Right

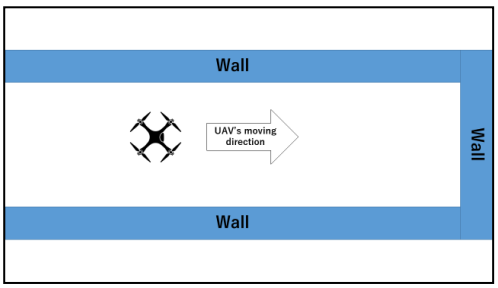

Figure 8. Situation 7: Stop

Hence, we will have seven situations with corresponding possible flight commands. Locations with stairs on the left or right will also be counted as conditions that can change direction in our dataset. One more thing to consider in making a dataset is that we don't want the UAV to predict turning left or right (for example, situation number 4, 5 or 6) too early, especially when UAV is still far from those positions. As a result, we decided to label the images from the position with distance to the above-mentioned location less than 1 meter, which we believe it's an appropriate distance for predicting directions.

B. Dataset

To gather data, we use a GoPro Hero 4 camera to record video footage of all the corridors in different buildings. In every building, we capture the videos with two different height (approximately 1m and 1.5m), as we consider that these are sufficient heights for any UAV to fly in any corridor. We also expect that the changes in observation's height may also increase the capacity of generalizing situations in multiple views. Video resolution is set to 1080p (1920x1080) with the frame rate of 30fps. GoPro Hero 4 camera has three types of FOV (Field of View): Narrow, Medium, Wide. Generally, FOV is calculated by the amount of space between the lens and the image sensor: the further the lens is from the sensor, the narrower the FOV. Each FOV's capture angle is shown in Fig. 9.

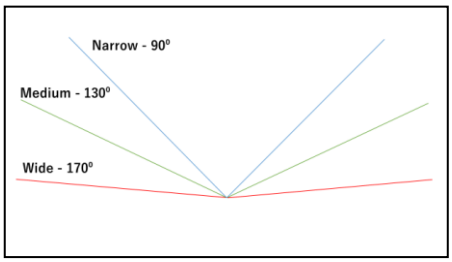

Figure 9. Three types of FOV(angle): Narrow (Blue), Medium (Green) and Wide (Red)

After compared captured images from different FOV settings (shown in Fig. 10), we set the camera's FOV to Wide mode, as we assume that the wider the view is, the more information we can have to predict the situation. The number of collected images is shown in Table I.

TABLE I. NUMBER OF IMAGES IN EACH SITUATION

No.	Situation	Number of images
1	Moving Forward	10434
2	Moving Forward or Turning Left	6292
3	Moving Forward or Turning Right	6459
4	Turning Left or Turning Right	1317
5	Turning Left	1695
6	Turning Right	726
7	Stop	515

Due to the limitations in finding locations with multiple directions, our dataset is imbalanced in some classes. To deal with the imbalance in training samples, we use some typical data augmentation method to increase samples before training the model: random zoom (1-1.5); random rotation (max angle: 15); random width

shifting (range: 0.2); random height shifting (range: 0.2); random shear transformation (max angle: 20). The examples of seven situations in our dataset are shown from Fig. 10 to Fig. 16.

Figure 10. Situation 1: Moving Forward

Figure 11. Situation 2: Moving Forward or Turning Left

Figure 12. Situation 3: Moving Forward or Turning Right

Figure 13. Situation 4: Turning Left or Turning Right

Figure 14. Situation 5: Turning Left

Figure 15. Situation 6: Turning Right

Figure 16. Situation 7: Stop

C. Training Process

This study applied the powerful image classification Deep Learning model – Convolutional Neural Network (CNN) combining the Transfer Learning technique to solve our problem.

CNN architecture is powerful in classifying images, which was used in many UAV related works [13], [21], [22]. However, CNN often contains millions of parameters, which is problematic for directly learning on a few thousand training images. To deal with this challenge, researchers have widely used the Transfer Learning technique in many kinds of research [21], [22]. The main idea of this technique is that the hidden layers of the CNN can be used as a feature extractor, which can be pre-trained on one big dataset (source task), and then re-used on other tasks (target task).

In this work, we apply this technique by augmenting several CNN models, which were powerful in the ImageNet Large Scale Visual Recognition Challenge [25] (ILSVRC): VGG16 [26], ResNet50 [27], MobileNet [28], DenseNet-161 [29].

We first removed the final layers of the original models, and then augmented them with some convolutional layers and fully connected layers at the end. The image's input size has been resized to match each model's original input layers, and categorical cross-entropy function has been used to train these classifiers. We split the dataset into three parts: 50% of the dataset for training, 25% for the validation process, and 25% for the testing model's performance. The initial learning rate is set to 0.001. ImageNet's pre-trained weight was applied to each model, and we performed training on 30 epochs with a mini-batch of size 32. The training will stop early if the loss function does not decrease after five epochs. We chose the model with the best accuracy, and

continue fine-tuning this model to increase the model's accuracy and performance on our dataset.

D. Performance Evaluation:

During the training, we used overall accuracy as a metric to evaluate each model's performance. For test data, we used three criteria that are widely used in ML researches: Precision, Recall, and F1 Score. They are computed by computing the number of True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). They are formulated as below:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (1)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2)$$

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

To evaluate the robustness of the proposed method on the real environments, we performed some experiments using a GoPro camera on a different location, which is not included in our dataset. The test location is chosen in terms of different objects, geometry, and lighting. The model was running on a host machine, which has an Intel processor, 32Gb RAM, NVIDIA GeForce RTX 2060 GPU running on Windows 10. Original images are captured at a resolution of 1080x720p, but their size is changed to match the input size before passing through the trained model. After processing the captured images through the classifier, situation prediction is estimated and shown on screen. We track the error predictions on locations with multiple directions on the way to identify the weakness of the proposed model on unknown environments.

IV. Result and Discussions

A. Results

After several experiments, by now, the augmented VGG16 gave the best result on our dataset with 95% accuracy on training data, and the average test accuracy was approximately 84%. The augmented VGG16 model's architecture is described in Fig. 17, and the training process is illustrated in Fig. 18.

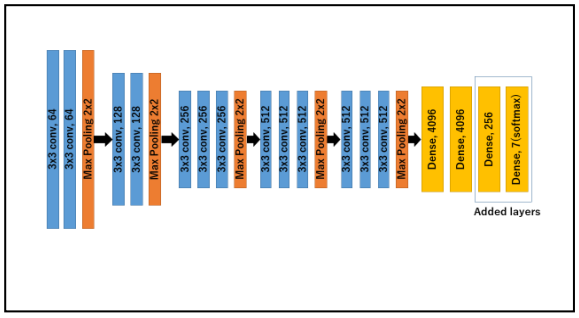

Figure 17. Augmented VGG16 architecture

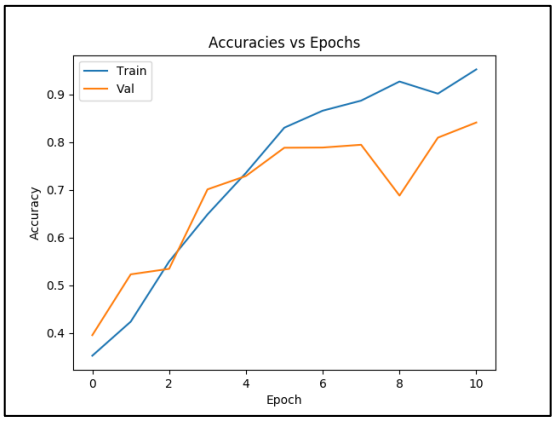

Figure 18. Visualization of training accuracy and validation accuracy on the training process (Augmented VGG16)

The Precision, Recall and F1 Score on the test data is shown in Table II. The performance of the model when furthered categorized by each situation can be illustrated in a form of a normalized confusion matrix in Fig.19.

TABLE II. Precision-Recall-F1 Score

No.	Situation	Precision	Recall	F1-score
1	Moving Forward	0.86	0.96	0.91
2	Moving Forward or Turning Left	0.97	0.83	0.89
3	Moving Forward or Turning Right	0.91	0.83	0.87
4	Turning Left or Turning Right	0.45	0.76	0.57
5	Turning Left	0.73	0.45	0.56
6	Turning Right	0.38	0.46	0.41
7	Stop	0.6	1.00	0.75

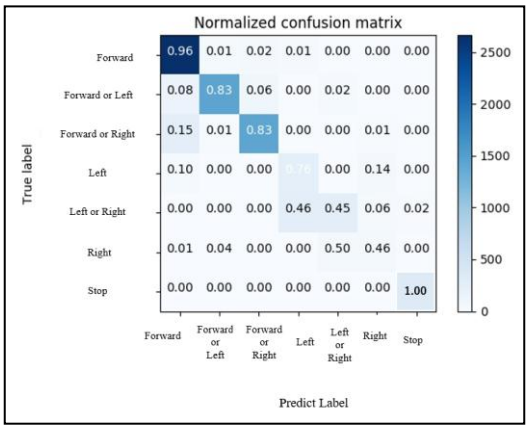

Figure 19. Normalized confusion matrix calculated on test data

The proposed model ran smoothly in real-time conditions (approximately 30 frames per second) with no delay in predicting the situation. Fig. 20 demonstrates the map of a test location, along with the model's predictions in each position that has multiple choices of moving directions (total 10 positions). Upon approaching these positions, sometimes, the model failed to predict the first time correctly. We demonstrated the correct results in blue color, and red color indicated the wrong predictions.

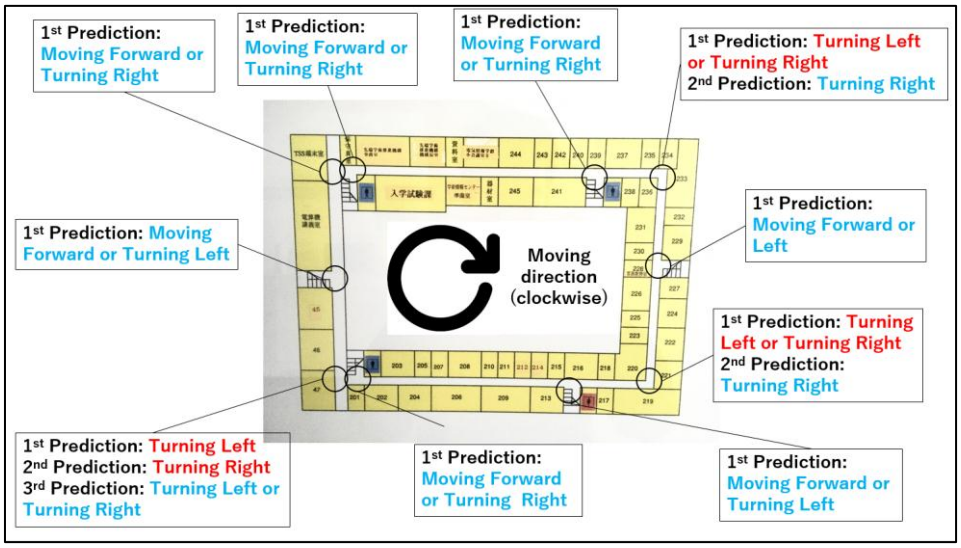

Figure 20. Positions with multiple directions in the test location.

B. Discussions

The proposed model reached 95% accuracy on the training data but gained only 84% on the testing data. From the training process shown in Fig. 18, we can see a sign of overfitting model on training. Although the data augmentation technique has been applied, the training dataset is still small and imbalanced in some classes, which may not be enough for the CNN model to generalize all the features in 7 different situations.

The Precision-Recall result and F1 Score in Table II gave us a more specific view on this problem. Situation 1 (Moving Forward), Situation 2 (Moving Forward or Turning Left), Situation 3 (Moving Forward or Turning Right) and Situation 7 (Stop) achieved the highest prediction rate and F1 Score (over 0.8 in each class), while three classes (Situation 4, 5 and 6) were not as high as the others (F1 Score is approximately 0.5-0.6 in these conditions). This can easily be understood as the number of training data samples in these classes is fewer than the others, which can be seen in Table I. Also, the captured images tend to have similarities when the UAV is getting closed to the wall in these three positions.

The real-time experiment also gave us a view on how well the model performs on an unknown location. Fig. 20 showed that in 7 out of 10 positions that have multiple directions, the model correctly predicted the situation at the first time. The other three positions are recognized in the second/ third time of prediction when the camera finally approached the location. It can be seen that the model demonstrated its weakness when dealing with Situation 4, 5, and 6, as explained above. Humans and different objects sometimes appeared on the way, but the model still predicted the current situation correctly.

V. Conclusion and Future Works

In this paper, we presented a vision-based approach for detecting possible directions in autonomous UAV exploration missions. Our method is based on a normal human's perspective view when piloting a UAV in the real environment. We presented the Deep Learning architecture with Transfer Learning adoption that analyzes and learns to classify each situation from a provided custom indoor dataset. Through our real-time experiments, we have shown that our approach performs well in different indoor locations, which gave a promising capability of deploying this model on autonomous UAV exploration.

In future works, additional images should be acquired, and more situations should be increased to make the model more practical and flexible in various situations. Image data were taken under different illumination, complex backgrounds with objects, and humans should also be considered as these conditions usually appear in real scenarios. We also plan to develop a pathfinding algorithm to deploy this approach on a real UAV.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

The first author proposes the ideas for the contribution of the paper, collects data, conducts experiments and analyzes the results. The second and third authors support and provide suggestions for the first authors on the whole research paper. The final results are discussed and approved by all members of the group authors.

References

[1] M. Kontitsis, K. P. Valavanis, and N. Tsourveloudis, "A UAV vision system for airborne surveillance," *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, New Orleans, LA, USA, vol.1, 2004, pp. 77-83.

[2] S. W. Chen, S. S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C. J. Taylor, and V. Kumar, "Counting apples and oranges with deep learning: A data-driven approach," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, April 2017.

[3] J. Tisdale, Z. Kim, and J. K. Hedrick, "Autonomous UAV path planning and estimation," in *IEEE Robotics & Automation Magazine*, vol. 16, no. 2, pp. 35-42, June 2009.

[4] T. Tomic et al., "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue," in *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 46-56, Sept. 2012.
[5] R. Bajaj, S. L. Ranaweera, and D. P. Agrawal, "GPS: Location-tracking technology," in *Computer*, vol. 35, no. 4, pp. 92-94, March 2002.
[6] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar, "Multi-sensor fusion for robust autonomous flight in indoor and outdoor environ-ments with a rotorcraft mav," in 2014 *IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 4974–4981
[7] K. Schmid, T. Tomic, F. Ruess, H. Hirschmller, and M. Suppa, "Stereovision based indoor/outdoor navigation for flying robots," in 2013 IEEE/RSJ *International Conference on Intelligent Robots and Systems*(IROS), Nov 2013, pp. 3955–3962.
[8] K. McGuire, G. De Croon, C. De Wagter, K. Tuyls, H. Kappen, "Efficient optical flow and stereo vision for velocity estimation and obstacle avoidance on an autonomous pocket drone," in *IEEE Robotics and Automation Letters 2*, pp. 1070–1076, 2017.
[9] P. Checchin, F. Gerossier, C. Blanc, R. Chapuis, L. Trassoudaine, 2010, "Radar scan matching slam using the fourier-mellin transform," in: *Field and Service Robotics, Springer*. pp. 151–161.
[10] J. Artieda, J. M. Sebastian, P. Campoy, *et al.* "Visual 3-D SLAM from UAVs," *J Intell Robot Syst*, vol. 55, 299, 2009.
[11] F. Caballero, L. Merino, J. Ferruz, *et al*, "Vision-based odometry and SLAM for medium and high altitude flying UAVs," *J Intell Robot Syst*, vol. 54, pp. 137–161, 2009.
[12] K. Alexandros and B. Christos, "Learning to fly by myself: A self-supervised CNN-based approach for autonomous navigation," 2018.
[13] A. Loquercio, A. I. Maqueda, C. R. del-Blanco, and D. Scaramuzza, "DroNet: Learning to fly by driving," in *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1088-1095, April 2018.
[14] H. Tulldahl and L. H åkan, "Lidar on small UAV for 3D mapping," in *Proc. SPIE - The International Society for Optical Engineering*. 9250, 2014.
[15] A. Bachrach, S. Prentice, R. He, P. Henry, A. S. Huang, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Estimation, planning, and map-ping for autonomous flight using an RGB-D camera in GPS-deniedenvironments," *Intl. J. Robotics Research*, vol. 31, no. 11, pp. 1320–1343, 2012.
[16] A. Bry, A. Bachrach, and N. Roy, "State estimation for aggressive flightin GPS-denied environments using onboard sensing," *International Conference on Robotics and Automation* (ICRA), 2012.
[17] P. Ram, V. Sachin, A. Shahzad, and C. Suman, and S. Pankaj, "Deep neural network for autonomous UAV navigation in indoor corridor environments," *Procedia Computer Science*, vol. 133, 2018, pp. 643-650, 2018.
[18] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcementlearning," arXiv preprint, arXiv:1509.02971, 20
[19] S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, and M. Hebert, "Learning monocular reactive UAV controling cluttered natural environments," in *Proc. IEEE International Conference on Robotics and Automation* (ICRA), May 2013.
[20] M. John, J. Kyle, T. Rachael, and K. Mykel, "Visual depth mapping from monocular images using recurrent convolutional neural networks," 2018.
[21] A. Giusti *et al*., "A machine learning approach to visual perception of forest trails for mobile robots," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 661-667, July 2016.
[22] N. Smolyanskiy, A. Kamenev, J. Smith, and S. Birchfield, "Toward low-flying autonomous MAV trail navigation using deep neural networks for environmental awareness," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, 2017, pp. 4241-4247.
[23] M. Zhang *et al*., "A high fidelity simulator for a quadrotor UAV using ROS and Gazebo," in *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, Yokohama, 2015, pp. 002846-002851.
[24] R. Huitl, G. Schroth, S. Hilsenbeck, F. Schweiger, E. Steinbach, Tumindoor, "An extensive image and point cloud dataset for visual indoor localization and mapping," in *Proc. 2012 19th IEEE International Conference on Image Processing (ICIP)*, pp. 1773–1776, 2012.
[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision 115*, 211–252, 2015.
[26] S. Karen and Z. Andrew, "Very deep convolutional networks for large-scale image recognition," arXiv 1409.1556, 2014
[27] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, "Deep residual learning for image recognition," pp. 770-778, 2016.
[28] H. Andrew, Z. Menglong, C. Bo, K. Dmitry, W. Weijun, W. Tobias, A. Marco, and A. Hartwig, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017.
[29] G. Huang, Z. Liu, K. Q. Weinberger, L. V. der Maaten, "Densely connected convolutional networks", in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, p. 3, 2017.

Duc Viet Bui was born in 1995 and received his B.E degree from Department of Computer Science, National Defense Academy of Japan in 2019. His area of interest includes computer vision, machine learning and artificial intelligence. He is a master student at Department of Computer Science in National Defense Academy of Japan.

Tomohiro Shirakawa received his PhD from the Department of Earth and Planetary Systems Science, Kobe University, Hyogo, Japan in 2007. He is presently working as a lecturer at the School of Electrical and Computer Engineering, National Defense Academy of Japan. Prior to the National Defense Academy, he worked for the Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, and for three years as a postdoctoral fellow of the Japan Society for the Promotion of Science. His research interests include cell biology, biophysics, living systems theory and biocomputing.

Hiroshi Sato is an Associate Professor of Department of Computer Science at National Defense Academy in Japan. He holds the degrees of Physics from Keio University in Japan, and Master and Doctor of Engineering from Tokyo Institute of Technology in Japan.
He was previously Research Associate at Department of Mathematics and Information Sciences at Osaka Prefecture University in Japan. His research interests include agent-based simulation, evolutionary computation, and artificial intelligence. Dr. Sato is a member of Japanese Society for Artificial Intelligence (JSAI), Society of Instrument and Control Engineers (SICE) and The Institute of Electronics, Information and Communication Engineers. IEICE). He was the editor of IEICE and SICE.