ICIC Express Letters ICIC International ©2019 ISSN 1881-803X
Volume **13**, Number **6**, June **2019** pp. **461–468**

MODELING AND IMPLEMENTATION OF OBJECT DETECTION AND NAVIGATION SYSTEM FOR QUADCOPTER DRONE

Aurello Patrik[1], Gaudi Utama[1], Alexander Agung Santoso Gunawan[2]
Andry Chowanda[1], Jarot Sembodo Suroso[3] and Widodo Budiharto[1,*]

[1]Computer Science Department, School of Computer Science
[2]Mathematics Department, School of Computer Science
[3]Information Systems Management Department, BINUS Graduate Program – Master of Information Systems Management
Bina Nusantara University
Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
[*]Corresponding author: wbudiharto@binus.edu

Received November 2018; accepted January 2019

Abstract. *This paper presents our research on the development of object detection and navigation systems using deep learning for quadcopter drone. The grand purpose of our research is to deliver important medical aids for patients in emergency situations and implementation in agriculture in Indonesia. We will exploit the drone technology for transporting items efficiently. In sending process, our drone must detect the object target, where the items will be delivered. Therefore, we need object detection module that can detect what is in video stream and where the object is by using GPS as well. To implement the module, we use combination of GPS, MobileNet and the Single Shot Detector (SSD) framework for fast and efficient deep learning-based method to object detection. The development of navigation systems and the ability of deep learning to detect and localize specific objects are studied by conducting experiments using drone camera. The average error from start to goal position is 1.94 meter.*
Keywords: Drone, GPS, Navigation systems, Deep learning, Computer vision

1. **Introduction.** Worldwide media and scientist attention have put Unmanned Aerial Vehicles (UAVs) in the spotlight. UAVs can be operated more economically than manned helicopters; they are less limited by weather conditions (although this varies by model) and easier to deploy. Safe and reliable outdoor navigation of autonomous systems is a challenging open problem in robotics, for example, [4] proposed DroNet: a convolutional neural network that can safely drive a drone through the streets of a city. Designed as a fast eight-layer residual network, DroNet produces two outputs for each single input image: a steering angle to keep the drone navigating while avoiding obstacles, and a collision probability to let the UAV recognize dangerous situations and promptly react to them. The challenge is however to collect enough data in an unstructured outdoor environment such as a city. Clearly, having an expert pilot providing training trajectories is not an option given the large amount of data required and, above all, the risk that it involves for other vehicles or pedestrians moving in the streets.

Deep learning is a fast-growing domain of machine learning, mainly for solving problems in computer vision. It is a class of machine learning algorithms that use a cascade of many layers of nonlinear processing. Deep learning has ability to learn multiple levels of representations that correspond to hierarchies of concept abstraction [1]. One of the implementations of deep learning is object localization and detection based on video stream. Object localization and detection are crucial in computer vision. Recent advances in object detection are mainly using deep learning such as Region-based Convolutional

DOI: 10.24507/icicel.13.06.461

Neural Networks (R-CNNs). From previous work, we used conventional machine learning approach like fast algorithm for object detection using SIFT (Scale Invariant Features Transform) as key point detector and FLANN (Fast Library for Approximate Nearest Neighbor) based matcher [2].

Technically, deep learning is based on the backpropagation algorithm, which is a method for training the weights in neural network. Backpropagation network has been known for its ability to learn from data and improving itself during training process, but its performance depends on the initial values. If backpropagation network algorithm is combined with genetic algorithm, we can achieve higher accuracy by defining the best initial values for the network's architecture [3].

We will use deep learning for object localization and detection in agriculture in our research. We use the drone technology for transporting items efficiently. While in sending process, our drone must localize and detect the object target. Therefore, object detection module is developed based on camera drone. In this paper, we exploit a fast deep-learning framework for object localization and detection, that is: Mobilenet and Single Shot Detector (SSD) for quadcopter drone. The quadcopter drone used in the experiments is shown in Figure 1. Here, we use OpenCV and make experiment using popular deep learning architectures, that is: GoogleLeNet, ResNet and VGGNet. The initial results will be used in our future research in delivering medical aids for patients in emergency situations by using drone.

Figure 1. Erle Drone used in this experiment [5]

The paper is organized as follows. We provide the problem statements and preliminaries that we answer with this paper in Section 2. Our proposed methods of tackling the problem provided in Section 2 are explained in Section 3. In Section 4, we provided our evaluation on the method we proposed in Section 3. We provide our conclusion in Section 5.

2. Problem Statement and Preliminaries.

2.1. **Deep learning.** Deep learning is an area of machine learning emerged from the intersection of Artificial Neural Networks (ANNs), artificial intelligence, graphical modeling, optimization, pattern recognition and signal processing. ANNs are a class of machine learning algorithms that learn from data and specialize in pattern recognition, inspired by the structure and function of the brain. The basic building block is a neuron. A neuron takes a weighted sum of inputs and calculates an activating function. Deep learning belongs to the family of ANN algorithms, and in most cases, the two terms can be used interchangeably. Deep Neural Networks (DNN) are categorized as unsupervised, supervised, and hybrid. The unsupervised learning does not use any task specific supervision information in the learning process. It generates meaningful samples by sampling from

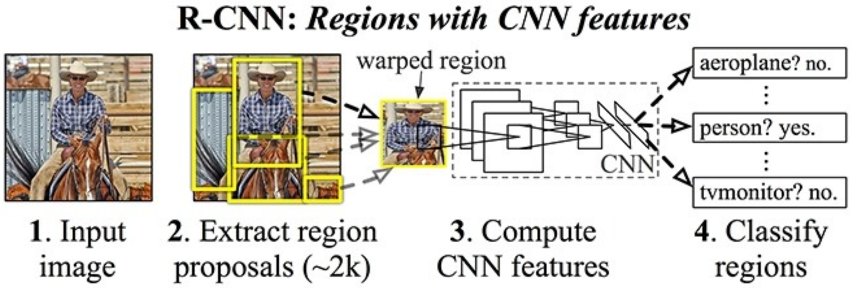

FIGURE 2. Deep learning model based on CNN [5]

the networks. Convolutional Neural Network (CNN) is a fundamental architecture in deep learning, called as LeNet [6] as shown in Figure 2. The CNN takes a small square and starts applying it over the image, and this square is often referred to as a window. This key component is named as the convolutional layer, which can capture the structure of an image. A convolutional layer connects each output to only a few close inputs, as shown in the illustration above. Intuitively, this means the layer will learn local features.

Recently, deep learning has been significantly developed and improved in computer vision, particularly in object recognition and classification. Referred as the black box approach, deep learning methods provide significant improvement to object recognition and classification problem. Deep learning allows the learning architecture to learn the important features that identify the object from a ton of images. Zhou et al. [7] implemented deep learning for scene recognition problem, and the researcher reached the accuracy of $94.42 \pm 0.76\%$ with more than 7 million datasets. Socher et al. [8], Krizhevsky et al. [9], Qi et al. [10], Simonyan and Zisserman [11], and He et al. [12] also proposed deep learning method to solve object recognition and classification problem. They trained millions of datasets to recognize hundreds of object classes. Most of them reached more than 80% of accuracy in both 3D and RGB descriptors with their proposed deep learning algorithm. Moreover, He et al. [12] claimed that the level of accuracy of a system with deep learning, in recognizing objects in ImageNet Classification, is surpassing the human capability. Currently, technology allows us to build and deploy a system with deep learning features in a mobile device (e.g., Tensorflow in Android and CoreML in iOS).

Several papers have proposed ways of using deep networks for predicting object bounding boxes [13]. Some of deep learning-based object detectors are Faster R-CNNs [14] and Single Shot Detectors (SSDs) [15] as shown in Figure 2.

For R-CNN, we need some steps for object detection such as extract region proposal, compute CNN features, and then classify regions. Furthermore, there is MobileNets architecture [16]. It is called as MobileNets because it is designed for resource constrained devices such as smartphone. If we combine both the MobileNet architecture and the Single Shot Detector (SSD) framework, we can get at a fast, efficient deep learning-based method to object detection.

2.2. **Global Positioning System (GPS).** Some organizations are using UAVs specifically because of this high level of interest. Companies such as Amazon, Google, and DHL have all launched substantial technical and, in some cases, coordinated public policy programs to bring drone delivery to fruition using GPS and important sensors [20]. For recording the object target, this research used the GPS which is radio navigation system and positioning by using satellite. This system is designed to provide the position and speed of three dimensions and information about time continuously. Our drone is equipped with GPS system and it can be used to record geolocation and to improve stability. When flying at high altitude, our drone becomes increasingly difficult to see and stability is therefore essential for controlling it. The GPS receiver calculates drone's position and helps it to remain stable against the wind and then improve drone stability. We developed program using Python in order the drone can also be controlled using PS-2 remote controller and autonomous mode.

3. Proposed Methods.

3.1. Architecture of the drone. We use Erle Drone [5], and as the heart of our drone system, it uses Erle-Brain 3 hardware autopilot and the APM flight stack. Erle-Brain 3 consists of a Linux-based embedded computer with full support for ROS (the Robot Operating System) and ROS 2.0 that integrated the sensors, camera, power electronics and abstractions necessary to easily create autonomous vehicles. ROS topics are an implementation of the publish-subscribe mechanism, in which the ROS Master serves as a well-known entry point for naming and registration. Each ROS node advertises the topics it publishes or subscribes to the ROS Master. Computer within the ROS network, is identified by its IP address (ROS_IP environment variable) [17]. The architecture of drone is shown in Figure 3.

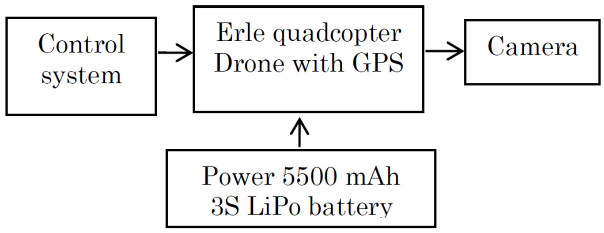

Figure 3. Block diagram of quadcopter drone

3.2. Object localization and detection. The family of popular object detectors in the deep learning are Single Shot Detector (SSD) that uses a single activation map for prediction of classes and bounding boxes and Faster R-CNN that implements different activation maps (multiple-scales) for prediction of classes and bounding boxes. Using multiple scales helps to achieve a higher mAP (mean Average Precision) by being able to detect objects with different sizes on the image better. SSD only needs an input image and ground truth boxes for each object during training. The MobileNet SSD was first trained on the COCO dataset and was then fine-tuned on PASCAL VOC reaching 72.7% mAP (mean Average Precision). We can therefore detect 20 objects in images (+1 for the background class), including airplanes, bicycles, birds, boats, bottles, buses, cars, cats, chairs, cows, dining tables, dogs, horses, motorbikes, people, potted plants, sheep, sofas, trains, and TV monitors. First, we train the training images, after that we got the model and will be used in testing. The SSD training objective is derived from the MultiBox objective, which is extended to handle multiple object categories and SSD model adds several feature layers to the end of a base network, which predict the offsets to default boxes of different scales and aspect ratios and their associated confidences.

3.3. Navigation systems. The structure of our system in ROS structure is shown in Figure 4.

The arrow represents whether the node publishes or subscribes to the topic, or represents which node calls a service or which node provides the service. If the arrow comes out from the node to a topic, it means the node publishes to the topic. If the arrow comes out from the topic to a node, it means the node subscribes to the topic. If the arrow comes out from the node to a service, it means the node calls the service. If the arrow comes out from the service to a node, it means the node provides the service.

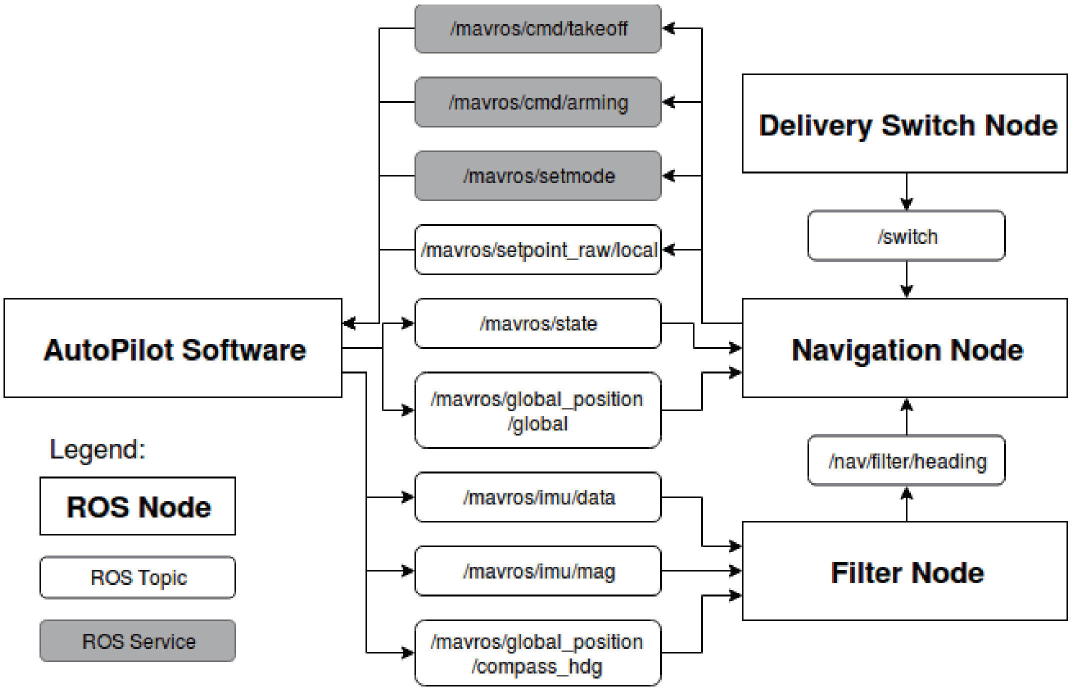

FIGURE 4. Navigation systems for the drone

3.4. **Algorithm.** For object detection system, the proposed algorithm was already discussed in [19]. For navigation systems, the program needs 2 positions, start position and goal position, so drone can fly to the destination.

Algorithm 1: Navigation Systems for Drone

```
initiate motors and sensors
input goal.latitude and goal.longitude
set home.latitude = current.latitude
set home.longitude = current.longitude
set default_altitude = 4
set drone_is_flying = False
wait until item loaded
arm_and_takeoff_sequence()
send_goal_coordinate(goal.latitude, goal.longitude)
wait until drone land and item unloaded
wait 5 seconds
arm_and_takeoff_sequence()
send_goal_coordinate(home.latitude, home.longitude)
#For arm_and_takeoff_sequence()
call ros service /mavros/set_mode to "GUIDED"
call ros service /mavros/cmd/arming
wait 5 seconds
call ros service /mavros/cmd/takeoff with altitude default_altitude
wait 5 seconds
#For send_goal_coordinate(latitude, longitude)
set destination.latitude = latitude
set destination.longitude = longitude
set drone_is_flying = True
#For a function that triggers every time current latitude and longitude
is reported from /mavros/global_position/global
```

```
If drone_is_flying = True then:
  if HubenyDistance between (current.latitude,
  current.longitude) and (destination.latitude,
  destination.longitude) <= 1.5 m then:
  call /mavros/set_mode to "LAND"
     set drone_is_flying = False
  else:
       bearing = find_bearing(current_coordinate, goal_coordinate)
        angle = find_direction_angle(heading, bearing)
        current_angle = angle
        set message as TwistStamped message.
        message.x_velocity = max_speed * sin(current_angle)
        message.y_velocity = max_speed * cos(current_angle)
        publish message to /nav/raw_local
  end if
end if
```

4. **Experimental Results.** In previous work we use AR Parrot drone for object detection successfully [19], and in this paper we use Erle Drone based on the Ubuntu OS Linux and the experiment with SSD algorithm, we use OpenCV 3.4 [18] and Python 3.5. The experimental results are shown in Figure 5 for camera drone. In our experiments, SSD algorithm shows to be superior compared with Faster R-CNN algorithm.

We conduct the experiment using GPS at the outdoor area as shown in Figure 6.

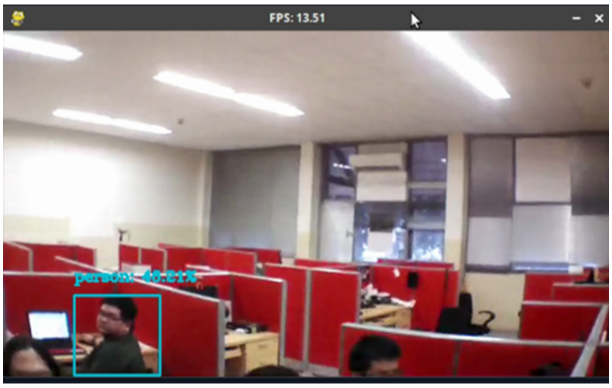

FIGURE 5. SSD object detector could detect object from the flying drone

FIGURE 6. Experiment using GPS for start to goal position [22]

Table 1. Error average for the experiment

Error position 1	2.5 m
Error position 2	1.9 m
Error position 3	2.0 m
Error position 4	1.8 m
Error position 5	1.5 m
Average error = 1.94 m	

Table 1 shows the error position and average error for the experiment in 5 times.

For topic /mavros/global_position/global, and it uses NavSatFix message, and it means that the Autopilot Software supplies the message in NavSatFix data structure. It supplies current coordinate in latitude, longitude, and altitude. For topic /mavros/setpoint_raw/local, it uses PositionTarget message, so the Navigation Node needs to supply velocity which needs to be applied to the Drone to moving to the target position. For topic /mavros/state, Navigation Node only needs the current flight mode of the drone, whether it is landing or in "GUIDED" flight mode. Topics with names /mavros/imu/data and /mavros/imu/mag are topics that provide data that were obtained from IMU system. For /mavros/global_pos-ition/compass_hdg, it provides data from external compass. Topic with name /nav/filter/ heading is a topic that provides filtered Drone heading.

Our experiment shows that Drone's landing position was good enough as the Drone lands within 3 meters from an absolute goal position, and users may still reach the Drone even with such deviation.

5. **Conclusion.** This paper presents the implementation of deep learning technology and GPS for navigation and object detection that can be fitted in quadcopter drone. Our method using MobileNet SSD Detector can be used as object detector with high-accuracy detection with average about 14 FPS. The navigation using GPS shows good result with average error position being 1.94 m. The resulting system is interactive and engaging and we are able to control the drone with low specification in hardware. Moreover, the Drone can correctly detect the common objects such as person, desk, or chair with high accuracy. For future work, we will use this system for drone applications in healthcare including delivery of medicine, defibrillators, blood samples, and vaccines [21].

Acknowledgment. We say thank you very much for Bina Nusantara University for supporting this research based on BINUS PUB grant 2017.

REFERENCES

[1] Y. Bengio, A. Courville and P. Vincent, Representation learning: A review and new perspectives, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.35, no.8, pp.1798-1828, 2013.

[2] W. Budiharto, Robust vision-based detection and grasping object for manipulator using SIFT keypoint detector, *International Conference on Advanced Mechatronic Systems*, Japan, pp.448-452, 2014.

[3] *Introduction to Deep Learning*, https://towardsdatascience.com/deep-learning-2-f81ebe632d5c, accessed in August 2018.

[4] A. Loquercio, A. I. Maqueda, C. R. Del Blanco and D. Scaramuzza, DroNet: Learning to fly by driving, *IEEE Robotics and Automation Letters*, vol.3, no.2, pp.1088-1095, 2018.

[5] *Erle Drone*, https://erlerobotics.com/blog/product/erle-copter-diy-kit/, accessed in February 2018.

[6] Y. Bengio, Learning deep architectures for AI, *Foundations and Trends in Machine Learning*, vol.2, no.1, pp.1-127, 2009.

[7] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba and A. Oliva, Learning deep features for scene recognition using places database, *Advances in Neural Information Processing Systems*, pp.487-495, 2014.

[8] R. Socher, B. Huval, B. Bath, C. D. Manning and A. Y. Ng, Convolutional-recursive deep learning for 3D object classification, *Advances in Neural Information Processing Systems*, pp.656-664, 2012.

[9] A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Advances in Neural Information Processing Systems*, pp.1097-1105, 2012.

[10] C. R. Qi, H. Su, K. Mo and L. J. Guibas, PointNet: Deep learning on point sets for 3D classification and segmentation, *IEEE Proc. of Computer Vision and Pattern Recognition*, vol.1, no.2, p.4, 2017.

[11] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, *arXiv:1409.1556*, 2014.

[12] K. He, X. Zhang, S. Ren and J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, *Proc. of the IEEE International Conference on Computer Vision*, pp.1026-1034, 2015.

[13] C. Szegedy, A. Toshev and D. Erhan, Deep neural networks for object detection, *Advances in Neural Information Processing Systems*, 2013.

[14] Girshcik et al., Faster R-CNN: Towards real-time object detection with region proposal networks, *Advances in Neural Information Processing Systems*, 2015.

[15] W. Liu et al., SSD: Single shot MultiBox detector, *IEEE Proc. of Computer Vision and Pattern Recognition*, 2015.

[16] A. G. Howard et al., MobileNets: Efficient convolutional neural networks for mobile vision applications, *arXiv:1704.04861*, 2017.

[17] F. Furrer, M. Burri, M. Achtelik and R. Siegwart, Robot Operating System (ROS), *Studies Comp. Intelligence*, vol.625, 2016.

[18] *Introduction of OpenCV*, www.opencv.com, accessed in August 2018.

[19] W. Budiharto, A. Patrik, G. Utama, A. A. Gunawan, A. Chowanda and J. Santoso, Fast object detection for quad copter drone using deep learning, *IEEE International Conference on Computer and Communication Systems*, Singapore, pp.192-195, 2018.

[20] J. Xu, *Design Perspectives on Delivery Drones*, RAND Corporation, Santa Monica, CA, https://www.rand.org/pubs/research_reports/RR1718z2.html, 2017.

[21] J. E. Scott and C. H. Scott, Drone delivery models in healthcare, *Proc. of the 50th Hawaii International Conference on System Sciences*, pp.3297-3304, 2017.

[22] *Binus Autonomous Drone*, https://www.youtube.com/watch?v=iSidObl2vBI, accessed in October 2018.