Automatic take-off landing quadcopter control on dynamics platform

Akmal Dutasatria and Surya Darma*
Department of Physics, Universitas Indonesia, Kampus UI Depok, 16424, Indonesia

*suryadarma@sci.ui.ac.id

Abstract. SAR (Search and Rescue) become an interesting topic discussed worldwide because of its urgency to save humans life. Recent natural disaster in Palu, Central Sulawesi Indonesia demonstrates the lack of mobility which can cause a problem for the SAR team reaching the victim within the isolated area. Uneasy job for the SAR team become more challenging in time response and in finding and saving disaster victims. Consider the fact, there is a need to develop more efficient and effective way to search and rescue the survivors. The using of Quadcopter can answer the problems. Unfortunately the Quadcopter still conventionally controlled by a remote and further it has to be on static ground whenever its take off or landing. This paper reports a Quadcopter fitted with dynamical autonomous take off and landing systems. Thus it can carry out the advantage from a tactical SAR vehicles or other moving platforms. Using object detection approach, The navigation system in an image-based control quadcopter or called visual servoing is the use of machine vision as a control of the closed loop position for movement. Images can be detected using YOLOv3 Real Time Object Detector, and tracks it by estimating the motion of objects in successive video frames. By using 3D reconstruction method the distance and position of the designated target and the quadcopter could be determined. Then the estimation of the relative position of the object is used as input for the control system on the quadcopter using PID controller. The controller commands the quadcopter to approach the target, while the image processing checks the relative position between the two. If the position satisfies the minimum landing parameter, then the controller commands a landing. While for take off we implement the controller to maintain its equilibrium and sufficient power to throttle.

1. Introduction
The use of YOLOv3 as the Real-time Object Detector were widely used in many application because of its robust and fast detection [1,2]. Featuring DarkNet-53 as it layer network, it has more layer than the previous version making it imprevis to residual blocks, upsampling and connection skip that the previous version suffers[1]. In addition YOLOv3 also surprisingly doesn’t need such high processing hardware to run making it suitable for used in device like odroid on quadcopter [3]. As one of machine vision that processes images to detect an object, we can try to implement it on quadcopter by complement it with kalman filter to tracks the landing pad by estimating the motion of it in successive video frames [4,5]. Then the estimation of the relative position of the object is used as input for the control system on the quadcopter. Position information is then sent by the image processing program to become a reference for the quadcopter control system program [6]. The quadcopter control system will reconstruct the quadcopter trajectory when taking-off and landing from the results of the program.
estimation calculation [7]. We expected to cut the operating time of the quadcopter to fly and land when it will carry out the search and rescue task.

2. Methodology
Our concept for the dynamical landing by autonomous system emphasize on how the quadcopter recognizing the target and then tracking it using computer vision until landing. While the quadcopter airborne, it will implement detection algorithm in which they moving or scanning the certain area with possibility of the target being there. To simplify this we can set rendezvous point in which the waypoint of the quadcopter and the moving platform are intersect so there can be high probability of the platform could be detected. There some parameters are required to set rendezvous point, for instance it required low-relative pose accurate positioning[8], which can be complied using GPS longitude and latitude coordinates. GPS navigation are less accurate compared to IMU navigation[9,10] so it will switch to vision-based navigation whenever the target are localized. When the position of the platform relative to quadcopter are acquired, it will be used as input of visual servoing that guides quadcopter to approach and land on the platform. To guarantee precise and successful landing we use kalman filter algorithm to reinforce our vision-based navigation.

2.1. Detection Work

![Detection Flow Diagram](image)

Tiny YOLOv3 primarily used to detect the image used the models that previously trained with DarkNet-53. We have 20000 models trained by which created several prediction to the refered target. The bounding box are created prior to the detection of the target which gives us relevant position of the target relative to the quadcopter depending of the size of the box. The size of the box represent the distance of the target. We measure it by finding the marker center of the bounding box while continuously calculate its width and height to give us relative distance of the target. Because of our target is dynamic we used kalman filter to predict the direction and movement of the platform.

In order to deal with uncertainty of the position measurement we use Kalman Filter algorithm to predict the actual position and speed of the platform. First we need to localize the image tag on landing platform previously using our computer vision to calculate the relative position of the quadcopter and the landing platform. It calculates frame-by-frame position of the tag within the image.
with some errors. Upon calculation it compared the measured position value and the predicted value that satisfy the predicted states that can be formulated into Kalman filter algorithm as follows

\[ \hat{X}_k = K_k Z_k + (1 - K_k) \hat{X}_{k-1} \]  

(1)

where \( \hat{X}_k \) is the state vector at time \( k \), \( Z_k \) is the sensor output; \( K_k \) is the Kalman Gain while \( \hat{X}_{k-1} \) is the state vector at time \( k-1 \). \( \hat{X}_k \) is representing the landing platform in its kinematic motion in the discrete time domain. The state vector at time \( k \) give us updated estimation of the target movement. While the sensor output will confirm the actual position. Since the previous state become the input of the next state, after a series of data (platform position from video frames), Kalman Filter also gives information about the dynamics of landing platform predictive states.

The updated estimation give us almost 80% mAP (mean Average Precision) with only less than a second of detection time. The colour distorsion and stability of camera have to be considered since they affect the accuracy of detections.

**Table 1.** The Average Precision (AP) of our model using CPU Intel i7-7500u.

| Type       | Tiny YOLOv3 |
|------------|-------------|
| AP (%)     | 78.34       |
| Weights    | 20000       |
| Times (ms) | 282.43      |

The model is trained to detect single object (i.e. Individual). Models trained using DarkNet-53 as Trainval to be tested by our test set.

![Figure 2](image-url) The implementation of detection algorithm to comply with the test.

In series of experiments more 20 times, the Tiny YOLOv3 shows the AP of 78.34% which mean a good percentage of detection capability. This result is better compared to SSD 300 (MobileNet V2) which has AP 72.01%.[11, 12]

2.2. **Proposed Control**

The output of camera vision extracted to specific values, then inputed to PID Controller and determine the control gain for each axis. The coordinates of the points on the frame and the distance with the quadcopter, will be used as an error value to be entered in the PID equation and calculate the error value continuously against the time \( e(t) \), which is the value between the set point (SP) and process variable (PV). SP will be the center of frame and PV is the center of target (Region of Interest/ROI). Then the value of \( e(t) \) is applied to the PID equation with reference to proportional, integral, and derivative constants to satisfy with the equation.

\[ u(t) = K_p e(t) + K_i \int_0^t e(t) \, dt + K_d \frac{de(t)}{dt} \]  

(2)
This merely give the control effort to the quadcopter so that it reaches the desired point, which in turn gives the flight controller a proportional pulse width modulation (PWM) to control the rotors according the PIDs. The $u(t)$ is a parameter that controls the velocity of the quadcopter.

3. Experiment Result

3.1. Detection and Tracking

![Tracking Errors](image)

**Figure 3.** System tracking response of each axes of evaluated using kalman filter

In Figure 3. There are three different graph that represent the tracking response of the landing platform for each axes. At the top the graph represent frame by frame displacement of the tag in x-axis within the image. This graph trend indicate reduced tracking error measurement at x-axes, while on center of the image the trend tends to be increased. This indicated that while the quadcopter approach to land on the platform the tag displacement within causing increasing error in y-axes because of the turning movement (rolling) get the position and match the direction of the moving platform, and in the others the z-axes displacement of the tag that indicates that quadcopter are reducing altitude near landing.

The outcome of tracking response on 30 m shows nearly 50% of error due to color distortion competed with range and camera stabilisation prior to wind disturbance. In Figure 3. We see the error reduced significantly when the object come within range. However the updated prediction meanwhile surpassing the detection parameter making it vulnerable to certain error. Hence no condition that meets the desired response. The IMU (Inertial Motion Unit) however shows the movements of the quadcopter as seen in Figure 4a show the position in X and Y of the quadcopter along with the measured location of the landing platform. As shown in Figure 4b, the quadcopter maintains the altitude while it is approaching the landing platform and if the distance between the quadcopter and the landing platform becomes less than 5 m, the quadcopter starts descending.
3.2. IMU Response

![Figure 4. Visualisation of the quadcopter movement when it airborne.](image)

(a) Orientation of the quadcopter heading in which the target is designated. b) The speed of the quadcopter during approaching the target.

**Table 2.** The desired quadcopter speed to approach and land in the platform.

| Distance (m) | Velocity (ms⁻¹) |
|-------------|-----------------|
| 5           | 1.16            |
| 10          | 2.68            |
| 20          | 4.54            |
| 30          | 5.36            |

**Table 3.** Mean Characteristic of actual Implementation

|        | Velocity (m/s) | Error (%) | Angular Velocity | Error (%) | Position (m) | Error (%) |
|--------|----------------|-----------|------------------|-----------|--------------|-----------|
| X      | 2.143          | 9.245     | 3.286            | 10.590    | 1.286        | 15.593    |
| Y      | 3.143          | 8.275     | 5.067            | 12.796    | 0.067        | 12.653    |
| Z      | 4.571          | 4.109     | 3.600            | 10.439    | 1.667        | 12.197    |

Table 2 summarizes the desired speeds of the quadcopter with certain distance when it approach the landing pad. The correlation of speed projection of the model depends on distances simulation which gives the successive approach upon landing based on visual based navigation. The position was corrected by the values from IMU. Table 3 shows the mean characteristic of actual implementation in a minimum noisy environment. It shows the differences of velocity from 4.109% in Z-axis until 9.245% in X-axis. With the maximum position error of 16%.

For the noisy measurement of the platform position and velocity, the quadcopter demonstrates similar approach and landing times as the no noise case. The largest landing error recorded with noisy measurements is 26 cm from the center of the landing platform, which shows that the method proposed in this work can be considered as a viable solution for autonomous landing on a moving platform.

4. Conclusion

In the process to build autonomous landing system in quadcopter that enables it to landing on dynamic platform requires complicated computation. We must consider some parameters in order to satisfy state-of-the-art computer vision algorithms, to support the detection we use GPS-based navigation in order to set rendezvous point to help the detectors find the target. We also use kalman filter algorithm
to reduce position measurement and estimation errors in vision-based navigation. The result was not satisfactory but still adequate to accommodate low-risk minimum disturbance mission. By then we have tried to optimize the application of computer vision algorithm and applying it on Quadcopter for SAR purpose.

Acknowledgments
This work was supported by Hibah PITTA (Publikasi Internasional Terindeks untuk Tugas Akhir Mahasiswa Universitas Indonesia) 2018 funded by DRPM Universitas Indonesia, Contract No. 2319/UN2.R3.1/HKP.05.00/2018.

References
[1] Redmon J and Farhadi A 2018 YOLOv3: An Incremental Improvement
[2] Li G, Song Z and Fu Q 2018 A New Method of Image Detection for Small Datasets under the Framework of YOLO Network Proc. 2018 IEEE 3rd Adv. Inf. Technol. Electron. Autom. Control Conf. IAEAC 2018 1031–5
[3] Chen J, Miao X, Jiang H, Chen J and Liu X 2017 Identification of autonomous landing sign for unmanned aerial vehicle based on faster regions with convolutional neural network Proc. - 2017 Chinese Autom. Congr. CAC 2017 2017-Janua 2109–14
[4] Beck H, Lesueur J, Charland-Arcand G, Akhrif O, Gagne S, Gagnon F and Couillard D 2016 Autonomous takeoff and landing of a quadcopter 2016 Int. Conf. Unmanned Aircr. Syst. ICUAS 2016 475–84
[5] Han S, Shen W and Liu Z 2016 Deep Drone: Object Detection and Tracking for Smart Drones on Embedded System I–8
[6] Kim J, Woo S and Kim J 2017 Lidar-guided autonomous landing of an aerial vehicle on a ground vehicle 2017 14th Int. Conf. Ubiquitous Robot. Ambient Intell. URAI 2017 228–31
[7] Domingos D, Camargo G and Gomide F 2016 Autonomous Fuzzy Control and Navigation of Quadcopters IFAC-PapersOnLine 49 73–8
[8] Ghomman J and Saad M 2017 Autonomous Landing of a Quadrotor on a Moving Platform IEEE Trans. Aerosp. Electron. Syst. 53 1504–19
[9] Gordon D, Farhadi A and Fox D 2018 Re 3: Real-time recurrent regression networks for visual tracking of generic objects IEEE Robot. Autom. Lett. 3 788–95
[10] Szafranski G and Czyba R 2011 Approaches of PID Control UAV Proc. Int. Micro Air Veh. Conf. 70–5
[11] . B B 2014 a Study on the Importance of Image Processing and Its Aplications Int. J. Res. Eng. Technol. 03 155–60
[12] Jeong J and Park H 2017 arXiv : 1705 . 09587v1 [ cs . CV ] 26 May 2017 Enhancement of SSD by concatenating feature maps for object detection