UNIVERSITY OF TORONTO

Institute for Aerospace Studies

Gazebo Simulation on Vision-Based Slung Payload Transportation System by UAV

MEng Project Summer 2021

Prepared by
Constantine Boyang Cheng
ID 1002417333
Userid chengboy

Supervised by
Prof. Hugh Liu
Aug. 23, 2021

Abstract

Slung payload transportation system by unmanned aerial vehicles (UAVs) has become a rapidly developing field of research. Various 3D simulation environments have been launched for evaluating the performance of slung payload transportation systems. This project proposed and developed a simulation environment to simulate a vision-based UAV-payload transportation system. Comparing to existing UAV-related simulated environments, this environment supports simulating an online vision-based payload pose estimator by installing cameras on the unmanned aerial vehicle to capture identifiers on the payload. The simulation environment is applied to evaluate the performance of a proposed vision-based pose estimator in a slung payload transportation system as well as the control system of the UAV in the system. The simulation results are considered a guideline of future developments in the control system and the estimator.

Table of Content

List of Tables

List of Figures

1. Introduction

Unmanned Aerial Vehicle (UAV) is an efficient and reliable method of transportation and becomes a focus in both research programs and industry applications. The field of slung payload delivery by UAVs, which is among the most popular usages of UAVs, focuses on attaching payloads using a tether to one or multiple UAVs. Various software-in-the-loop (SITL) simulation platforms have been developed for evaluating the performance of aerial robots during taking-off, landing and completing surveying or scanning missions to guide future developments. [1, 2, 3] The simulation platforms are supported by autopilot programs that are used to creating missions and publishing commands to UAVs. [4] Additional modules and plugins are provided by simulation platforms for achieving tasks involving internal and external disturbances such as box-like payloads and wind shears.

Chau [5] has proposed a vision-based payload pose estimation system. The system estimates the pose of the payload attached to a UAV based on images captured by cameras on the UAV containing identifiers on the payload in real-time. However, there is no existing simulation environment with visual components that targets simulating slung payload transportation systems. Also, existing simulation environments do not support the application of different identifiers on each side of the payload or provide a straightforward method to switch identifiers on the payload, which increases the difficulty of verifying the estimation system.

To fulfill the simulation requirements, a Gazebo model containing a UAV with cameras installed and a payload with identifiers is developed for simulating vision-based single-UAV slung payload transportation systems in this project. The model is applied to evaluate the default PID control system of PX4 [4], the autopilot software used to command the transportation system, to identify edge cases of the control system. A method is developed to apply and conveniently replace identifiers as the texture on the payload for evaluating the performance of the estimator developed by Chau [5]. The performance of the control system and the estimator are analyzed for possible future developments.

2. Related Works

Previous studies on simulating UAV systems with slung loads are aiming at verifying the performance of the control system onboard when a payload is added to the aerial vehicle. Based on the objective, numerical simulation tools are utilized for fulfilling the requirements. Tartaglione et al. [6] conducted numerical simulations using Simulink to evaluate the robustness of the control system through the payload tracking error and the trajectory analysis. Similarly, the numerical simulation performed by Shi et al. [7] is based on an online flight evaluation toolbox developed by Shi et al. [8] However, although numerical simulations are

efficient and commonly used for verifying UAV systems, the visual-based estimator cannot be evaluated based on tools that only support numerical inputs and responses.

For simulating a vision-based UAV system, a 3D graphical simulation platform is expected instead of numerical simulation tools. 3D robotics simulation tools and the ground control software supported by PX4 are compared in the study by Hentati et al. [9] on several criteria including the types of vehicles and sensors supported and the ease of development. In this project, several additional criteria, including the computational and memorial costs, the availability of external plugins and the similarity to the real-world environment, are considered when selecting the simulator to fit the project requirements as well as take the hardware limitations into consideration.

Qian et al. [10] have implemented a Gazebo model integrating a quadrotor, a simple box-like payload, and a cylinder-like tether connecting the quadrotor and the payload. The work of Qian demonstrates that sufficient components are provided by Gazebo to simulate a slung load transportation system. However, the work of Qian does not support applying identifiers on the payload to enable vision-based estimations. Based on the study of Qian, we propose a model with vision components equipped on the UAV and identifiers applied to the payload as the texture to simulate a vision-based slung payload transport system. Also, the proposed model is designed to adapt the latest version of ROS and Gazebo.

3. Project Description

The project aims on building a simulation environment to evaluate the performance of a slung payload transportation system as well as a vision-based payload pose estimator installed on the system based on a 3D robotics simulator. To fulfill the requirement, it is expected to develop a simulation environment that contains a quadrotor, a payload with identifiers on the surface, a tether connecting the quadrotor and the payload, and a visual component installed on the quadrotor that can capture images of the payload during flights. The payload is expected to be allowed to perform 3 degrees of freedom movements to enable simulations of all possible real-world motions.

To evaluate the performance of the PX4 controller, it is necessary that the simulation environment is available to assign missions containing challenging operations to the slung payload transportation system to determine the edge cases that may make it impossible for the default controller to stabilize the system, which will be used as guidelines for future enhancements on the control system. In addition, plugins creating external disturbances such as winds are expected to be supported by the simulation environment to verify the stability of

the controller under corresponding circumstances.

To enable vision-based pose estimations, the identifiers on the payload are expected to be unique on each side of the payload and easily detectable. Also, the identifiers are required to be visible by the visual component on the UAV. At the same time, a scheme is expected to be designed to support routinely identifier switches on the surface of the payload to facilitate the test case generating process.

To verify the simulation environment, the performance of the control system and the estimator are theoretically analyzed for each test case, and the simulation results are compared with the theoretically expected performance to check the consistency and thus the reliability of the environment. During simulations, the slung payload transportation system is commanded to follow a path containing a series of waypoints in the simulated world. The length of the real path is computed to evaluate the performance of the PX4 control system since an ideal controller will stabilize the system on the shortest path towards each waypoint. The difference between the ground truth and the estimation of the payload pose is recorded to verify the payload pose estimator. The simulation results are analyzed for proposing potential enhancements on the control system and the estimator.

4. Development of Simulation Environment

In this project, the PX4 Autopilot Platform is used to host and conduct simulations. The three 3D graphic robotics simulators supported by PX4 Autopilot, jMAVSim, Gazebo and AirSim, are compared and evaluated, and ROS-based Gazebo is selected as the simulator utilized in all simulation processes. A Gazebo model formed by a quadrotor, a box-like payload and a tether is developed. A camera is applied at the bottom of the quadrotor. Also, a method is proposed to apply ArUco markers, a square marker developed by Garrido-Jurado et al. that are used as identifiers, to the payload as the texture. [11]

4.1 Comparisons of 3D Robotics Simulators

jMAVSim is a simple 3D robotics simulator that supports simulations of single or multiple quadrotors in a virtual world. The simulator supports software-in-the-loop (SITL) simulations through a connection with PX4. [3] According to the GitHub documentation, wind speed is configurable during simulations. However, based on the source code of jMAVSim, attaching payloads on UAVs is not supported, which renders the simulations of a slung payload transportation system using jMAVSim unavailable. Also, there is no approach to install sensors on quadrotors by default and thus impossible to simulate a vision-based system. [12]

Gazebo is a robotics simulator that supports simulations of arbitrary customized robots. [1] SITL and ROS integration are supported by Gazebo. Robot models are defined in the XML format as .sdf files. Exporting models from 3D design software is supported while the list of properties of each model is defined by the user, which makes the model highly configurable. [13] Multiple pre-defined sensors and plugins, including camera model and wind plugin, are provided by Gazebo, which enables the simulations of vision-based systems under customized circumstances. The simulated world is also configurable and can be defined in `.world` files in the XML format.

AirSim is a Microsoft-developed simulation plug-in of Unreal Engine, a 3D game engine, that can be used in 3D robotics simulations. The simulator interfaces with ROS and is available for SITL simulations. Comparing to the other two simulators, AirSim provides a more graphically realistic virtual world and thus requires higher computational and memorial costs while executing. [2] Based on the experimental results, a disk space larger than 74 GB is required for installing AirSim, which forms the largest drawback of the simulator since the hardware requirement is not capable to be satisfied by FSC in the duration of this project. In addition, the payload can be visually attached to the UAV while the dynamics between the payload and the UAV are required to be built manually by modifying the source code of the simulator, which renders AirSim less configurable than Gazebo.

Based on the experiments, Gazebo is the most suitable simulation platform due to the high freedom on model configurations, easy implementations, and relatively small computational and memorial cost.

4.2 Structure of Gazebo Model

The Gazebo model contains a UAV, a tether, and a box-like payload. One of the default quadrotors provided by PX4, the IRIS quadrotor, is utilized in the model. Four 10 x 4.7 propellers are applied to the model. [14] In the model, the weight of the quadrotor is defined as 1.5 kg including the battery.

The default box model provided by Gazebo is applied to construct the payload component. The shape of the payload is defined as a cube, which allows each side of the payload to contain the same number of ArUco markers with the same size. The length, width and height of the payload is fixed to be 2 meters to simplify the implementation of the payload pose estimator since the distance between the UAV and the payload is estimated based on the real-world size of the ArUco markers and the size of markers in the camera. [5] The weight of the payload is considered a variable for evaluating the performance of the PX4 controller.

The tether component is based on the default cylinder model of Gazebo. The radius of the cylinder is set to be small to approximate a stick. Two ball-like joints are applied to connect the tether to both the drone and the closest side of the payload to the drone. Ball-like joints allow 3 degrees of freedom movements of both the tether and the payload for modelling the real-world motions. Two formations of the tether are proposed in the designing process. One design is considering one thin cylinder as the tether, which is accepted as the final design, while the other is a series of thin cylinders connecting by ball-like joints. The single-cylinder tether design is preferred over the multi-cylinder design for avoiding unexpected reactions among cylinders. The length of the tether is defined as the distance between the drone and the payload to avoid the tether extending into the drone or the payload, which may reduce the stability of the model.

The Single-Drone-Payload Model is shown in Figure 1. The picture at the left shows the default state of the model while the right picture illustrates the state of the model after the UAV taking off. The closest side of the payload becomes the upper side after the system taking off.

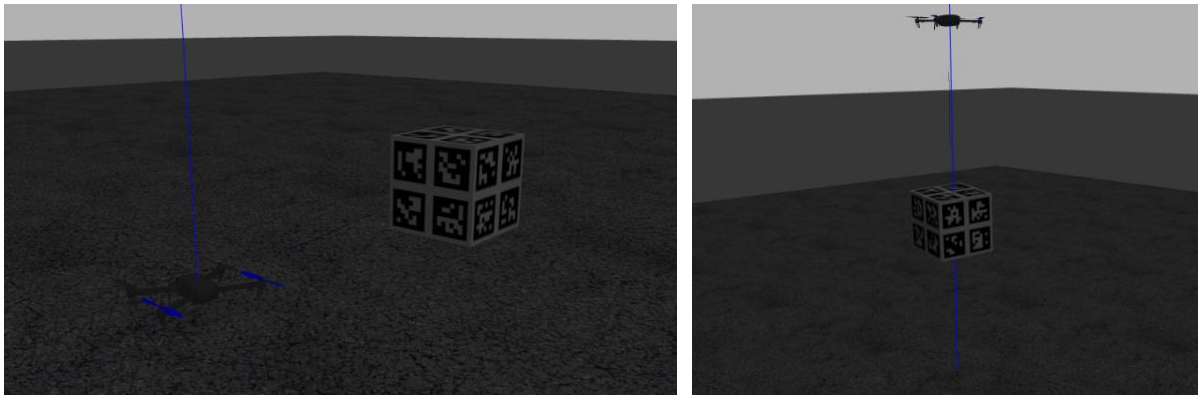

Figure 1 Structure of Model

4.3 Visual Component

The pre-defined Gazebo camera plugin is utilized as the visual component of the system. The camera supports LGB image captions. The field of view of the camera is configurable. Also, it is available to enable distortions on the camera. However, since the distortions coefficients are applied through post-processing the images captured by the camera and cutting off the sides of the images, which functions differently from real-world wide lens cameras, the field of view of the resulting image will be smaller than the non-distortion cases.

Two cameras are installed on the front and the bottom of the UAV model, respectively. The bottom camera captures the upper side of the payload during flights. The front camera is used for possible future developments on enabling detections of the payload pose during take-off

and landing. Since the tether connecting the payload and the UAV is attached near the central point of the bottom of the UAV, the bottom camera is placed at the central point to expect a higher probability of the payload to be contained in the camera frame. It also avoids the body of the UAV, such as the landing gears, to be captured by the camera and disturb the estimation process to install the bottom camera at the central point of the bottom of the UAV. Figure 2 is a sample image captured by the bottom camera.

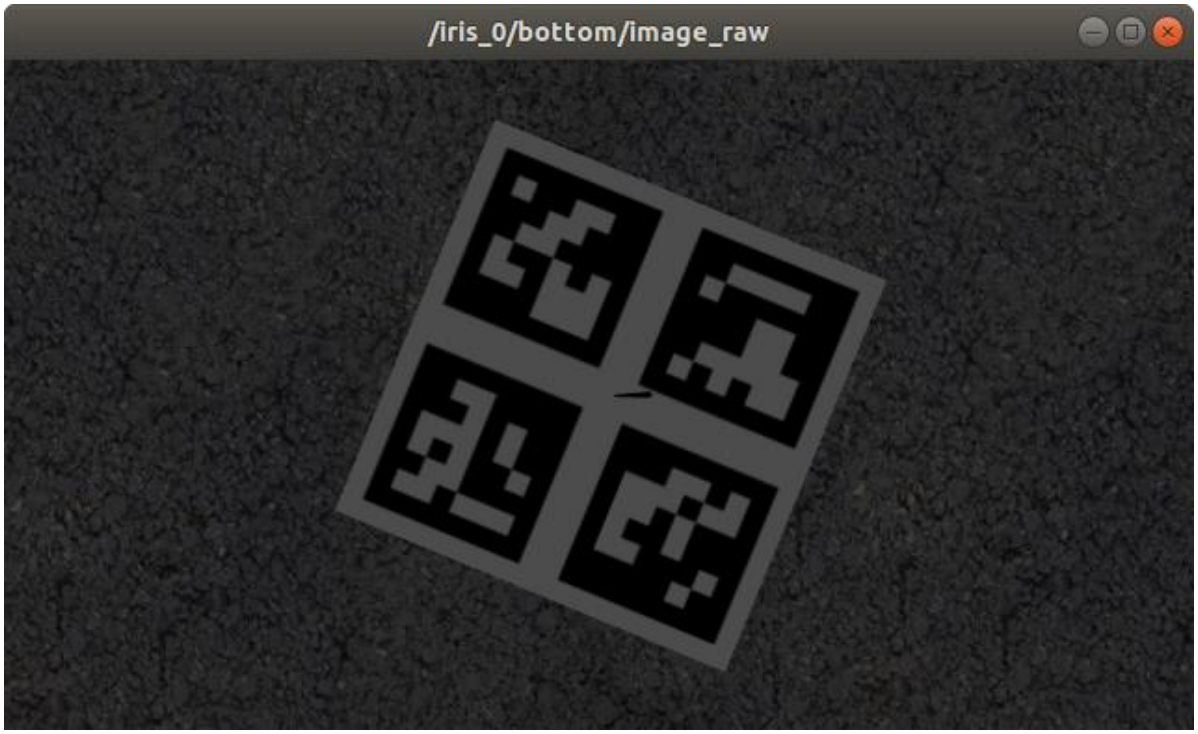

Figure 2 Sample Image Captured by the Bottom Camera

In the simulations, the bottom camera regularly takes picture of the upper side of the payload. The identifiers contained in the pictures are then detected by the estimator and the payload pose is estimated based on the detected identifiers.

4.4 ArUco Marker Generation on Payloads

To allow pose estimations of the payload, identifiers are expected to be placed on the surface of the payload. In the proposed model, ArUco markers are applied as identifiers of the payload. ArUco marker is a square marker containing a matrix of binaries surrounded by a black border. The black border benefits the detection and localization of the markers while the usage of binary matrices allows a computationally cheap identification process. [11] OpenCV provides a series of functions to generate and identify ArUco markers, which simplifies the implementation of the estimator. [15] Examples of ArUco markers are provided in Figure 3.

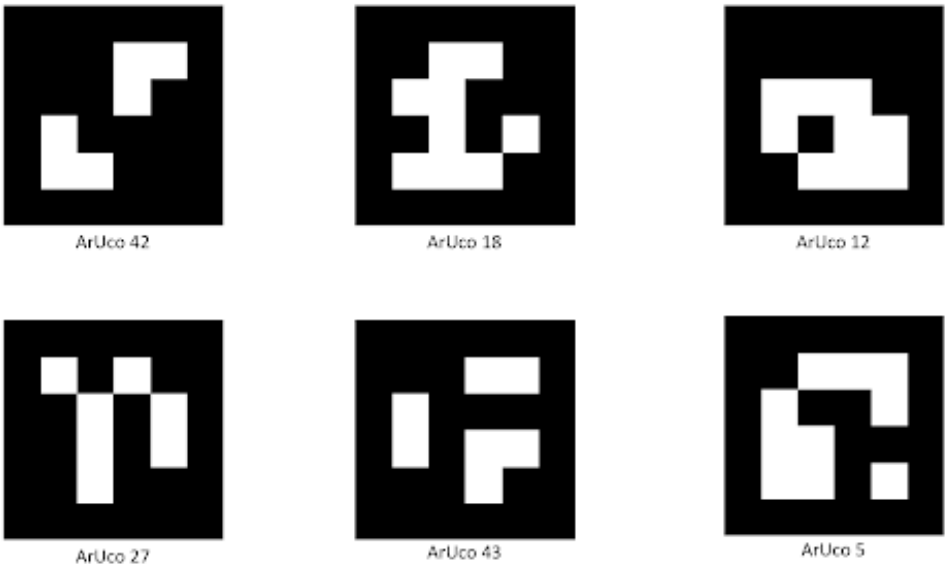

Figure 3 Examples of ArUco Markers [16]

To estimate the pose of the payload, different ArUco markers are expected to be placed on different sides of the payload. At the same time, according to Chau, the estimation accuracy is enhanced by placing multiple ArUco markers on each side of the payload. [5] Applying multiple identifiers on each side of the payload allows error rejections and makes it possible to make estimations if only part of the payload is displayed in the frame of the camera while at least one identifier is fully presented.

Blender, a 3D design software, is utilized to create the texture file for applying ArUco markers on the payload as the texture. Blender supports unwrapped designs, which allows designing the texture of all sides of an object using a single image. [17] To create the texture, 24 ArUco markers are generated and realigned in the format required by the unwrapped designer of Blender, which is saved as an image. After loading the image into the unwrapped designer of Blender, the texture can be applied to the payload and saved as a `.dae` file recording the mesh map of the texture and the path to the texture image in the XML format. The section in the `.dae` file specifying the path to the texture image can be replaced to switch the ArUco markers on the payload, which makes it possible to generate a proto texture file using Blender once, and routinely create different `.dae` texture files based on the proto file to avoid repeated manual designing works. An example of the unwrapped design interface of Blender is shown in Figure 4.

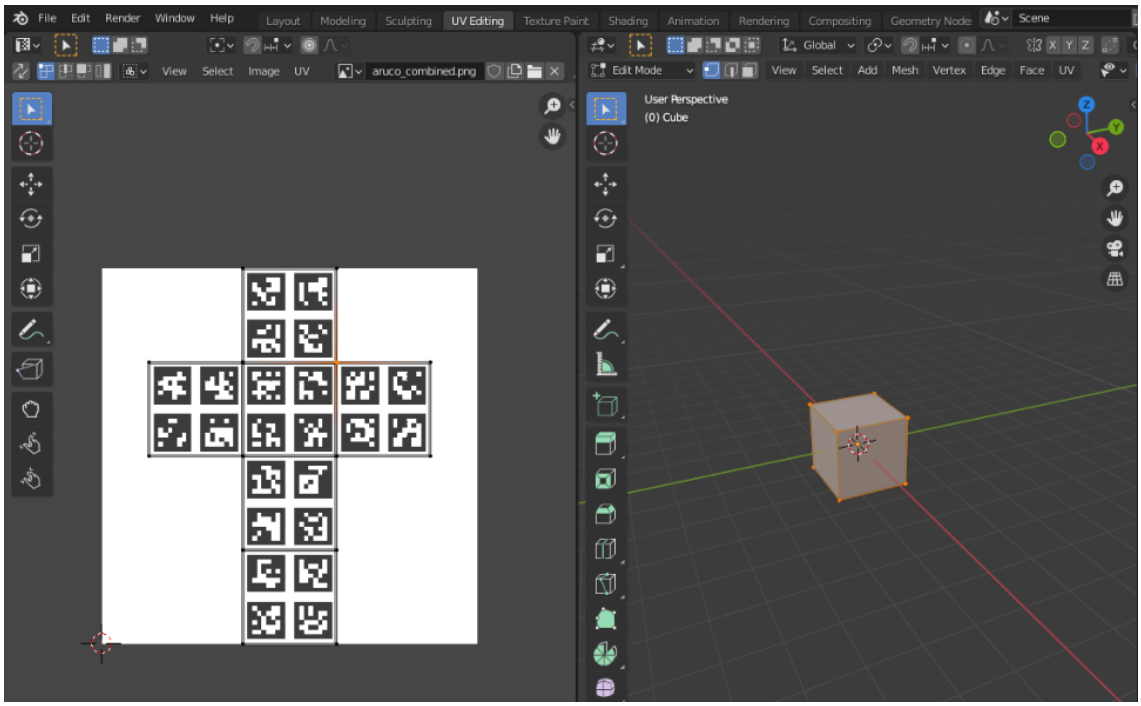

Figure 4 Unwrapped Design of Blender

In the Gazebo simulator, a 0-degree yaw angle is defined as the state when the model is towards the positive x-axis direction, which points to the right on the 2D Cartesian coordinate system. So, when generating the texture image, all ArUco markers are rotated 90 degrees clockwise to be aligned with the 0-degree yaw angle definition of the simulator.

5. Simulations

In the experiments, two paths are defined as the missions of the UAV to simulate cases that may fail the PX4 controller. At the same time, the mass of payload, the distance between the payload and the UAV, the distribution of identifiers on the payload and the distortion coefficient of the bottom camera of the UAV are considered parameters to analyze the performance of the control system of the UAV and the payload pose estimator under different circumstances. The wind plugin of Gazebo is also introduced to apply external disturbances to the slung payload transport system during simulations. QGroundControl [18] is utilized as the ground control station for applying commands and missions to the system.

5.1 Path Defining

Two paths, including a circular and a zigzag path, are created as the missions of the UAV-payload system. Each path contains a series of 3D waypoints with different altitudes. The missions of the system include flying through each waypoint on a particular path, regularly taking pictures for the payload and conduct online estimations on the payload pose. In the

simulations, the UAV takes off at the original point, flies through all waypoints on the path, and returns to the original point after completing the mission.

The circular contains 41 waypoints. The combination of sections between each pair of neighbouring waypoints approximates a circle. The main usage of the circular path is to analyze the performance of the control system of the UAV on travelling through a curved path. Each pair of neighbouring waypoints are assigned different altitudes to require the UAV to perform an ascending or descending when travelling to each waypoint, which allows evaluations of the performance of the control system on vertical motions. The circular path and the altitudes of each waypoint are shown in Figure 5.

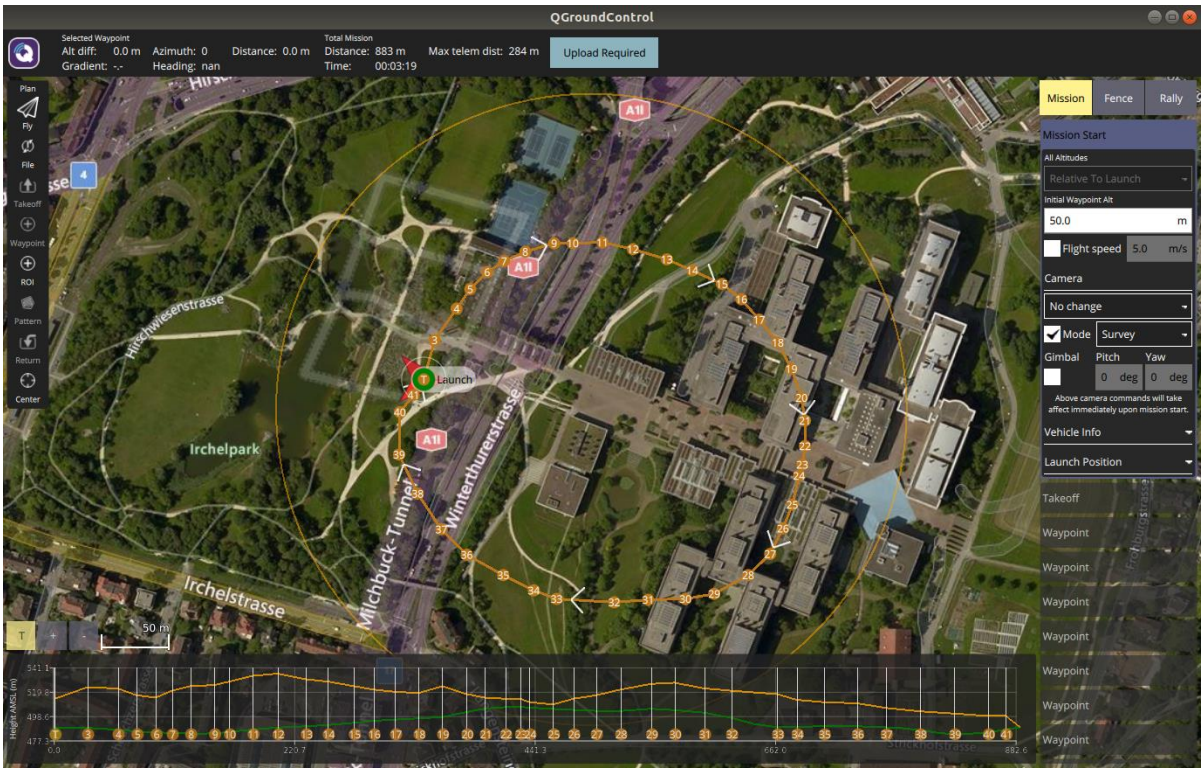

Figure 5 Circular Path

The zigzag path is formed by 7 waypoints. A sharp turn with an angle greater than 150 degrees are expected to be performed by the UAV when reaching each waypoint. This path aims on checking the performance of the control system of the UAV at sharp turns. Similar to the circular path, each pair of neighbouring waypoints are assigned different altitudes for the UAV to perform a vertical motion when moving to each waypoint. The zigzag path is shown in Figure 6.

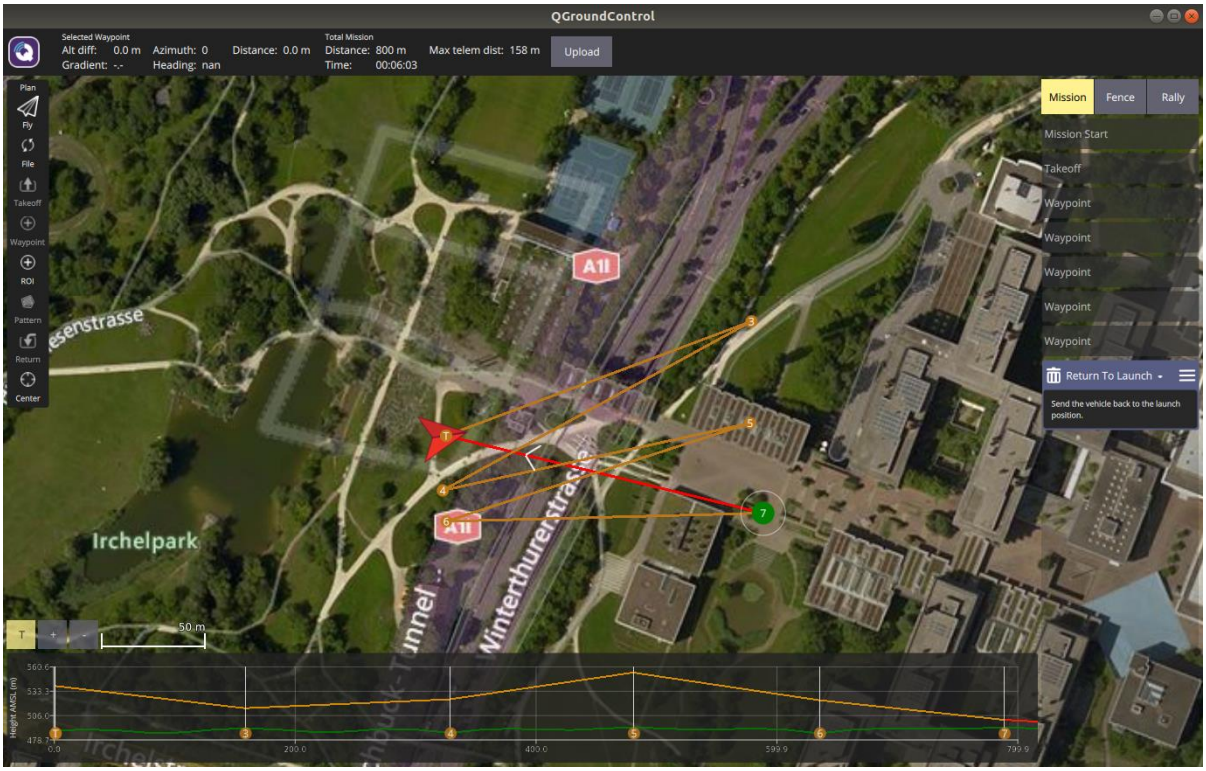

Figure 6 Zigzag Path

5.2 Payload-Related Parameters Setting

In the simulations, the mass of the payload is considered a parameter to verify the stability of the PX4-provided control system of the UAV with different load weights. Since the weight of the UAV model is set to be 1.5 kg, the mass of the payload is assigned to be from 0.5 kg to 1.5 kg with a 0.05 kg increment for each test case. Theoretically, the increase of the weight of the payload will reduce the stability of the system since the PID controller is designed based on the quadrotor itself.

Since the vision-based payload pose estimator can only make estimations when there is at least one ArUco marker fully presented in the camera frame, test cases with different distances between the payload and the UAV are created to evaluate the effects of the motion and the visual size of the payload on the estimation accuracy. A short distance between the payload and the UAV results in a short tether and thus enhances the stability of the payload, but the visual size of the payload will be large and reduces the probability for the camera to capture a complete ArUco marker when the payload is swinging. A large distance reduces the stability of the payload but increases the probability of the payload to fully presented in the camera frame when the payload is relatively stable with respect to the camera. Three distances, 1.6 m, 2.6 m, and 3.6 m are selected.

Two distributions of identifiers are proposed to be applied on the surface of the payload. The

ArUco markers are only applied to the closest surface of the payload to the UAV, which is the upper side of the payload during flights, in one test case. In the other test case, different ArUco markers are pasted on all sides of the payload. Theoretically, the estimation accuracy may be slightly enhanced by applying identifiers to all sides of the payload to increase the probability of the camera capturing at least one identifier. These test cases aim on verifying the necessity of applying identifiers on all sides of the payload to potentially reduce the computational cost for the estimator without significantly reducing the estimation accuracy. The two distributions of identifiers on the unwrapped payload are illustrated in Figure 7.

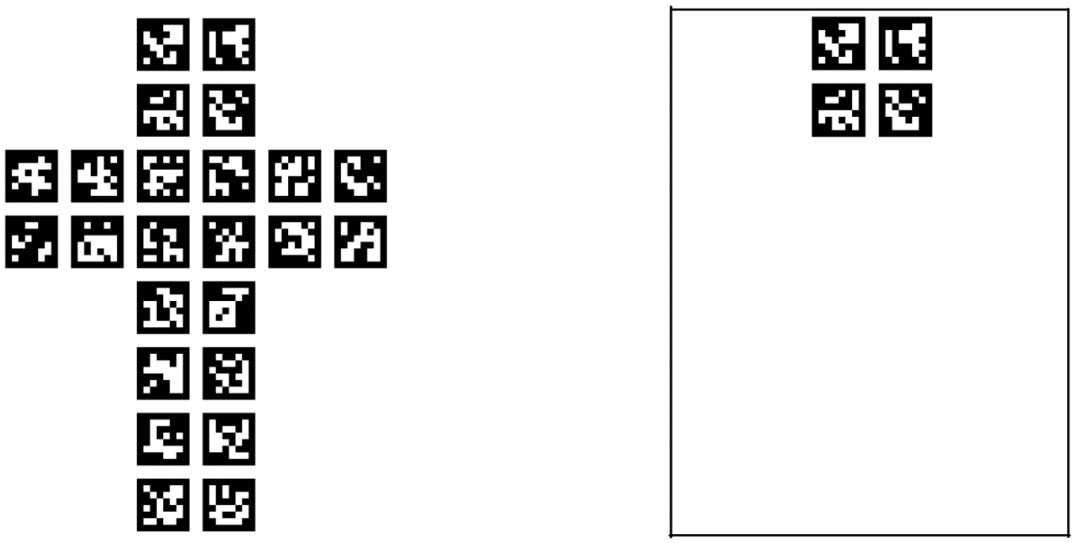

Figure 7 Distribution of Identifiers on Unwrapped Payload

In addition, the distortion coefficient is applied to the bottom camera of the UAV to evaluate the potential effect of image distortions on the accuracy of the estimation. The distortion coefficients used in the simulations are -0.3, -0.001, 0.0, 0.0 and -0.01 as k1, k2, p1, p2 and k3, respectively, which results in a barrel distortion.

5.3 Usage of Wind Plugin

The Gazebo wind plugin is used to simulate winds and wind shears in the experiments. [19] The wind and wind shear created by the plugin function as a parallel force applied to all components in the simulated world. The mean and covariance of the wind speed and direction can be configured. For creating a wind shear, the start time and duration of the wind shear are also available to be defined in addition to the speed and direction. A normal distribution is generated based on the defined speed and direction of the wind and the wind speed and direction at each time step during simulations are generated randomly based on the distribution.

In the simulations, the wind is defined following the positive y-axis with a speed of 4 m/s in the windy test case. For the test case involving wind gusts, the gust is set to be 5 m/s towards the positive direction of the x-axis, y-axis, and z-axis. The wind gust starts at the 10^{th} second and lasts for 100 seconds. It is expected that with the increase of the wind speed, the stability of the system will be negatively affected.

5.4 Summary of Parameters

For evaluating the performance of the PX4 control system, the mass of payload and the wind speed are considered parameters since they are the two main factors that may affect the stability of the system. The test cases for analyzing the performance of the PX4 control system of the UAV where the flight missions are completed are summarized in Table 1:

Table 1 Summary of Test Cases for Controller Evaluation

Mass of Payload (kg)	Wind Speed along Positive y-axis (m/s)
0.5 – 1.5 with a 0.05 increment	0
0.5 – 1.5 with a 0.05 increment	4

In the simulations for evaluating the performance of the payload pose estimator, the distance between the payload and the UAV, the distribution of ArUco markers on the payload and the distortion coefficient are regarded as parameters. In addition, the speeds of wind and wind gusts are considered parameters to evaluate the effect of winds on the motion of the payload and thus the estimation accuracy of the pose. The test cases for evaluating the payload pose estimator are listed in Table 2. The first row in the table is the base case of the simulations.

Table 2 Summary of Test Cases for Estimator Evaluation

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture	Distortion
1.6	0	0	Upper Side	No
2.6	0	0	Upper Side	No
3.6	0	0	Upper Side	No
1.6	4	0	Upper Side	No
1.6	4	5	Upper Side	No
1.6	0	0	All Sides	No
1.6	0	0	Upper Side	Yes

5.5 Experimental Environments

In all simulations explained in this report, Ubuntu 18.04, ROS Melodic, PX4 v1.12.1, Gazebo 9, MAVROS v1.8.0, MAVLink v2021.3.3, QGroundControl v4.1.3 and OpenCV v3.2.0 are

utilized.

6. Results and Analysis

6.1 Performance of PX4 Controller

The length of the real flight path of the UAV is used to evaluate the PX4 control system installed on the UAV since the UAV is expected to follow the shortest path towards each waypoint when completing the missions. A control system that may lead to overshoots or oscillations will lengthen the real flight path. Hence, a shorter simulated flight path represents a more desirable performance of the control system.

The simulation results are summarized in Table 3. The ground truths for the lengths of the circular and zigzag paths are 940.2577 m and 1025.315 m, respectively.

Table 3 Simulation Results of PX4 Control System

Mass of Payload (kg)	Wind Speed (m/s)	Length of Real Flight Path for Zigzag (m)	Length of Real Flight Path for Circular (m)
0.5	0	**1044.857**	**971.8430**
0.5	4	1045.057	972.6948
0.95	0	1051.477	977.2611
0.95	4	1144.737	1372.310
1.3	0	The system touched the ground during flights and failed to complete the mission.	
1.3	4		
1.5	0	The system failed to take off due to the over-weighted payload.	
1.5	4		

From the simulations, it is observed that when the mass of the payload reaches 0.95 kg, the system is stable when there is no wind but unstable when the simulated world is windy. Also, the system becomes unstable without wind and cannot take off when the payload reaches 1.3 kg and 1.5 kg, respectively. Therefore, although test cases are created with the mass of payload assigned to be from 0.5 kg to 1.5 kg with a 0.05 kg increment, the flight missions are completed only at the four aforementioned cases since the purpose of the simulations is detecting edge cases that the control system may become unstable, and thus only the length of real flight paths for the four masses above are summarized in Table 3.

From Table 3, it is obvious that a smaller mass of payload or wind speed results in a shorter length of real flight path and thus a more stable system during the flights. Given a wind-free

simulated world, the control system becomes unstable when the mass of the payload reaches 1.3 kg, which is close to 1.5 kg, the weight of the UAV. Therefore, it is concluded that the performance of the control system on stabilizing the UAV and the whole system is desirable.

6.2 Performance of the Vision-based Estimator

The average difference between the ground truth and the estimation of the payload pose over all time steps is considered the criteria for evaluating the performance of the vision-based estimator. The ground truth of the payload pose is retrieved by applying `gazebo_ros_p3d`, [20] a Gazebo plugin that publishes the inertial pose of any object in the simulated world, to the payload. Since the trend of changes of the estimation accuracies on x, y, z positions and all Euler angles of the orientation are similar over test cases, the average simulation errors on the y position and the pitching angle over timesteps are selected and summarized for analyzing the reliability of the estimator in Table 4.

Table 4 Simulation Results of Vision-based Payload Pose Estimator for y and Pitch

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture Distri-bution	Distor-tion	Circular Path		Zigzag Path	
					y Diff (m)	Pitch Diff (rad)	y Diff (m)	Pitch Diff (rad)
1.6	0	0	Upper	No	0.6833	0.08733	**0.3092**	**0.03562**
2.6	0	0	Upper	No	1.056	0.1217	0.3882	0.06995
3.6	0	0	Upper	No	1.754	0.1860	0.5979	0.08287
1.6	4	0	Upper	No	0.6989	0.09200	0.3195	0.04302
1.6	4	5	Upper	No	0.7103	0.09143	0.3108	0.04009
1.6	0	0	All	No	**0.6595**	**0.08468**	0.3160	0.03704
1.6	0	0	Upper	Yes	0.7396	0.1010	0.3453	0.06931

From the results above, it is obvious that when the distance between the payload and the UAV increases, the estimation accuracy significantly drops. Also, the estimation accuracy decreases when distortion coefficients are applied to the bottom camera of the UAV. The wind and wind gust and the distribution of identifiers on the payload do not exert any apparent effect on the performance of the payload pose estimator comparing to the length of the tether and the camera distortion. Only a slight decrease in the estimation accuracy is observed when winds and gusts are introduced in the simulated world. Applying identifiers to all sides of the payload slightly enhances the performance in the circular path cases, but an opposite result is observed from the zigzag path cases.

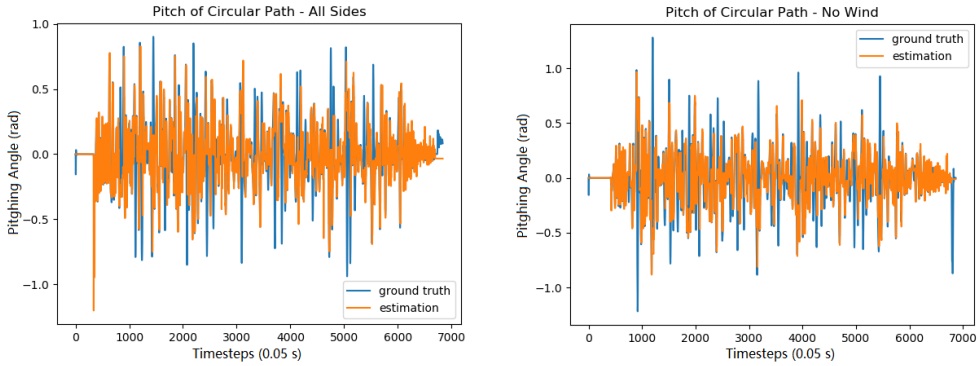

Figure 8 Simulation Results on Pitching Angle for Circular Path

Figure 8 above shows the ground truths and estimations of the pitching angle of the payload for the two test cases that result in the best performance of the estimator at the circular path. In the diagram, the "All Sides" case refers to the case that the identifiers are applied to all sides of the payload while the "No Wind" case represents the case recorded at the first row of Table 4. The diagram demonstrates that the main cause of the estimation error is the swings of the payload. When the system is following a curved path or making sharp turns, large pitching angles are recorded, which represent strong swings of the payload. Among the large-pitching-angle cases, a smaller pitching angle is estimated compared to the ground truth, which is possibly due to the payload being out of the camera frame and thus the estimation is unable to be conducted.

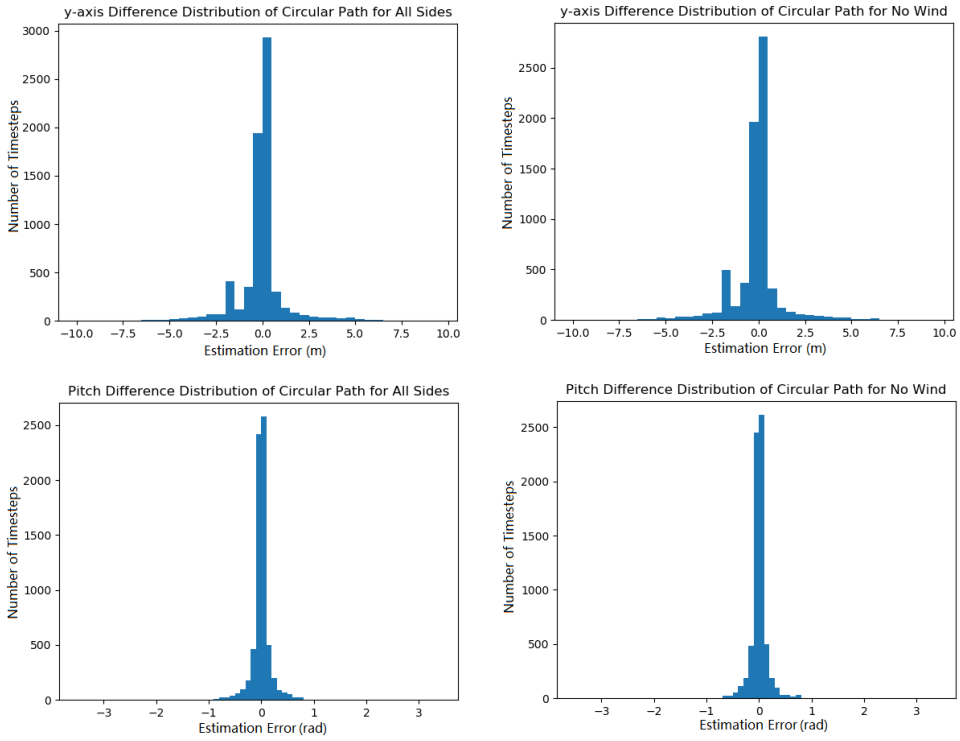

Figure 9 Distribution of Estimation Errors at Time Steps for Circular Path

Figure 9 illustrates the distribution of estimation errors at each time step of the y position and the pitching angle for the two test cases that result in the best performance of the estimator at the circular path. The diagram demonstrates that at more than 70% of time steps, the errors on the y position and the pitching angle are smaller than 0.5 meter and 0.1 radian, respectively, which indicates that the estimator has a strong performance when at least one identifier is fully within the camera frame. A spike is shown at around -2 m error for the y position in both cases, which represents the estimation error during taking-off and landing. Since the payload cannot be captured by the bottom camera of the UAV when the altitude of the system is below the distance between the UAV and the payload during taking-off or landing, it is expected that a large estimation error is presented.

6.3 Effects of the Properties of Payload

From the simulation results summarized in Chapter 6.1, there is a negative correlation between the mass of the payload and the stability of the system during flights. The PX4-provided PID control system for the IRIS quadrotor model is originally tuned based on the specifications of the quadrotor, which may not be stabilized for a slung payload system with a significantly larger weight than the quadrotor. Therefore, it is expected that the stability of the control system reduces with the increase of the weight of the payload and the PID control is required to be redesigned for supporting heavy-payload cases. Also, the simulation results are consistent with the theoretical expectations on the trend of the stability of the system at the increasing of the mass or the wind speed, which shows that the simulation environment is reliable.

From the simulation results shown in Chapter 6.2, a negative correlation is presented between the length of the tether, which is the distance between the UAV and the payload, and the average estimation accuracy over time steps for both simulation paths. As discussed in Chapter 5.2, a large distance between the UAV and the payload results in a small visual size of the payload in the camera, which allows a high probability of the payload to be within the camera frame when the payload is relatively stable, but stronger swings of the payload during flights. The simulation result demonstrates that the benefits of increasing the length of the tether on reducing the visual size of the payload do not overwhelm the drawbacks in reducing the stability of the payload. Hence, a shorter tether is preferred for a more stable payload during flights to enhance the estimation accuracy.

According to the simulation results, the distribution of the identifiers on the surface of the payload does not exert any obvious effect on the estimation accuracy. Since the tether is attached to the central point of the upper side of the payload and the tether is a cylinder-like rigid body, the probability of the payload to perform a flipping with pitching or rolling angle larger than 90 degrees can be neglected, which ensures that the upper side of the payload to be

visible by the camera during flights. Therefore, applying identifiers at all other sides of the payload is not necessary given the identifiers on the upper side are available to be captured by the camera throughout flights, and the simulation results are theoretically reasonable.

6.4 Effects of Wind

The simulation results show a clear negative correlation between the wind speed and the stability of the control system of the UAV. Similar to the installation of the payload, an external force is applied to the UAV when the wind is introduced, which is not taken into consideration on designing the default PID controller of PX4. So, the default PX4 controller is expected to not support the cases with high wind speeds, which corresponds to the simulation results.

On the contrary, the performance of the payload pose estimator is not apparently affected by the winds and wind gusts. Since the wind provided by the Gazebo plugin applies parallel forces to all objects in the simulated worlds, the motion of both the payload and the UAV may be aligned due to the inertia when the difference between the weight of the payload and the UAV is small. Hence, the payload is available to be captured by the camera in a windy simulated world given the payload is in the camera frame prior to any change of wind speeds or the arrival of wind gusts, and thus the winds do not significantly affect the performance of the estimator. In the real world, the wind may not be parallel, the performance of the estimator may then be apparently affected by winds in such cases.

6.5 Effects of the Visual Component

From the simulation results, when distortion coefficients are applied to the camera, the performance of the payload pose estimator decreases. As described in Chapter 4.3, distortions are applied to the captured images while the sides are removed in the processed images on Gazebo cameras, so the resulting field of view is reduced when distortion coefficients are applied. Therefore, the probability of the payload being out of the camera frame increases, which is one potential reason for the decrease in the estimation accuracy. At the same time, image processing steps for applying distortion coefficients in the visual component and the corrections of images from the distortion in the estimation processes may result in information loss, which may also potentially exert a negative effect on the estimation accuracy. Hence, to enhance the performance of the estimator, a non-distortion camera is preferred.

7. Conclusion

This project proposed and developed a simulation environment for verifying a slung payload transportation system by UAV based on Gazebo. An IRIS quadrotor provided by the PX4 Autopilot is utilized to carry a box-like payload via a cylinder-like tether. QGroundControl is

used as the ground station to command the system during simulations, which allows user-defined arbitrary paths for simulating edge cases. A camera is installed at the bottom of the UAV while ArUco markers are applied on the surface of the payload as identifiers, which allow vision-based payload pose estimation. The Gazebo wind plugin is used to simulate parallel winds and wind gusts.

The simulation results demonstrate that the quadrotor is stable for most test cases and thus the performance of the PX4-based PID control system is desirable for stabilizing the UAV. The control system failed to stabilize the system only when the weight of the payload is close to or exceeds the weight of the UAV in a windy simulated world. However, the payload is not stabilized by the control system.

The estimator demonstrates a desirable performance on simulating the payload pose when at least one identifier on the payload is fully captured by the camera. Most errors in estimation are derived from the swings of the payload out of the camera frame. Also, the estimation accuracy decreases when a distorted camera is utilized. Future development will be focusing on enhancing the stability of the payload and expanding the field of view of the camera to increase the probability of the payload being within the camera frame.

Overall, the simulation environment makes it available to simulate the performance of a control system as well as a visual-based estimation system. The performance of the control system and the estimator from the simulation results are consistent with the theoretical analysis, which demonstrates the reliability of the simulation environment. The simulation environment is expected to not benefit the development of the control system and the estimator in this project, but similar vision-based projects related to slung payload transportation systems in the future.

8. Future Development

The Gazebo wind plugin currently used in simulations applied a parallel force as the wind to all components in the simulated world. However, in the real world, the wind may not be parallel. A frictional drag exists between the air and the ground and slows down the wind near the ground or any other surface. [21] So, the wind speeds at different heights are normally different. At the same time, whirlwinds, such as dust devils, cannot be simulated using parallel forces. Therefore, the wind plugin is expected to be enhanced to supporting different wind speeds over altitudes as well as non-parallel winds.

Future research on the location of the camera can be conducted by analyzing the common directions of swings of the payload during flights. For example, if most swings of the payload

are along the x-axis, the camera can be placed horizontally on the x-y plane since the length of the frame of the camera is 1.78 times longer than the width. Also, to expand the field of view, multiple cameras can be placed in parallel at the bottom of the UAV while the estimations will be made based on the combined images captured by all bottom cameras. Also, multiple cameras can be installed at the sides of the UAV to estimate the pose of the payload during taking-off and landing.

The simulation results demonstrate the high performance of the control system on stabilizing the UAV. However, the stability of the payload is not considered since the PID control is tuned based on the properties of the UAV, which results in heavy swings of the payload during flights and decreases in the accuracy of the vision-based payload pose estimation. To stabilize the payload, the controller can be enhanced by either tuning the PID controller based on the overall performance of the system or introducing an LQR control for trajectory tracking. For the LQR tracking method, the integrals of the tracking error on the speed and Euler angles of the payload can be defined as states and assigned the same weight as the corresponding tracking errors of the UAV in the Q matrix.

Furthermore, the simulation results show that a relatively short tether results in a more stable payload and thus a higher estimation accuracy. Future works can be conducted to find the optimal tether length for different sizes of payload.

Acknowledgement

This project is completed as the MEng Project of Summer 2021 with the supervision of Prof. Hugh Liu and the assistances and collaborations of Mr. Helson Go and Mr. Erik Chau at the Flight Systems and Control Lab of the University of Toronto Institute for Aerospace Studies.

Reference

[1] N. Koenig, et al., “Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator”, *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol.3, pp. 2149-2154, 2004.

[2] S. Shah, et al., “AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles”, *Field and Service Robotics*, pp. 621-635, Nov. 2017.

[3] H. Willee, et al., “jMAVSim with SITL”, *PX4 User Guide*, 17-Jun-2021. [Online]. Available: https://docs.px4.io/master/en/simulation/jmavsim.html. [Accessed 16-Aug-2021].

[4] PX4 Autopilot. [Online]. Available: https://px4.io. [Accessed 16-Aug-2021].

[5] E. Chau, “On-board Computer Vision-Based System for Rigid Body Slung Payload Pose Estimation”.

[6] G. Tartaglione, et al., “Model Predictive Control for a Multi-Body Slung-Load System”, *Robotics and Autonomous Systems*, vol. 92, pp. 1 – 11, June 2017.

[7] D. Shi, et al., “Harmonic Extended State Observer Based Anti-Swing Attitude Control for Quadrotor with Slung Load”, *Electronics*, vol. 7, no. 6, p. 83, May 2018.

[8] D. Shi, et al., “A Practical Performance Evaluation Method for Electric Multicopters”, *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 3, pp. 1337-1348, June 2017.

[9] A. Hentati, et al., “Simulation Tools, Environments and Frameworks for UAV Systems Performance Analysis”, *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 1495-1500, 2018.

[10] L. Qian, et al., “Guidance and Control Law Design for a Slung Payload in Autonomous Landing A Drone Delivery Case Study”, *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 4, pp. 1773-1782, Aug. 2020.

[11] S. Garrido-Jurado, et al., “Automatic Generation and Detection of Highly Reliable Fiducial Markers under Occlusion”, *Pattern Recognition*, vol. 47, no. 6, pp. 2280-2292, June 2014.

[12] R. Chiappinelli, “jMAVSim source code”, [Source code]. Available: https://github.com/PX4/jMAVSim/tree/master/models. [Accessed: 18-Aug-2021].

[13] “Make a Model”, *Gazebo*. [Online]. Available: http://gazebosim.org/tutorials?tut=build_model. [Accessed: 18-Aug-2021].

[14] “Iris - The Ready to Fly UAV Quadcopter”, *ArduCopter*. [Online]. Available: http://www.arducopter.co.uk/iris-quadcopter-uav.html. [Accessed: 18-Aug-2021].

[15] Doxygen, “Detection of ArUco Markers”, *OpenCV*, 02-Apr-2021. [Online]. Available: https://docs.opencv.org/4.5.2/d5/dae/tutorial_aruco_detection.html. [Accessed: 18-Aug-2021].

[16] L. Itti, “Simple demo of ArUco augmented reality markers detection and decoding,” *JeVois Smart Machine Vision Camera*. [Online]. Available: http://jevois.org/moddoc/DemoArUco/modinfo.html. [Accessed: 18-Aug-2021].

[17] A. Verma, “Blender 2.9: Uv mapping – simply explained,” *All3DP*, 09-Mar-2021. [Online]. Available: https://all3dp.com/2/blender-uv-mapping-simply-explained. [Accessed: 18-Aug-2021].
[18] QGroundControl. [Online]. Available: http://qgroundcontrol.com. [Accessed: 19-Aug-2021].
[19] H. Willee, “Gazebo Simulation”, *PX4 User Guide*, 17-Jun-2021. [Online]. Available: https://docs.px4.io/master/en/simulation/gazebo.html. [Accessed: 19-Aug-2021].
[20] J. Hsu, “gazebo_ros_p3d.cpp File Reference”, *gazebo_plugins Documentation*, 24-Jan-2021. [Online]. Available: http://docs.ros.org/en/melodic/api/gazebo_plugins/html/gazebo__ros__p3d_8cpp_source.html. [Accessed: 19-Aug-2021].
[21] J. D. Anderson, “Wall Boundary Conditions,” in *Fundamentals of aerodynamics*, 6th ed., New York, NY: McGraw Hill Education, 2017, pp. 241–242.

Appendix

Table A-1 Simulation Results of Vision-based Payload Pose Estimator for x and Roll

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture Distri-bution	Distor-tion	Circular Path		Zigzag Path	
					x Diff (m)	Roll Diff (rad)	x Diff (m)	Roll Diff (rad)
1.6	0	0	Upper	No	0.6623	0.2169	0.4867	0.1329
2.6	0	0	Upper	No	0.9858	0.2751	0.5833	0.2621
3.6	0	0	Upper	No	1.829	0.3267	1.126	0.1912
1.6	4	0	Upper	No	0.6612	0.1733	0.4900	0.1309
1.6	4	5	Upper	No	0.6559	0.1664	0.4698	0.1341
1.6	0	0	All	No	0.6534	0.1737	0.4799	0.1697
1.6	0	0	Upper	Yes	0.7150	0.2095	0.5038	0.1725

Table A-2 Simulation Results of Vision-based Payload Pose Estimator for z and Yaw

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture Distri-bution	Distor-tion	Circular Path		Zigzag Path	
					z Diff (m)	Yaw Diff (rad)	z Diff (m)	Yaw Diff (rad)
1.6	0	0	Upper	No	0.1583	0.06654	0.1158	0.03168
2.6	0	0	Upper	No	0.2247	0.1630	0.1365	0.1100
3.6	0	0	Upper	No	0.4321	0.2782	0.2293	0.2375
1.6	4	0	Upper	No	0.1656	0.06975	0.1170	0.03773
1.6	4	5	Upper	No	0.1652	0.07138	0.1146	0.03847
1.6	0	0	All	No	0.1591	0.06696	0.1188	0.06952
1.6	0	0	Upper	Yes	0.2454	0.05780	0.2202	0.03341

Table A-3 Resulting Std-Dev of Vision-based Payload Pose Estimator for x and Roll

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture Distri-bution	Distor-tion	Circular Path		Zigzag Path	
					x Std (m)	Roll Std (rad)	x Std (m)	Roll Std (rad)
1.6	0	0	Upper	No	1.313	0.5391	0.6927	0.3862
2.6	0	0	Upper	No	1.920	0.5962	1.101	0.6429
3.6	0	0	Upper	No	3.369	0.5960	2.003	0.4246
1.6	4	0	Upper	No	1.304	0.3833	0.6955	0.3738
1.6	4	5	Upper	No	1.297	0.3703	0.6557	0.3835
1.6	0	0	All	No	1.293	0.3804	0.6877	0.4415
1.6	0	0	Upper	Yes	1.414	0.4116	0.7648	0.4035

Table A-4 Resulting Std-Dev of Vision-based Payload Pose Estimator for y and Pitch

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture Distri-bution	Distor-tion	Circular Path		Zigzag Path	
					y Std (m)	Pitch Std (rad)	y Std (m)	Pitch Std (rad)
1.6	0	0	Upper	No	1.261	0.1506	0.4833	0.05685
2.6	0	0	Upper	No	1.866	0.1844	0.7434	0.1239
3.6	0	0	Upper	No	2.868	0.2667	1.061	0.1170
1.6	4	0	Upper	No	1.338	0.1491	0.4946	0.06801
1.6	4	5	Upper	No	1.333	0.1512	0.4903	0.06511
1.6	0	0	All	No	1.228	0.1621	0.4924	0.07015
1.6	0	0	Upper	Yes	1.351	0.1521	0.5166	0.09791

Table A-5 Resulting Std-Dev of Vision-based Payload Pose Estimator for z and Yaw

Distance (m)	Wind Speed (m/s)	Gust Speed (m/s)	Texture Distri-bution	Distor-tion	Circular Path		Zigzag Path	
					z Std (m)	Yaw Std (rad)	z Std (m)	Yaw Std (rad)
1.6	0	0	Upper	No	0.2577	0.1430	0.1634	0.03344
2.6	0	0	Upper	No	0.3678	0.2710	0.2034	0.2166
3.6	0	0	Upper	No	0.7127	0.4333	0.3493	0.3323
1.6	4	0	Upper	No	0.2752	0.1046	0.1705	0.03692
1.6	4	5	Upper	No	0.2707	0.1007	0.1668	0.04054
1.6	0	0	All	No	0.2587	0.1288	0.1663	0.2443
1.6	0	0	Upper	Yes	0.2923	0.09017	0.1925	0.03717