8 May 2019

From: Asst Prof Evangelista
To: MIDN 2/c Credle
Subj: COMMENTS ON EW502 PROPOSAL
Encl: (1) Markup of EW502 proposal

1. Some comments on the EW502 proposal are attached.

2. **Overall comments:**

a. The proposal has some typos (e.g. "que" meaning cue), formatting inconsistencies, and appears not to follow the IEEE format exactly. Several of the figures are difficult to read / unclear what they are meant to communicate.

b. **In some places, comments given on earlier drafts appear not to be incorporated.**

c. The purpose of the research still comes through as garbled. The main idea comes from wanting to start Autonomous Drone Racing (e.g. AlphaPilot challenge) at USNA and finding it would be too big an undertaking for a single midshipman or capstone team to take on. Part of what we want is a road map for a serious USNA entry, and serious work at the first part of that road map that makes sense as an honors project - which you have selected as identifying gates and plotting a course and corrections to get through the gates. The proposal leaves out potential military (e.g. NSW funding source) interest in reducing the cognitive load on an operator by making such tasks easier (also of interest to drone race pilots).

d. Another dimension to the work that is only briefly mentioned is the interest in how the autonomy should work. Should a computer be naively turned loose on the gate problem and be allowed to operate as a black box? Should we study how human pilots operate, which is of interest because USNA has a unique School of Drones program where pilots would be available to study?

e. MIDN 1/c Wenberg provided information on visual servoing including a pile of references; I cannot tell where these are in the proposal and how this work is related to those previous USNA efforts.

3. Many of the comments that seem to have been ignored from previous drafts address the issues above. Possibly the disconnect is due to confusion about EW502 templates etc. To simplify this in future work under EW495/EW402, use of the Latex workflow with revision control in Git / shared Github repository will be required. Future deliverables

should be provided in advance to the advisor well before they must be signed or turned in.

D EVANGELISTA

Copy to:
EW502 instructor
File (Credle capstone 9020)

Use IEEE format.

Autonomous Drone Racing

A. Credle and D. Evangelista

?

Abstract

purpose garbled

Autonomous racing drones are not only beneficial to the drone racing sport, but also in autonomous vehicles and in military drones flying through windows, chimneys, etc. Research in this field is often based around either race gate recognition or flight controls, but this project will incorporate both. We will create a process for a drone to modify its given flight path based on the visual recognition of the gate in order to fly through the gate. The processor will identify the gate, locate it in relation to the drone's current path, and modify the path to fly through the gate. This project will demonstrate this process by using simulations and physical experiments. Simulations will consist of individual tests of recognition, location, and path modification processes, and then tests for all three together. Physical experiments will test all three process together in a real environment. Success will be measured by the difference in distance between the path and the center of the gate, as well as the difference in trajectory at the same point. The total cost of the project is $29,666- with $1586 of that being new equipment. This project is scheduled to have the coding and simulation done within the first semester and physical experiments done by the end of the second. The biggest risk involved is with the drones crashing, so the physical experiments have more time than simulations to ensure safety. If successful, this will be the first step in creating an autonomous racing drone at USNA.

naive AI vs what human does?

format inconsistent

use consistent terms everywhere

I. Background and Motivation

UAS

The concept of drone racing is straightforward: a group of people fly unmanned aerial vehicles (UAV) through gates, the first one to the finish line wins. The UAVs, or drones, come in a variety of sizes, ranging from an inch in diameter to over a foot, and are flown through communication with a remote controller. Races can take place in a variety of conditions including time of day and location. Gates can be configured in an unlimited number of patterns and often come in circular or rectangular form, though they are not limited to these shapes. Limitations are often placed on the power and size of drones that are raced, as owners are often expected to bring their own equipment [1]. While the concept of racing is not new, drone racing is one of the fastest growing sports in the world [2]. Its fame is quickly growing, and sports broadcasting companies like ESPN are signing contracts to broadcast competitions [3]. There are no limitations on age, gender, ethnicity, or physical prowess, making drone racing available to everyone.

what does this add?

Figure 1: Two drones racing

?

The concept of autonomous drone racing is even newer, though its potential goes beyond that of its sport. In concept, a human drone racer goes through a series of ques and maneuvers to race the drone. If these ques and maneuvers are broken down into systematic steps, it is possible to automate the process and have the drone fly itself through the course. In modern autonomous vehicles, the specific location of the vehicle is often known, through GPS, a tracking system,

↳ what does AlphaPilot assume?

? cue.

Where are Weinberg's visual servo refs?

etc. While this method of localization works for vehicles with static environments, it does not take into account the location of its surrounds, or the vehicles location in relation to specific objects.

By mixing both an assumed path and gates for the drone to fly through, the exact location of the drone is not required; its relative location to the gates is all that is needed to fly a successful race. This concept will be critical for future autonomous systems, such as self-driving cars, that are required to navigate based on an assumed path mixed with the vehicles relative to the environmental objects or barriers (Fig. 1) [4]. For cars, the environmental objects would include lane lines, other vehicles, curbs, objects in the road, etc [5]. By integrating objects of the environment into the core of the navigation process, autonomous vehicles will be able to keep both passengers, pedestrians, and wildlife safe. Autonomous vehicles of the future must be able to navigate based on both an assumed path and a relative location in order to efficiently and safely navigate dynamic environments.

What does this add?

Figure 2: An autonomous car sensing its environment

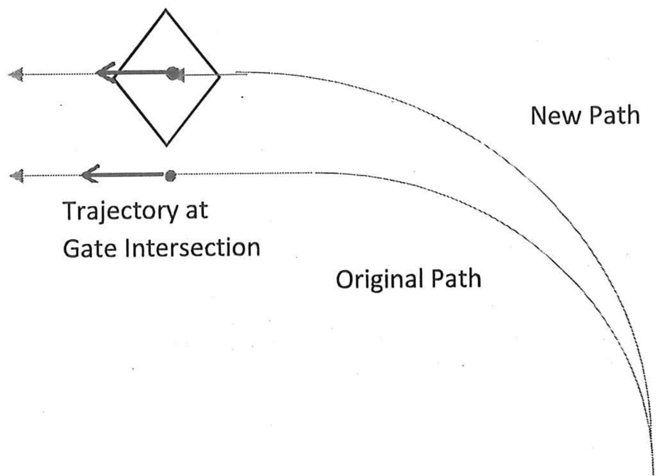
As the first autonomous drone-racing project at USNA, this will open the door to future drone research for midshipmen. EW281 and EW282 provide opportunities for midshipmen to experiment with drones on a hardware and flight-testing level, but this project will allow for future drone development on the software and autonomy level. Additionally, this research is only the first step in creating a fully autonomous racing drone that rivals a human's performance. Future steps include path optimization, recognition with visual uncertainties, racer collision avoidance, etc. This research will be one small step in the larger picture of fully autonomous drone racing.

Setup research Q here?

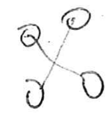
II. Problem Statement

Given a three-dimensional flight path, drone accelerometer readings, and the image of a drone-racing gate in a three dimensional environment, this research intends to begin the process of developing autonomous drone flight through a drone-racing course. Given a three-dimensional path that does not fly through the gate, we will find and test the feasibility of a guidance system that creates a similar path that does. This system will be tested by placing a small quadrotor in multiple scenarios in relation to a race gate(s), with varying pre-determined paths, and directing the guidance system to fly the drone through the race gate(s). Success of this system is based on the ability to autonomously correct the path and fly the drone through the center of the gate without contact. This will be measured by the difference in distance between the path at its intersection with the gate, and the center of the gate. The competing objective will be to maintain the same trajectory at the intersection with the gate as if the path had not been modified. The final output of this project will a guidance system that could be implemented on any racing drone with a camera and IMU and fly through race gates autonomously.

previous edits ignored..



Top view?



is it really so hard to draw a drone?

Figure 3: Depiction of flight path correction (original path in green, new path in blue, trajectory at gate intersection as red arrow, gate intersection point as red dot)

Make so it reads in B&W. don't use word art. edits ignored.

III. Literature review

This paper is the one of the first developments of a drone autonomous flight program, so the background research encompasses papers that focus on multiple aspects of autonomous flight. The main categories that these papers include navigation, flight controls, and visual recognition. This research focuses on the visual recognition of the gate and navigation, but these flight controls is necessary for the background research because it affect the performance of navigation and is necessary for basic demonstration.

In order to understand the movement patterns of the drone, [6] provides a set of equations that could be used to accurately model the induced drag and thrust forces. These equations were derived from proofs using properties of physics, and then the coefficients were identified through flight experimentation. This provides suitable estimations for the forces acting on the drone, which allows for control of the drone through the gate. This is connected to [2], which created a state space model that could accurately predict the movement of a small drone with aggressive flight. By determining these equations through modeling and live testing, these state equations can be coupled with the flight force equations from [6] to form an accurate estimate of the drone's location. Coupling these with a flight controller will allow the drone to have a basic understanding of its location within the general space. This research intends to use visual servoing to adjust a drone's flight paths, and in order to understand the drone's relation to the predetermined flight path, [6] and [7] provide an accurate location.

While [6] and [7] provide what is believed to be an accurate location of the drone in-flight, its location in relation to the preprogrammed flight path is not always certain. [8] provides the groundwork for a concept known as "nano mapping", which refers to modeling a 3D data structure depicting obstacles around the drone. This technique is different from a traditional mapping approach because traditional mapping is based off a common world frame. [8] is based solely on the frame of the drone, allowing for navigation based off the drones currently location, rather than its location relative to a known point. This concept is also crucial to the development of this research because the coupled flight path planning approach requires both the common world frame and the drone world frame.

Different methods for seeing the world around the drone exist, including range finding technology (used for altitude measurements), single camera, and multi-camera approaches. [7] is based on a single camera approach, while [4] found that a three camera setup was optimal for uncertain terrains. [9] is based on autonomous navigation of drones in a wooded area where the location of the trees is unknown before flight. The drone does not have a preprogramed path, but rather has a program to recognize key features along a route and to estimate the likelihood of needing to turn. While the racecourse for this research is assumed to be known, a single camera approach will be utilized for hardware simplicity; estimating the likelihood that the computer is seeing a gate, however, may be utilized in this research. If a neural network could be constructed to follow a path through a forest, a similar neural network could be formed to fly through an aerial path.

One of the main challenge that arises with flying through the gate is recognizing multiple gates. In small drone racing courses, it is likely that the drone will be able to see multiple gates within its field of view. [10] worked through this problem and found reasonable neural network parameters to mitigate this issue. By assuming the next gate in the racecourse would be the largest gate in the drone camera view (using a single camera approach), [10] removed the neural network layer of all image analysis and left only the recognition of the largest circle gate. By removing this process in the image analysis, [10] found a significant reduction in computation time while only reducing the accuracy of gate recognition by less than 10% (464ms to 34ms reaction time, 82.4% to 75.5%).

The other main challenge that arises after recognizing the proper gate to fly through is implementing a visual servoing program to direct the drone through the gate. [11] created a process of gate recognition that located the center of the gate in relation to the location of the center of the cameras view and redirecting the drone towards the center of the gate. This research is the closest process to the research in this paper, as it specifies a visual recognition process that is simplified into the computational level of an inflight processor.

While many of these papers are useful in the development of an autonomous drone, it should be noted that there lacks a consistency among drone researchers in the assumptions and underlying assumptions. As described earlier, some researchers utilize a single camera operating system, while others use multi-camera systems, and even some include range-finding technology. Many of the subjects of these papers delved into different subsections of drone autonomy, so it is reasonable that their methods varied. For research such as [6], [7], and [8], their focus was more for drone flight in general, so this critique does not apply as much as it does for projects such as [9], [10], and [11], which had varying sensor and visual capabilities. As the latter 3 projects had more to do with direct visual recognition, it would have been more advantageous for the autonomous drone community had the projects been embarked on in a similar manner.

These projects center around the research in this proposal in two ways: by giving basic flight control and navigational equations to use in baseline location estimations and flight controllers, and by providing simplistic approaches to visual recognition to model my approach. While my approach uses both a whole world frame and a drone view frame to form a novel approach to gate recognition, many of the approaches depicted in [9], [10], and [11] can be used to model a realistic approach to melding the two frames. This project will take these approaches and form a simple and novel method to visual gate recognition and path modification.

IV. Demonstration Plan

In this project, it is important to ensure that factors not related to gate recognition and visual servoing do not hamper the performance of the drone. This research intents to recognize, locate, and fly through a drone-racing gate, so all other aspects of drone flight will be controlled. Initial testing will be conducted in drone flight simulators to allow for repeatability and a constant testing environment. This will ensure that gate recognition and visual servoing processes can be refined and optimized before moving to actual flight-testing.

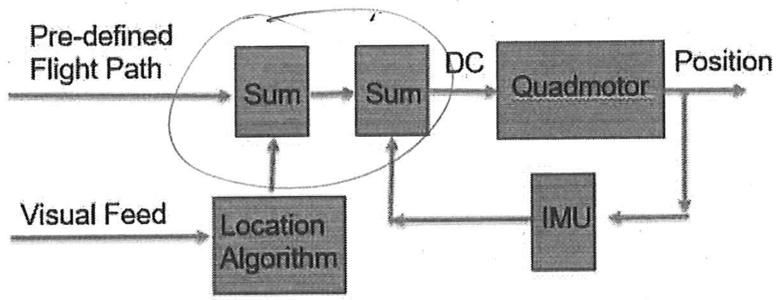
In order to demonstrate the effectiveness of the recognition and flight-path correction processes, a mix of simulations and experiments will be required. First, simulations will be run to test each of the "recognize, locate, and correct path" processes. Secondly, a simulated course will be run in order to demonstrate ideal drone capabilities with mixed gate recognition and path correction programming. Last, a physical experiment with a small, single camera quadcopter with an internal IMU will with a variety of racecourse configurations to ensure its practicality with real world environmental factors. Experimentation will occur indoors so that these factors will be minimal and repeatable.

optic flow left out?
visual servo left out?

A. Simulation or computational studies

Simulation testing for the first two sections of the demonstration is necessary because of its control, replicability, and generality. In the drone flight simulators, a myriad of flight controls can be controlled with the click of a button. Every movement of the drone will be calculated and ideal. This is optimal for initial demonstrations for two reasons. First, it allows environmental factors to be controlled; the simulation can be programmed to fly the drone with specific and repeatable factors such as altitude loss, wind, drag, etc. Secondly, simulation testing can be replicated at a high frequency. If thousands of test runs are collected within an hour, weak points in the different processes tested can be identified efficiently.

Simulation testing will consist of two main section: individual testing, and group testing. Individual testing will isolate each of the "identify, locate, and modify" processes and analyze their performances. For the identify process, a rectangular gate, approximately 4 feet wide and 3 feet tall, will be inserted into the simulation environment. The simulated drone will also be placed in the environment, approximately 5 to 20 feet away from the gate at 0 to 150 degree angles from the orthogonal vector of the gates plane. The processor will be tasked with identify the gate only, and it will be optimized until it has at least a 95% success rate. Next, the locate process will be tested. The same simulator will be run, with the same gates, except the processor will be tasked with estimating the approximate location of the gates target point in reference to the drone. The target point will be one of five points at which the drone intends to cross the gates plane, be it at the center of the gate or one of the corners. The processor, knowing the dimensions of the gate and the target point, will be able to calculate the angle offset from the orthogonal, and the distance from the target point. This process will be tested for every target point, from a spectrum of angle offsets, and will be optimized until it has at least a 95% success rate. This is the first step in the Location Algorithm in Fig 4. Finally, the modify process will be tested. The simulation set up for this test will be different, as the angle offset and distance from target point will be known. The processor will take the original path (which does not fly through the target point) and calculate the trajectory of the drone on the original path as it would pass through the target point. Then, the processor will create a new path with the same trajectory at the target point and the smallest deviations in trajectories along the rest of the path. This is second part of the Location Algorithm and the first sum block in Fig. 4. The simulation will then fly the drone through the new path and the drones location and trajectory at the target point will be logged. This process will be repeated and optimized until a suitable deviation is achieved. ?



?

Figure 4: Block diagram for the drone and controller

The group testing will consist of all three "identify, locate, and modify" processes. The drone will be placed in the simulator with a gate, and the processor will be tasked with identify the gate, calculating the distances and angle to the target point, and modifying the drones flight path to hit the target point. The drone will then fly the new path, and trajectory data will be collected to analyze. This entire process will be repeated and optimized to ensure the next phase, experimental work, can be completed without great risk to the drone or observers.

coordinate w/ Marcello?

B. Experimental work

The second phase of demonstration for this project includes proof-of-concept experiments of a drone flying through single and multi-gate racecourses. First, a single gate course will be created in order to test "locate, identify, and modify path" with real world factors. The drone will be set to take off from the ground and climb to 1 foot of altitude. The drone will then recognize a gate placed within its field of view, modify its flight path to fly through the gate, and fly the flight path. The drone will continue through the gate and fly a straight path for 2 feet, slow to a hover, and land on the ground. This test will be run from as many approach angles as possible with as many original flight paths as possible. Trajectory data will be collected and analyzed in order to optimize the drone's performance in real world conditions.

Next, a multi-gate course will be created (2-5 gates) in which the drone will fly through the gates in sequential order. Sequential order is important as it shows the recognition programs capability of distinguishing which gate is closer, in turn recognizing the correct order of gates. While initial multi gate testing will be conducted with the gates in a straight line (Fig. 5), the gates will eventually be configured in staggered patters (Fig. 6).

? Top view side view?

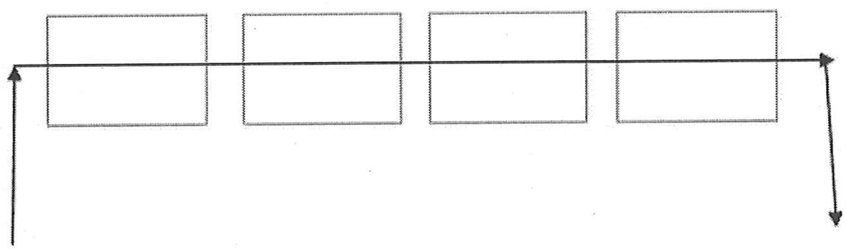


Figure 5: Depiction of Multi-gate course, straight-line configuration

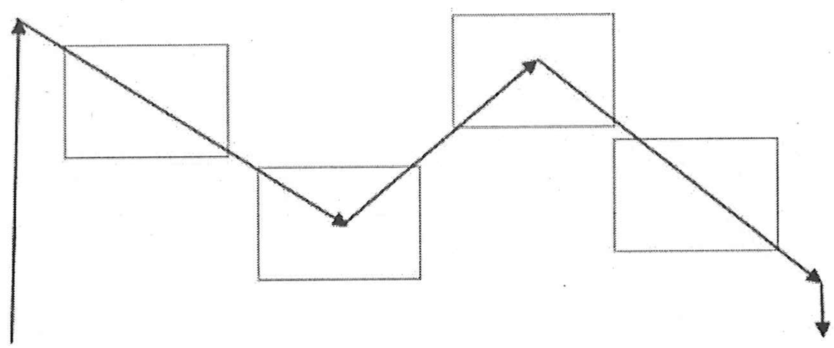


Figure 6: Depiction of Multi-gate course, staggered configuration

In the event that autonomous drone flight through muti-gate courses is successful before the end of the spring semester, further optimization is possible ~~through retinal tracking~~. The proposed plan for gate recognition includes analyzing an entire video feed, although an experienced drone racer would not. Experienced pilots look to specific sectors in the field of view to focus on specific ques for flight. If time permits, a retinal scanner will be used to track the eye movements of drone pilots flying racecourses. By analyzing pilot's vision, the visual recognition processes could be optimized to perform with less processing power, thereby increasing processing speed. If time does not permit this stretch goal, it could be a Capstone for a future midshipman. Source?

cues.

C. Property Measurement

The primary measurement for the simulation and experimental testing will be the deviation from the modified path and the target point in the plane of the gate. This will be measured using data collected from the simulation software and from the Optitrack system during the experimental testing. The competing measurement will be the difference in

previous edits ignored

the trajectories between the original path and the modified path at the target point. This measurement will be a three-dimensional vector, so each individual vector part will be compared.

As this is only the first step in creating a fully autonomous racing drone, this project's objective is not to produce a full speed autonomous racing drone. Previous research in the gate recognition field showed that at full speed (5 ms^{-1}), correct gate recognition with multiple gates in the field of view had a 75%-85% success rate. This project will be flying the drone at a much slower speed (0.5 ms^{-1}), and initial experimentation will use a lit gate in a dark room. This is to ensure gate recognition is not hampered by the environment so that it can be assumed the gate will be recognized. With this assumption, gate recognition will not be measured as it has been with previous research. Further measurements can be done with future capstone groups testing this process at higher speeds.

D. Technical risks and mitigation

The main technical risk involved with this research is processing speed. In the event that the gate recognition and path modification processing is too complex and lags, the drone will not be able to complete its course. In order to mitigate this risk, the approach to building the program will be to build from simplicity, rather that simplifying something complex. ~~This~~ will allow for the simplest and fastest processing, mitigating the risk of lag. ?

Due to the innate troubles with high-speed drone flight, there is always the risk of crashing a drone. In our experiments, ~~we plan~~ to minimize this risk by keeping speeds significantly lower than racing speed (0.5 ms^{-1} vs 5 ms^{-1}). In addition, drone flight will be at 1 foot of altitude, so in the event a hard shutoff is required, the drone will only fall a foot onto a carpeted floor. This should allow for a safety shutoff without damaging the drone. W.C.

The only other issue that may arise is low battery life, which will be mitigated by purchasing multiple drones and batteries, as accounted for in the budget section.

E. Time risks and mitigation

The timeline for this project can be found in encl. 1. The main portions of the projects include relearning python, writing code, integrating inputs, simulation testing, single gate experimental testing, and multiple gate experimental testing. The biggest time risk is in the coding and input integration. While not ideal, experimentation and simulation can be done simultaneously; however, both require working code. If the control algorithm takes longer to create than planned, it would push back simulations and experimentation. This is mitigated by allow more than ample time to run physical experiments, so if both simulation and experimental testing are pushed back, experimental testing can take a time hit without suffering.

F. Justification of special high risk activities

This research will be conducted using Python, as it allows for high level control of the gate recognition and path correction processes. While learning a language is risky, it is advantageous for this project because most other drone flight projects on the yard utilize Python, so cross-referencing will be much easier if I use the same language. It can also be assumed that Python is likely the best language to use for this type of project since other projects are using it as well. In addition, I have completed SY202, the introductory course for Cyber majors at USNA. This course taught the basics of the Python language, so I feel confident in my python coding abilities, and ~~will~~ likely only need a refresher in the language. The first 3 weeks of the timeline have been dedicated to refreshing on python, which should be sufficient to complete this project.

G. Budget

In order to conduct the physical experiment demonstrations, a set of drones and race-gates are necessary. The drone required a single camera, an IMU, and a flight controller. The DJI Tello drone was selected as it has all of the required parts and is easily programmable. The DJI Tello is already being used on the Yard for Capstone projects, so there are

refer to Conift 2019

readily available resources to help in the event that the drone breaks or malfunctions. In addition, the Vortex 180 drone was selected because it is a standard sized racing drone, it fits all of the requirements for this research, and it is already in use in USNA SWAT-C and EW281. This drone will allow the research to get as close to true drone racing conditions as possible. Finally, the TBS gates were selected as they are sized for indoor drone racing and come with programmable lights. The lights allow for easy gate recognition during the first stages of experimental work as the lights in the testing room can be turned off and the gate lights turned on. This removed the background environment from the drone's field of view, making gate recognition easier. Finally, an operating budget of $500 was accounted for in event that more drones/ drone parts are required, or for retinal tracking technology if time permits the stretch demonstrations. This budget also accounts for expenses associated with attending a drone conference or professional drone racing competition. As costs for such events are unpredictable, it is assumed $500 will likely cover these costs. The total labor, overhead, and material costs can be seen in Table IA, and the breakdown of the out of pocket costs for materials can be seen in Table IB.

TABLE I

(A) LABOR, OVERHEAD, AND MATERIALS COSTS.

LABOR	Category	Hours	hourly rate	Cost
	Midshipman	336	$25	$8,400
	Faculty	64	$60	$3,840
	Staff	45	$40	$1,800
Labor Sub-total				$14,040
OVERHEAD	Category	Base Amount	Rate	Cost
	Fringe Benefits	$14,040	35%	$4,914
	Facilities	$14,040	50%	$7,020
	General Services	$14,040	15%	$2,106
Overhead Sub-total				$14,040
MATERIALS	Category			Cost
	In-stock Items			$0
	New Items			$1,585.96
Materials Sub-total				$1,585.96
TOTAL COST				$29,666
OUT-OF-POCKET COST				$1,586

(B) OUT OF POCKET COSTS.

Part Number	Estimated Unit Cost	Quantity	Cost
Tello	$99.00	4	$396.00
Vortex 180	$269.99	2	$539.98
LED Micro Racing Gates	$74.99	2	$149.98
Operating Budget			$500.00

Huh? Less than?

V. CONCLUSION

In conclusion, this research will create a process for a drone to modify its flight path based on the visual recognition of racing gates. This is the first steps in the process to create a fully autonomous racing drone. Future steps will include path optimization and processing optimization. The process for this research will identify the gate, locate it in relations to the drone, and modify the flight path to fly through the gate. The processor will have a visual feed, the three dimensional acceleration data of the drone's flight, and the pre-determined three-dimensional path. The process will output a new three-dimensional path for the drone to follow in order to fly through the gate. This project will demonstrate this process by using the Liftoff Flight Simulator and physical experiments to refine and optimize the program and measure results. Success will be measured by the accuracy of the new three-dimensional path as it passes through the gate. Accuracy is based on the difference in distance of the path and the center of the gate. The biggest risk involved with this research is damaging the drones during the experimental demonstrations, but this risk will be mitigated by extensive use of the simulator first.

Acknowledgements

We thank Prof Esposito, MIDN 2/C Ethan Marcello, and the EW502 class for helpful comments which were incorporated into this proposal. We thank MIDN 1/C Canlas and MIDN 1/C Cunniff for their advice and suggestions.

References

cite them

[1] Red Bull UAE, "Drone racing: The Sport of the Future," Red Bull, 18-Oct-2018. [Online].
[2] J. Condliffe, "Is Drone Racing A Sport Yet?," *MIT Technology Review*, 19-Sep-2016. [Online].
[3] A. Marshall, "ESPN Has Decided Drone Racing Is a Sport Because Internet," Wired, 03-Jun-2017. [Online].
[4] "How do Self-Driving Cars Work?," IoT For All, 05-Oct-2018. [Online].
[5] S. Rayej, "How Do Self-driving Cars Work?," Robohub, 03-Jun-2014. [Online].
[6] J. Svacha, K. Mohta, and V. Kumar, "Improving quadrotor trajectory tracking by compensating for aerodynamic effects," *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 860–866, 2017.
[7] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and imu," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 404–411, 2017.
[8] P. R. Florence, J. Carter, J. Ware, and R. Tedrake, "NanoMap: Fast, Uncertainty-Aware Proximity Queries with Lazy Search over Local 3D Data," *Preprint!*, 2018.
[9] A. A. Zhilenkov and I. R. Epifantsev, "The use of convolution artificial neural networks for drones autonomous trajectory planning," *Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2018*, pp. 1044–1047, 2018.
[10] S. Jung, S. Hwang, H. Shin, and D. H. Shim, "Perception, guidance, and navigation for indoor autonomous drone racing using deep learning," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2539–2544, 2018.
[11] S. Jung, S. Cho, D. Lee, H. Lee, and D. H. Shim, "A direct visual servoing-based framework for the 2016 IROS Autonomous Drone Racing Challenge," *Journal of Field Robotics*, vol. 35, no. 1, pp. 146–166, 2018.

Appendix

Enclosure 1: Timeline

Timeline-Fall AY2020

Week (M)		**Note**	**Planned Activities**	**Actual Hrs**
1	Aug 19	Class begins on M	Relearn Python (learn and practice)	4
2	A 26		Relearn Python (learn and practice)	4
3	Sep 2	M= Labor Day T ← M	Relearn Python (learn and practice)	4
4	S 9		Create Pseudo-code for flight controller	6
5	S 16		Integrate visual input into code	3

6	S 23	Ac Reserve	My 6 week deliverables is: Pseudo-code for flight controller Create code to detect gates	4
7	S 30	6 wk grades due Tues	Integrate pre-determined path into code	5
8	Oct 7		Integrate acceleration data reading into code	5
9	O 14	M = Columbus	Create code to calculate location of gate	4
10	O 21		Finish creating code to calculate location of gate	4
11	O 28	Ac Reserve	My 12 week deliverable is: input integrated code Begin control algorithm creation	4
12	Nov 4	12 wk Grades due Tues	Continue control algorithm creation	4
13	N 11	M = Vets Day	Finish control algorithm creation	4
14	N 18		Test gate recognition in simulation	5
15	N 25	Tgiving Thurs-Fri	Test gate location in simulation	5
16	Dec 2	R= last day class	Test path modification in simulation	5

Timeline-Spring AY2020

Week (M)		Note	Planned Activities	Actual Hrs
1	Jan 6	Begins Tues (M sched)	Modify code based on simulation testing	4
2	J 13		Modify code based on simulation testing	4
3	J 20	M= MLK	Set up experimental testing	3
4	J 27		Begin experimental testing (single gate)	4
5	Feb 3		Continue experimental testing (single gate)	4
6	F 10	Ac Reserve	My 6 week deliverable is: experimental testing data Analyze experimental data	3
7	F 17	M = Pres Day 6 wk grds W	Modify code based on experimental testing	4
8	F 24		Modify code based on experimental testing	4
9	M 2	M = Columbus	Set up experimental testing for multiple gates	3

SB	M 9			
10	M 16		Begin experimental testing (multiple gates)	4
11	M 23		Continue experimental testing (multiple gates)	4
12	M 30	Ac Reserve	My 12 week deliverable is: multiple gate experimental data Analyze experimental data	3
13	Apr 6	12 wk Grades T	Modify code based on multiple gate experimental data	5
14	A 13		Submit poster to MSC for printing, work on Capstone presentation	5
15	A 20		Capstone day	5
16	A 27	T= last day of class	Schedule technology transfer with adviser	3