*pacer drone*

Evan Arthur   
*Colby College*  
Waterville, Maine, USA

edarth21@colby.edu   
Eli Decker  
*Colby College*  
Waterville,Maine, USA

edecke20@colby.edu

Tate Ewing  
*Colby College*

Waterville,Maine, USAtewing21@colby.edu

***Abstract*—This electronic document is a “live” template and already defines the components of your paper [title, text, heads, etc.] in its style sheet. *\*CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract*. (***Abstract***)**

***Keywords—component, formatting, style, styling, insert (****key words****)***

* Introduction (*Heading 1*)

In Spring 2017, Nike-sponsored athletes attempted to break two hours in the marathon through the pursuit of marginal gains that were hoped to push the runners under the elusive 2-hour barrier. An important innovation made was the implementation of a pace car that emits laser lines which allows the runner to maintain perfect pace. No longer would the runners have to constantly check their watch to make sure they are at the correct pace, breaking up their stride and slowing them down in the process. Instead, the mental challenge of maintaining pace is eliminated, allowing the runner to simply follow a line on the ground. Our goal was to replicate this experience on a running track. Because it is impossible to use a pacer car on a track, we decided the most effective option was to create a drone capable of flying around a track at a constant speed while carrying a laser emitting a pace line on the track behind it. We chose to use the DJI Tello drone because of its affordable price and easy to use SDK for sending commands and receiving the video stream.

* Related Work

Because a track lane is defined by two painted lines, our work was inspired by the much more common task of car lane detection. A car lane detection system attempts to identify the two road lanes on either side of it in order to stay in between them. This problem is well documented and has been implemented in commercial vehicles available to the general public. Because no one to our knowledge has attempted to make a drone fly around a track, we were not able to find any directly related work. We looked specifically at Kemal’s curved lane detection system on Hackster.io designed for cars. He used a sliding window algorithm to detect the lane and subsequently give a curve radius that can be used to alter the car's path. Though designed for a different purpose, the problems are similar enough that his method was transferable from a car on a road to a drone on a track.

* *Maintaining the Integrity of the Specifications*

Identify applicable funding agency here. If none, delete this text box.

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

* Method

Our project can be divided into three main problems:

* Connecting to and sending directions to the drone.
* Taking the drones video stream and computing where the drone should go.
* Maintaining a constant pace throughout.

Because we chose to use a DJI Tello, connecting and sending directions to the drone was handled by connecting our computer to the Wi-Fi signal emitted by the drone. Combined with the SDK provided by DJI, we were able, in theory, to send commands to the drone and receive the drones video stream in almost real time.

The DJI Tello drone has a built in Wi-Fi connection, and the SDK allows the user to control the drone over Wi-Fi and receive the video stream from the drone’s front-facing camera. In theory, we should easily be able to access the video stream, process the video to determine controls for the drone, and send the instructions to the drone.

Our first step was to configure a lane detection algorithm to determine the radius of the track’s curve so we could control the drone. After receiving the video stream, we put each frame through a pipeline in order to identify the track lines and the accompanying curve radius in order to send the correct flight commands to the drone. Because our computer will potentially not be able to handle each frame and maintain real-time, we implemented a threaded camera that allows us to always be looking at the most recent frame:

* First, we resize and recolor the video.
* Then we un-distort the image based off of a collection of photos taken by the drone of a chess board so that we know how the camera distorts a known object.
* We apply a perspective warp so that what is directly in front of us is mapped so that it appears as if we are looking down on the road without perspective
* We apply a sobel filter that turns all pixels that are below our threshold black and all ones above the threshold white, leaving us with an image that is all black instead of white lines. Obtaining these thresholds was a process of trying a variety of values and keeping the most effective ones that captured the whole line, but no extra noise.
* We implement a sliding window algorithm to identify the lines and fit a curve to mimic the line. (describe the sliding window)
* We calculate the curve radius based off of the curve found by the sliding window algorithm
* We translate the curve radius into a yaw value to fly the drone at and send it to the drone

WRITE ABOUT ALGORITHM SPEED AND ACCURACY IMPROVEMENT HERE

As driverless cars have become a hot area for research, we were able to leverage simple lane-finding algorithmic techniques used in self-driving cars. Luckily, a track is even more of a closed system than a road so there are some built-in assumptions we could make when analyzing our video stream.

By flying at a similar height, the whole time, the relative size of the lane lines will remain fairly constant, which means we can alter the perspective of the camera to make an overhead view centered on the inside lane lines. Also, tracks always have painted lane lines for humans to easily see, which means we could do sobel filtering on the video stream to help separate the lines from the track surface and turn it into black and white (binary) for easy processing.

After filtering, we used a sliding window algorithm. Each window takes the average of where the white pixels are to come up with an approximation for where the line in the window must be (after sobel filtering the line is in white, the track surface is in black). The size of the windows was fine-tuned to maximize detection of lanes when curving while balancing the tradeoff of minimizing interference such as number markings on the track. We determined the size of the windows by examining the minimum number of windows required to detect the lane pixels without getting distracted by noise. The stack of sliding windows then use their combined averages to approximate the radius of the curve based on the centering of white pixels in each window.

Once we had the algorithm dialed in, it was time to address how to control the drone with the curve radius. Although the SDK advertises various methods and features that would perfectly suit the task, it was not as simple as anticipated. The communication between the drone and computer presented some frustrating challenges.

The DJI Tello accepts a variety of commands, which it receives over a UDP connection in JSON format. Writing code for the UDP connection and parameterized functions to easily test different values for controlling the drone would be a tedious process. Fortunately, we were able to leverage the djitellopy library (<https://github.com/damiafuentes/DJITelloPy>), which establishes the socket connection and formats the drone commands with convenient, object-oriented design.

For the most basic commands, the djitellopy worked without hiccup. We were able to command the drone to move forward, backward, side-to-side, and yaw. When attempting to use some of the more advanced methods, however, we encountered some issues. For example, listed in the Tello SDK Version 1.3.0 is the go\_x\_y\_z\_speed method, which commands the drone to fly in a curve, passing through the points (x1, y1, z1) and (x2, y2, z2) and a given speed in cm/s. This method would be especially useful for our task, because it is easy to calculate two points on a curve given the radius. Even after much experimentation the Tello drone responded with an error message every time.

Instead of using go\_x\_y\_z\_speed, we turned to send\_rc\_control, which allows us to control forward and backward speed, side-to-side speed, and yaw rate, allegedly in cm/s. We found that the actual speed of the drone was not consistent with the indicated cm/s unit. This inconsistency made it extremely difficult to control the drone based on a curve radius given in meters. After extensive trial and error, we determined a range of values for forward movement and yaw rate that fly the drone in an appropriate curve.

Given the periodic inconsistency of the algorithm’s detected curve radius and the speed/yaw units, the best solution is to make the drone fly in a curve when the radius is between 250 - 350 meters, and otherwise fly straight. When the algorithm accurately detects a curve, the radius is somewhere in this range. Most tracks have a similar curve radius, so it should work for most applications.

* Results

We were not able to measure how accurate the drone follows the lane on an actual track, per the technical limitations discussed in the methods, but we were able to gauge the accuracy of our lane detection algorithm--to some degree. Our algorithm calculates the curve radius of the lines detected, so to measure how accurate these calculations are we compared our results with the standard track curve radii. See below the graph of the algorithm's output.



By looking at the graph, you can tell that it is not very consistent. Comparing it with the graph of the standard track's radii exposes the algorithm’s inaccuracy.



The detected curve radius bares little resemblance in shape to the standard curve radius, and the scale of the measurements is completely off. This would suggest that the algorithm is completely unusable. When watching a live processing of the lane detection, it appears to be quite accurate.





In both the straightaways and the curved sections of the track, the curves appear to be tracked accurately. This leads us to believe that there is problem with the conversion from calculated pixels to meters.

* Conclusions and Future Work

While some of the results were inconclusive, the research showed that this is a promising area of further exploration. In many places in the world, network signals are unreliable or inaccurate and more stable computer vision techniques for autonomous drone movement provide a possible solution. The curve-mapping for drone use and computer vision techniques in this paper could be expanded upon to come up with more robust systems by utilizing neural networks for recognizing lane lines in varied lighting and weather conditions. Additionally, recognizing and utilizing other markers on the track such as start lines could be used for pace correction without requiring a GPS connection.

Reference

• List the papers cited in your report. Follow the IEEE format at the end of the

* *Authors and Affiliations*

**The template is designed for, but not limited to, six authors.** A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

* *For papers with more than six authors:* Add author names horizontally, moving to a third row if needed for more than 8 authors.
* *For papers with less than six authors:* To change the default, adjust the template as follows.
* *Selection:* Highlight all author and affiliation lines.
* *Change number of columns:* Select the Columns icon from the MS Word Standard toolbar and then select the correct number of columns from the selection palette.
* *Deletion:* Delete the author and affiliation lines for the extra authors.
* *Identify the Headings*

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

* *Figures and Tables*
* *Positioning Figures and Tables:* Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

|  |  |  |  |
| --- | --- | --- | --- |
| **Table Head** | **Table Column Head** | | |
| ***Table column subhead*** | ***Subhead*** | ***Subhead*** |
| copy | More table copya |  |  |

* Sample of a Table footnote. (*Table footnote*)
* Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*
2. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.
8. **IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**