Autonomously Track and Land with a UAV

Jonathan Edwards

Computer Engineering

University of Canterbury

Christchurch, New Zealand

jme161@uclive.ac.nz

Richard Green

Computer Science Engineering

University of Canterbury

Christchurch, New Zealand richard.green@canterbury.ac.nz

Abstract - This paper proposes a solution to the problem of having a UAV automatically land on a tracked object. The method proposed by this paper is to use a combination of fiducial marker tracking, depth detection and natural feature tracking to determine the location of landing and guide the UAV to a safe landing. …

Keywords - component, formatting, style, styling, insert

mirror clip was added so that the drone could look vertically down

I. INTRODUCTION

Over the years, research in the Unmanned aerial vehicles (UAVs) industry has increased considerably. More importantly, delivery companies and hospitals have taken a particular interest in UAVs. As UAVs helps speed up the delivery process and removers the human factor from the equation. Having a user fly a drone requires a lot of skill and training for the drone to fly successfully without crashing. Many UAV research papers state the importance of equipping drones with artificial intelligence to help humanity with our daily lives. In particular it can be difficult for a user to identify the landing location, to continuously track an object, and to land safely every time. There are already several different types of drones that are equipped with artificial intelligence on the market today, but these can be quite expensive.

There are many research papers that have been published and describe the different ways an object can be tracked by using a drone. Techniques published range from colour detection to corner detection and to template matching.

This paper proposes an alternative method, the method consists of programing a low-cost drone with Artificial intelligence that will allow it to land, track, and detect objects autonomously. The proposed tracking and detecting algorithm detects for ArUco (fiducial) markers. This allows the drone to be able to detect and track fiducial marker(s) that are within the drone’s field of vision (FOV). If a marker is detected, the algorithm approximates the x, y, and z axis distances as well as the vertical and horizontal angles between the drone and marker reference frame . These angles are fed into a closed loop PID controller which allows the drone to manoeuvre autonomously in the x, y, and z axis while keeping the marker centred in the camera’s frame. Once the drone is fully aligned, the proposed method will cause the drone to descend and land once it becomes less than 20 cm above the marker.

II. BACKGROUND

1. *Colour detection*

Three papers were seen using the HSV method for being able to track and detect objects [1,2,3]. It entails going from the Red Green Blue (RGB) colour space to the Hue Saturation Value (HSV) colour space. RGB is defined by listing how much red, green and blue is contained in a single value [4]. Each value ranges between 0 – 255. This is an additive method where the more of each colour is added, the brighter it becomes. When combining the amount of red, green, and blue, it is extremely difficult in dictating how much of each colour to use as seen in Figure 1. HSV on the other hand is comprised of 3 parameters, hue, saturation, and value [4]. The hue value ranges from 0 – 180 and the saturation and value can range between 0 – 255. controls the These values determine the output colour that is seen as shown in Figure 1. The HSV method uses thresholding to and is much better at handling lighting differences.

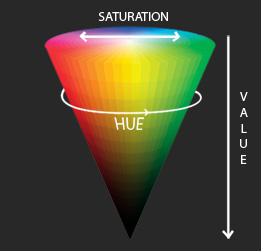
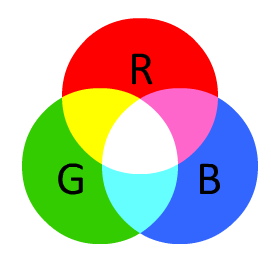


Figure 1 – RGB to HSV colour space’s [4]

1. *Corner detection*

Another approach is to use the Harris Corner algorithm to extract corners and to infer features of an image [5]. The image is first converted to grayscale before running through the algorithm. The detected corners are then filtered, and the strongest corners are then drawn onto the original colour image for display. Corners are valuable as it is typically much easier to match the same corner from two slightly different frames than it is to match an edge or patch of colour as can be seen in Figure 2.

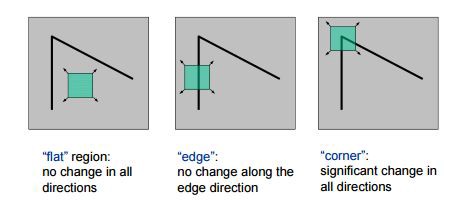


Figure 2 – Corner detection method [5]

Another paper proposes autonomously landing a UAV on a Helipad [6]. This paper used a corner detection algorithm (similar to the one above) to estimate the attitude between the Helipad and the camera, Figure 4 shows their final result. To achieve their goal, they used both corner detection and corner matching algorithms. Corner matching is similar to Corner detection but instead looking for all possible edges in a frame, it looks for similarities between corners and their quadrants as can be seen in Figure 3. Once computed their method is able to track and approximate the orientation of the H symbol by using Pose estimation [6].


                        figure
                    

Figure 3 – Corner matching [6]


                        figure
                    

Figure 4 – Final result with using Corner detection + Corner matching [6]

1. *Template matching*

Another approach is to use Template matching for tracking and detecting objects. Multiple papers use the ArUco (fiducial) marker algorithm to approximate the distance between the drone and the camera [7,8,9]. The fiducial marker algorithm detects for a black and white image within the cameras FOV and determines the unique ID that corresponds to the marker. Fiducial markers are objects that are used to provide a point of reference. These papers also all use the ArUco library, this library contains dictionaries that store different bit size markers [7,8,9]. Figure 5 illustrates an example of a fiducial marker that has a unique ID of 23.

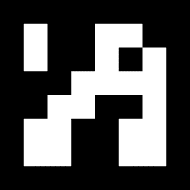


Figure 5 - 6x6 fiducial marker with ID 23

Template matching can either be kept in the RGB or HSV colour space or be converted into the greyscale colour space. Efficient template matching requires the image to be converted from the RGB colour space to the grey-scale colour space. The fiducial marker algorithm allows the greyscale image to be identified. Pose estimation is also used for estimating the marker’s position and attitude relative to the camera.

1. *Limitations of prior research*

Limitations should be given in each sub-section

A paper discussed an approach for detecting for a Helipad object [6]. Since this approach used was scanning predetermined edges, the results were more accurate than the Colour detection method [1]. However accuracy was variable as the experiment never got tested on a UAV and on a linear stage instead.

\* A problem with template matching and trying to land was that the markers were to big and often got lost [7].

- Fix = Add multiple fiducial markers

However, the algorithm did not attempt to find the object once it had been lost. The speed at which the object could move impacted the chance of it being lost and therefore the speed must be constrained to the drone’s performance capabilities [1].

Another consideration was that the drone couldn’t see obstacles outside of its view and could easily crash if it were to hit something [3].

Another method is to use complex algorithms such as Speeded Up Robust Features (SURF) [15] and Maximally Stable Extremal Regions (MSER). These algorithms produced improved results, but the method was very complex compared to template matching.

Another paper used a more accurate calculation to maintain a certain distance from an object [3].

the HSV method can also be sensitive to noise or disturbances in the environment [2]

The biggest limitation that paper [1] faced was that by using a HSV Colour detection method instead of using a greyscale method was that their accuracy decreased. The papers colour detection algorithm(s) also needed a good constant light source. In order to go Pose estimation could not be used HSV and the RGB method had to be used instead. Thus, finding the orientation of an object is harder when not converting to the greyscale colour space.

The template matching approach as proposed in papers [7,8,9], was by far the simplest and more accurate method. As the biggest limitation that these papers faced were calibration setup, tracking control and time delay between frames.

Fiducial markers can be vulnerable to motion blur, especially when used on a drone. Various types of fiducial markers exist, with circular markers being the most resistant to motion blur [8].

Detecting and tracking an object is highly determined by the image processing method that is used. This paper uses the template matching approach [7,8,9] rather than the HSV colour space approach [1,2,3] and Corner detection & matching approach. Overall, the primary goal of this paper was to also prioritise reliability and accuracy over flexibility and performance.

III. METHOD

1. *Drone and Hardware*

The DJI Tello Drone has a small horizontal built-in FPV camera that can transmit either video or photos back to the user. It is also low cost, compact and robust quadcopter that was designed for the STEM community as well as hobbyists. The drone includes a first-person view (FPV) camera, an ultrasonic sensor, a 3-axis gyroscope & accelerometer, an inertial measurement unit (IMU), and Wi-Fi connectivity. The drone’s camera can transmit either 720p video at 30fps or 5MP photos, has a viewing angle of 82.6 degrees. The drone can be controlled by using either a smartphone or a computer over Wi-Fi, has a flight time of up to 13 minutes or a max flight distance of 100m at max flight speed 8m/s [10]. An  Intel processor powers the drone which allows it to capture high-quality footage and be enclosed in a small compact form. An image of the drone can be seen in Figure 6. A mirror clip was also added so that the drone could capture images in the downward direction. An image of the drone with the mirror clip attached can be seen in Figure 7.



Figure 6 – DJI Tello drone

A picture containing bicycle

Description automatically generated

Figure 7 – DJI Tello drone with mirror clip

The drone will be programmed and communicating with a laptop that runs the Windows operating system. The laptop is a 64-BIT machine and is controlled by the Intel Core i7-9750H, it has 16 GB of RAM and is clocked at 2.60 GHz.

1. *Software*

The drone will be programmed in Python 3 (Version 3.8.10) and will be programmed in the Visual Studio Code IDE. Plotting was done in the Matlab IDE and (Version R2019b – academic use) is used. The DJI Tello Drone DJITelloPy interface (Version 2.1) is used, this controls the communication over a Wi-Fi link connection to and from the laptop. This interface allows the drone to stream real time video back to the laptop. The interface also controls how data is sent to and from the sensors that are on the drone. The interface also comes equipped with a set of commands that can be fed through the drone to make it move in a specific direction [11].

Open CV is an open-source computer vision and machine learning software library for python [12]. The open-source library contains more than 2500 algorithms that are all extensively well documented and come with sample code. This software library is useful for identifying objects, tracking objects, and for classifying objects. OpenCV (Version 4.5.1) will be used as well as the ArUco (fiducial) marker library.

1. *Image Processing*

This tracking algorithm is constantly updated when a marker is detected. Prior discussed tracking an object by using a color and edge detection method [1,6]. The problem stated in section II was that both these methods are complex and difficult to get high accuracy. This paper over comes these limitations by focussing on functionality and accuracy by using an easy to implement algorithm.

The proposed image processing algorithm in this paper consists of image conversion, marker detection, data processing and a PID controller. The proposed algorithm as shown in Figure 9, relies on being able to identify a unique ID [16], and the algorithm must be fast enough in order to avoid bottlenecking. The algorithm that is used in this paper has the same structure as the one a previous student used for his COSV-428 project at the University of Canterbury [16]. De Gouw’s paper in 2019 proposed a limitation in the video feed between the drone and his laptop. The paper reported that the drone was capable of outputting 30fps at a resolution of 640x360. Instead, the algorithm was only getting between 1.1 – 1.5 frames per second and meant that some aspects of accuracy were lost [16]. This paper managed to fix the frame rate to be at 30fps by upgrading the hardware of the drone, and also re-implementing & fixing some bugs in the software’s algorithm in Figure 9. This improvement helped improve the accuracy of the experiment.

Camera calibration is important and necessary as it prevents an image from being distorted (normally only has to be done once). Radial distortion caused by imperfect lens and thus a possible blurry image is produced, also as seen in Figure 8. If the camera is not correctly calibrated, it could affect the algorithm to not perform correctly. In this paper, the camera on the drone is calibrated with a checkerboard test similar to Jiang’s paper [7].

A picture containing text, shoji

Description automatically generated

Figure 8 - Image distortion

The proposed algorithm in Figure 9 of this paper uses the grey-scaling with template matching method (ArUco (fiducial) markers [13]. The algorithm is easier to implement, and there are less errors in the accuracy [7,8,9,16]. Detecting for objects using the HSV colour space method (section *II.A*) and the Corner detection (section *II.B*) are both possible [1,2,3,6]. A limitation found in papers [1,2,3,6] was that it required a lot more trial and error to get something working accurately. Another was that they it was more complex for them to find the pose estimation geometries. Thus, why this paper chose the simpler more accurate approach using ArUco markers.

Diagram

Description automatically generated

Figure 9 – Flowchart of the implemented Algorithm [16].

1. *Marker detection*

The ArUco marker is a synthetic square marker composed by a wide black border and an inner binary matrix determines its unique identifier (ID). The black border facilitates its fast detection in the image and the binary codification allows its identification and the application of error detection and correction techniques. There are a range of different binary sized marker’s that can be chosen from, as seen in Figure 5. This paper uses a 6x6-bit marker with ID 23 as well as a Helipad looking marker that contains fiducials within fiducials [9], this can be seen in Figure 10.

A limitation that a paper faced was that fiducial marker got lost when descending [7]. This paper overcame this limitation by adding multiple fiducial markers as did another paper [9], as seen in Figure 10.

Since the DJI Tello drone comes equipped with a horizontal FPV fixed camera, a mirror clip was designed to be attached so that the drone could look vertically down [18], as can be seen in Figure 7. Because of this the fiducial marker was mirrored as can be seen in Figure 10.

*A picture containing text, clock

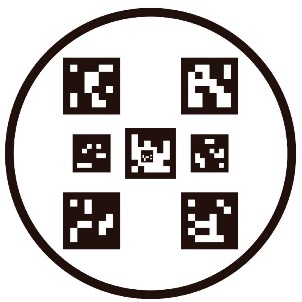
Description automatically generated* 

Figure 10 – Helipad setup, original vs mirrored

1. *Tracking strategy*

This paper uses two vector methods to be able to centre the drone with the marker. The Translational vector method was used for finding the centre point angle between the drone’s camera and marker, as seen in Figure 11. The alpha value calculates the angle between the centre of the drone and marker [16]. This is calculated using Equation 1. A Closed Loop (CL) PID controller as seen in Figure 12 is used to recenter the drone’s X and Y axis in respect to the marker’s position inside the camera frame [7].

Diagram

Description automatically generated

Figure - Centre Point angle calculation [16]

|  |  |
| --- | --- |
|  | [1] |

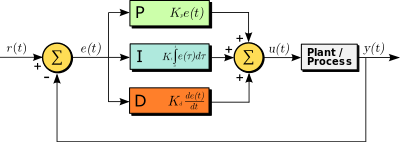


Figure - Closed Loop PID controller

The other method used for finding the yaw centre angle was computed by using the Rodrigues vector method, as seen in Figure 13. The method is first converted to the Euler matrix form by using RQ decomposition. This then provided the Euler angles about the X, Y and Z axis for the marker and only the Z axis rotation matrix was used in this paper as shown by Equation 2 [7]. A CL PID controller was also implemented to allow for smooth angle adjustments between the drone’s Z axis and the marker.

A picture containing text, sky

Description automatically generated

Figure – Rz Yaw angle calculated

|  |  |
| --- | --- |
| Understanding the math behind rotating around an arbitrary axis in WebGL -  Stack Overflow | [2] |

IV. RESULTS

A picture containing text

Description automatically generated

A picture containing text

Description automatically generated

A screenshot of a computer screen

Description automatically generated with medium confidence

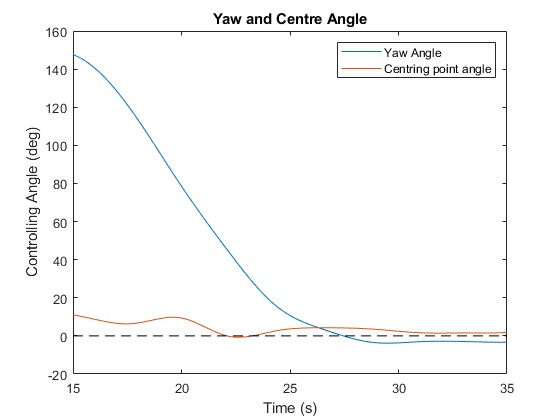
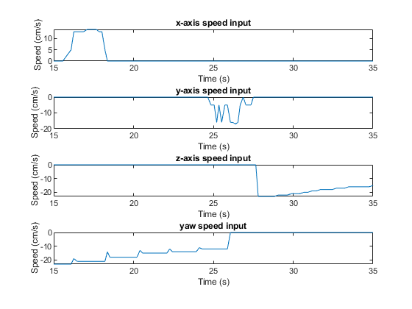
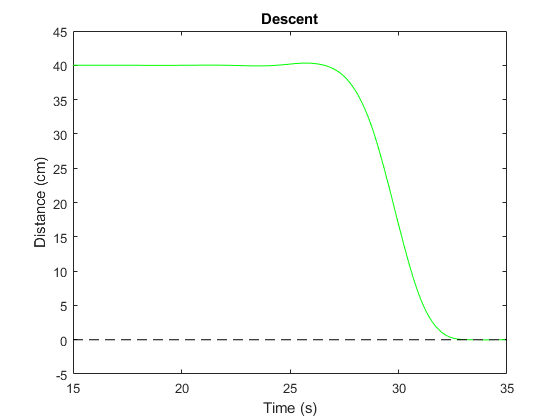
A picture containing diagram

Description automatically generated

Limitation

A picture containing text

Description automatically generated



A PID controller was used to smoothly control the movement of the drone. The controller used two separate sets of gains for moving the drone to the desired distance and for positioning the marker in the centre of its view. The integral and derivate gains for both sets were set to a value of 1, as they were found to have very little effect due to the slow loop speeds. The proportional gain for the distance movement was set to 5. The proportional gain for the centring movement was set to 15.

* A very short introductory context that repeats the research question and helps to understand your results.
* Report on data collection, recruitment, and/or participants. For example, in the case of clinical research, it is common to include a first table summarizing the demographic, clinical, and other relevant characteristics of the study participants.
* A systematic description of the main findings in a logical order (generally following the order of the Methods section), highlighting the most relevant results.
* Other important secondary findings, such as secondary outcomes or subgroup analyses (remember that you do not need to mention any single result).
* Visual elements, such as, figures, charts, maps, tables, etc. that summarize and illustrate the findings. These elements should be cited in the text and numbered in order. Figures and tables should be able to stand on its own without the text, which means that the legend should include enough information to understand the non-textual element.

[How to Write the Results Section of a Research Paper (kolabtree.com)](https://www.kolabtree.com/blog/how-to-write-the-results-section-of-a-research-paper/)

V. CONCLUSION

how your research improved on (or compares with) prior research

* Summing up results (one/few sentence)
* Compare paper the 2 students papers
* Compare results with other people’s research & results.
* More focused about accuracy rather than performance
  + Do 3 experiments to make it a fair test
  + Chose a method which gave more accurate results. Testing the functionality of the prototy[ed system.
  + **Functional prototypes** are designed to imitate the functions of the actual product as closely as possible no matter how different they look from the actual product. These types of prototypes are produced for the products which are dependent on the function rather than the display.

1. *Future Research*

* Describe what would/could be done differently for next time.
* Next stage is to use a Harris corner Detection

The Harris Corner Detector is a function for finding corners in an image by looking for sections of the image that have a large variation in intensity in every direction.

(Future - Helipad) – as this paper was targeted towards having an accurate and functional system rather than focusing on performance.

* Improvement in PID controller [8]

VI. REFERENCES

|  |  |
| --- | --- |
| [1] | It Nun Thiang, Dr.LuMaw, Hla Myo Tun. “Vision-Based Object Tracking Algorithm With AR.Drone”. International Journal of Scientific & Technology, Volume 5, Issue 06, pp. 135-139 June 2016. |
| [2] | Chi-Tinh Dang, Hoang-The Pham, Thanh-Binh Pham and Nguyen-Vu Truong, "Vision based ground object tracking using AR.Drone quadrotor," International Conference on Control, Automation and Information Sciences (ICCAIS),Nha Trang, 2013, pp. 146-151. |
| [3] | Ioannis Dramouskas, Isidoros Perikos, and Ioannis Hatzilygeroudis. “A Method for Performing Efficient Real-Time Object Tracing for Drones”. Advances in Smart Systems Research, Vol. 6, pp. 55-64, 2017. |
| [4] | “HSV vs. RGB”. Internet: <https://handmap.github.io/hsv-vs-rgb/>, 26 Oct 2016, [Accessed 11/04/21] |
| [5] | “Introduction to Harris Corner Detector”. Internet: [Introduction to Harris Corner Detector | by Deepanshu Tyagi | Data Breach | Medium](https://medium.com/data-breach/introduction-to-harris-corner-detector-32a88850b3f6), 16 Mar 2019, [Accessed 16/04/21] |
| [6] | Cosimo Patruno, Massimiliano Nitti, Ettore Stella, and Tiziana D’Orazio. “Helipad detection for accurate UAV pose estimation by means of a visual sensor.”. International Journal of Advanced Robotic Systems (IJARS), Vol. 14, pp. 1-15, Sept. 2017. |
| [7] | Ziwen Jiang. “An Autonomous Landing and Charging System for Drones.” M.Eng. thesis, University of California, Berkeley, 2019. |
| [8] | Vision based control and landing of Micro aerial vehicles - [FULLTEXT01.pdf (diva-portal.org)](http://kau.diva-portal.org/smash/get/diva2:1332929/FULLTEXT01.pdf) |
| [9] | [Sensors | Free Full-Text | An Onboard Vision-Based System for Autonomous Landing of a Low-Cost Quadrotor on a Novel Landing Pad (mdpi.com)](https://www.mdpi.com/1424-8220/19/21/4703?type=check_update&version=2) |
| [10] | ‘Tello”. Internet: <https://www.ryzerobotics.com/tello>, [Accessed 21/03/21] |
| [11] | “Tello SDK 1.3.0 documentation”. Internet: <https://terra-1-g.djicdn.com/2d4dce68897a46b19fc717f3576b7c6a/Tello%20%E7%BC%96%E7%A8%8B%E7%9B%B8%E5%85%B3/For%20Tello/Tello%20SDK%20Documentation%20EN_1.3_1122.pdf> [Accessed 21/03/21] |
| [12] | “About OpenCV”. Internet: <https://opencv.org/about.html>, [Accessed 14/03/21] |
| [13] | “Detection of ArUco Markers”. Internet: <https://docs.opencv.org/master/d5/dae/tutorial_aruco_detection.html>, [Accessed 10/03/21] |
| [14] | M. F. Sani and G. Karimian, "Automatic navigation and landing of an indoor AR. drone quadrotor using ArUco marker and inertial sensors". 2017 International Conference on Computer and Drone Applications (IConDA), Kuching, 2017, pp. 102-107. |
| [15] | [RobustVisual\_2015.pdf (robesafe.es)](http://www.robesafe.es/personal/bergasa/papers/RobustVisual_2015.pdf) |
| [16] | Daniel de Gouw and Richard Green, “Tracking Fiducial Markers with a Drone”, Computer Vision Lab, University of Canterbury, Tech. Rep., May 2019 |
| [17] | James Fairbairn and Richard Green, “ Automatic UAV landing”, Computer Vision Lab, University of Canterbury, Tech. Rep., April 2015 |
| [18] | “Mirror Image: Reflection and Refraction of Light”. Internet: <https://www.livescience.com/48110-reflection-refraction.html#:~:text=A%20reflection%20appears%20to%20be,direction%20from%20which%20it%20hit>. [Accessed 21/03/21] |

Ref 14 & 15 are used for future work