TECHNICAL RAPPORT

Technical Rapport: Use of a Single Drone in Crowd Control

Roel Kuiper[1] | Christopher Buch Madsen[1] | Andrew Nijssen[1] | Abel Oakley[1]

[1]FNWI, Universiteit van Amsterdam, Amsterdam, Science Park 904 1098 XH Amsterdam, Nederland

Correspondence
Andrew Nijssen
Email: andrew.nijssen@student.uva.nl

Funding information

This report examines the use of drones as a potential alternative to the use of helicopters in crowd control. Furthermore the research conducted is focused on practical and cost-effective employment of existing technologies.

1 | INTRODUCTION

Drones are now widespread and offer various possible applications that differ widely such as the predator drones used in the Syrian conflict (thenation.com) to non-lethal applications such as toy drones. Often due to reasons relating to guaranteeing the safety of government security personnel and the way government contractors can maximise their profits: equipment tends to become more sophisticated and more expensive. It is the authors' opinion that since government expenditure is financed by tax payer money, there is a moral imperative to minimise costs relating to government spending wherever possible. Since drones are a potentially inexpensive resource as compared to a helicopter, and can be operated without direct contact of government personnel, it is the authors' opinion that they are a cost-efficient and safe prospect for future use in crowd. To ensure public safety on national holidays such as Kingsday, the Dutch police employs the use of helicopters to monitor the massive crowds in Amsterdam. The overhead view allows the police to disperse personnel efficiently throughout the city. Additionally police officers monitor the live video footage from the helicopter in order to assess any potential dangers. We believe that drones could assist in crowd monitoring in the future. Our research has focused on drone surveillance of a predetermined area whilst making note of the number of police officers in that area. The goal of which is to assess the use of drones to monitor crowds in the future.

Therefore our **research question** is:

* Equally contributing authors.

The cost-effective and efficient employment of drones in crowd control situations.

The following hypotheses were constructed to gather data:

Hypothesis 1: A single toy drone can take off, fly in a square and land autonomously.

Hypothesis 2: A single autonomously flying toy drone can surveil and keep track of the number of people in a predetermined area and store that information.

In order to identify police officers in a crowd several methods can be employed. Since much work research has focused on object recognition, and our research focus lies in practical application of existing methods. The hypothesis shall be researched by employing previously established methods of object (e.g colour) recognition. One way of using video imagery to identify objects is the colour-space, a way of describing colours in an image[3]. The simplest colour space is the RGB space, each pixel represents a combination of Red, Green and Blue values [3]. However the tresholding in the RGB colourspace is that brightness is shared by all three RGB colours whereas in the HSV colourspace (hue, saturation, value) the brightness is only encoded in V [3]. This allows for thresholding just the H and S values and to eliminate illumination problems so that an object can be identified under varying lighting conditions. Tools such as OpenCV, an open source computer vision library, can be configured to work with toy drones such as the Parrot AR Drone with HSV colour recognition.

2 | METHOD

2.1 | Algorithm

```
Autonomous flying routine of a drone in a square

Step 1: if ar-drone is on ground: take off

Step 2: Increase y-axis (strafe right) velocity

Step 3: Stop y-axis velocity

Step 4: Rotate 90 degrees anti-clockwise

Step 5: Go back to step 2
```

2.2 | Software and Hardware

To reduce costs an AR-Drone version 2 Elite, equipped with a front-facing camera and a downward-facing camera at a cost of 299 euro [2] was used in our experiments. In addition the researchers made use of their private laptops to programme and control the aforementioned drone. The OS systems of the computers used were Unix (Mac OS X Sierra), Windows 10 and Ubuntu. We settled on the use of OpenCV which has a strong focus on real time (robotic) applications.

Our code is written in C++ using the integrated development environment (IDE) Microsoft Visual Studio 2015, with the compliment of puku0x's C++ library for AR-drones, aptly named CV Drone [1].

2.3 | Object Recognition

For making the drone recognize and follow a specific color we used a proportional-integral-derivative controller (PID). A PID controller is an algorithm which calculates an error value $e(t)$ as the difference between a and a measured process variable. It then applies a correction based on proportional, integral and derivative terms.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

The PID controller function is translated to C++ language for the velocity in both the x and the y direction.

```
vx = kp * error_x + ki * integral_x + kd * derivative_x;

vy = kp * error_y + ki * integral_y + kd * derivative_y;
```

2.4 | Procedure

The area in which the drone was operated is the central hallway in D building located on ground level at Science Park.In order to simulate crowd control we needed to get the drone to patrol a designated area. For this designated area we used a square, we defined the square in velocities and rotations. During the entirity of the flight the camera of the drone was facing inwards, toward the centre of the square. To get the drone to recognize the amount of people we colour recognition was employed. The drone detected objects of a designated colour, using HSV values, and counted the number of objects. The colour used to recognise different objects was red, as this is an easy colour to differentiate from the environment in general and especially in the environment we were working in. To prevent the drone from detecting too small objects we used a minimum size for to recognise objects. This ensured that the detected objects were people.

3 | RESULTS

Using the Parrot AR-Drone version 2 Elite, we created an autonomous flight routine that has the drone fly in a regular quadrilateral (square) pattern whilst the camera always faced the square . With the CV Drone library we pieced and stringed together code to use the forward facing camera to recognise and keep count of the total number of objects within the square, given a specific colour. Furthermore we were also able to have our drone track, and autonomously follow a given object by employing colour recognition using the downward facing camera.

This following link is a showcase of the aforementioned autonomous flight routine created for the drone:
https://www.youtube.com/watch?v=8J5Nb2dBNwQ

FIGURE 1 The downwards facing camera of the drone, capturing a red ball and recognizing the red ball by it's colour (red).

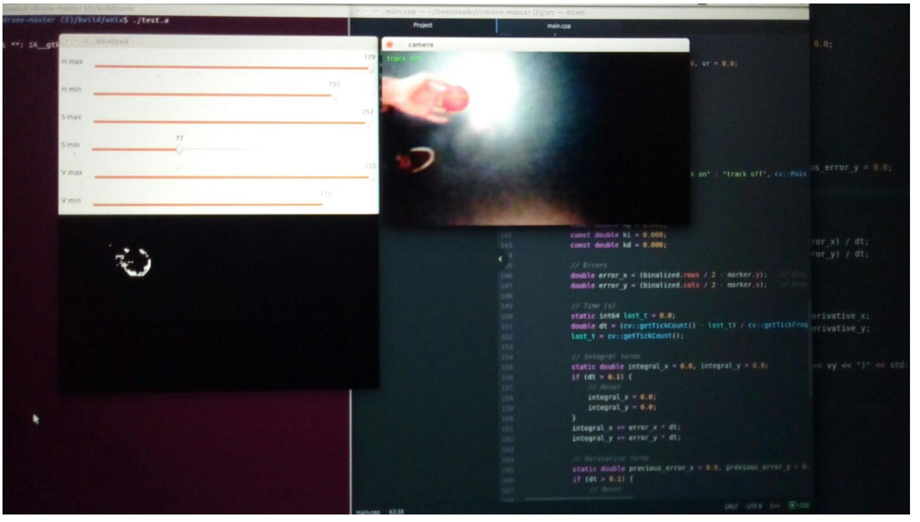
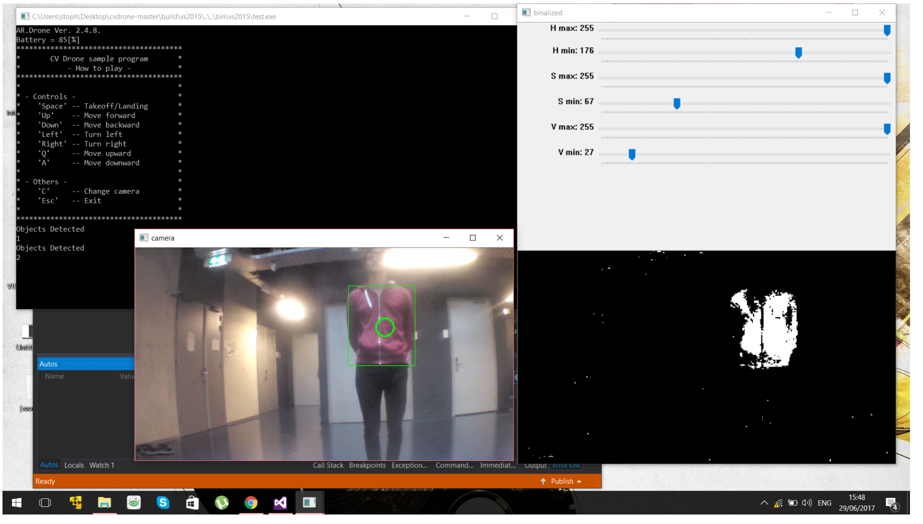


FIGURE 2 The drone autonomously tracking and following the red ball by recognizing it's colour (red) with the feed of the downwards facing camera.

FIGURE 3 The forward facing camera feed of the drone recognizing the red cardigan by it's colour (red).



4 | DISCUSSION

Several factors proved technically difficult perhaps due to the fact that the drone used wasn't in the best condition. We encountered large error margins. This resulted into having to adjust the velocities of our autonomous flight routine continuously throughout the day. Minor incidents such as low battery power, changing batteries, bumping into a wall and different ground texture (through illumination changes) proved to be very influential. Early in our project we focused on programming entirely in javascript as we anticipated that the use of a more advanced programming language such as C++ might have been too difficult. We started off working in an AR-drone library using Node.js (Javascript). However due to unanticipated restrictions in camera use we decided to use OpenCV. A larger timeframe would have allowed us to better assess the potential for autonomous drones in surveillance. Due to zoning laws we were restricted to conduct our research inside the University of Amsterdam. Limiting the application of our data to the outside world. Future research should focus on object recognition in the open air with uncontrolled weather conditions. In the open air gps can be employed to navigate which would allow for greater precision and solve the technical difficulties we encounterd between trials. Future research should also focus on employing algorithms to disperspe police officers effectively over an area, and allow drones to provide the instructions to said officers on the ground.

5 | CONCLUSION

The question as to whether or not drones can be used as an alternative to monitor crowds was researched. The first hypothesis in which a single toy drone was employed to fly in a square and take off, and land autonomously showed that this is indeed possible. However over many trials much work was put into calibrating the speed and time necessary to make a proper square. Over 100+ trials it wasn't possible to have the drone take off and land in a perfect square without manual recalibration. However our findings and end product we conclude that a toy drone can be calibrated to surveil an area and detect objects based on their colour and size. The small cost of a toy drone indicate that in the future drones of similar or better quality can be used to ensure public safety. In a crowd control situation data collected by drones can be used to identify the number of objects with a specific colour, such as blue uniforms of Dutch police officers. Relaying that information back to officers on the ground in order to disperse personnel more effectively. The ability to use the downward facing camera to identify and follow small objects was a happy surprise to the researchers since the camera has a low resolution. Drones could then also be employed to follow targets from above using object recognition, again eliminating the need for expensive helicopter surveillance.

REFERENCES

[1] CV-DRone;. Accessed: 2017-06-21. `https://github.com/puku0x/cvdrone`.

[2] Parrot Drone 2.0;. Accessed: 2017-06-19. `parrot.com/fr/drones/parrot-ardrone-20-gps-edition#parrot-ar-drone-20-gps-edition`.

[3] Brahmbhatt S. Practical OpenCV. Apress; 2013.