Poster Abstract: High Speed Object Tracking Using Edge Computing

Joel Dick, Caleb Phillips, Seyed Hosein Mortazavi, Eyal de Lara
University of Toronto
Toronto, Canada

ABSTRACT

The use of unmanned aerial vehicles (UAV), or drones, has in recent years seen explosive growth due to lower costs and technology advances in mobile computing, batteries, sensors, and control systems. Drones are now used in a multitude of applications, from natural resource exploration, the film and entertainment industry, to urban surveillance, and defense. The image processing demands of these applications requires higher powered computing capabilities than those available locally to the drone, prompting the offloading of these tasks to the cloud. However, the latency requirements of the cloud are beyond those acceptable for many applications. This paper proposed the use of a server on the network edge to optimize both processing capability as well as latency for applications requiring real-time communication between a drone and a cloud server. We propose to test the limits of this model by implementing a system for real-time tracking of golf drives on a golf course.

KEYWORDS

Edge Computing, Computational Offloading, Drone, Unmanned Aerial Vehicles, Golf

ACM Reference format:
Joel Dick, Caleb Phillips, Seyed Hosein Mortazavi, Eyal de Lara. 2017. Poster Abstract: High Speed Object Tracking Using Edge Computing. In *Proceedings of SEC '17, San Jose / Silicon Valley, CA, USA, October 12–14, 2017,* 2 pages.
https://doi.org/10.1145/3132211.3132457

1 INTRODUCTION

Within recent years, there has been an explosive proliferation in the use of commercially available unmanned aerial vehicles (UAVs), commonly known as drones, for a variety of imaging applications. These applications include natural resource exploration, the film and entertainment industry, urban surveillance[2][3], and defense. Many of these application require high performance computing power for the processing and analysis of high definition video. To preserve battery life and limit their size, UAV systems generally run on lightweight, lower power processor units with only the minimal computing power necessary for control of the drone and communication to a base station. This has prompted the need to offload computationally intensive tasks, such as advanced image processing and analysis, to the cloud. However, many applications require real-time feedback that is faster than the round trip latency to remote cloud servers, and therefore the use of lower latency solutions are required.

SEC '17, October 12–14, 2017, San Jose / Silicon Valley, CA, USA

ACM ISBN 978-1-4503-5087-7/17/10…$15.00
https://doi.org/10.1145/3132211.3132457

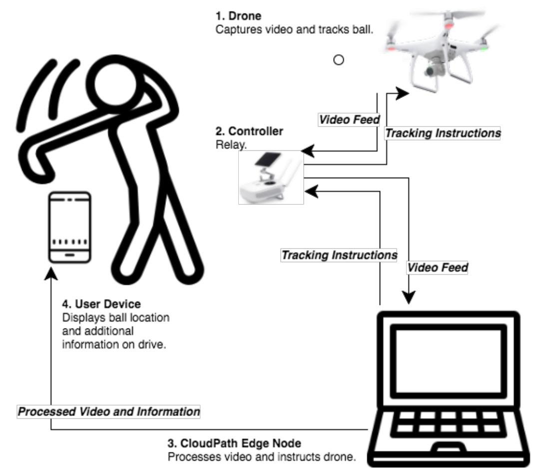

Figure 1: Architecture

Our proposal is to offload computationally intensive but time sensitive computing tasks to the edge of the network, allowing for low latency, near real-time communication between a drone and server. Our system will leverage CloudPath[4], a multi-tier edge computing platform, to act as a nearby low latency edge server to which the drone can offload its computation. To test this model, we have chosen to implement a system that detects and tracks a golf ball trajectory in real time on the golf course utilizing two way communication between a drone and the CloudPath node for image processing and analysis. We use a drone to shoot high definition video of the ball's flight. The video is then processed and analyzed on a local CloudPath server to detect the trajectory of the golf ball, and that data used to provide control parameters to the drone, as well as real-time data to the player on a mobile device. The architecture of the system is shown in Figure 1. Our solution highlights a unique application for edge computing and the integration of mobile, UAV, and cloud technologies for computationally intense real-time computing.

The proposed application is demanding in both processing and latency requirements. Golf drives can go as far as 350m at speeds in excess of 300 km/h in professional tournament play. Even meeting the requirements of amateur play would be quite demanding for mobile computing systems. To capture the entire drive in a single frame, the camera equipped drone would need to shoot images of the golf ball from over 150 meters away. Given a resolution of

4K video (4096 pixels) an object as small as a golf ball would still occupy less than one pixel on the camera sensor. This is not an impossible task, but to detect an object as small as that against a complex backdrop requires processing power well in excess of that available to the drone alone, and necessitates offloading onto the cloud.

Alternatively, to get a better view of the ball, the drone and camera can dynamically track the movement of the golf ball by flying close to the ball and following its path, or rotating the camera on a gimbal to track the angle of the ball in flight. This requires two way communication between the drone and the server. Several frames need to be sent from the drone to the server, those images need to be analyzed for the movement of the ball, an update to the flight path and gimbal orientation needs to be calculated, and new flight instructions need to be sent back to the drone, all within the time it takes for the ball to move out of the limited frame of the camera. At the typical speeds of a golf drive, this leaves only about 150ms for computation and two-way communication from the drone to the server. Typical latency to remote cloud servers is in excess of 100ms. Therefore, using a server that is located a single network hop away from the user is the only feasible option.

2 IMPLEMENTATION

Our proposed solution makes use of a DJI Phantom 4 Pro equipped with a 1-inch 20 megapixel CMOS sensor camera. The camera is capable of shooting video at up 30, 60, or 120 frames per second at resolutions ranging from 720p HD (1280 x 720 pixels) to 4K (4096 x 2160 pixels). However, there is a tradeoff between resolution and frame rate with higher quality resolutions limited to lower frame rates. For this application, we favored higher resolution over faster frame rate to make detecting of the ball easier. Once the ball is detected and locked onto, tracking its movement is easier even at a lower frame rate. The camera has a 84-degree field of view and is mounted on a three axis gimbal, allowing us to capture the entire drive at a distance of about 125 meters, or fly in and follow the ball at a closer distance while rotating the gimbal to keep the ball within the frame.

The drone communicates with a DJI remote controller handheld device using 2.4 GHz or 5.8 GHz radio. The handheld has a built in display running a modified version of Android with 4GB RAM memory and 16GB of storage. Our system will use a custom Android application developed with the DJI Mobile SDK that will run on the handheld device to control communication between the drone and the CloudPath server. Our CloudPath node runs on a standard 2015 Macbook Pro. The image processing and analysis application will be written in Python 2.7 and use the OpenCV library[1] for image processing functions, and will run on the CloudPath server.

At run-time, the drone and CloudPath systems will engage in two-way communication via the handheld device. The camera will take continuous video of the golf ball before the player takes a shot. The video data will then sent to the handheld device which will relay the data to the CloudPath server. The CloudPath application will lock onto the position of the ball. If movement is detected in the ball, the CloudPath application will calculate if the drone needs to move or if the camera angle needs to be adjusted to keep the ball within the frame. A new flight path and camera position will be calculated and relayed back to the drone via the handheld device. After the drive is completed, the CloudPath application will analyze the entire footage of the ball's flight and presents it the the user via the handheld device overlaid on the map of the golf course.

3 CONCLUSION

Our proposed application tests the limits of the edge computing model for image processing tasks for drone applications. Our implementation demonstrates that CloudPath can be used to optimize both processing capability and maintain an acceptable latency for applications requiring real-time communication between a drone and a cloud server running the computation intensive parts of the application. The use of this system for tracking golf drives on a golf course in real time is but one application of the edge computing paradigm, but our solution highlights a unique application for edge computing and the integration of mobile, UAV, and cloud technologies for computationally intense real-time computing.

REFERENCES

[1] Gary Bradski. 2000. The OpenCV Library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer* 25, 11 (2000), 120–123.
[2] Ning Chen, Yu Chen, Sejun Song, Chin-Tser Huang, and Xinyue Ye. 2016. Smart Urban Surveillance Using Fog Computing. In *Edge Computing (SEC), IEEE/ACM Symposium on.* IEEE, 95–96.
[3] Ning Chen, Yu Chen, Yang You, Haibin Ling, Pengpeng Liang, and Roger Zimmermann. 2016. Dynamic urban surveillance video stream processing using fog computing. In *Multimedia Big Data (BigMM), 2016 IEEE Second International Conference on.* IEEE, 105–112.
[4] S. H. Mortazavi, M. Salehe, C.S. Gomes, and E. de Lara. 2017. CloudPath: A Multi-Tier Cloud Computing Framework. In *2017 IEEE/ACM Symposium on Edge Computing (SEC).* to appear.