DyCOCo: A Dynamic Computation Offloading and Control Framework for Drone Video Analytics

Chengyi Qu, Songjie Wang, Prasad Calyam
Email: cqy78@mail.missouri.edu, {wangso, calyamp}@missouri.edu; University of Missouri-Columbia, USA.

***Abstract*—Unmanned aerial vehicles (UAV) or drone systems equipped with cameras are extensively used in different surveillance scenarios and often require real-time control and high-quality video transmission. However, unstable network situations and various transport protocols may result in impairments during video streaming, which in turn negatively impacts user's quality of experience (QoE). In this paper, we propose a dynamic computation offloading and control framework, named DyCOCo, based on image impairment detection under various available network bandwith conditions. Our DyCOCo framework demo features IoT devices in a testbed setup on the GENI infrastructure. Our demo results show that our DyCOCo approach can efficiently choose the suitable networking protocols and orchestrate both the camera control on the drone, and the computation offloading of the video analytics over limited edge computing/networking resources.**

***Index Terms*—computation offloading, drone video impairment recovery, GENI testbed**

I. Introduction

Unmanned aerial vehicles (UAVs) are being extensively used in different scenarios in urban and rural area surveillance scenarios. They typically are embedded with high-resolution cameras and edge computation resources. As shown in Figure 1, a UAV system consists of air-to-ground wireless links and ground-based wired networks. In most air-to-ground scenarios, UAVs are connected to ground control stations (GCS) at the infrastructure gateway through a wireless network e.g., based on Wi-Fi, 4G/LTE or 5G technologies. In edge networks, a variety of environmental conditions (e.g., codec) may affect the performance of the video streaming between the UAV and GCS. This in turn affects the performance of video streaming in terms of frame blurring, stalling and distortion.

To ensure user quality of experience (QoE) [1] is satisfied through intelligent coordination in the drone video analytics, we present a novel dynamic decision making strategy which can be used during the process of computation offloading scenarios involving UAVs and GCS in edge computing setups. The intelligent video processing considers the trade-offs in selection of computation location using a novel function-centric computing (FCC) architecture [2]. Using this approach, we decouple the application pipeline into several standalone functions/microservices that can be executed independently on different computing nodes/sites. We have conducted experiments to evaluate how our scheme utilizes state-of-the-art computation offloading techniques to Pareto-optimize trade-off performance (i.e., frames-per-second) vs. QoE factors (accuracy rate and tracking rate) during drone video analytics.

This material is based upon work supported by the National Science Foundation under Award Number: CNS-1647182 and the Army Research Lab under Award Numbers: W911NF1820285 and W911NF1910181. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the Army Research Lab.

978-1-7281-2700-2/19/$31.00 2019 © IEEE

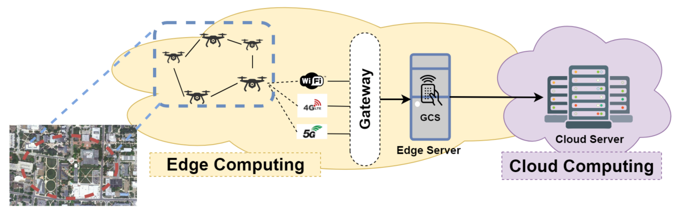


Fig. 1: Overview on UAV video analytics with Edge and Cloud resources

II. Computation Offloading and Control

We propose a novel dynamic function-centric computation offloading framework for drone video streaming analytics viz., "DyCOCo" that can be used on multi-UAV/GCS/edge-cloud scenarios as shown in Figure 1. This framework is deployed on a GENI testbed, which features physical IoT devices on the ground (acting as UAVs) to connect to edge and cloud servers using Docker services and RESTful API through a wireless network. The framework allows us to not only facilitate trade-offs in performance vs. processing speed of drone video analytics, but also aids in decision-making among the edge, cloud or FCC computing paradigms for data processing. Furthermore, we also consider network failure and the resultant recovery time in UAV video transmission to avoid packet loss and minimize waiting time. For near real-time processing and high-quality video analysis, our framework makes dynamic decisions on changing networking protocols (i.e., TCP/HTTP, UDP/RTP, QUIC) for video data transmission as well as resolutions to avoid impairments such as blurring and stalling during video streaming, as detailed in Figure 2.

Notably, based on our literature survey of [3] and related experiments, unreliable wireless network quality and inappropriate networking protocol/codec choices often results in video impairments, which could directly influence the object tracking ability and recognition accuracy during video transmission. Our proposed approach shown in Figure 3a addresses this issue by dynamically changing the networking protocols, switching between high resolution/low resolution for video capture, or

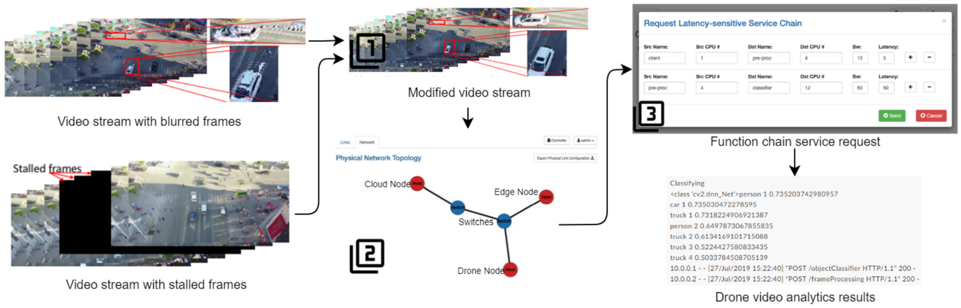
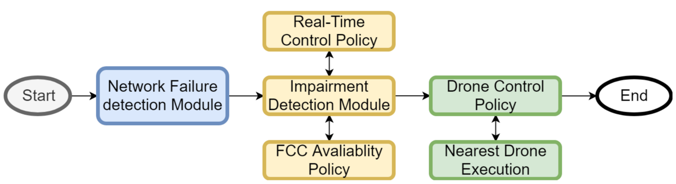


Fig. 2: Illustration showing processes of our DyCOCo framework for UAV video analytics on 1) impairment detection and control module to modify video stream, 2) physical network topology settings to use cloud resources and 3) requesting service chain to allocate resources and fulfil real-time control policy

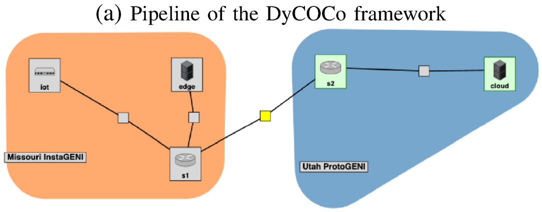
changing camera direction for effective assessment of the scene in real-time.

III. Demonstration Scenario

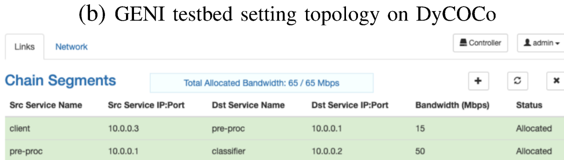
Our experiments consider several drone surveillance video streams from a VisDrone dataset [4] with different VGA resolutions (1080p: 1920 x 1080; 720p: 1280 x 720; 480p: 854 x 480; 360p: 640 x 360) in the DyCOCo pipeline (see Figure 3a) to count and track moving objects (e.g., cars, trucks and pedestrians). The analytics application used in our experiments is publicly available at [2].



(a) Pipeline of the DyCOCo framework



(b) GENI testbed setting topology on DyCOCo



(c) Function chain setting GUI on DyCOCo

Fig. 3: DyCOCo prototype illustrations

Our geo-distributed drone/GCS/cloud testbed setup includes 2 micro-processors, 1 edge server (without HPC capability) reserved in a GENI rack [5] at the Missouri InstaGENI site and 1 cloud server reserved at Utah ProtoGENI site as shown in Figure 3b. The two GENI sites are connected by having an OpenFlow virtual switch on each site. Micro-processors are NVIDIA Jetson Nano [6] with a configuration of 128-core Maxwell GPU, Quad-core ARM A57 CPU and 4G RAM. Edge server has 12 cores Intel Xeon CPU and 16 GB RAM and has the same settings on a cloud server at the Missouri InstaGENI site. Cloud server has 2 X 20 cores Intel Xeon Silver CPUs, 192 GB of EEC RAM and 12 GB NVidia Tesla P100 GPU. All the servers above support Docker containers and can execute video analytics functions independently, without running an entire application pipeline.

In the first part of the demo, we show how UAVs with on-board computation resources can be controlled to execute a part of the video analytics functions as well as buffering the video data (i.e., during network partitions) instead of offloading all the computation to a GCS or the cloud. We demonstrate that our DyCOCo framework can handle network failure events without compromising the object recognition accuracy as well as tracking performance. In the second part of the demo, we demonstrate how different transport protocols and various bandwidth generate impairments in the video stream. We also show how our DyCOCo framework detects impairments and rectifies them through: (a) camera control directed by navigation control to get the necessary high-resolution video streams in the video farming, or (b) change the networking protocol/codec without compromising the tracking performance. Next, we instantiate a geo-distributed UAV/GCS/cloud topology on a GENI testbed and allocate diverse resources to each node (see Figure 3b). As the final step, we setup latency and bandwidth requirements to each function/task in the video analytics application, and show how our DyCOCo framework identifies the most appropriate network topology and allocates required resources with our *MRSCO* tool [2], as shown in Figure 3c.

References

[1] P. Calyam, M. Haffner, E. Ekici, and C. G. Lee, "Measuring interaction qoe in internet videoconferencing," in *IEEE/IFIP MMNS*, 2007.

[2] D. Chemodanov, C. Qu, O. Opeoluwa, S. Wang, and P.Calyam, "Policy-based function-centric computation offloading for real-time drone video analytics," in *IEEE LANMAN*, July 2019.

[3] M. Seufert, R. Schatz, N. Wehner, and P. Casas, "Quicker or not? -an empirical analysis of quic vs tcp for video streaming qoe provisioning," in *22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Feb 2019, pp. 7–12.

[4] P. Zhu, L. Wen, X. Bian, L. Haibin, and Q. Hu, "Vision meets drones: A challenge," *arXiv preprint arXiv:1804.07437*, 2018.

[5] B. Mark, J. Chase, and L. Landweber, "Geni: A federated testbed for innovative network experiments," *Computer Networks*, 2014.

[6] Nvidia, "Nvidia jetson - the ai platform for autonomous machines," Jul 2019. [Online]. Available: https://developer.nvidia.com/embedded/develop/hardware