ABSTRACT

UAVs (unmanned aerial vehicles) equipped with high-end cameras have become increasingly popular among consumers. UAVs have been traditionally considered for applications such as disaster response and surveillance, while emerging applications include live-event broadcasts, precision agriculture, and augmented-reality games. Consumer UAVs today use fixed-bitrate video streaming where users configure the resolution (4K or 1080p). However, applications with real-time streaming that deploy UAVs in the wild will require adaptive video streaming to tackle uncertain wireless link capacities and meet their video quality requirements. This paper is a first step toward the design of adaptive video streaming algorithms that can provide significant gains for UAV streaming. Our system SkyEyes leverages two novel aspects to aid UAV streaming: content-based compression and video rate adaptation based on location sensors and client buffer status. Our system prototype, while far from complete, exhibits promise – we believe that adaptive video streaming is indeed crucial for real-time UAV applications.

1. INTRODUCTION

UAVs (unmanned aerial vehicles) or drones equipped with cameras have become increasingly popular in the consumer market, for example, DJI and Yuneec recently released a consumer drone that captures 4K video. Hobbyists use them to take aerial photography, and several new applications have emerged, such as live-event broadcasts, precision agriculture, and immersive augmented-reality games with drones. Commercial drones hold great promise to make surveillance and search-and-rescue missions easier and safer. With the widespread use of UAVs, they must effectively tackle the challenge of reliable and stable streaming, especially as the number of sensors gathering data on UAVs grow.

Consumer UAVs today use fixed-bitrate video streaming algorithms where users configure the resolution (4K or 1080p) to view the video on a smartphone-like screen. However, emerging drone applications such as live-broadcasts and gaming require high-definition real-time streams at low latency and jitter. Imagine a scenario where a UAV camera captures the aerial shots of soccer ball game. Other drone applications such as military surveillance or search-and-rescue process the video in real-time to complete their mission. The video data link is critical to the mission success. Imagine a scenario where a UAV patrols a remote area and is operated remotely by a ground crew so that the drone camera becomes the “eyes in the sky.”

UAV cameras stream the captured video to one or more ground stations over a wireless channel. The quality of the wireless link varies based on the drone speed and distance of the ground-station. Civilian use-cases are limited to the line-of-sight channel because of FAA regulations. But, in some military scenarios, UAV might only be connected to the ground station over a satellite data link. The captured streams need to be transmitted over varying link capacities at highest quality. Modern video streaming applications for mobile and web clients use adaptive techniques to combat variations in the throughput of the underlying channel. Adaptive video streaming algorithms and systems have been intensively studied in research communities [1, 2] and in industries, such as Apple, Netflix, and Microsoft. In particular, several video streaming clients have developed adaptive HTTP-based video streaming protocols, such as DASH [3], to perform channel-based video rate adaptation. However, existing adaptive video-streaming techniques do not work well for fast-mobility scenarios of UAVs and their emerging applications.

To solve this problem, we design SkyEyes, a system that adapts video streaming algorithms to meet drone-based application requirements. SkyEyes uses two novel aspects. First, to tackle high-mobility scenarios, the system adapts the video resolution and bit-rate based on location information, such as inertial sensors and GPS coordinates. The client buffer information when available is also used to adapt video throughputs. Second, the system uses a content-aware compression technique to transmit compressed feature representation of the objects or regions that the target application queries. The system compresses and transmits only the relevant video portions. Such content-aware compression is crucial under severe link capacity constraints when video throughput adaptation does not suffice.

This paper incorporates the above ideas into an Android-based prototype attached to a Yuneec drone. We evaluate SkyEyes with adaptive video streaming over an actual soccer field. Our system SkyEyes improves the transmitted video
quality by predicting and selecting highest video bitrates compared to conservative behavior of existing throughput-adaptive algorithms. The time when video freezes during playback is less than 1% even when throughput varies rapidly. This suggests that SkyEyes can be timely and responsive allowing real-time feedback from ground-stations for successful UAV missions. Results from content-based compression are promising when the target objects of interest occur over less than 50% of the entire video. Still, much work remains to test the SkyEyes performance for widespread use.

2. MOTIVATION

2.1 Applications

Disaster Response: UAVs can reach remote areas at fast speeds in disaster and emergency situations, such as earthquakes, tornadoes etc. They can be safely deployed in situations that would pose a risk to emergency responders on foot. By using night-vision or infrared cameras, UAVs can pinpoint locations of survivors in the dark. Operations center can use the live images transmitted from the UAV to the ground station to decide and plan search-and-rescue.

Surveillance & Intruder Detection: UAVs have been considered in surveillance of large areas or large gatherings, where it is difficult to maintain an overview of the entire area with just ground security personnel. UAVs can be deployed to scan agricultural fields, where they can alert the farmers to intruders or animals in the field and spray pesticides after identifying pest-infested areas. UAVs deployed in these applications need to transmit regions of interest from the live video stream to enable the ground station to act upon them.

Live Event Streaming: Sports fields and concert venues deploy more than 20–30 cameras to get full coverage of the event. Video shots from UAVs provide a unique aerial perspective that is not possible with static cameras. Media companies, such as Fox news, have begun to use drones to cover such live broadcasts that require high-definition video streams to be transmitted reliably to the ground station.

Immersive Gaming Applications: Some emerging applications of UAVs include augmented reality based games such as first-person-view (FPV) drone racing and drone Pokémon Go. In FPV drone racing, the player’s drone competes on a track with interesting obstacles and the player controls the drone by using FPV goggles to see the view from drone’s perspective. Similarly, the player in drone Pokémon Go app chases the Pokémon character with a drone that the player controls based on the drone’s video perspective. These gaming applications require live streaming at low latency to ensure timely response to their players.

2.2 Challenges

Several aspects make video streaming from UAVs challenging. First, UAV cameras capture video while they move. This video does not achieve high compression efficiency because video compression algorithms, such as H.264, rely on spatio-temporal motion-compensation. In addition, UAVs are equipped with increasing number of sensors including multiple cameras and thermal imagers, that add to the growing volume of data. Second, the wireless link capacity can vary rapidly when the drone moves at high speeds (>20m/sec) where the trajectory might be determined on the fly. In this scenario, conventional throughput-based adaptation don’t guarantee good performance as predicted throughputs will have large errors from the time-series based prediction. Finally, the drone application requirements vary a lot – while some require low latency and jitter, others require high video-quality, and yet others are only interested in detecting target objects of interest.

3. SYSTEM DESIGN

The system consists of two modules as illustrated in Fig. 1. The adaptive video streaming module ensures high-quality stable video streaming. It adapts video bitrates to rapid fluctuations of wireless-link capacity as drone moves by incorporating predictions based on location sensors. Further, it ensures the video does not freeze for timely response to the ground station by incorporating playback buffer status in the adaptation algorithm. The content-aware compression module ensures only the relevant video portions are transmitted when the application’s target objects or queries...
3.1 Adaptive Video Streaming

Two key factors determine the quality of UAV streaming: the available link throughput and the receiver playback buffer at the ground station. The throughput of the drone-server wireless link is largely determined by path loss, though it is susceptible to fluctuations from short-term fading. Link throughputs vary rapidly as the drone moves at high speeds. If the drone streams a fixed-bitrate video and the channel quality degrades as drones move far away, then the received playback buffer might remain empty waiting for the high-quality video segments to arrive. The streaming video will freeze at the instants when the receiver playback buffer is empty.

Throughput prediction. Adaptive algorithms typically rely on time-series based throughput prediction to adapt video bitrates to the available link throughputs. Such algorithms work well when the throughput varies slowly so that future throughputs are similar to the past throughput values. We design a novel throughput prediction module to account for the fast throughput variations from drone mobility. In addition to using previous throughput values, our module leverages the current location coordinates and projected drone trajectory to predict the available link throughput.

Buffer-based adaptation. Adapting throughputs does not suffice when the predictions are incorrect and the receiver playback buffer becomes empty thus stalling or freezing the video. For real-time applications, frozen video could result in delayed action from the ground station that can lead to failed missions. Incorporating feedback from receiver playback buffer can significantly reduce re-buffering events and frozen video screens that degrade UAV stream quality.

Our adaptive video streaming algorithm adapts the bitrate of video segments based on the predicted link throughput and the receiver playback buffer status. The algorithm is described in Alg. 1. When the receiver playback buffer at the ground has the minimum required video length \( T_B \), the ground station requests the UAV to stream the next segment at highest bitrate available. The highest available bitrate is estimated based on the prediction of both drone location and throughput from the previous segment duration. On the other hand, when the receiver playback buffer is lower than the minimum required video length \( T_B \), the ground station conservatively requests a lower bitrate to fill its buffer to the minimum level. The proposed algorithm runs at the rate of the segment length (e.g. our algorithm will run every 1 second for 1-second long segments).

**Algorithm 1 Adaptive video streaming**

1: Predict link throughput based on the drone location coordinates in the next step and the throughput values in the previous time step.
2: Check the number of segments in buffer, \( B_t \).
3: If \( B_t < T_B \) (the required minimum video length in buffer), request the video quality that can fill the buffer to \( T_B \). If infeasible, request the lowest quality.
4: Else, request the highest possible bitrate video given the predicted link throughput, i.e. maximize the received video quality.

3.2 Content-aware compression

Video compression schemes, such as H.264/AVC, rely on motion-compensated interframe prediction to exploit correlation among successive frames and thus work well when the background is relatively static. UAV captures videos while moving. The captured video does not yield high compression efficiency compared to videos captured from static cameras. UAV applications, such as surveillance, search-and-rescue, precision agriculture, are typically interested in specific objects, such as faces, persons etc. We leverage this application requirement to design content-aware compression.

Conceptually, content-aware compression scheme will only transmit frames that contain target objects that UAV application is interested in or has sent queries for. UAV can process each frame of the incoming video to detect and match to the target objects of interest and only transmit frames that contain matched objects. However, the object recognition pipeline can be computationally intensive for the UAV processor.

We propose a design where the UAV processes each incoming video frame to detect objects and extract image features that will be used for matching. While traditional object recognition pipelines rely on SIFT and SURF features, they are not compact. We instead rely on compressed histogram of gradients (CHoG) features because they are easy to compute and uses only about 60 bits per feature, on average [4]. This design effectively breaks down the object recognition pipeline and UAV transmits only the compressed features for each frame to the ground station. The ground station then matches the features against target objects of interest to identify if the video frame is relevant. If the video frame is relevant, it is added to the transmit queue. The process then falls back to adaptive video streaming algorithms which fetch the relevant video segments based on available link throughputs. The algorithm is illustrated in Fig. 1.

4. PRELIMINARY EVALUATION

Experimental Setup. Our initial prototype (Fig. 2) is implemented on an android-based smartphone attached to a Yuneec Q500 drone that streams video to a laptop. The laptop located on the edge of the soccer field serves as a ground station in this case. We implement the adaptive video streaming module by modifying the open-source code available from the VLC player [5]. The phone stores and streams a soccer video dataset at various bitrates [6] – we use the same soccer video instead of live capture from the drone camera to compare different video streaming schemes fairly. To stream live, a real-time video encoder converts the captured video to segments of different video qualities with the DASH format [3]. We evaluate video streaming algo-
Adaptive Video Streaming. Our field experiments give us the following key insights: fixed-bitrate video streaming performs poorly in UAV streaming and conventional throughput-adaptive algorithms fail to keep up with the rapid fluctuations in link throughput as the drone moves at speed. The proposed adaptive video streaming module outperforms both by sending high bitrate video segments while keeping the video freezing time minimum.

We measure the video quality of fixed-bitrate streaming while the drone hovers at different locations on the field (location A, B and C in Fig. 3). The available link throughput varies significantly as the drone moves away from the ground station. For example, location C supports streaming of 35 Mbps high-quality 4K video streams with no freezing screens at a distance of about 8 m from the ground station. However, the same 35 Mbps stream freezes frequently when the drone moves farther away. Thus, over the drone trajectory shown in Fig. 3, fixed-bitrate streaming at 35 Mbps experiences 77.8% freezing duration on average (see Fig. 4). On the other extreme, fixed-bitrate streaming at 9 Mbps on the same trajectory experiences only 3.5% freezing duration on average. In fact, location B can support streaming of 9.6 Mbps streams with no freezing screens. These results suggest the value of adaptive video streaming. In fact, our proposed video streaming module experiences less than 1% video freezing (Fig. 4) while sending highest bitrate video segments (Table 1).

Next, we compare conventional throughput-adaptive video streaming to our proposed algorithm in Section 3.1. Fig. 5 illustrates a 30 second trace when link throughput (black line) varies rapidly as the drone moves between locations B and C at approx. 20m/s speeds. Conventional throughput-adaptive algorithm (red line) relies on previous throughput to predict the current throughput and thus, its prediction is too aggressive if throughput decreases rapidly. Our proposed algorithm (blue line) uses the drone’s projected location coordinates and the receiver playback buffer status to predict the throughputs more accurately. Thus, our proposed algorithm experiences no video freezing screens (Table 2), while conventional throughput-adaptive algorithm experiences freezing screens 26.67% of video duration. Further, our proposed algorithm sends higher percentage of high-bitrate video segments as shown in Fig. 6. For example, it receives 51% of the video segments at 16 Mbps and more, while the conventional throughput-adaptive algorithm only receives 23% of the video segments at 16 Mbps. Similar results are observed in field experiments as shown in Table 1.

Content-based compression. We consider a face recog-

<table>
<thead>
<tr>
<th>Adaptive Alg.</th>
<th>freezing time (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>tput adapt</td>
<td>26.67</td>
</tr>
<tr>
<td>SkyEyes.</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2: Freezing time comparison of conventional throughput-adaptive and our proposed algorithm when throughput varies rapidly (Fig. 5).
nition module to recognize faces in the captured video from the UAV. In our initial prototype, we find content-based compression provides gains over H.264 video compression when the target objects of interest occur over less than 50% of the entire video. Comprehensive evaluation will be presented in our future work.

5. RELATED WORK

Adaptive video streaming. HTTP-based adaptive video streaming (DASH) has been widely used by video streaming companies, such as Youtube, Netflix, Hulu, to provide high video quality to end users. Each video is segmented into chunks of 2–10s and multiple copies of each chunk are stored in the video server at different bitrates. When the client requests a new segment, a video bitrate is selected based on the estimated channel condition. Conventional throughput-based adaptive video streaming [3] doesn’t perform well when throughput fluctuates rapidly [2] because the predictions are not accurate. Buffer-based adaptation [7] promises to improve video quality by requesting lower-bitrates video segments when buffer is empty and thus reducing re-buffering events. However, the reactive nature of this adaptation does not work well either when the link quality changes frequently. Our proposed approach in this paper uses the best of both approaches while designing a more accurate throughput prediction module based on drone’s location coordinates to improve video quality.

Adaptive video streaming with scalable video coding (SVC) adapts better to a fast-varying channel [8]. However, each video segment must be encoded to multiple layers. The video segment must be encoded to multiple layers. The enhancement layers after downloading the base layer. Today encoding live video streams into SVC format is computationally intensive at the UAV processor and therefore not considered in this paper.

Content-aware compression has been shown to be promising under constrained wireless link capacities for video monitoring [9] and mobile visual search [4]. In Vigil [9], the authors propose a content-aware compression algorithm to determine the most valuable frames from cameras within a cluster relevant to the target queries over a real-time tracking and surveillance application. The results minimize communication bandwidth by more than 2–5x. Similar gains are reported for mobile image-based retrieval queries in [4]. This paper proposes a content-compression approach for UAV-streaming while ensuring its computations do not tax UAV processor.

6. CONCLUSIONS

This paper is an early effort towards designing adaptive video streaming for UAV applications. Our proposed system SkyEyes predicts available link capacities as UAVs move rapidly based on drones’ location sensors and adapts video bitrates to match the receive playback buffer constraints. Further, it leverages content-based compression to identify target regions of interest to stream when link capacities are low and application requirements are known. Our initial results promise significant performance gains.

While much remains to be done, we believe there is promise to pursue a longer-term research engagement. In particular, we intend to thoroughly evaluate in our future work if the proposed algorithms can keep up with real-time video encoding. Drone manufacturers have recently expanded the computational capabilities of their platforms making real-time video processing promising for our future work.

7. REFERENCES


Figure 6: Comparison of video segments’ quality in conventional throughput-adaptive and proposed algorithm for SkyEyes when throughput varies rapidly (Fig. 5). SkyEyes receive higher bitrate video segments with less than 1% freezing time.