TECHNICAL PAPER

Tech Source
An EIZO Group Company

Challenges in Relaying Video Back To Mission Control

Using a customizable H.264 hardware encoder is essential to delivering the high compression ratio and guaranteed speed needed for mission critical manned and unmanned video streaming applications.

By Christopher Fadeley, Software Engineer, Tech Source

The change in emphasis to a more avionics approach to the military theatre has introduced a new set of challenges in terms of technological feasibility. It is often requested to have an active video feed sent large distances back to ground, with low latency and in high definition. These criteria must be met at all times. The video feed is typically used for monitoring and/or recording of the mission, so if the video feed arrives too late or in poor quality, the resulting images are of no use to control staff and the mission can be compromised.

The real challenge is dealing with the low bandwidth available to stream video. All transmissions must be sent wirelessly and ground control may be a long distance away, especially when it involves remote controlled Unmanned Aerial Vehicle (UAV) or intelligence gathering missions (Figure 1). Wireless transmission is usually performed in an atypical method with limited bandwidth, like a cellular relay or satellite transmission. This often puts a very strict limitation on the data transfer rate of the video.

Figure 1:
UAVs need the feed streamed remotely in as timely a fashion as possible, otherwise the mission may be compromised. (Copyright iStock.com-JoeLena-credit required)

Raw video by definition is lossless, but is also highly wasteful in the amount of data it takes to display. A 1080p 30fps raw video has a data rate upwards of 200MB/s (megabytes per second). Raw video has its place in the military field in the form of live local viewing and GPGPU processing on the fly. But in applications where the video needs to be streamed remotely, raw video is not feasible.

The solution is to compress the video, but this comes with its own set of challenges. Compressing video can cause a drop in quality. The bigger challenge is that video must be sent in a timely fashion and compression introduces latency. Latency is the time between the camera capturing the video and the time that data is actually displayed or recorded remotely or locally. Compression takes time, especially if high quality and/or resolution are needed.

The best way to ensure low latency with reasonable compression is to use dedicated compression hardware. And given the military field, the hardware must be ruggedized to survive in harsh environments and still consume low power. Keeping that low power while compressing with expected results can be difficult. Hence the hardware must also be computationally efficient.

Encoding

The current standard for compressing video is H.264/MPEG4-Part 10 AVC. This process of compression is also called encoding. H.264 is now the most widely adopted advanced video codec in part due to its high compression ratio and highly configurable options. The strict bandwidth limitations make the configuration of H.264 a necessity. H.264 can encode to either a constant or variable bitrate. Constant bitrates are easy to understand. The video being encoded is always encoded at the user defined bitrate with no regard for the video being captured. This allows an end user to strictly define their video stream to their known bandwidth ensuring no transmission overflow since the video is guaranteed to be a certain size.

Variable bitrate is much more complex since it encodes based on what the actual video data is. A video stream with high motion (every frame different from the last) is very difficult to encode. Compression at a basic level relies heavily on the concept of redundant data through time. For example, a zip-file looks at the binary data of a file and sees where data is being repeated. If the file being compressed is a document containing the letter 'A' repeated 1000 times, the encoder can analyze this data and store the file as 'A:1000' instead of the letter being repeated so many times. This saves valuable space.

In the case of video, a repeated black screen is very easy to encode since it is the same pixel data over and over. Variable bitrate encoding can be set to a target bitrate which it tries to meet; however, it is used only as a median. This means when a video is easy to encode, a variable bitrate will reduce the bitrate and in turn optimize the bandwidth available. The same goes for sudden difficult to encode video (fast moving motion video) where the bitrate may increase to ensure a decent quality.

If a video stream is difficult to encode, there is a chance the variable bitrate encoder will surpass the available bandwidth. And this could result in potential loss of video (dropped frames or buffered delayed frames). Therefore, constant bitrate settings are most often used for the military space. Variable bitrate encoders do have the option to enact a strict maximum size in order to limit this potential for overflow, but this can cause more strain on the encoder which is unnecessary for most military applications and ultimately the variability of the bitrate is an unnecessary addition to an already complex system.

H.264 also has various profiles or ways in which the encoding is handled – the 3 primary profiles are Baseline, Main and High. Baseline is computationally simple and fast to encode. Main and High add more features (like B-frames) making the resulting compression ratio better, but at the cost of computation time and hence latency.

Time is of the Essence

Compression ratios keep getting better and better, but this always comes at the cost of simplicity and processing time. And anything that becomes more computationally difficult typically takes longer to process.

The process to receive video from a sensor and send it back to ground is quite complex as figure 2 shows. First, raw video is captured by the sensor and sent to a processing board. Video is then sent through a series of decoders to present the data as actual video data to the system.

This actual video data is then sent to the on-board hardware accelerated encoder. The video is then encoded to H.264.

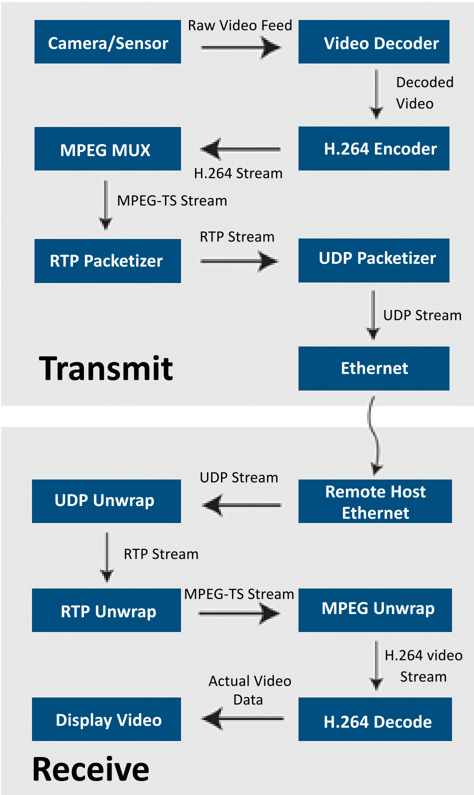

Figure 2:
Block diagram shows the full pipeline of sending video remotely and then displaying at control.

The resulting compressed video is then muxed with audio or any other streams into an MPEG-TS2 container. This muxed container then is split into RTP (Real-time Transport Protocol) packets. The RTP packets are then split into UDP packets. The UDP packets are sent out to Ethernet. The UDP packets are then sent back to a remote location in a method solely dependent on the environment.

Back at ground, the UDP packets are captured and the reverse procedure occurs (UDP → RTP → MPEGTS).The MPEGTS stream must then be unwrapped and the H.264 video is decoded into a presentable format. This now presentable pixel format is rendered to the screen to be viewed. All these steps are expected to be performed as fast as possible, preferably (but unrealistically) with no latency.

The main "time" costs in the above system are the time to encode, the time to relay back to ground and the time to decode. As previously mentioned, the time to relay back to ground is solely dependent on the environment of the aircraft and the means to send back the data.

The encoding time is manageable based on hardware and settings. A dedicated H.264 hardware encoder is an absolute requirement to properly meet user expectations. CPUs are too slow due to their fundamentally serial approach. And the GPGPU approach is possible but is ultimately wasteful in power and still not as fast as a dedicated encoder. The only way to have both a high compression ratio with guaranteed speed is to have dedicated hardware to perform the tasks.

With a dedicated encoder, the time to encode and wrap into a streaming format is minimized. As mentioned before, the real user of time is the process of sending the encoded data back to ground. But this may be difficult to improve upon and depends on the communication method between the capture and the display sites.

This encoded video must also be decoded on the receiving side. This means the encoded stream must be extracted from the MPEG-TS mux and decoded back to regular pixel data before being displayed. This is essentially reversing the whole encoding process already performed and in turn takes a similar amount of time to the encoding process. Hence, like the encoding process, this too must be optimized as much as possible to ensure a timely display of remote video. Many of the latest GPUs (graphics processing units) have built-in decoders that display applications can use.

The Reality

Video needs to be sent back to control in a timely fashion. Otherwise, the entire mission could be compromised. The reality of the situation is the video being shown at ground will always be "late" when compared to what is actually happening. The goal is to limit how "late" the video is by as much as possible. If the video is too "late", it is ultimately useless and decisions made back at ground are being made with a faulty reality. The result of these decisions based on incorrect data can be devastating.

The dedicated encoder on board must then be optimized for the specific environment. As previously mentioned, the strict bandwidth limitations is the real challenge. And with these strict bandwidth limitations, the video will never look perfect. And if it is a high motion video, it may not even be close. Hence if the video isn't time sensitive (monitoring the video instead of controlling back at ground), certain optimizations can take place at the encoding layer which adds extra time (in the magnitude of ms) but potentially ensuring better quality.

As previously mentioned, implementing a high profile encode may take a few extra milliseconds (5-20ms), but may be the necessary addition to make the video useable. Filters (like temporal motion filters) can also be applied which analyze the video frames for items like motion and sharp contrast and then computationally alleviate these issues.

Again, any addition to the pipeline will always add some degree of extra time.

The Tradeoff

Encoded video will always have the tradeoff between size, quality, power consumption and speed. The smaller the size, the worse the video will look. And the more processing that needs to be performed to improve this quality adds to the final latency and power. Further developments in the encoding field will continue to improve this situation, but this tradeoff will always be an inherent issue.

For now and for some time H.264 is the standard for compressing video and military applications must work within both compression level and the bandwidth limitations to ensure successful implementation.

Alleviations and Implementation

Based on implementation and application need, there is the possibility to implement a system which highly alleviates this tradeoff simply via brute force with multiple streams. If the encoding product has multiple dedicated encoders on board which can be customized individually and bandwidth availability permits, then both encoders can be utilized to meet all needs.

For example, one encoded stream can be set to encode at a the full 30 frames per second at low quality ensuring every frame is sent timely and with no frame loss. And the other encoded stream can be set to encode at a higher quality but at a lower frame rate (5fps for example) and with more leniency for buffering. This way a video feed can be analyzed in real time with no frame loss for live use and the second higher quality stream can be recorded (locally) or referenced live if there is a sudden need for high visual fidelity.

A few card and box level encoders with this dual encoding capability are available in the market today. One such example of a product is the Tech Source Condor VC 100x (figure 3). This rugged XMC card is highly configurable and is an extremely low power hardware encoder that is used in several current programs to achieve the low encoding latency with very high efficiency. With two independent encoders, it achieves the dual mode configuration that is discussed here.

Another configuration is to utilize a combination of raw and encoded video utilizing the same feed. For example in a manned aircraft, the raw video can be captured and analyzed locally for live motion tracking and radar display while the encoded version can be sent back to ground for mission control, recording and analysis.

Priorities and requirements must first be properly evaluated in order to implement a successful and effective system.

Figure 3

The VC100x XMC H.264 encoder allows video to be sent remotely and in a timely fashion.

Conclusion

There will always be latency in sending video from aircraft back to ground. There is simply no way around this reality. The solution is to alleviate as many bottlenecks as possible in the pipeline prior to implementation. A dedicated hardware encoder is an absolute necessity to limit this delay. Only a dedicated customizable encoder optimized for high efficiency with low power consumption can be used in both manned and unmanned avionics streaming.

Tech Source
Altamonte Springs, FL.
(407) 262-7100.
www.techsource.com