H.264 Considerations

Nine Key Considerations When Selecting a Video Compression Algorithm

With the emergence of H.264 and other types of compression and the growing abundance of sales and marketing claims, it is increasingly difficult to determine what type of compression is right for your application. To make sense of all the competing claims, you first need to understand the basics of how compression works.

Types of Compression

There are two basic types of compression: frame-by-frame compression and temporal compression. Frame-by-frame compression takes a full picture for each frame, compresses it, and sends each picture, one after another, in a stream. The most popular frame-by-frame compression is Motion JPEG (MJPEG). It is widely used because it produces the highest quality video, is very simple to decode and is non-proprietary. MJPEG's Achilles' heel, however, is that to achieve high image quality on every video frame, it produces relatively large files, which means it requires more bandwidth to transmit and more storage to record.

Temporal compression algorithms were developed to stream video using less bandwidth by trading off image quality, and in many applications, bandwidth and storage consistency. In simple terms, temporal compression works like this: whereas MJPEG will take 50 "full" pictures to make 5 seconds of 10 frame-per-second (fps) video (5 seconds X 10 fps = 50), temporal compression algorithms will take 10 or fewer "full" pictures, usually referred to as Key Frames, to reproduce 5 seconds of 10 fps temporal compression video. This is done by using mathematical calculations to predict what may change between the Key frames, and then sending only what is changing instead of sending the "full" picture. If there is little or no motion between frames, temporal compressions will simply reproduce the Key Frames ,which in the example above is a 10:50 ratio or 80% more efficient than MJPEG. The efficiency of temporal compression comes with a caveat: bandwidth and storage requirements are highly variable depending on the scene recorded. To illustrate this, think about a camera that is panning, tilting, or zooming: 100% of the scene is changing with each frame, forcing the temporal compression to work overtime to interpret what is happening and then calculate what images to send. The result is a large increase in bandwidth and storage required but with marginal video quality to show for it.

So is temporal compression better than a frame-by-frame compression for your application? That will depend on a variety of considerations, but before we dive into those considerations, let's look at the most promising of the temporal encoding algorithms, H.264.

What is H.264?

There have been volumes written about this so we won't go into much detail, except to say it is the most sophisticated temporal encoding algorithm available in the market today. In some cases, it can deliver images that approach MJPEG quality. It does this by using more sophisticated mathematical algorithms than its temporal predecessors like MPEG-2, MPEG-4 and H.263. These mathematical algorithms are better able to predict motion changes between Key Frames so the video is more likely to be accurate with less blurriness and "blockiness" than the older temporal compression techniques.

There are 8 types of H.264, *referred to as profiles*: *

Profile Type	Typical Applications	H.264 Video Quality	Bandwidth
High Stereo Profile	Two-view stereoscopic 3D video	Highest	High
High 4:4:4 Predictive Profile (Hi444PP)	Ultra High Quality broadcast applications that demand lossless video	Highest	High
High 10 Profile (Hi10P)	Professional applications that use interlaced video	Medium	Medium
High Profile (HiP)	Hi Definition and Megapixel Broadcast and disc storage applications. HD DVD and Blu-ray Disc, use HiP	Medium	Medium
Extended Profile (XP)	Mainstream consumer broadcast and storage applications that require high compression and higher reliability	Medium	Medium
Main Profile (MP)	Mainstream consumer broadcast and storage applications	Medium	Low
Baseline Profile (BP)	Low-cost applications that require additional error checking	Lowest	Low
Constrained Baseline Profile (CBP)	Low-cost applications like video conferencing and mobile applications	Lowest	Low

* Wikipedia http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC

Each manufacturer customizes its H.264 profile to suit its needs, which means no two H.264 profiles are necessarily created equal. The decision on which to go with is driven primarily by three variables:

1. Desired image quality
2. Available bandwidth/storage
3. Available processing power

Make a note of those variables as they will ultimately impact your decision on which compression methodology is best for your application. Most H.264 camera manufacturers utilize either Baseline, Constrained Baseline, or in some cases, a much lower performance profile with many quality features simply "turned off" because they do not have the processing power in the camera to support the higher quality features. We all know there is no "silver bullet" when it comes to technology. If a compression algorithm uses less storage or processing it has to compromise in other areas, and experts who understand those tradeoffs learn to design around them. Since IQeye cameras have significant on-camera processing power, IQinVision has chosen the higher performance Main Profile for its H.264 encoding, which provides superior video quality for the same bandwidth compared to the Baseline Profile.

Compression Considerations

In order to help determine what compression is right for you, we have developed a list of nine considerations that relate to either the User Requirement or the Video Environment. This article covers the first two, the other seven will be addressed in two subsequent articles.

1

RESOLUTION [User Requirement]

Do you need consistent, high quality video? If the best video quality--that is, forensic or high detail--is paramount, then MJPEG is the right choice since video quality will be constant in most scenes. If you are considering H.264 to accomplish the best video quality, it does allow you to select constant bit rate (CBR) or variable bit rate (VBR) to attempt to achieve this purpose. CBR is essentially a bandwidth throttle and when used, H.264 will sacrifice image quality to limit bandwidth. IQinVision recommends always utilizing VBR for security surveillance applications, otherwise you could turn high quality video into blurry, block images, especially with higher resolution HD/megapixel cameras. With H.264 compression, as resolution increases, image quality can vary substantially. The same holds true for bandwidth variations so make sure your network designers account for the worst case scenarios like challenging scenes where the bandwidth will increase dramatically.

2

FRAME RATE [User Requirement]

What frame rate do you need? This is a very important consideration as the wrong answer can result in a lot of unnecessary expense. In the past, many project specifications were based on the camera's ability, not the customer's needs. Today, more customers are asking. "If I have an incident, what do I want to see?" Let's say a customer wants good images of a person walking down a street, and based on the camera and lens they have, they will be in the field of view for 10 seconds. Recording at 30 fps results in 300 images of that person. Whether that is enough or too little depends on the customer. Predicting bandwidth and storage as frame rate varies with MJPEG is quite simple as the relationship between the two is straightforward. There are many calculators that can help with this, simply multiply the average file size by the number of frames. With H.264 it is much more complex and somewhat counter-intuitive. Since H.264 has to interpret the changes between Key Frames, it is more efficient at higher frame rates because there is less change from frame to frame so the camera works more efficiently. The converse is that with lower frame rates there is more change between Key Frames that will require more bandwidth and storage to process, and more importantly, will degrade image quality. While image degradation will vary depending on how a manufacturer has implemented their version of H.264, a basic rule of thumb is that H.264 may be appropriate for applications that require 15 fps or higher depending on other User Requirement and Video Environment considerations.

In closing, here are some other important rules of thumb for the first two decision considerations discussed in this article:

- *Select H.264 when your requirements dictate that saving bandwidth is more important than consistent image quality or predictability in bandwidth or storage needs.*
- *Select H.264 when your requirements are to record at 15 fps or higher and other User Requirement and Video Environment considerations also support that choice.*
- *Select MJPEG when you need very high quality with consistent and predictable bandwidth and storage.*

Look for a discussion of the following considerations in future articles as well as recommendations based on such considerations for what compression methodology is best for your application:

3. **WEATHER** [Environment]
4. **LIGHTING** [Environment]
5. **SCENE MOTION** [Environment]
6. **OBJECT SPEED** [Environment]
7. **CAMERA MOTION** [Environment]
8. **RECORDING** [Requirement]
9. **LIVE VIEWING** [Requirement]