White paper

Latency in live network video surveillance

Table of contents

1. Introduction

In the network video surveillance context, latency is the time between the instant a frame is captured and the instant that frame is displayed. This is also called end-to-end latency or sensor-to-screen latency. This transporting process includes a long pipeline of steps. In this white paper we will try to dissect these steps. We will first look into those which will affect latency and finally give recommendations in how to reduce latency.

2. What is latency?

The definition of latency depends on the context, where there will be variations in the meaning. In network technology, latency is commonly perceived as the delay between the time a piece of information is sent from the source and the time the same piece of information is received at its final destination.

This paper discusses latency in network video surveillance systems. Here we define latency as the delay from when an image is captured by a camera until it is visible on a video display. There are several stages required in this process: capture, compress, transmit, decompress and display of the image. Each stage adds its own share of delay, which together produces the total delay, which we call end-to-end latency. This end-to-end latency can be divided into 3 major stages impacting the total system latency:

1. Latency introduced by the camera (image processing / encoding latency)
2. Latency introduced by the network (transmission latency)
3. Latency introduced by the receiver side (client buffer, decoder latency, and display latency).

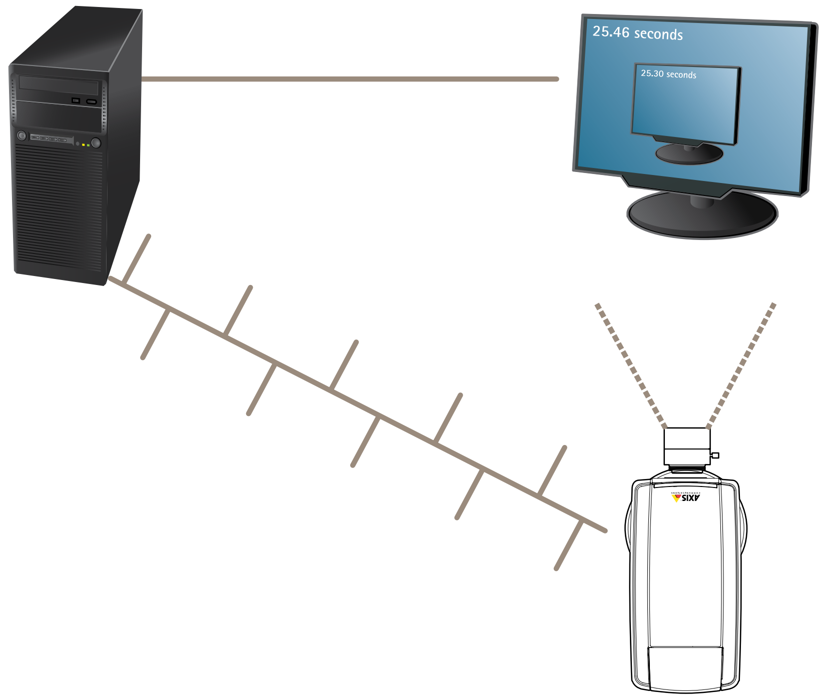
Each of these latencies needs to be considered when designing the video solution in order to meet the latency goal of the video surveillance system.

3. How do we measure latency?

Latency is usually expressed in time units, e.g., seconds or milliseconds (ms). It is very hard to measure exact latency as this will require the clock on the camera and the display device to be synched exactly. One simple way (with reservation for minimum deviation from the exact values) is by using the timestamp overlay text feature. This method measures the end-to-end latency of a video surveillance system that is, the time difference between the capture of one image frame in the lens to when that same frame is rendered on a monitoring device.

Note that this method will produce a possible error of up to one frame interval. The possible error of one frame interval depends on the fact that the timestamps used to calculate the latency are only collected at frame capture. We will therefore only be able to compute the latency with the factor of the frame rate. Hence, if we have a frame rate of 25 fps, we can calculate the latency as a multiple of 40 ms. If we have a frame rate of 1 fps, we can calculate the latency as a multiple of seconds. This method is therefore not recommended for low frame rates.

> Turn on timestamp in the overlay by using (%T:%f)
> Place the camera in an angle so it captures its own live stream output
> Now take snapshots of the live stream output to compare the time difference between the time displayed in original text overlay and the time displayed in the screen loop



From the picture above you can see the time difference is 460 ms - 300 ms which gives us an end-to-end latency at 160 ms.

4. What affects latency?

4.1 Latency in the camera

4.1.1 Capture latency

Let us take a look inside the video camera. Images are made from pixels captured by the camera sensor. The capture frequency of a sensor defines how many exposures the sensor delivers per time unit, i.e. how many frames/ number of images it can capture per minute. Depending on which capture rate you choose you will have different capture latency. By setting the capture rate to 30fps, meaning the sensor will captures one image/frame every 1/30 th of a second, you are introducing a capture latency of 33.3 ms.

4.1.2 Latency during image enhancement

After capturing the raw image, each image frame will go through a pipeline of enhancement processing. These steps, such as de-interlacing, scaling and image rotation, will add latency. The more enhancement you want, the higher the cost in latency in the camera. At the same time the enhancements also affects the total data being produced, leading to effects in the network latency. Below are a few parameters that will affect latency.

Image rotation
Rotation of the video stream to either 90° or 270° degrees adds an additional load to the encoding processor. The pixels will have to be rearranged and buffered before they are sent to the decoder, causing delay.

Resolution
Higher resolution means more pixels for the processor to encode; the increase in processing time for a higher resolution vs a lower resolution is balanced by a faster processing unit in high resolution cameras and thus usual insignificant. But higher resolution does result in more data per frame. i.e., more packets to be transmitted. In a network with limited bandwidth it might lead to delay during transmission which in turn will lead to the need of larger buffer at the receiver side, causing longer latency.

Multiple streams
If more than one kind of stream is requested from the camera (different frame rates or resolutions), the processing of an additional kind of stream will add latency as all streams must be encoded by the same processor.

4.1.3 Compression latency

After the image has been processed it will be encoded to compress the amount of data that need to be transferred. Compression involves one or several mathematical algorithms that remove image data. This takes time depending on the amount of data to process. The delay introduced in this step is called compression latency.

There are three aspects of compression that will affect latency.

Complexity of Compression algorithms
More advanced compression algorithm will produce a higher latency. H.264 is a more advanced compression method than MJPEG, but the difference in latency during encoding is only a matter of a few microseconds. On the other hand, on the decoding site the variation may be bigger. (The H.264 data stream produced by Axis video products requires the decoder to buffer at least one frame, while MJPEG decoding requires no buffer.)

Effectiveness of the compression method
Most common encoding schemes used in Axis cameras are MJPEG and H.264. Both MJPEG and H.264 introduce latency in the camera. H.264 is a compression encoding that, when applied, minimizes the throughput of a video stream to a greater extent than when compared with MJPEG. Which means using H.264 will produce fewer data packets to be sent through the network, unpacked and rendered in the receiver end. This will, of course, have a positive effect on reducing the total latency.

The choice of bitrate
Video compression reduces video data size. However, not all frames will be the same size after compression. Depending on the scene, the compressed date size can vary. In other words, the original compressed data is streams of Variable Bit Rate (VBR), which result in variable bitrate being outputted into the network. One needs to take the constraints of the available network such as bandwidth limitations into consideration.
The bandwidth limitations of a streaming video system usually require regulation of the transmission bit rate. In some encoders, the choice of VBR and Constant Bite Rate (CBR) is presented. By choosing CBR you will guarantee the network receives a limited amount of data so it will not be overloaded, leading to network delay and the need of a larger buffer in the receiver end further on in the system.

In Axis cameras, choosing H.264 will provide you the choice to select CBR or VBR. From firmware 5.60 the choice is between Maximum Bit Rate (MBR) and VBR. However, Axis has always recommended using networked video with VBR where the quality is adapted to scene content in real-time. It is not recommended to always use CBR as a general storage reduction tool or fix for weak network connections, since cameras delivering CBR video may be forced to erase important forensic details in critical situations.

When choosing a compression method one should take all three aspects mentioned above into consideration. On one hand an advanced encoding algorithm will take longer time to encode and decode, on the other hand it will reduce the data volume being sent through the internet, which will in turn shorten transition delays and reduce the size of receiver buffer.

4.1.4 Buffer latency

Because images are handled one frame at a time, only a limited amount of data can be compressed at a time, short-term buffers between the processing stages are sometimes needed. These buffers also contribute to the latency in the camera.

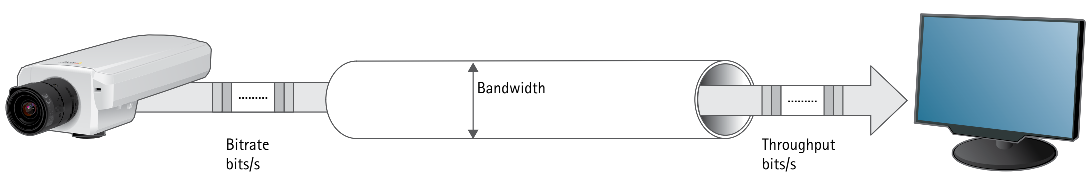
4.1.5 Audio latency

In some cases the video stream is accompanied by audio. The audio encoder needs to wait for a certain amount of samples before a block is available to begin the encoding of audio adding additional delay in the camera side. The sample rate and block size is different in different audio encoding algorithms.

4.2 Latency in the network

After the image is captured, processed and compressed, the video data will travel through a network before it reaches the client side for rendering. To understand how the network will affect latency we need to first understand some basic concepts in video networking, namely the definition of Bandwidth, Throughput and Bitrate. Seen in the picture below if we imagine the link/network between the camera and the monitor to be a pipe, then the Bandwidth is how thick that pipe is. The Throughput measures how much data actually comes through the pipe per time unit. The Bitrate is how much data is being carried out to the pipe per time unit.

Basic concepts: Bandwidth, Throughput and Bitrate



Network latency is proportional to bitrate and inversely proportional to bandwidth.

The Bandwidth is how much data the network between the camera and the monitor can potentially handle. It is the maximum capability of your link. It depends on the length and the infrastructure of the link, i.e. switches, routers, cables, proxies, etc. If we increase the capacity of the network, more data will be able to pass through, leading to lower latency.

The Throughput is the actual achieved speed of your data transfer. It depends on if you are sharing the link with others. It depends also on the electromagnetic interference on the cables in the link, as well as the QoS configured on the ports that may cap throughput.

Bitrate is the number of data in bits that are processed per unit of time. In video surveillance, bitrate is defined by the amount of data generated by the camera to send through the network per unit of time. The bitrate depends on many factors; it depends very much on the filmed scene, the processing done in the camera and the video stream settings. When the camera is producing more data to be transmitted, you can expect higher network latency if the bandwidth is limited.

The total latency in the network depends on three major factors. The infrastructure of the link between the camera and the video viewing device which determines the bandwidth, the amount of data produced of the camera which determines the bitrate, and the choice of transmission protocol.

4.2.1 The infrastructure

The network is the most unpredictable source of the end-to-end latency. Switches, routers, cables, proxies...everything in the network between senders to receiver will affect the total end-to-end latency. In a local area network, the latency in the network could be only a few ms, which is insignificantly low and can be ignored. However, if the video stream is to be transmitted over the internet with unspecified routes, the network latency will be hard to predict and could in many cases be the main contributor to the end-to-end latency.

With careful network management and bandwidth allocation, the unpredictable factors in the network latency can become more predictable. The link between the camera and view device needs to have a guaranteed throughput. In a LAN (Local Area Network), this could be done by making sure there are as few hops as possible in the link. The link should not by shared with other traffic such as Voice over IP (VoIP) or other protocols that will be prioritized over video by default, or other demanding services that will overload the link. If the link is over a WAN (Wide area network), the QoS needs to be guaranteed in each hop, i.e. routers and switches. This can also be accomplished by leasing a point-to-point route through your local internet provider. Below is a list of things that affect the throughput.

Configurable	Cost-related
Overhead of the package (protocol-dependent example: V-LAN header)	Processor speed and port buffer of the switches and routers
Proxies and firewalls	Cable type or wireless
QoS of each link in the whole route	
Burst mode or not. Enabled => higher speed	
MTU - the size of the video payload	

4.2.2 Video stream data amount

As mentioned in the previous chapter, which image enhancement and compression method is selected in the camera also affects the network latency, since they affect the amount of video data being produced. Sending a smaller amount of data will clearly take less time.

4.2.3 The transmission protocols

The video frames from the camera are passed on to a transport protocol application, usually RTP or HTTP. Transmission to the rendering client is done over an IP network. The transmission is either through reliable TCP, which is a connection-oriented protocol, with re-transmission for lost data packets or though UDP, which is a simpler protocol that does not guarantee delivery and provides no facility for retransmission of lost packets.

In Axis cameras, there are the following options when choosing to encapsulate the encoded data stream for transmission. Recommendations on which encapsulation to use are also listed in the table.

Topology	Recommended Axis video packets encapsulation modes
LAN / fewer hops and directly managed nodes	MJPEG / HTTP / TCP
LAN / fewer hops and directly managed nodes	H.264 or MJPEG / RTP / RTSP / HTTP / TCP
LAN / fewer hops and directly managed nodes	H.264 or MJPEG / RTP / RTSP / TCP
WAN / several hops where you do not have full control over the nodes	H.264 or MJPEG / RTP / Unicast / UDP
WAN / several hops where you do not have full control over the nodes	H.264 or MJPEG / RTP / Multicast / UDP

Normally it will take longer to transport a packet using TCP than through UDP, because of the extra connection setup, the acknowledgement messages, and re-transition of packages when a loss is detected. On the other hand, with UDP the user will experience artefacts or interruption in the video stream when packets are lost on the way. TCP will yield jitter on packet loss, UDP will yield artefacts and/ or interruptions on packet loss. If data loss and temporary quality degradation is acceptable, the UDP could be a choice for networks with low bandwidth.

If you are using TCP, there will be more packets to be sent; to support this you need a better bandwidth. If you know there is a lot of congestion in the network, then select UDP as your transmission protocol. Since packet loss is accepted, at the same time it will also lead to packet loss resulting in lower quality of image.

4.3 Latency on the Client side

After the video is received on the client side of the video system, it is unpacked, reordered and decoded and a media player is used to render the video. Each step also contributes to the total latency generated on the client side. The computer itself plays an important role in the overall client side latency. The CPU capacity, the operative system, the network card and graphic card also affects the outcome of latency. Usually MJPEG is the method with lowest decoding and display latency because data can be drawn on screen as they arrive because there are no time codes. H.264 and other video compression standard assign time codes to each picture and require them to be rendered accordingly.

4.3.1 Play-out buffer

Real networks are often very large and complicated, with bursting traffic behavior and packets arriving in different orders. To compensate for variations introduced by network transport, a buffer is used on the client side. It makes sure that the packets get into the right order and buffers enough data so the decoder doesn't "starve"; uniformed frame rate is displayed in the viewer. This buffer is often called play-out buffer or jitter buffer. When used, this buffer contributes to relatively high latency in the client side.

It is important to stress that different viewer applications have different play-out buffer size. With VLC the default play-out-buffer is set to 20 ms, Quicktime 5 sec. In most viewers, the buffer size could be changed. But it is important to keep in mind that reducing the buffer will increase jitter. The user needs to find the balance between jitter and tolerable latency.

4.3.2 Audio buffer

In playback, audio streaming is also more sensitive to hiccups or delays than video streaming. A single delayed audio packet generates an annoying crack in the soundtrack. The audio has to be lip-synchronized with the video. This requires the need to set up a large play-out buffer when video is accompanied with audio. This will of course increase the end-to-end latency.

4.3.3 Decompression

The next source of latency is the time required for the decompression process. Depending on what encoding method is used, the decoding will vary in time. The decoding latency depends very much on what hardware decoder support is present in the graphic card. It is usually faster to decode in hardware than in software. Generally, H.264 is harder to decode than MJPEG. When it comes to decoding in H.264, the latency also depends on the profile chosen in the encoding phase. Base is the easiest to decode; main and high will take longer. The H.264 data stream produced by Axis video products requires the decoder to buffer at least one frame.

4.3.4 Display device refresh rate

The display device's refresh frequency also plays an important role. For TV the refresh rate could be up to 1 sec. For computer monitor frames the refresh rate is around 14-15 ms, whereas special gaming monitors have a refresh rate of 4-5 ms.

5. Reducing latency

It is important to keep in mind that designing a system to meet low-latency goals will require other tradeoffs. The user needs to decide what the acceptable latency is and find the optimum balance between video quality and cost of the surveillance system. It is either decrease video quality or invest in better hardware and software solutions. With this in mind, there are a few simple recommendations to reduce the end-to-end latency.

5.1 Camera side

Resolution
Choose a lower resolution if possible. Higher resolution implies more data to be encoded, this may lead to higher latency.

Enhancements
Image enhancements (rotating, de-interlacing, scaling, etc.) may also add latency. Reduce these enhancement will reduce latency.

Encoding
Make sure the encoder provides the level of control over latency that your system requires. There needs to be a balance between the amount of data and the capacity of network infrastructure. If the video is sent through a network with limited bandwidth, choose H.264 as the encoding method. This will lead to lower bitrate due to harder compression. Choose baseline profile if the network can manage the bitrate, as baseline will be easier to encode and decode. Motion JPEG is better from a latency standpoint if the network can handle the ~10 times higher bitrate.

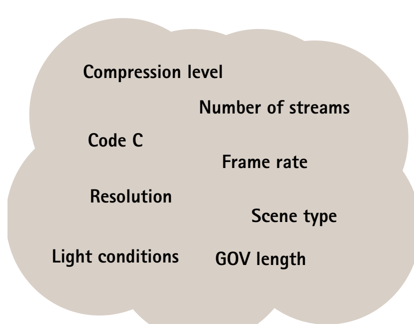
Number of streams
Limit the number of streams from camera with different settings. Each unique combination of settings such as resolution, frame rate and compression will required its own individual encoding process, adding load to the processor, causing delay.

Frame rate
Use as high frame rate as possible. As frames are encoded and decoded one frame at a time the buffers will delay at least one frame. With higher frame rates the delays caused in buffers will be reduced. For a stream with 30 fps, each frame will take 1/30 of a second to capture. We can then expect a latency of 33 ms in buffers. For 25 fps we will have a delay of 40 ms.

Audio
Audio needs a higher playback buffer leading to longer latency if lip-synch is required with the video. By removing audio, you will remove an appreciable amount of latency.

Bitrate
To reduce the latency in the camera we need to reduce the amount of data being generated, and outputted onto the link to be transferred to the other end for viewing. All boils down to the bitrate the camera generate.

Factors that affect bitrate



5.2 Network

Many of the recommendations mentioned above are aimed at limiting the total data volume being sent though the network. In most cases, a limited network is the largest contributor to the end-to-end latency. If the network has a high capacity then many of the above recommendations are not needed.

Make sure that your network has a good quality of service, and that all the hops within the network are configured to suite your video demand. Make sure that your bitrate over the network is guaranteed to be able to deliver the data output from the camera.

5.3 Client side

There is much to be done on the client side to reduce the end-to-end latency. Improvement in the client side will make the most impact on the total latency.

Processor and graphic card
The CPU plays a central role in the client side latency. Make sure that you have a good processor with enough capacity to process the video stream and handle other requests simultaneously. Make sure you have a good graphic card updated with the latest firmware with good support for decoding.

Viewer/VMS
Selection of viewer: make sure that your viewer doesn't have an unnecessarily long play-out buffer. If it does, try to change it. Some viewers will have up to a few seconds of buffer. The video buffer compensates for variations introduced by network transport. The buffered frames are decoded and played out at a constant time interval, achieving a steady video stream. Use a sufficiently sized play-out buffer in the media player on the receiver side to control latency, but be aware of the cost of video jitter.

In AXIS Media Control, you have the choice in increase or decrease this buffer to find the optimal value between the trade-off of Jitter and latency.

Display
Use display with as short refresh rate as possible. Another important step in getting a more pleasant live view is to adjust the screen frequency to a multiple of the capture frame rate of the camera. An example would be 60Hz for 30 fps mode or 50 Hz for 25 fps mode. However, this does not affect the latency. Be sure to keep the graphic card's driver updated.

6. Conclusion

The process of live streaming in IP video surveillance is capturing in the camera's device, packaging and transporting through the network and unpacking in the receiver to display. Each of these steps can add more or less latency.

On the camera side the process is largely a shooting, enhance processing and compression and packaging. Roughly speaking, each frame takes a time gap of 1 / 30s exposure. It then requires a millisecond (ms) to scale and encode the image. The encoded image is then chopped up and packaged for the network. Finally, an image is outputted onto the network every 33 ms. The time it takes for this process in the camera is under 50 ms. It varies slightly depending on whether the frame is an I- or P-frame and which camera it is (PTZ excluded). The variation is typically around 10 microseconds. In the big picture of end-to-end latency, the camera only contributes to a small fraction of the total latency.

The network latency can be very large or very small. It is the most unpredictable factor in the end-to-end latency equation. Invest in a good network between the cameras and the client makes the network latency more predictable. Network latency depends very much on the data to bandwidth ratio. Although a lot of configuration in the camera can be made to reduce the latency, the main goal of these configurations is to reduce the amount of data generated, hence reducing packets in the network.

In the client side, data is received and buffered to be sorted and queued out to the graphics card and monitor. The receiving buffer in the client is the part that affects the latency the most, even up to several seconds. With a big buffer the chance of jerky video stream is reduced; video will be able to play evenly. However, that comes with a cost of added latency. A small buffer holds "fresh pictures" with short latency, but risk of jerky video stream.

To reduce latency is always a question of cost. Reduce the quality of the video or invest in a good network and good client side hardware and software. Usually the first choice is not preferred. Focusing in improvement in the two later choices will be a better return of investment in the context of latency.

63380/EN/R1/1504

About Axis Communications

Axis offers intelligent security solutions that enable a smarter, safer world. As the global market leader in network video, Axis is driving the industry by continually launching innovative network products based on an open platform - delivering high value to its customers and carried through a global partner network. Axis has long-term relationships with partners and provides them with knowledge and ground-breaking network products in existing and new markets.

Axis has more than 1,900 dedicated employees in more than 40 countries around the world, supported by a network of over 75,000 partners across 179 countries. Founded in 1984, Axis is a Sweden-based company listed on NASDAQ Stockholm under the ticker AXIS.

For more information about Axis, please visit our website www.axis.com.

AXIS®
COMMUNICATIONS