WHEN SURVEILLANCE MEETS INTELLIGENCE

AI-powered Analytics in Video Management Systems – from Theory to Practice

Abstract

A Video Management System (VMS) refers to a software platform that enables the handling of all aspects of deployed cameras. VMS platforms are typically used for security and surveillance applications, although nowadays, with the spread of IP cameras their use extends to other application domains.

In recent years with the evolution of machine learning, the use of artificial intelligence (AI) for video analytics is surging. AI-based video analytics can process the video content to deliver insights and extend the possibilities of the video management system (VMS) beyond its traditional use. This creates a perpetual motion of growing demand for computational capacity that in turn inspires the evolution of more applications that need even more AI computational resources.

This document is aimed at providing practical guidelines for system designers on how to add AI into existing systems as well as designing new systems with AI.

In the coming sections we shall address some of the main KPIs that are of concern in the various building blocks of a VMS and attempt to provide practical guidelines for how to determine the main properties to optimize those KPIs.

Contents

1. What is a VMS?

A Video Management System (VMS) refers to a software platform that enables the handling of all aspects of deployed cameras. VMS platforms are typically used for security and surveillance applications, although nowadays, with the spread of IP cameras their use extends to other application domains.

In security and surveillance applications, VMS platforms are often used to monitor large areas such as commercial buildings, industrial sites, and public spaces. These systems can be configured to detect motion or other events, trigger alarms, and record video footage for later review. The footage may be reviewed by security personnel in real-time or offline, to investigate incidents or identify potential security risks.

VMS platforms typically consist of several components, including video acquisition, streaming over networking infrastructure, video storage and analysis and display endpoints. Each of these components can be applied at one or more locations along the processing pipeline as will be described. The video acquisition devices are typically IP cameras. The network hardware and other hardware elements that are used in the VMS ecosystem are typically compute boxes, sometime called gateways, that have a combination of components like connectivity hardware for reception and transmission across a network, compute elements for analyzing the video content, disks for storing and retrieving the contents and display walls for observation by human operators.

The software applications used in VMS platforms vary depending on the specific system being used. Some VMS platforms provide basic video recording and playback capabilities, while others include more advanced features such as facial recognition, license plate recognition, and object tracking. Some VMS platforms may also integrate with other security systems, such as access control or intrusion detection systems, to provide a more comprehensive security solution.

In addition to security and surveillance applications, VMS platforms may also be used in other industries such as transportation, industrial automation, retail and even healthcare. In transportation, VMS platforms may be used to monitor traffic flow or ensure safety on public transportation systems. Industrial automation typical usage of such platforms is for non-intrusive process control and quality assurance and in healthcare it may be used to monitor patients in medical clinics.

Overall, VMS platforms provide a flexible and scalable solution for managing different scales of camera deployments. With the ability to integrate with other security and surveillance systems, VMS platforms can help provide a more comprehensive security solution and enhance situational awareness for security personnel.

2. VMS and AI

In recent years with the evolution of machine learning, the use of artificial intelligence (AI) for video analytics is surging. AI-based video analytics can process the video content to deliver insights and extend the possibilities of the VMS system beyond its traditional use. This creates a perpetual motion of growing demand for computational capacity that in turn inspires the evolution of more applications that need even more AI computational resources.

The use of AI is not limited to a single task or a specific purpose and is becoming a dominant building block in all stages of the processing pipeline. However, in the different stages it serves a different purpose, addressing a different property of the overall system.

AI-based analytics is a game changer in this domain and can be applied much more than it has been used in the past. With advances in AI capacity and compute power, AI is no longer the bottleneck that needs to be used sparsely, allowing to optimize the system KPIs and set new goals, outperforming old deployment standards.

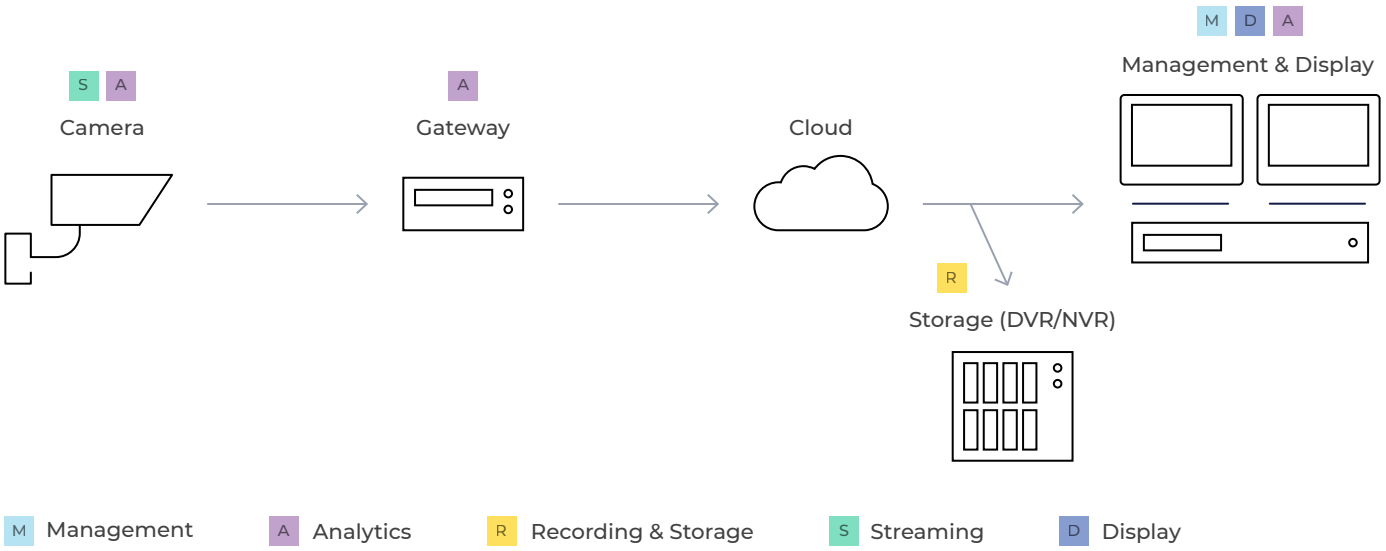

Figure 1: *Example Deployment Setup*

The motivation behind the use of analytics along the processing pipeline may vary. More specifically, one can consider a typical deployment pipeline going from a camera to a gateway, to a Digital Video Recorder (DVR) or Network Video Recorder (NVR) for storage and then fed to some security room with a display wall as illustrated in Figure 1. In such a setup, analytics may provide several benefits as follows:

1. **Improve network utilization:** whether AI-analytics is camera attached, i.e., is handled by the camera, or gateway-bound, it can be used to lower per-channel bandwidth by streaming relevant content only.
2. **Lower latency:** gateway-bound analytics can also serve to reduce overall latency by offloading central processing, in such cases in which the central unit is overloaded with multiple requests that are queueing up to be processed.

3. **Improve storage space utilization:** AI-analytics bound to the storage entity, i.e., close to where data is stored and retrieved, can serve to improve the utilization of the storage space, eliminating the need to store unnecessary content, and better controlling the access bandwidth to the storage elements.
4. **Enhanced safety:** when applied in the context of the management entity, AI-based analytics can serve for spotting the relevant regions of interest, selecting camera sources that trigger events, drawing the operator's attention or trigger manual intervention. It may as well auto-trigger actions of end-point actuators, for example to grant or prevent entrance in an access control implementation.

3. Important Properties of a VMS Deployment

VMS is a general term for a wide range of system deployments that vary in functionality, configuration, scale and purpose. The following section covers some of the key properties of VMS systems, specifying some of the key aspects that should be considered when designing and deploying AI-powered VMS.

3.1 Multiple Functions

Complete video management solutions involve multiple functional entities. According to the deployment scenario they may be centralized or de-centralized and therefore not co-located.

These are the fundamental entities of a typical VMS system:

- Management
- Analytics, monitoring & event triggering
- Streaming
- Recording & Storage
- Display

It is important to note that while all the above entities have an important role in a complete VMS deployment, not all of them are mandatory for a viable system. In fact, apart from the management entity, any combination of the others is valid.

3.2 Different configurations

The deployment scheme of cameras is also quite diverse resulting in multiple optional configurations. Each of these configurations implies different settings and different KPIs. We will try to address some typical configuration types.

Some select deployment configurations are shown in the following figure.

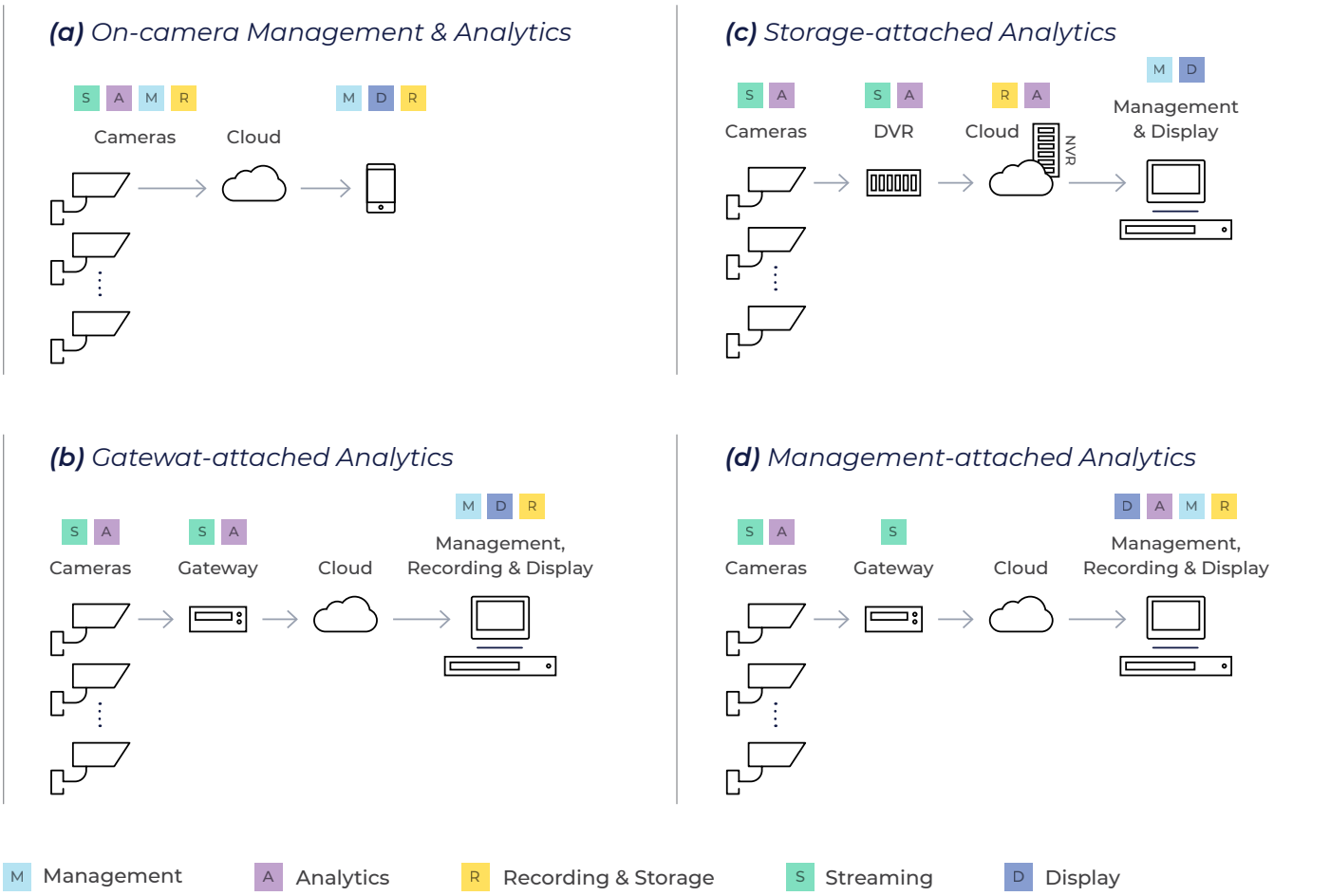

Figure 2: *Common deployment scenarios*

3.3 Multiple scales

Scalability is a fundamental property to be addressed in large scale video management system design. It is inspired by the growing demand to cover wider areas and more scenarios and events, as well as the growing traction of non-intrusive systems.

Scalability is translated into multiple parameters that eventually determine the system requirements and may in turn impact the deployment configuration. At the platform level, the **number of channels** (aka 'streams') is the primary factor. At the stream level, scalability can be attributed to stream properties. In the case of analytics, it is typically the number of **operations per second** per channel which is determined by:

- Resolution, measured in **pixels-per-frame** (spatial-domain scaling)
- Frame rate, measured in **frames-per-second** (time-domain scaling)
- Functionality scaling, measured in **operation-per-pixel**, which in AI processing case, is determined by the neural network model computation intensity.
- Pipeline scaling, measured by the **number of tasks** for the processing pipeline, representing its complexity. For instance, a typical processing flow may involve multi-stage processing of the source stream to reach the desired output. Complex pipelines are beyond the scope of this write-up but involve a combination of multiple tasks, scheduled to run in a sequence. These may run sequentially or in parallel and may involve dynamics that are dependent on the specific data observed in the video stream. For example, an Automatic License Plate Recognition (ALPR) pipeline, requires the identification of a vehicle, the identification of the vehicle's license plate, and the recognition of the characters in the license plate.

3.4 Different venues

The large and diverse nature of VMS deployment scenarios is derived from the different venues in which cameras are deployed.

Deployment	Unmanaged	Managed
Scale (no. of endpoints)	Small (up to 20)	Medium-large (over 50)
Management	De-centralized	Centralized
Environment	Indoor	Indoor / outdoor / mixed
Functions / usage	Typically single function	Mostly multiple use cases Allows additional services

Due to the above properties, we can expect that larger scale deployments are more likely to be managed and that is the focus of the content from this point onwards.

3.5 Building blocks / components

The following tables summarize the entities as described above, and the different deployment configurations, along with their main functions, most important resource they require and the main KPIs to optimize to best serve their purpose.

Function	Main Resource	Main functions	KPIs (acronym) [units]
Management	Connectivity	Command & control	Round trip time (RTT) [s]
Analysis	Compute	Event triggering Indexing	Per-pixel capacity [TOPS/pix] Total capacity [TOPS]
Streaming	Bandwidth	Encoding / decoding	Bandwidth [Mbps] Transcoding rate [Mbps]
Recording & Storage	Storage	Store & retrieve	Recording time [hr] Access speed [Gbps]
Display	Graphics	Multi-frame display	Rendering speed [Mbps]

Table 1: *VMS entities and main KPIs*

Configuration	A	B	C	D
Description/ purpose	Camera attached	Standalone analytics gateway	Storage server / NVR / DVR	Management & Display server
CPU	ARM A-class	Atom / Celeron class x86	Core class I5 – i7	Xeon class
GPU	No	No	No	Yes
AI (TOPS)	5-10	20-40	50-100	100-200
Typical no. of channels	1-2	4-8	16-32	64-128
Typical end-product example	Outdoor camera for traffic monitoring	Camera pole	Small-medium store	Airport perimeter security
Typical AI application	ALPR	Object detection	Metadata	Re-ID

Table 2: *Typical deployment configurations*

4. Design Guidelines

This section will address practical aspects of design and deployment of analytics in each of the formerly mentioned entities in a possible VMS deployment. As described, the needs of the various deployment scenarios may vary according to venue, scale and other constraints imposed by the specific case. Consequently, some of the decisions may vary respectively.

4.1 Streaming entity

In any camera deployment, the source data is the raw video as acquired by the endpoints. This raw data, if not processed, imposes very demanding bandwidth requirements that need to be eventually processed, transferred, stored and retrieved.

The streaming entity serves as the data plane of the VMS, primarily purposed to manipulate the raw data, and ensure its delivery from one point to another along the deployment infrastructure.

The information that the raw data holds, however, is the actual purpose of deploying all these endpoints in the first place. With the application of artificial intelligence, the amount of information that a video stream carries can be delivered in much less bandwidth than the raw video; let alone insights drawn from that information and actions taken in response to specific events.

Furthermore, along with the meaningful data comes unnecessary data content which could be harmful in some cases. It may, unintentionally yet unnecessarily, include information that is not relevant for the subject matter and could potentially be an impediment to people's privacy.

Equipping the streaming entity with the capability to identify the content of the stream both in the spatial, as well as the temporal domain, and determine in real-time what part of the raw data is unnecessary and what part should be delivered, when and how, allows potential reduction in the average bandwidth consumed and better control over personal identifiable information carried across the system.

In the context of the streaming entity, AI gives the opportunity to determine **what** content is being transferred and **when**.

Including AI in the streaming entity may allow few possible configurations and possible intelligence models that may be applied to them:

Mode	Description	Intelligence	Computational intensity implications
Continuous	All content is streamed, unwanted content removed on the fly	→ Detect objects → Segment instances → Anonymize based on boxes or masks	→ All frames processed on the fly → Minimal latency
Filtered	Store & forward allowing time-domain analysis for deeper exploration	→ Removal of no activity → Frame rate adjustment according to activity level → Re-identification across video segments and/or across different video sources → Summarization and frame skipping → Inpainting of removed segments for pleasing output	→ May operate on groups of frames → Can apply only part of the processing pipeline to key frames → Other intense tasks run only on relevant frames → Higher allowable latency → May include additional metadata based on time-domain (e.g. tracking)
Triggered	Streaming only upon predefined events	→ Entry to / exit from restricted areas → Object counting → Object density	→ Dynamic processing based on activity intensity → Can trade-off compute intensity and triggering quality (miss / false thresholds; latency)

The key factors when introducing intelligence into the streaming entity are:

→ No. of streams / channels [#]

→ Per channel frame rate [fps]

→ Encoding capacity [bps]

→ Latency [s]

→ Bandwidth [bps]

More specifically, a typical 4K30 stream, amounts to ~7.5 Gbps of raw data. If encoded without data analysis it may reach up to 45 MBps with H.264 (or ~22 Mbps with H.265). By introducing analysis of the stream, the number of streams that can fit into a given bandwidth can be increased, or alternatively, a gateway box with a given encoding capacity can handle more streams simultaneously.

For example, in a typical deployment, with proper video coverage, >90% of the video has no meaningful information, which means that only <10% of the video should be streamed. Furthermore, when important

information does appear, not all the cameras will capture that important information simultaneously. Almost all areas are covered with 2-4 overlapping zones thereby reducing the simultaneous nature of events by at least 50%. An even more advanced solution, that applies AI also for spatial reasoning, may recover more bandwidth if delivering only the active region of the frame which may further reduce the input bandwidth by a factor of 2-4.

In such a case applying proper analytics can handle x10-20 more streams compared to a similar scenario in which full frame is uploaded. Such an approach may save the expense for upgrading the connectivity infrastructure due to the bandwidth demand, resulting in ~$20-30 saving per camera per month (50-70% saving of the monthly cost). In a 10,000 camera deployment, this amounts to >$2M saving per year.

The capacity of the analytics is obviously a function of the analytics deployed. It may vary widely depending on the resolution, frame rate and number of stages in the processing pipeline.

A high accuracy state-of-the-art object detector running at Full HD input resolution, and 30 fps may require 1-3 Tera Operations per Second (TOPS) per stream (~x10s Kops/pix/stream is a reasonable figure of merit). Typically, a multi-stage operation is applied, requiring higher peak capacity to address real-life use case[1].

Design Rules

- Input Bandwidth constraint: $N \cdot W \cdot H \cdot R \leq$ *Camera connection bandwidth*
- Output Bandwidth constraint: $N \cdot \sigma \cdot W \cdot H \cdot R \leq$ *Downstream link bandwidth*
- Video Encoding constraint: $N \cdot W \cdot H \cdot R \leq$ Encoder bandwidth
- AI capacity constraint[2]: $N \cdot R \cdot [W \cdot H \cdot C_1 + N_{ROI} \cdot W_{ROI} \cdot C_2] \leq$ Available TOPS

N – number of streams

W – frame width (pixels)

H – frame height (pixels)

R – frame rate (fps)

σ - compression ratio (%)

C_1, C_2 – AI workload compute capacity of the 1st, 2nd stage in the pipeline respectively (ops / pixel)

N_{ROI} – Number of regions of interest (ROI)

W_{ROI}, H_{ROI} – Average width & height of an ROI box (pixels)

1 Refer to Hailo Model Explorer for further details about model compute requirements.

2 This equation assumes a typical 2-stage pipeline for analytics (it can easily be extended for more advanced pipelines with additional stages)

4.2 Analytics entity

The analytics entity is the most recent entrant to traditional VMS solutions, emerging from the rapid evolution in this field. In terms of possible deployment configurations, the analytics entity can be included either as an add-on software component in one or more of the existing nodes, or as a separate hardware processing component.

In the former case, the analytics relies on the hosting hardware platform and its capabilities, while in the latter a new type of node is introduced. This node, which is sometimes called an AI-box or an AI-gateway, enables adding more compute power with the purpose of offloading the next stage in the pipeline thereby improving the overall system utilization.

In spite of the additional cost of another piece of hardware, In many cases the introduction of such a gateway improves the overall system capacity resulting in a better total cost of ownership (which can be quantified as the cost per stream for the overall system). In other cases, it is the most practical approach for upgrading an existing installed base without replacing or rewiring existing nodes.

The analytics entity is AI-centric and gives the opportunity to trigger events requiring further action only when needed, thereby lowering system load

The main factors that govern the properties of the analytics component are the following:

- Per channel frame rate (fps)
- Decoding capacity (pixels/sec)
- Event rate (events/sec)
- Latency (sec)
- Accuracy (% miss-rate; % false alarms)

Design Rules

- AI capacity constraint: $Actual\ TOPS = N \cdot W \cdot H \cdot C \cdot R \leq Available\ TOPS$
- Number of streams constraint: $N = min\left(\frac{D}{B}, \frac{Available\ TOPS}{W \cdot H \cdot C \cdot R}\right)$

N – number of streams

W – frame width (pixels)

H – frame height (pixels)

C – AI workload compute intensity (ops/pixel)

R – frame rate (fps)

D – Decoder bandwidth (bits /sec)

B – Stream bandwidth (bits / sec)

For instance, an analytics box which addresses simultaneously 32 channels of FHD at 30 fps will require an actual AI capacity of ~50 TOPS (assuming 1.5 TOPS per channel for analytics).

This also implies a decoding capacity that can support 32 channels. An H.264 encoded stream will typically reach ~10 Mbps.

A platform supporting 320 Mbps decoding bandwidth is applicable in this case.

If we wish to consider 4K streaming, the analytics will scale nearly linearly while decoding ratio scaling is sub-linear and in the case of 4K the stream bandwidth will typically reach ~50Mbps.

In this case, the same platform with 320 Mbps will be able to handle only ~6 channels, even though the analytics alone can handle more than that.

(Another option, is to enjoy the benefit of a more advanced encoding such as H.265 which provides a x2 factor on the encoding/decoding performance)

4.3 Recording & Storage entity

Contrary to the analytics function, storage is a longstanding feature of any video deployment, facilitating the ability to record some or all of the streams over time. Its most important property is obviously the storage time and cost for a given capacity.

Other important parameters are the bandwidth into the system, storage access time, and finally the ability to search & retrieve required information.

Applying AI in a storage entity gives rise to multiple use cases such as the ability to dramatically improve the utilization of the available storage resources by making the storage content-aware. It also may be leveraged for on-the-fly indexing enabling metadata to be attached to the video for easier retrieval, and free text search to find events of relevance.

AI supported storage allows to determine **what** content is being stored, and **when** to suspend and resume

To summarize, the key factors of the storage entity include:

- No. of streams
- Per channel frame rate (fps)
- Storage duration (hours)
- Storage space (terabytes)
- Access (Read & Write) speed (sec)
- Search speed (sec)

Design Rules

→ *Actual Bandwidth ~ $N \cdot \sigma(C) \cdot (W \cdot H \cdot R) \leq$ Write speed*

→ *Storage duration ~* $N \cdot \left(\frac{\sigma(C) \cdot (W \cdot H \cdot R)}{\text{Storage space}} \right)$

N – number of streams

W – frame width (pixels)

H – frame height (pixels)

C – AI workload compute intensity (ops/pixel)

R – frame rate (fps)

σ - compression ratio (%)

4.4 Display entity

In most VMS deployments, there is some sort of human interface. It can either be for the initial deployment phase where health checks are conducted by the solution provider, or after the initial deployment, for monitoring. Either as a complementary aid mainly used for debugging purposes or as an integral part of the overall system deployment in the form of display monitor wall in some facility management control room.

Beyond the obvious function that an AI-driven component can add to a display, such as local processing of the currently displayed view for extracting metadata, or tracking of objects, AI can be tasked to manipulate the display, to pop up relevant streams as soon as something meaningful is happening in them, or highlight specific areas in the image while dimming others based on activity.

AI supported display component can be used to determine **what** content is being displayed, and **when** it is displayed

What determines the performance of the display entity are the following factors:

→ Number of displays

→ Input stream resolution (pixels)

→ Display resolution (pixels)

→ Decoding speed (pixels/sec)

Design Rules

$$\rightarrow \frac{M \cdot (D_W \cdot D_H)}{W \cdot H} \cdot R \leq D$$

M – number of display monitors

W – Stream frame width (pixels)

H – Stream frame height (pixels)

M_W – Display monitor width (pixels)

M_H – Display monitor Height (pixels)

R – Refresh rate of display monitor (fps)

D – Decoding speed of video stream (pixels/sec)

5. Further Remarks

While analytics is rapidly paving its way into mainstream applications, there is still a lot of room for innovation. With the acknowledgment that AI can be used to improve both usability and cost of video management systems, new capabilities arise. This may change the way of usage of VMS altogether.

The rapid adoption of large language models (LLMs) in generating and interpreting speech can easily be imagined for multiple obvious needs such as prompt-based searching, summarization and transcription, free-form rule statement and many more.

The recent improvements in AI powered image generation may be used to introduce new level of anonymization, abstract generation of output video feed, in-painting and out-painting of the subject video for a more pleasing display etc.

Not only can these be utilized within existing entities as described throughout this paper, but it may also allow redefining of the core nature of the management entity by harnessing the power of scene understanding to ease the deployment process by system integrators. Auto-segmentation of the scene observed by an installed camera may automatically generate proposals for region aware analytics, self-supervision of ill-positioned camera and may expedite the configuration process altogether.

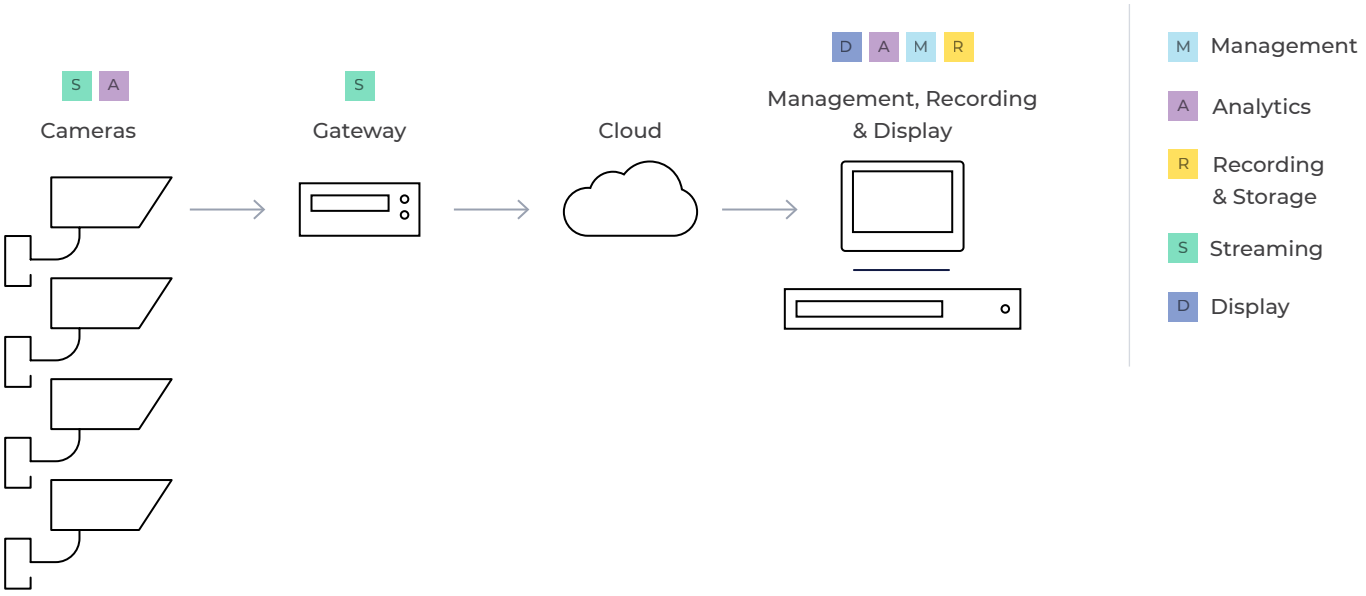

Figure 3: *A management attached deployment would typically require tens of TOPS in the management entity in addition to the camera TOPS*

6. Summary & Conclusions

The emergence of advanced analytics enables gradual transition from existing platforms to new and advanced ones. In this short paper we made an attempt to cover some of the basic principles of adding AI to VMS deployments in order to improve usability and cost-efficiency.

In short, new capabilities inspired by the AI revolution bring about new possibilities both to dramatically improve existing features and to imagine new features. These all require intense AI processing which has now become a necessary resource when building such systems.

The good news is that, thanks to new infrastructure, this resource is not scarce any longer and is readily accessible to solution providers when deploying new systems or upgrading existing ones.

In a fully AI-supported system, which contains intelligence both in the camera and in the management entity (Figure 3), to support all elements of the system, we would expect a minimum of 1-3 TOPS per camera to support streaming and a basic level of analytics, and for 32 channels with FHD resolution at 30 fps, additional 50 TOPS will be required in the VMS server, supporting the management, recording, analytics and display functionality.