Automatically assessing properties of dynamic cameras for camera selection and rapid deployment of video-content-analysis tasks in large-scale ad-hoc networks

Richard J.M. den Hollander, Henri Bouma [1], Jeroen H.C. van Rest, Johan-Martijn ten Hove, Frank B. ter Haar, Gertjan J. Burghouts

TNO, Oude Waalsdorperweg 63, 2597 AK The Hague, The Netherlands

ABSTRACT

Video analytics is essential for managing large quantities of raw data that are produced by video surveillance systems (VSS) for the prevention, repression and investigation of crime and terrorism. Analytics is highly sensitive to changes in the scene, and for changes in the optical chain so a VSS with analytics needs careful configuration and prompt maintenance to avoid false alarms. However, there is a trend from static VSS consisting of fixed CCTV cameras towards more dynamic VSS deployments over public/private multi-organization networks, consisting of a wider variety of visual sensors, including pan-tilt-zoom (PTZ) cameras, body-worn cameras and cameras on moving platforms. This trend will lead to more dynamic scenes and more frequent changes in the optical chain, creating structural problems for analytics. If these problems are not adequately addressed, analytics will not be able to continue to meet end users' developing needs. In this paper, we present a three-part solution for managing the performance of complex analytics deployments. The first part is a register containing meta data describing relevant properties of the optical chain, such as intrinsic and extrinsic calibration, and parameters of the scene such as lighting conditions or measures for scene complexity (e.g. number of people). A second part frequently assesses these parameters in the deployed VSS, stores changes in the register, and signals relevant changes in the setup to the VSS administrator. A third part uses the information in the register to dynamically configure analytics tasks based on VSS operator input. In order to support the feasibility of this solution, we give an overview of related state-of-the-art technologies for autocalibration (self-calibration), scene recognition and lighting estimation in relation to person detection. The presented solution allows for rapid and robust deployment of Video Content Analysis (VCA) tasks in large scale ad-hoc networks.

Keywords: Autocalibration, self-calibration, dynamic sensors, PTZ cameras, video content analysis, body-worn cameras.

1. INTRODUCTION

Video analytics is essential for managing large quantities of raw data that are produced by video surveillance systems (VSS) for the prevention, repression and investigation of crime and terrorism. Procuring and operating video analytics is difficult. Video analytics is highly sensitive for conditions of, and changes in aspects of the scene, of the camera (garbage in equals garbage out) and of the video analytics and processing chain [20]. Fortunately, a lot of knowledge concerning the proper use of video analytics has been gathered – and is still being accumulated – in scientific, industrial, consultancy and end user organizations. They typically employ specialized staff who require up-to-date trainings and specialized tools (e.g. a Rotakin doll) to procure, install, configure, maintain and operate video surveillance systems (VSS) and video analytics. Figure 1 contains some examples of factors that influence the quality of video analytics and corresponding measures to control and mitigation them, grouped by scene, camera and processing chain.

However, there are several trends that influence future needs with regard to VSS: larger areas are being covered and they are to be used in more operational conditions (e.g. threat levels), a move from static VSS consisting of fixed CCTV cameras towards more dynamic VSS deployments over public/private multi-organization networks, and an increasing variety of types of sensors, including pan-tilt-zoom (PTZ) cameras, cameras on moving platforms and body-worn cameras. These trends will lead to more (dynamic) scenes and more frequent changes in the optical chain. If these trends are not adequately

[1] henri.bouma@tno.nl; phone +31 888 66 4054; http://www.tno.nl

R.J.M. den Hollander, H. Bouma, J.H.C. van Rest, J.-M. ten Hove, F.B. ter Haar, G.J. Burghouts, "Automatic assessing properties of dynamic cameras for camera selection and rapid deployment of video content analysis tasks in large-scale ad-hoc networks," Proc. SPIE, Vol. 10441, (2017). http://dx.doi.org/10.1117/12.2268797

and timely addressed, analytics will remain fragile, i.e. video analytics that will fail quickly and often in real-life situations, and will therefore not continue to address end users' developing needs.

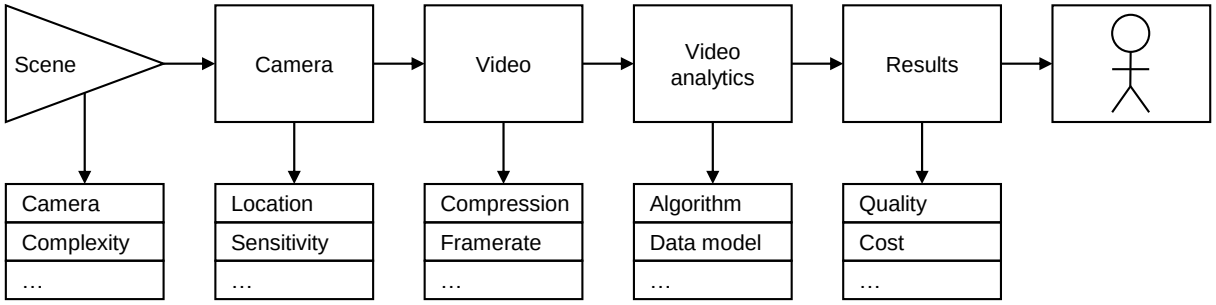

Figure 1: The extended processing chain of video analytics. The camera monitors the scene and generates video footage. This is analyzed by video analytics which generates metadata.

There are several different strategies to deal with this challenge. A first is extensive and frequent manual calibration, which may require introducing objects into the scene, such as large checker-boards with patterns or a Rotakin doll, as well as manually annotating certain points or lines in the footage. If done properly, this method can yield the most accurate results, but visiting all relevant scenes may not be economically feasible and sometimes even dangerous because of safety reasons, or impossible because the sensor has already left the environment.

A second alternative strategy is to try to foresee a larger set of operational circumstances, and to design the analytics so robust that it can handle almost everything. This is very difficult ("garbage in is garbage out"), and is not very realistic for larger surveillance systems in the context of security. Camera properties may drift and larger systems are prone to changes due to maintenance. In addition, it is rather difficult to foresee future use cases, e.g. due to changing threats.

A third strategy, which is investigated in this paper, is to attempt to just as dynamically *apply* this knowledge, at least partially in an automated form, i.e. also *during* the configuration and operation of a VSS, as opposed to only at installation and configuration time. For example, in earlier work we have described how auto calibration could help the automatic finding and tracking of people at large airports [3][13]. We coin this strategy "Managed analytics" (MA), as it is analytics that is semi-automatically managed; the operational performance of the video analytics subsystem(s) is the target to be optimized. Here the primary video analytics for finding and tracking people is supported by secondary (or supporting) video analytics that performs auto-calibration.

In this paper, we present a three-part solution for managing the performance of complex analytics deployments. The first part is a register containing meta data describing relevant properties of the optical chain, such as intrinsic and extrinsic calibration, and parameters of the scene such as lighting conditions or measures for scene complexity (e.g. number of people). A second part frequently assesses these parameters in the deployed VSS, stores changes in the register, and signals relevant changes in the setup to the VSS administrator. A third part uses the information in the register to dynamically configure analytics tasks based on VSS operator input. In order to support the feasibility of this solution, we give an overview of related state-of-the-art technologies for autocalibration (self-calibration), scene recognition and lighting estimation in relation to person detection. The presented solution allows for rapid and robust deployment of Video Content Analysis (VCA) tasks in large scale ad-hoc networks.

The outline of this paper is as follows. Section 2 elaborates on managed analytics and it gives an overview of the three-part solution. Section 3 describes the relation between primary video analytics and supporting analytics. Section 4 presents an example of the second part that signals relevant changes in the orientation by using preset verification. Preset verification is important to determine when the intrinsic and extrinsic calibration are valid. Section 5 also shows an example of the second part that assesses parameters in the VSS by using facial resolution assessment. Facial resolution assessment is important to determine when and where face recognition can be applied. Section 6 describes an example of the third part, which uses calibration information in the register to configure a behavior-analysis application. Finally, Section 7 summarizes the conclusions.

2. MANAGED ANALYTICS FOR MODERN VSS

2.1 Managed analytics

Managed Analytics (MA) is analytics that is semi-automatically managed: the factors that determine the quality of the output of analytics are being monitored and controlled in order to be able to provide a sufficiently constant and high level of operational performance of the video analytics subsystem(s). Table 1 shows some factors that could influence the quality of video analytics.

Table 1: Examples of factors that influence the quality of video analytics, grouped by scene, camera, video analytics and processing chain, and corresponding control- and mitigation measures.

Factor group	**Relevant aspects**	**Factors that influence aspects**	**Control- and mitigation measures**
Scene	Lightning, complexity (occlusion)	Weather, day/night, number of people and their behavior	Connect analytics to weather station; active lighting; control number of people in location or their behavior
Camera	Camera position, ad hoc addition of new sensors	Framerate, compression, position, orientation, focal length, coverage or sensitivity of sensor	Installation, configuration and maintenance (e.g. with Rotakin doll); tampering detection; auto calibration
Video analytics & processing chain	Data transfer, data pre-processing	Changes in algorithm type, pre-processing or data representation	Version control; Test with representative datasets

As illustrated in Figure 2, MA requires the right types and amount of accurate contextual information. At first glance, MA may therefore seem more complex than regular video analytics. However, the complexity of the video analytics subcomponents in a managed analytics set-up can be reduced. Basically, MA is a new generic template, a design pattern, for VSSs that allows for a step-change in performance and robustness.

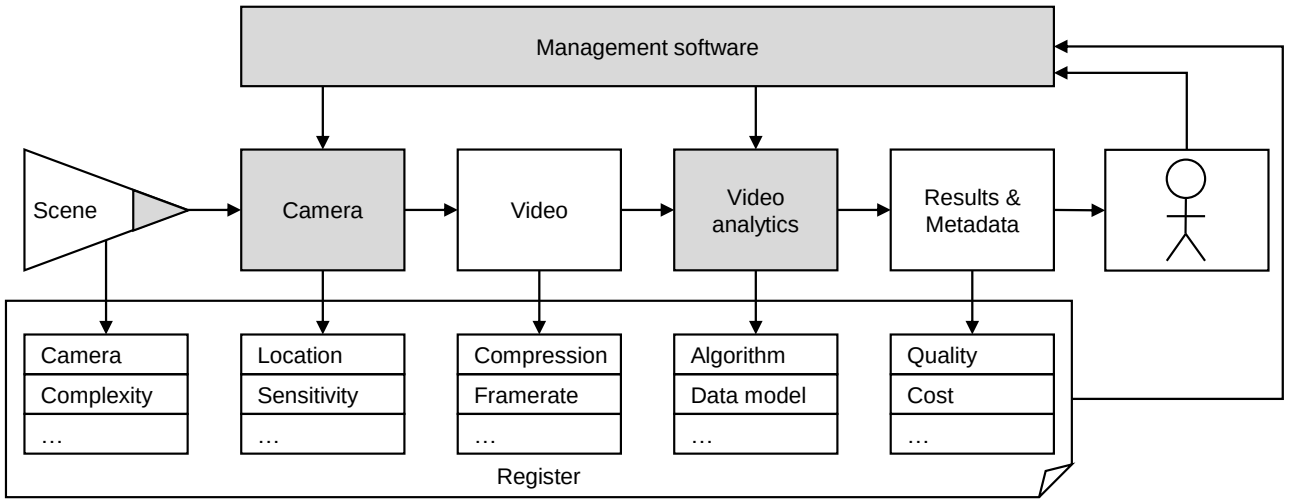

Figure 2: Managed analytics uses (partly automated) management software that monitors and controls the factors that influence the quality of the output of video analytics.

2.2 Managed analytics scenarios and use cases

There are several types of scenarios where MA should be particularly advantageous. In practice, they can occur simultaneously around the same VSS. Hereby, we show four examples where MA can be advantageous.

- *Deployment and planned growth*: Deployment and planned growth in terms of coverage or functionality typically happens in bursts. A new airport terminal is being built, or there is a roll-out of new functionality. MA might help during such planned developments in order to facilitate quick configuration of the new elements, and also to stay in control of the conditions under which they are being deployed and used for, in the beginning. This helps prevent teething troubles and manage end users' expectations. Especially when new functionality is being deployed, end users that are not experienced with analytics may have very different expectations from what they will actually get.
- *Regular use*: MA could be used in large-scale CCTV deployments where maintenance and moving (low) threat patterns lead to continuous changes in the CCTV conditions. Consider a set-up where one organization is responsible for the deployment and maintenance of sensors and another is responsible for the capabilities created with the sensors, i.e. to use them. MA can help to efficiently and objectively assess the current status of the scene, sensors and processing chain, which can facilitate the cooperation between these parties.
- *Mobile deployment*: The variety of camera platforms is increasing, and they are increasingly mobile, e.g. pan-tilt-zoom (PTZ) cameras, cameras on moving platforms and body-worn cameras. Besides the obvious geometrical differences concerning position, orientation and focal length, because of their mobility they can also be used - sometimes within the span of minutes- in wildly different operational conditions. E.g. in the case of a bodycam: in a private home and later that day at the forefront of a counter terrorism action.
- *Crisis*: The third scenario is the case of a crisis, such as a terrorist attack. LEAs suddenly (during or post incident) gain (legal) access to existing surveillance cameras and / or corresponding footage that are normally managed by other organizations, such as public transport organizations, retail organizations or event management organizations. The capabilities that LEAs need in crisis situations are very different from the needs required by more normal circumstances, and may therefore require analytics that is more robust, more scalable, and / or has more or different functionalities.

There are several generic use cases for MA that are useful in each of these scenarios.

- *Dynamic capabilities*: Use MA to dynamically (i.e. runtime, not design-time) design new capabilities composed of resources (cameras, ICT-hardware and personnel). This is obviously a desirable capability in the crisis scenario. In addition, this can also be used for self-service delivery in regular use if regular end-user benefit from a high level of flexibility.
- *Design to budget*: Use MA to dynamically adapt the quality and the workings of the VSS to available resources (e.g. processing power, personnel, etc.).
- *Quality monitoring*: Use MA to continuously monitor the quality of the output of a VSS in order to warn administrators in the case of unexpected performance drop.
- *Supported configuration*: Use MA to efficiently (and / or continuously) configure new or mobile VSS components (e.g. cameras), thereby reaching an increased operational availability and performance.

2.3 Comparing MA with traditional VSS deployments

If we want to describe the differences of MA w.r.t. traditional VSS deployments empirically then a parameter space needs to be described that we can plot our findings in. The parameters should be related to key performance indicators (KPIs) that are relevant to the owner of the VSS, and it should cover at least functionality and cost. These high-level parameters can be further decomposed in more specific parameters describing the workings of a VSS. Functionality can be decomposed in information value (including accuracy and timeliness) and level of deterrence, both also described in terms of geographical coverage, flexibility and fitness for purpose. Costs can be decomposed in required ICT (processing units, network and storage), required manual labor, both in terms of operational use (OPEX) and investments (CAPEX).

Metadata accuracy and required manual labor are important to describe how MA compares to traditional VSS deployments. We believe these two parameters impact the business case of a VSS more than the others, especially with continuously dropping costs of ICT.

2.4 Functional building blocks

MA requires several elements to be in place (Figure 3).

- *Register*: There needs to be a register containing the metadata regarding the scene, the sensor, the optical chain and the processing chain.
- *MA engine*: A managed-analytics engine (MA-Engine) is required that implements the intelligence to support the use cases of the previous section. This engine also includes a front end, to keep the human in the loop where need, and to keep him in control of the VSS.
- *Primary video analytics*: The primary video analytics is the collection of video analytics that supports the capabilities that are directly used by the end user (e.g. intrusion detection, aggression detection, suspect search or face recognition).
- *Supporting (video) analytics*: The supporting (video) analytics is the collection of analytics that supports the primary video analytics. There needs to be supporting (video) analytics (e.g. auto calibration, tampering detection, crowd density detection, illumination detection, etc.) and perhaps even additional sensors such as e.g. weather stations to assess the parameters of the scene. For example, primary behavior analysis may be dependent on a calibration of the camera or a color or intensity transformation (e.g. for contrast enhancement). There can be overlap between primary and supporting analytics. For example, crowd density detection may be directly useful for the end user, but it can also improve aggression detection.

Both the supporting and the primary analytics could perhaps also be combined into a form of "app store" that allows end users and administrators to dynamically select analytics for their tasks. It would also help to implement (open source) a reference architecture to demonstrate how these elements could actually function. Industry and / or end users could use that reference architecture to implement their own versions.

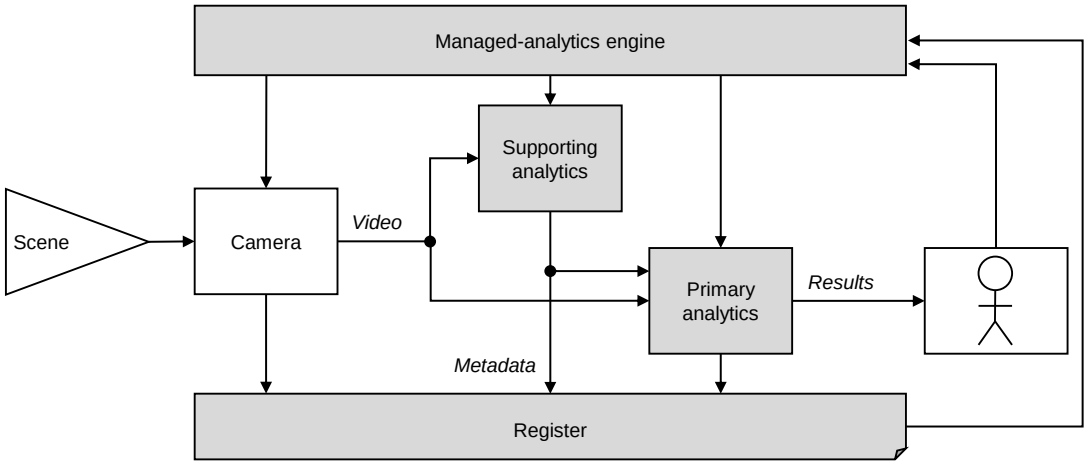

Figure 3: Key elements of managed analytics.

MA also requires non-technical elements. First, a metadata standard is required for describing and communicating sensor and analytics parameters. Existing standards already go a long way, but not all aspects that required for MA are currently standardized. Second, a set of good practices needs to be developed, collected, described and disclosed that describe useful functional connections between different technical components of MA. This could be developed in the form of a certification scheme which is based on the metadata standard, and would help the video analytics ecosystem work together in an efficient and effective manner. Third, for specific supporting analytics, a set of relevant and useful test data sets should be made available.

2.5 Emerging functions of supporting analytics

The supporting (video) analytics are instrumental to the functioning of MA, because they allow the MA-Engine to extend its internal model of the world (including the surveillance system itself) and successively to use that model to optimize its perception performance. For example, in order to adapt to changes (e.g. a local camera failure), the system detects when the model must be updated, computes the update and intervenes in the optical and/or processing chain to maintain or reach required operational performance (e.g. rotate a nearby camera and/or notify the end user of a changed coverage).
The supporting analytics together support the following functions (Figure 4). These functions emerge from their cooperation, they don't have to be implemented as explicit or separate modules:

- *Detect changes*: One key element to apply supporting analytics efficiently, is to recognize when a new world model must be computed or when an existing model can be used. This element monitors the quality of the model in order to be able to provide a high level of operational performance of the primary video analytics. Changes in the camera setup or scene conditions may necessitate the update of the camera register. When the perceived world does not change, the current model is still valid and its model can still be used. However, when something changes, one should validate whether this is a known situation – which was analyzed before – or it is a new situation, which requires the computation of a new model. The first element signals that a change has occurred and whether a known model is valid. An example of this element is preset recognition.
- *Compute update*: Another key element is the actual computation of a new model and an update of the model in the register. When the existing models are not valid, a new model must be computed to handle the new situation. An example of this is auto-calibration [13].
- *Apply supporting analytics*: The third key element is the use of the model to intervene in the optical and/or processing chain. For example, 'compute update' would compute an illumination variation and 'apply supporting analytics' would apply an illumination correction.

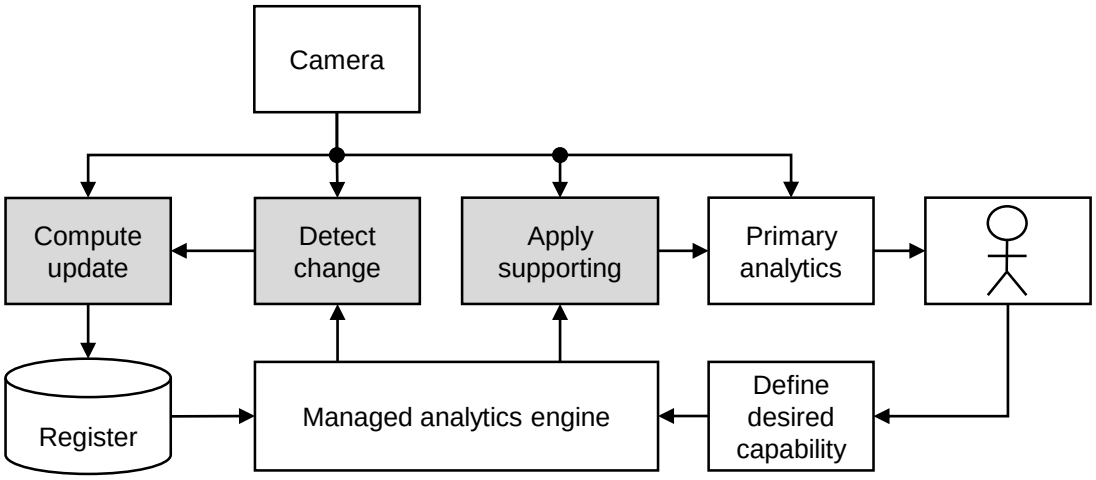

Figure 4: Three elements of supporting analytics: compute update, detect change and apply supporting analytics.

3. RELATION BETWEEN PRIMARY AND SUPPORTING ANALYTICS

In this section, we will give examples of primary analytics (Sec. 3.1) and supporting analytics (Sec. 3.2) with an emphasis on the relation between the two.

3.1 Primary analytics

The four key questions in the surveillance domain are who, what, where and when [10]. In slightly more detail, a primary function of surveillance, and by extension, of video analytics for security purposes, is to generate situational awareness concerning entities such as persons and vehicles, their activities and behavior, the way they relate to each other in situations and how they change over time in scenarios [21]. This information is used before (preparation and prevention), during (repression) and after (investigation and recovery) threats and incidents.

Primary analytics is therefore concerned with the analysis of video into more abstract information as described above. Here follow some examples, for the questions pertaining "where?" and "what are they doing (activity)?".

Where are they?
There are solutions for real-time forensic searches based on multi-camera tracking and re-identification [3][17]. The re-identification algorithms use information from the complete appearance, which is suitable for typical CCTV resolutions. Another strategy to recognize where people uses face recognition [11]. Face recognition requires sufficient resolution on the face. The primary re-identification algorithms can benefit from information regarding illumination characteristics of the scene to cope with illumination variations. And the primary algorithms for face recognition can benefit from facial resolution assessment to efficiently apply face recognition only in cameras and regions with sufficient resolution.

What are they doing?
Capabilities have been demonstrated, such as loitering detection, detection of suspicious behavior of pickpockets [5], threat detection of cargo theft on a parking lot [6] and long-term behavior analysis for compound security [23]. In some cases, rapid detection of the crime or incident – while it is occurring – is beneficial. For example, intrusion detection and aggression or violence detection [2] can help to intervene in an early stage. The primary behavior analysis can benefit from supporting algorithms for autocalibration to convert pixels to meters and perform measurements in real world coordinates as opposed to image coordinates or perform measurements in pixel coordinates at the right scale.

3.2 Supporting analytics

Illumination detection and correction
Retrieving objects in images from multiple cameras is an active research topic within image processing. In very controlled environments the assumption will hold that the pixel values of an object are comparable between other cameras, but in general this is not the case due to illumination variations in the scene and variations in types of configurations between cameras. One strategy is to match features that are lighting invariant [17], but this may lead to the loss of information, e.g. gray clothes may become similar to black clothes. Another strategy addresses the correction of photometric variations based upon changes in background areas to correct foreground areas [28]. This approach can also be separated in the three elements of supporting analytics: detect when the illumination changed, compute a new illumination correction, and apply the illumination correction to the incoming images using the primary video analysis.

Scene recognition and preset verification
Scene recognition has been investigated for several applications such as video summarization and scene-based localization. Scene selection can be used to reduce a complete video to only the unique environments. Selecting distinct scenes in a single continuous video made by a body-worn camera or pan-tilt-zoom (PTZ) camera has some similarities to shot-boundary detection (e.g., in movies and TV programs) [4]. Scene localization for a mobile camera has for instance been performed by matching local image regions [30]. Scene selection and scene-based localization can be examples of primary analytics that assist the user directly to find relevant parts in the recorded video. Similar technology can also be used as supporting analytics, for example to verify whether the pose of a PTZ camera conforms to a given preset pose. In case the historic preset recording and a current recording match precisely, the camera is confirmed to be in preset mode. In contrast, non-matching scenes indicate that the current recording does not correspond to the preset mode; in this case certain camera parameters from the preset condition cannot be used. In cases where the scene must be calibrated (e.g., for behavior analysis), it is important to know whether the orientation of the camera matches one of the known camera calibrations, or whether it has an unknown orientation that requires the computation of a new calibration, for example with an auto-calibration algorithm. Verifying whether a camera is in a specific preset mode, is therefore a prerequisite. In Section 4, we describe a method for preset recognition in more detail.

Facial resolution assessment
One of the key applications for surveillance cameras is the identification of individuals. An important requirement for the ability to recognize individuals is sufficient pixel resolution across the face. When the camera is in a known preset mode, the camera calibration can be used to derive the expected facial resolution at different positions in the scene. Only when individuals are at positions for which faces contain sufficient resolution, identification may be possible. At other positions for which there are insufficient pixels across the face, identification need not be considered. This also enables the selection of cameras in a large-scale network that need to be inspected for e.g. suspect search [17], thereby saving time for the law enforcement operators. In Section 5, we describe a method for facial resolution assessment in more detail.

Autocalibration

Once the camera is known to be out of preset mode or it is newly deployed, a new calibration procedure is required. The cost of performing a regular (manual) calibration for cameras is high, especially when there are many cameras in the environment; all initial deployments of cameras without any calibration will require that they are initialized efficiently. There are several options to perform the calibration. The first option is to perform the calibration manually in the scene, e.g., with a Rotakin doll, checkerboard patterns, or measurements. The second option is to manually indicate sizes and distances on the screen without accessing the scene. The third option is to generate a calibration fully automatic with supporting video analytics, which is called autocalibration. Automating the calibration allows for a short configuration time, and the use of video analytics in a wider range of scenarios, including ad-hoc crisis situations and large scale surveillance systems. Autocalibration can be entirely based on pedestrian detections [9] in surveillance video. The intra-camera geometry estimation entails an estimate of the tilt angle, focal length and camera height, which is required for the conversion from pixels to meters and vice versa [13]. In Section 6, we describe a method for aggression detection, which requires a calibration of the cameras.

4. SCENE RECOGNITION AND PRESET VERIFICATION

The scene recognition for confirmation of preset positions is supporting analytics that can be used to conclude whether two images show an identical scene. One of the images is a historic recording of the preset position, where it was known that the camera was in preset. The other image is a current recording for which it is to be decided whether it conforms to the preset position or not. The method should confirm the preset when there are only changes due to lighting, weather variation or moving objects between the images. On the other hand, when the camera's pose or focal length is changed with respect to the preset, the method should reject the preset condition.

4.1 Method for preset verification

In order to compare and match two images, a search has to be performed for (locally) similar regions. Since moving objects can appear in the scene, global similarity measures are not adequate for this task. Local region matching has been used frequently for stereo and object matching, and many local features have been designed with varying degree of invariance to camera position, pose or lighting changes. We apply the SIFT descriptor [16] which has a high degree of invariance for lighting changes. In principle, it is also invariant to small pose changes but this is not necessary for our application: when a change of camera pose produces zero feature matches between images, we can already conclude that the preset condition is not valid.

The method for preset detection is summarized below:

- Compute SIFT keypoints for image 1 and 2 (in grayscale).
- Compute matches between the descriptors of the keypoint sets.
- Count the number of matches that have similar keypoint locations (= are separated less than 3 pixels).
- Accept the preset condition when the number of matches is at least 4, otherwise reject the preset condition.

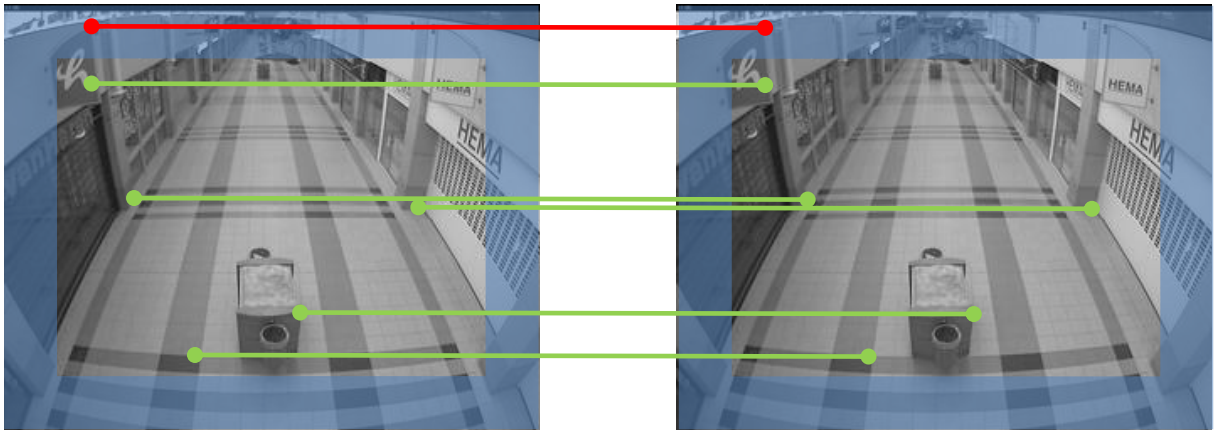

Figure 5: Illustration of the matching for preset verification between two images. Some example keypoint matches between the images are shown here, indicating locally similar regions (horizontal lines in green and red). Only matches inside the selected ROI (gray area) are counted (green lines).

When selecting matches between the two images, only those that are away from the image boundary are considered. Surveillance footage usually contains some timestamp or other metadata in the video frame that can produce spurious matches on the text. We therefore select a region of interest (ROI) that contains the inner region of the frame, and only allows matches based on the imaged scene itself. The thresholds for keypoint distance and number of matches have been determined empirically on the test set.

4.2 Experiments and results

The test set is constructed from surveillance footage from 37 different cameras that were positioned in a shopping mall and subway station. An example images from the shopping mall is shown in Figure 6. Since only few cameras were panning/zooming during the recordings, we have generated an artificial positive and negative training set form the footage in the following way. The positive set was generated by repeatedly drawing two images from a single camera within a 24 hour timespan. In total, the number of selected positive pairs was 100k, which equals on average 2700 pairs for each camera. However, some images contain actual pan/tilt/zoom activity of the camera, so these pairs were manually removed from the training set. The final number of positives was 97801. The negative set was generated by repeatedly drawing two images from different cameras; this set contains 100k pairs.

Table 2 contains the confusion matrix that resulted from running the preset detection method on the test set. Nearly all positives were correctly identified. Some errors were made due to reflections in the camera lens that significantly obscured the images. In addition, one error was made when a gate appeared in view that obscured most of the structure used for matching with the other image. Although these conditions could happen more often, our random sampling of scenes from all surveillance footage ensures that the test set is fairly representative. The obtained true positive rate on this test set is therefore 99.97%. The false alarm rate is 0%; no non-matching image pair contains enough accidental matches for erroneously detecting a preset condition.

Table 2: Confusion matrix for preset detection on the test set.

	Preset confirmed	**Preset rejected**
Positive pairs	97775	26
Negative pairs	0	100000

5. FACIAL RESOLUTION ASSESSMENT

One of the key applications for surveillance cameras is the identification of individuals. An important requirement for the ability to recognize individuals is sufficient pixel resolution across the faces. This holds for both manual and automatic means for identification. Experiments with human face recognition has shown that 30 to 40 pixels is the minimum horizontal resolution for recognizing *known* faces [26]. The identification of *unknown* faces in surveillance video will

therefore require at least this amount of resolution, and likely even more. Automatic face recognition systems also require a minimum of 30 to 50 pixels across the width of the face [27].

When the camera is in a known preset mode, the calibration information can be used to derive the expected facial resolution at different positions in the scene. Only when individuals are at positions where their faces contain 40 pixels or more, there is a chance of identification based on the facial resolution. Other cameras and positions where there are less than 40 pixels contained on the face, need not be considered for identification. Facial resolution assessment is supporting analytics that can select the relevant regions and cameras with sufficient resolution, which helps to scale VCA tasks to larger camera networks.

5.1 Method for facial resolution assessment

In recent work [13], it has been shown that automatic person detection can be used to achieve an autocalibration of the camera. Here we show an alternative approach towards finding facial resolutions that is easier to compute. The input also consists of person detections, but the model for the relation between the position of a detection and its width is linear instead of the higher-order relation for its height. The linearity for the width is the result from imaging a constant width object that moves on a plane with a pinhole camera; it is comparable to the way a railroad track is imaged as two straight lines disappearing at the horizon. The non-linear relation for the height is caused by the non-zero pitch angle and it is especially visible near (under) the camera. Although we could apply a face detector directly on the video frames, this may miss certain faces and is less reliable than a full-body person detector. The typical face width is taken as 25% of the person-detector width.

The method for facial resolution assessment in a video contains the following steps:

- Computation of person detections in the video
- Computation of the average face (bounding-box) width per image line (y-coordinate)
- Robust fit of a line through the (y-coordinate, face width) data in the upper half of the image
- Calculation of y-coordinate for 40 pixel face width

5.2 Experiments and results

We assume that the camera contains no roll angle; face widths are therefore equal across the image width. In addition, lens distortion is assumed to be relatively small since otherwise the linear relationship would be violated. An example of the method's output for a video is shown in Figure 6. It can be seen that the average face width is quite irregular, even after averaging nearly 35,000 person detections. As detections are less reliable due to perspective effects (causing the persons to appear rotated), a linear fit is performed on the detections in the upper half of the image only. Although this assumes that the upper half indeed contains sufficient person detections, it is not a restrictive assumption for all the tested cameras.

The results for the 18 cameras showed that the estimation of the identification region (yellow area in Figure 6) is adequate, except for cases where there are insufficient person detections in certain image regions. This may happen for instance when pillars or apparatus are obstructing the flow of people, or certain regions are outside the regular person flow, see Figure 7.

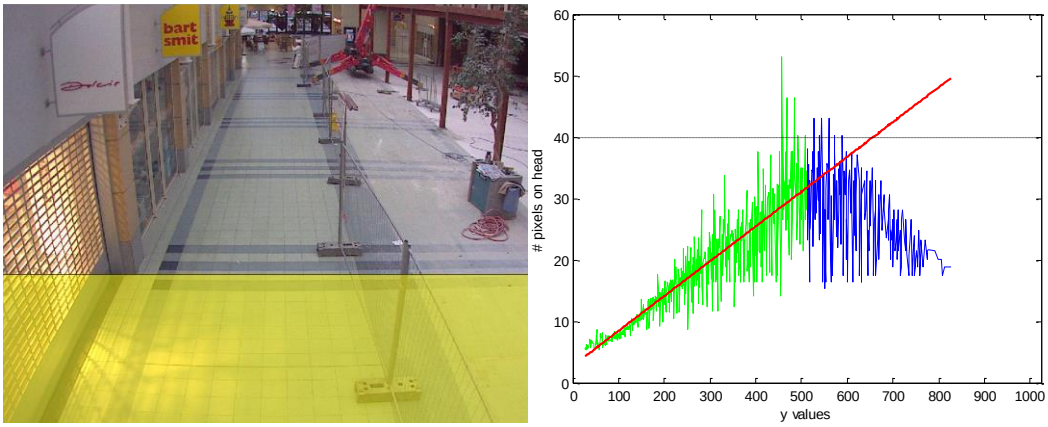

Figure 6: The average number of pixels of the face width for each image line, as computed from person detections. A line is fitted through the average values of the upper half of the image (green) while disregarding the lower half (blue). The intersection point of the line with the 40 pixel boundary (dotted line) at y=657, is indicated in the video frame on the left: all faces that would appear in the shaded area (yellow) are candidates for identification.

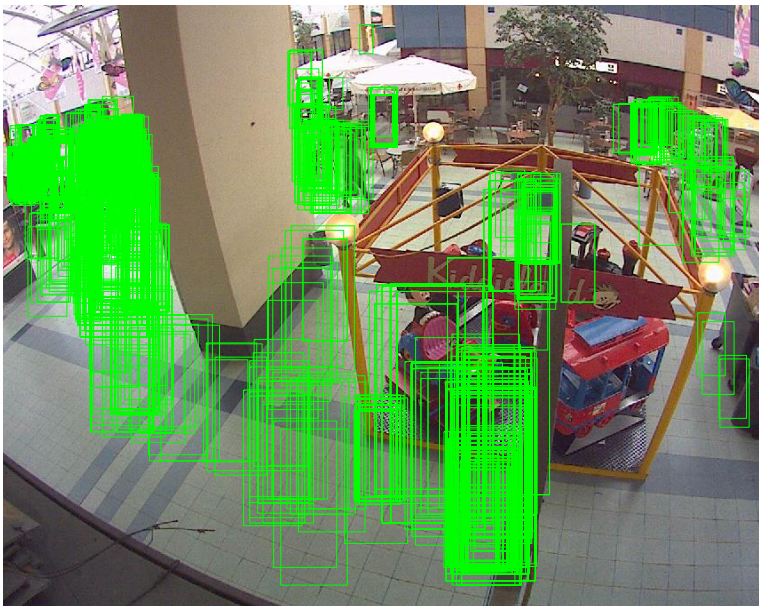

Figure 7: An example scene where there are only few detections; this hampers a reliable line fit.

6. AGGRESSION DETECTION

In this section we examine the benefit of managed analytics, especially of auto-calibration, to aggression detection in surveillance camera footage. Aggression detection is an example of primary analytics that can benefit from a calibration to perform analysis at the appropriate scale, thereby reducing computational demands, and false positives. Typical aggression detection strategies depend on prior knowledge concerning the scene (crowdedness, lighting conditions), the optical chain (camera location and orientation) and processing pipeline, making this an ideal candidate to demonstrate the effects of managed analytics.

Factors that contribute to the occurrence of violent incidents include fatigue (e.g. late at night, multi-day events), intoxication (e.g. drugs or alcohol), mixed social identities (e.g. mixed soccer hooligans) and poor crowd management (e.g. obstruction in people flows). In the Netherlands 83% of the municipalities experience the problem of such incidents during night-life activities [14]. According to information received from the police, when such incidents threaten to occur,

the authorities are occasionally late at the scene. The direct impact of such incidents includes property damages, wounded people and occasionally even fatalities. When this happens to public (civil) servants, the punishments are, in The Netherlands, extra harsh. The indirect impact of an incident, especially in the case of extreme violence and / or if it is picked up by media, is significant, including public naming and shaming of (supposed) perpetrators (up to vigilante justice), silent marches, and additional preventive and repressive security measures and policies. Authorities responsible for crowded night-life districts or crowded events can be aided with alerts concerning potential violent incidents. Video analytics can detect real-time aggression in live surveillance camera footage by automatic behavior analysis. This may in a later development stage be combined with aggression detection in the audio domain.

6.1 Related work

In other research on aggression detection in surveillance scenes, several techniques have already been tried. The first attempts are based on low-level color visual cues for track warping and motion [18], and multiple-frame feature point detection and tracking with crowd motion vectors and the Kanade-Lucas-Tomasi tracker [24]. However, extensive tests for performance are missing and the features aggression detection had not yet been extensively researched. In 2011 it seems that the tipping point had been reached, when papers start noting high scores on aggression detection in crowds. Chen e.a. [8] reached an accuracy of 96% with a not real-time approach based on the optical-flow context histogram and a sequence classification with random forest, Bayesnet, and SVM. Many more extensions on the optical-flow algorithm have been made, e.g. a Gaussian model of optical flow (to include uncertainty) [31] with an accuracy of 82.79% on a crowd violence dataset while having a processing speed of 36ms per frame of video. A different paper used an extension that makes use of the eigenvalues from second-order statistics of the histograms of optical flow vectors [22], reaching an accuracy of 82.2% and achieving real time. An additional real-time application is where optical flow is combined with Horn-Schunck reaching an accuracy of 71.2% [1]. A recent method used linear SVMs in combination with local features and spatio-temporal positions in the form of Improved Fisher Vectors, reaching accuracies of 96.4% [2]. Furthermore, even fluid mechanics have been successfully applied to the field of violence detection in crowded scenes [19]. We will use the proven capability of the optical flow as a feature. Optical flow is a good feature of punching arms, kicking legs and pushed or falling bodies, interlocking bodies during struggles and bodies in abnormal positions like falling or lying horizontally. We will show how the primary video analysis application (aggression detection) benefits from supporting analytics (calibration). Our approach can predict aggression in real-time at high accuracy given a real-life dataset.

6.2 Method for aggression detection

The aggression-detection method consists of a detector to find the relevant regions in the video frames where there is significant motion potentially indicating aggressive acts (moving object detection, MOD). Feature extraction is based on motion (optical flow) and appearance (normalized cross correlation) is followed by a classifier with a random forest to create histograms of the features. The classifier is a support vector machine (SVM) with a chi-squared (χ^2) kernel. This method is trained on real life footage from a surveillance system already in place, and the data was carefully selected to contain variations in illumination, distance to the camera and group size. This section provides a description of the MOD, its relation to supporting analytics (calibration), optical flow, and the learning procedure.

- *MOD*: In the first step the regions of interest (ROIs) are detected by the MOD algorithm which detects boxes in areas with significant change compared to the previous frame.
- *Calibration*: The candidate boxes for MOD are defined by a grid. These boxes are assumed to be square (width and height of the box are equal to the height of a person) because when two people fight or are entangled, that area in the image is best recognized as a square. The calibration can be generated automatically with an autocalibration method (as a type of supporting analytics conforming managed analytics) or with manual calibration. For automatic calibration, we used the autocalibration approach of Hollander et al. [13]. For manual calibration, the user indicates the size of the person in the back of the scene and at the front of the scene. Together with a polygon where the persons in the scene are expected, a calibration grid is computed by linear interpolation, with a given horizontal stride (10% of the box width) and vertical (10 steps in the depth direction). At every frame, all candidate boxes are analyzed and those where the change surpasses a threshold are further analyzed (Figure 8). The manual calibration reaches a few percent better performance than the automatic calibration (comparison results not shown). For a large number of cameras or for regularly moving PTZ cameras, manual calibration would not be feasible, but in our experiments we calibrated 53 videos manually.

- *Optical flow*: We used the OpenCV implementation to compute optical flow at a regular grid inside each box that was detected by the MOD. The flow in the box is normalized by the size of the box to obtain scale-invariant features.
- *Learning*: A random forest quantifies the features into histograms, which are classified by an SVM with a Chi-2 kernel. The labels provided to the SVM for training are derived from intersection with ground-truth boxes in the videos, of respectively aggression and normal situations. Implementation details can be found in the work of Burghouts e.a. [6].

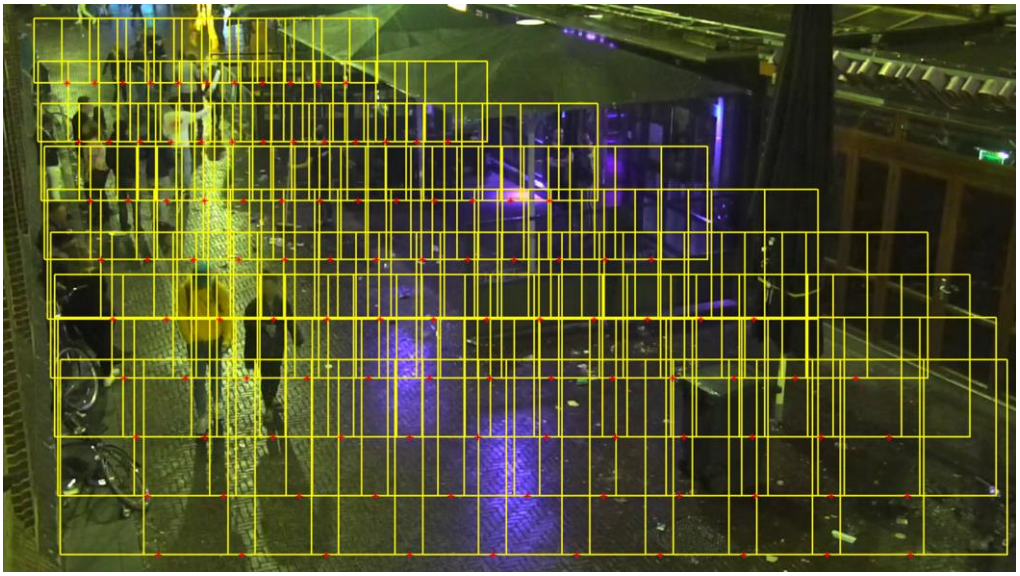

Figure 8: Candidate boxes on the manual calibration grid in one of the 53 videos. A selection is shown with a horizontal strive of 50% for visualization purposes. Faces and text in the image were redacted for anonymization.

Relevance of autocalibration for this type of aggression detection
Calibrating a viewpoint by manual annotation took us approximately 10 seconds and 8 mouse clicks per camera viewpoint. This estimate excludes time spent to verify or correct the annotations. In practice these are often needed because this is very monotonous work, and prone to mistakes. This may not seem like a lot of time, but when this needs to be done for every time a camera is moved or added it can add up, especially in times of crisis (see section 2.2 for the scenarios), also for relatively small deployments, and especially if also other types of supporting and primary analytics are to be considered.

6.3 Experiments and results

The dataset contains 14 cameras and is collected from a city center, from a surveillance camera system without sound recording. Videos were recorded at 15 fps and the resolution is 1024x720 pixels. In total, 53 videos were collected, of which 30 without an aggression incident and 22 with one real aggression incident. The videos were annotated by boxes in each frame, yielding 22 aggression annotations in total. The videos have variations such as day and night, and incidents happening close-up in the middle of the view, as well as far away from the camera, sometimes with many people in the view and sometimes with few people, and clutter from disco lights etc. When the manual annotation has overlap with the automatically generated MOD box the label for this box is ‘aggressive’ and otherwise it is ‘normal’. For evaluation, we use the same criterion as for evaluation: the false positives (FP) are non-overlapping with the aggression annotations and the true positives (TP) are overlapping. The dataset is absolutely not well balanced and it contains a huge amount of negatives. For the training and evaluation, a two-fold cross-validation is applied, with folds determined at the video level.

The grid leads to a huge amount of candidate boxes. Typically there are more than 500 candidate boxes in one image. The total number consists of boxes for each of the 53 videos and for each frame in the video. The MOD detections are an effective way to filter the many boxes in the image to the few relevant ones that relate to aggression. In Figure 9, it can be seen that with 1000 FP boxes, on average 50 boxes are detected on each of the 22 aggression incidents. That means that for this operating point, half of the detections (1000 FP and approximately 1100 TP) relates to aggression.

The aggression classifier provides an effective way to reduce the number of MOD boxes even further. In Figure 10, it can be seen that with a total of 20 FP, on average 16 TP per incident are detected (blue line) and that 20 out of 22 incidents are

detected (red line). This is the selected operating point. At 8 false positives, still 20 aggression incidents are detected, but with a lower average number of detections on each incident (8 detections).

The method is configured at the selected operating point (see above). When the classifier produces a confidence that surpasses this threshold, our method silences for one second. Typical aggression incidents last longer than just a few frames, so it makes sense to silence for a short period, to prevent bursts of detections. The results are shown in Figure 11. The figure shows that 20 out of 22 aggressive incidents are detected at the cost of only 4 false positives.

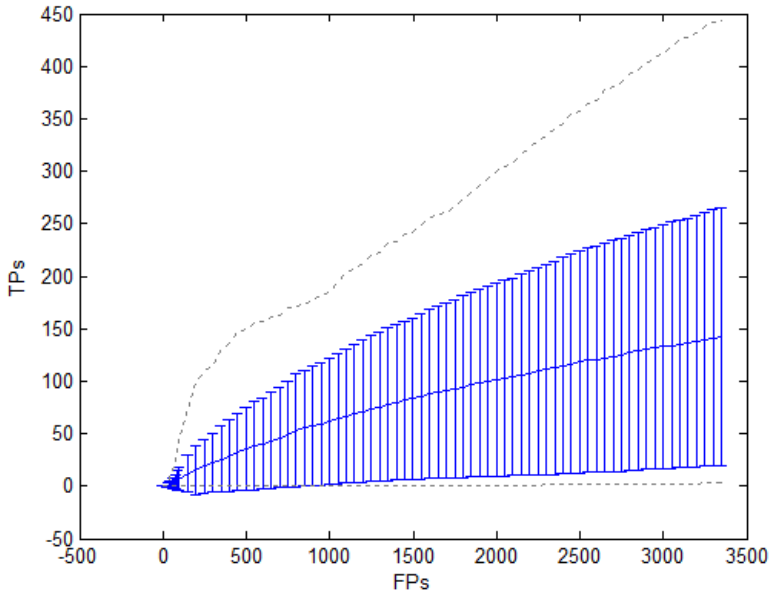

Figure 9: MOD performance: true positive detections per incident (TP) versus total false positives detections (FP).

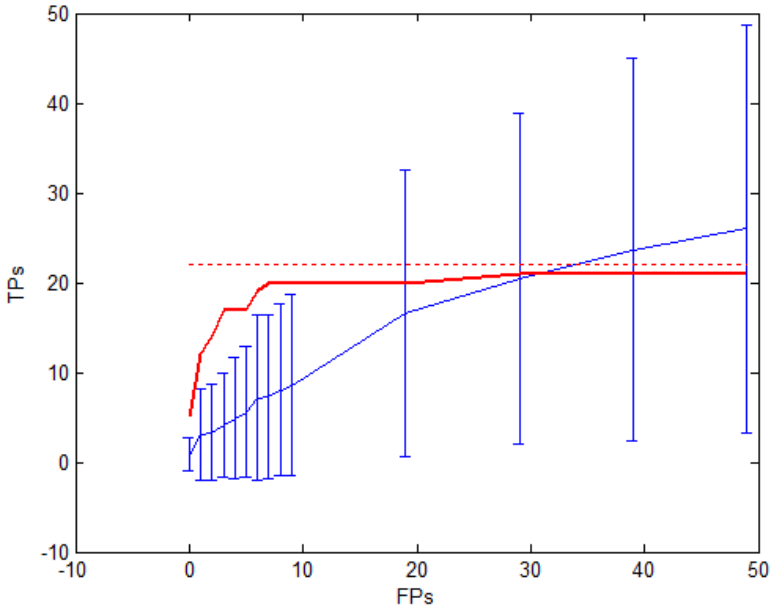

Figure 10: Classifier performance: The blue line shows the true positive detections per incident (TP) versus total false positive detections (FP). The red line shows the amount of detected incidents (out of 22, red dashed line).

The method achieves an F1-score of 88.7% (accuracy of 99.8%). Because of the nature of the real-life dataset with large amounts of true negatives, the performance is not comparable to related work where datasets are often more or less balanced (see for instance [12]). The supporting analytics (calibration) is essential for the primary analytics (aggression detection) to generate boxes of the appropriate size. For surveillance applications the operational applicability depends on how well the method deals with sparse incidents and mostly normal situations. The algorithm works in real-time. It may be concluded that the method's performance is accurate and promising for a variety of surveillance and monitoring applications. The primary analytics depends on a calibration for the bounding-box generation.

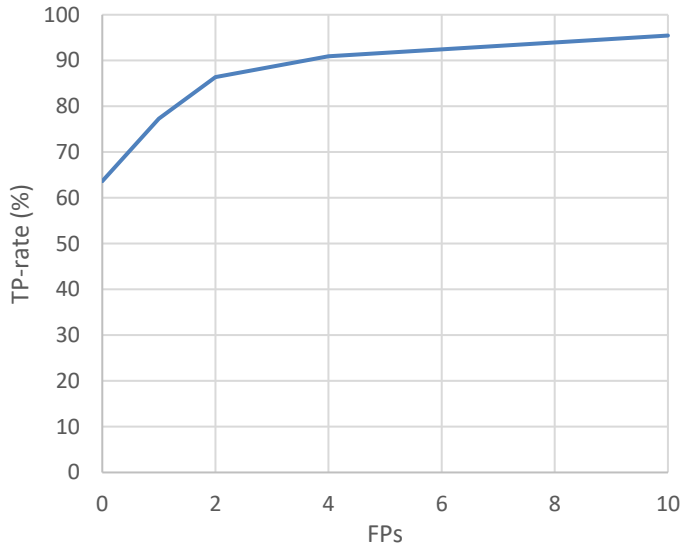

Figure 11: Performance of our system after selecting the operating point and adding a silence time of 1 second after a detection. At 4 false positives (FP), 91% of the aggression incidents is detected (20 out of 22).

7. CONCLUSIONS

In this paper, we presented a three-part solution for managing the performance of complex analytics deployments. The first part is a register containing meta data describing relevant properties of the optical chain, such as intrinsic and extrinsic calibration, and parameters of the scene such as lighting conditions or measures for scene complexity (e.g. number of people). A second part frequently assesses these parameters in the deployed VSS, stores changes in the register, and signals relevant changes in the setup to the VSS administrator. A third part uses the information in the register to dynamically configure analytics tasks based on VSS operator input. In order to support the feasibility of this solution, we give an overview of related state-of-the-art technologies for autocalibration (self-calibration), scene recognition and lighting estimation in relation to person detection. The presented solution allows for rapid deployment of Video Content Analysis (VCA) tasks in large scale ad-hoc networks.

ACKNOWLEDGEMENT

The work for this paper was conducted as part of the program 'Security' in the project 'Sensors and systems for security' in the Dutch top sector 'High Tech Systems and Materials'. We thank Daniel Maaskant for his internship related to aggression detection at TNO.

REFERENCES

[1] Arceda, V., Fabian, K., Gutierrez, J., "Real time violence detection in video with ViF and Horn-Schunk," Proc. Latin American Caribbean Conf. for Eng. Techn., (2016).

[2] Bilinski, P., Bremond, F., "Human violence recognition and detection in surveillance videos," IEEE Adv. Video and Signal-based Surveillance AVSS, 30-36 (2016).

[3] Bouma, H., Rest, J. van, Buul, K., Jong, J. de, Havekes, A., "Integrated roadmap for the rapid finding and tracking of people at large airports," Int. Journal of Critical Infrastructure Protection IJCIP 12, 61 - 74 (2016).

[4] Bouma, H., Baan, J., Haar, F., et al., "Video content analysis on body-worn cameras for retrospective investigation," Proc. SPIE 9652, (2015).

[5] Bouma, H., Baan, J., Burghouts, G., et al., "Automatic detection of suspicious behavior of pickpockets with track-based features in a shopping mall," Proc. SPIE 9253, (2014).

[6] Burghouts, G., Schutte, K., Hove, J., et al., "Instantaneous threat detection based on a semantic representation of activities zones and trajectories," SIVP 8(1), 191-200 (2014).

[7] Burghouts, G., Schutte, K., Bouma, H., Hollander, R. den, "Selection of negative samples and two-stage combination of multiple features for action detection in thousands of videos," Mach. Vision Appl. 25(1), (2014).

[8] Chen, Y., Zhang, L., Lin, B., Xu, Y., Ren, X., "Fighting detection based on optical flow context histogram," IEEE Innovations in Bio-inspired Computing and Applications, (2012).

[9] Dollar, P., Appel, R., Belongie, S., Perona, P., "Fast feature pyramids for object detection," IEEE Trans. Pattern Analysis and Machine Intelligence 36(8), 1532-1545 (2014).

[10] Garofolo, J., Contestabile, J., Powell, J., Corso, J., et al., "First workshop on video analytics in public safety," NIST Internal Report 8164, (2017).

[11] Grother, P., Quinn, G., Ngan, M., "Face in video evaluation (FIVE) Face recognition of non-cooperative subjects," NIST NISTIR-8173, (2017).

[12] Hassner, T., Itcher, Y., Kliper-Gross, O., "Violent flows: Real-time detection of violent crowd behavior," IEEE CVPR Workshops, (2012).

[13] Hollander, R. den, Bouma, H., Baan, J., Eendebak, P., Rest, J. van, "Automatic inference of geometric camera parameters and inter-camera topology in uncalibrated disjoint surveillance cameras," Proc. SPIE 9652, (2015).

[14] Lemmers, L., Hasselt, N. van, "Alcohol en uitgaansgeweld; de stand van zaken," Trimbos institute AF1310, (2014).

[15] Li, X., Cheng, Y., Zhang, T., "Robust extrinsic calibration from pedestrians," ACM Image Communication 55, 106-120 (2017).

[16] Lowe, D., "Distinctive image features from scale-invariant keypoints," Int. J. Computer Vision 60(2), (2004).

[17] Marck, J., Bouma, H., Baan, J., et al., "Finding suspects in multiple cameras for improved railway protection," Proc. SPIE 9253, (2014).

[18] Mecocci, A., Micheli, F., "Real-time automatic detection of violent-acts by low-level colour visual cues," IEEE ICIP, (2007).

[19] Mohammadi, S., Kiani, H., Perina, A., Murino, V. "Violence detection in crowded scenes using substantial derivative," IEEE Adv. Video and Signal-based Surveillance AVSS, (2015).

[20] Rest, J. van, "Surveillance and video analytics: factors influencing the performance," EU JRC100399, (2016).

[21] Rest, J. van, Grootjen, F., Grootjen, M., Wijn, R., et al., "Requirements for multimedia metadata schemes in surveillance applications for security," Multimedia Tools and Applications MTAP 70(1), 573–598 (2014).

[22] Ribeiro, P., Audigier, R., Pham, Q., "RIMOC, a feature to discriminate unstructured motions: application to violence detection for video-surveillance," Computer Vision and Image Understanding 144, 121-143 (2016).

[23] Schutte, K., Burghouts, G., Stap, N. van der, Westerwoudt, V. et al., "Long-term behavior understanding based on the expert-based combination of short-term observations in high-resolution CCTV," Proc. SPIE 9995, (2016).

[24] Saxena, S., Bremond, F., Thonnat, M., Ma, R., "Crowd behavior recognition for video surveillance," Adv. Concepts for Intell. Vision Systems, 970 – 981 (2008).

[25] Shalnov, E., Konushin, A., "Convolutional neural network for camera pose estimation from object detections," Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., (2017).

[26] Sinha, P., Balas, B.J., Ostrovsky, Y., Russell, R., "Face Recognition by Humans," Face Recognition: Advanced Modeling and Methods, Academic Press, (2006).

[27] Tistarelli, M., Li, S.Z., Chellappa, R., "Handbook of Remote Biometrics," Springer, (2009).

[28] Torres, J., Schutte, K., Bouma, H., Menendez, J., “Linear colour correction for multiple illumination changes and non-overlapping cameras,” IET Image Processing IPR 9(4), 280-289 (2015).
[29] Wieringa, F., Bouma, H., Eendebak, P., et al., “Improved depth perception with three-dimensional auxiliary display and computer generated three-dimensional panoramic overviews,” J. Medical Imaging 1(1), (2014).
[30] Wang, J., Zha, H., Cipolla, R., “Coarse-to-fine vision-based localization by indexing scale-invariant features,” IEEE Trans. On Syst. Man and Cyber 36(2), 2006.
[31] Zhang, T., Yang, Z., Jia, W., Yang, B., Yang, J., He, X., “A new method for violence detection in surveillance scenes,” Multimedia Tools and Applications, 75(12), 7327 – 7349 (2016).