A decentralized Privacy-sensitive Video Surveillance Framework

Tobias Senst [1*], Volker Eiselein [1*], Atta Badii [2#], Mathieu Einig [2#], Ivo Keller [*], Thomas Sikora [*]

[*] *Communication Systems Group, Technical University Berlin*
[1] {senst,eiselein}@nue.tu-berlin.de

[#] *Intelligent Systems Research Laboratory, University of Reading*
[2] {atta.badii, m.l.einig}@reading.ac.uk

***Abstract*—With the increasing spread of accurate and robust video surveillance, applications such as crowd monitoring, people counting and abnormal behavior recognition become ubiquitous.This leads to needs of interactive systems taking into account a high degree of interoperability as well as privacy protection concerns. In this paper we propose a framework based on the ONVIF specification to support the work of video operators while implementing a privacy-by-design concept.We use an OpenGL-based 3D model of the CCTV site where we display the results of the video analytics in an avatar-based manner and give an example application on mugging detection.To place the automatically detected scene information, such as people detections and events, an automatic camera calibration is used which effectively reduces the deployment effort.**

***Index Terms*—Video Surveillance; Privacy Protection; ONVIF; Calibration; Mugging Detection**

I. INTRODUCTION

Nowadays, Video Surveillance has become an ubiquitous tool which does not only exist as a vague and theoretical idea in research communities but also finds its application in numerous scenarios all around the globe. Not all technical issues have been solved in these days but the number of cameras deployed increases as steadily as the performance of the algorithms which are used for analysis of the recorded video streams.

However, from ethical perspectives it has to be ensured that the common need for security in public spaces does not interfere with the individual rights in modern societies. This makes it necessary to introduce systems which are capable of not only enhancing the possibilities of police and law-enforcing agencies but which also refraining from revealing too much of an innocent individual's identity and their personal information to the video operator.

From a practical point of view, Video Surveillance systems are also at a turning point. On a large CCTV-surveilled site, such as an airport or a major train station, hundreds of cameras can exist, and it is virtually impossible to have security staff watching all the recorded CCTV footage round the clock in order to detect all suspicious events. Instead, it is mostly clear that computers will be used for an automatic analysis of CCTV videos. Their results can then be used to help video operators by drawing their attention to potentially interesting events.

Yet, it is important to note that a CCTV management system providing solutions for these problems should be capable of prioritizing alarms as in general automatic analysis will not work under all conditions and thus may generate false alarms. Also not all alarms will refer to equally relevant dangers or threats in the scene. It is therefore essential to develop mechanisms which allow video operators to focus on their work while receiving notifications from a lot of cameras and still being able to concentrate on the most important events. Summarizing these features, a modern Video Surveillance management system should be designed according to the following concepts:

- **Privacy protection:** A video operator should not be able to identify individuals unless this is inevitable according to the current security situation. With respect to the context, it might even be necessary to wait for an explicit permit from the court before identities in the scene may be revealed.
- **Usability:** Despite a potentially huge number of CCTV cameras and analytics engines, video operators need to receive clear indications of possibly harmful situations and must be able to identify quickly how to respond to these. Alarms must be prioritized and operators need to overview under all circumstances which alarms are to be checked first.
- **Conformity to open standards:** The system must be able to work with analytics engines from different brands and manufacturers. Analytics results must be exchanged among these submodules and also be displayed in the management system.
- **Extendability / Scalability:** The system is likely to work with a large number of cameras and analytics engines and must still guarantee that video operators do not lose track of the most important events in the scene. Deploying more cameras or performing new types of analyses should be possible with small effort and should not affect current analyses.
- **Easy deployment:** As no two deployment sites will be the same, it should be possible to set up the system quickly in different environments, using different analytics modules etc. The different modules might be on sev-

eral computers on the site and should thus communicate over a standardized Ethernet connection and exchange all their data via standardized protocols.

In the Video Surveillance domain, today a huge number of algorithms can be identified which serve for many different applications. Algorithms analyzing very densely crowded scenes are mostly influenced by the physics of liquid dynamics [1]–[3]. Moving pedestrian streams can be identified using coherent motion patterns and thus allow to indicate an abnormal event, such as accidents or fightings. In lesser crowded scenes, people can be detected and tracked with model-based approaches, such as [4]–[6]. These allow to count the number of people in the scene and also give important knowledge about individuals' paths for abnormal behavior detection such as mugging.

Similarly, in scenes with few people, automatic Action Recognition or Human Behavior analysis based on spatio-temporal [7], [8] or textural [9]–[11] appearance models can be performed if the camera resolution is sufficiently high.

While the above methods all focus on persons observed in a scene, many applications also exist for recognition of static objects. E.g. Left-Luggage detection is often done by Background Subtraction algorithms, such as e.g. [12], [13].

Concerning Privacy aspects in Video Surveillance, different approaches exist. General overviews such as [14] and [15] list a detailed discussion of needs such as integrity and confidentiality of data.

Following [16], the change from analog to digital CCTV systems leads to easier access and analysis of video data. CCTV network operators are now almost free to choose which analysis tasks are to be run in real-time and which of them might be better suited on large server arrays. This leads to a privacy-by-design approach in which smart cameras perform multiplexing of the recorded data and separate the behavioral part from the part containing personal data. According to the context, it is then possible to limit the access to the respective data.

II. ARCHITECTURE

The aim of the proposed Framework is to support end users (video operators) in their work and to include privacy protection techniques by design. Therefore the architecture is set up to account for an end-to-end tool chain which allows realizing all relevant analyses.

Privacy protection is inherent to the system as the most-viewed scene representation is an abstract 3D model of the site which also provides the user with a fast and comfortable overview of the current events. By means of automatic camera calibration, the video analytics results can be displayed as 3D objects in the correct position within the 3D site model.

A. ONVIF

The Open Network Video Interface Forum (ONVIF) has developed a network layer of IP security devices described by web services based on the Organization for the Advancement of Structured Information Standards (OASIS). The definition

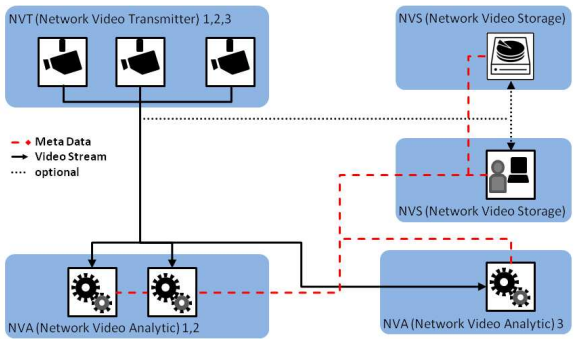

Fig. 1. Architecture of the ONVIF based network. The optional video stream enables to store the source video data stream. For a higher privacy-preserving capability the prototype only uses abstract scene representations.

of the client is given by Web Services Description Language. Thus the integration of the client is platform-independent and allows the framework to combine a diverse set of video surveillance devices unrelated of their brand.

The ONVIF 2.2 standard [17] categorizes devices into four classes: firstly the network video transmitters (NVT) which provide one or more video streams, such as a camera. Network video analytics (NVA) are devices used to analyze video, audio or metadata and extract additional information. Network video displays (NVD) provide the visualization of the media streams up to the interfaces between system and human operators. Network video storages (NVS) are devices used for recording streamed media and metadata as well as the capability of accessing the data in a structured manner.

Figure 1 shows an application scheme of ONVIF devices allowing the combination of cameras, network archives and analytics modules in order to exchange all relevant data in a well-standardized and simple manner. Each device implements a specific set of ONVIF-specified web services. The figure shows an optional video stream connection between the IP cameras and the video storage and video display. To enable a higher privacy preserving capability the proposed prototype is based on an abstract scene representation and does not store any video data.

In this paper we will focus on the receiver and the video analytics device web service which are important to implement the NVA and NVD, i.e. the video analytic application and the graphical user interface. The receiver service provides access to the configuration of the assigned received video streams to the appropriate video analytic algorithms. It is based on a list of configuration objects, which contain information such as the streaming protocol, the connection mode and the media URI. Different receiver configurations can be identified by receiver tokens.

The video analytics device service provides structured access to the parametrization of the video analytic algorithm implemented by the NVA. The main element of the service is the AnalyticsEngineControl object, which comprises tokens and descriptions for the NVA as well as the possibility of activation and deactivation. The AnalyticsEngineControl contains

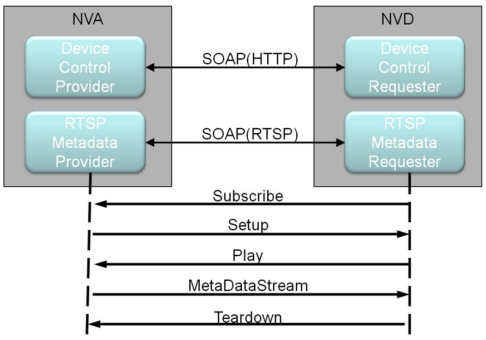

Fig. 2. Communication scheme between devices in the proposed prototype.

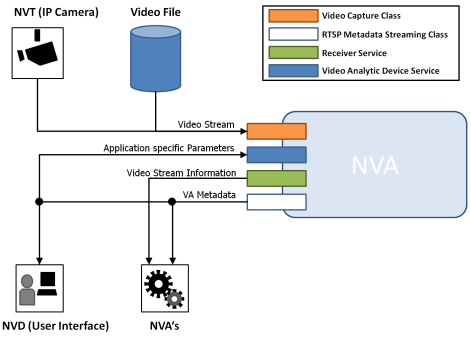

Fig. 3. Interfaces of the video analytics modules of the proposed prototype. The NVT supports IP cameras or video files in order to be able to build test systems for predefined scenarios.

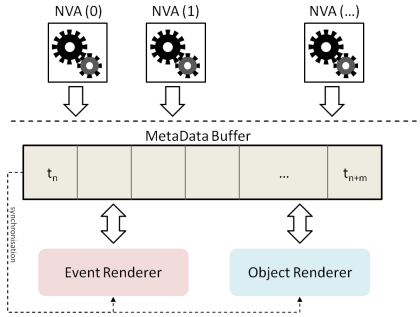

Fig. 4. Synchronization of received meta data by the graphical user interface.

the token for the receiver and the algorithmic parametrization on which the analysis will be applied. This means the web service interface for the AnalyticsEngineControl allows the GUI or any other NVD devices to discover the NVA regarding their video sources and to perform actions such run, stop and reset on each algorithmic device.

The parametrization of the algorithmic module is assembled by the VideoAnalyticsConfiguration. This allows the end-user (or the deployment engineer) to modify the algorithm-specific parameters externally, such as thresholds, learning rates etc.

The data structure and the exchange between devices are based on the Simple Object Access Protocol. The prototype uses two ways to transfer data between devices, see Figure 2.

- Device control data such as described above that exchanges configuration and parametrization data is transferred via the web service device interfaces over HTTP, encapsulated in SOAP and within the ONVIF specification.
- Real-time capable data such as metadata from video analysis results e.g. object trajectories and intrusion events are transferred packet-based within an explicit Real-Time Streaming Protocol (RTSP) stream, encapsulated in SOAP and within the ONVIF specification.

Each NVA provides a metadata stream which gives access to the respective video analytic results. The ONVIF metadata description specifies two different kinds of analytic structures to assemble the analysis results: The EventStream type and the VideoAnalyticsStream type is defined as events and analytics based schemata respectively. In contrast to [18] the proposed system is based on both schemes.

B. Implementation

The data transfer of the NVA is performed on three different ways: Video data from video files or video streams over RTSP and HTTP are received by a video capture module. The application-specific parameters and data transmitted to discover the NVA device from the 3D GUI are exchanged via the receiver and video analytic device web services interfaces. The results of the video analytics algorithm as e.g. recognized people in the scene or detected events are transmitted via the metadata streaming over RTSP.

To transmit complex data, in our case people positions, ONVIF specifies the VideoAnalyticsStream type which consists of a set of frame elements containing the information for a single video frame. The content of a frame is assembled with an object-based representation where features related to one particular object can be assigned by their object identifier.

To transmit deterministic data such as people tracks, the trajectories have to be disassembled by the NVA and composed by the GUI client module. The RTSP is designed for real-time data streaming and based on the UDP protocol which has no handshaking. Thus the reliability of the metadata transmission is not given. As a consequence the client has to deal with missing data packets resulting into lost events or frames.

Furthermore it is not ensured that the received metadata packages are in temporally correct order. Therefore, the framework supports a central metadata buffer which manages all connected NVAs, see Figure 4.

This concept enables the GUI to interfere between events and analytic frames and to apply additional filters. It is possible to display data from a specific set of cameras or analytic results according to a given topic. It is also possible to maintain a strict user-role-base access model which supports the overall privacy protection by the system.

The buffer size is restricted which leads to a policy of object removal from the buffer if a preset time threshold has exceeded.

If a new metadata message has been received and deserial-

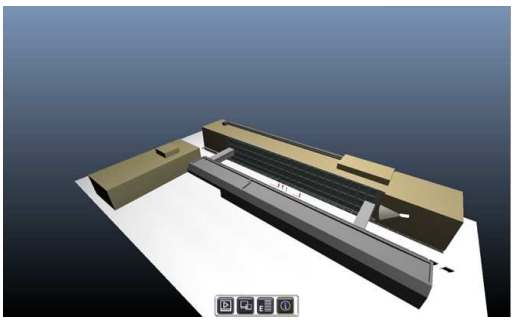

Fig. 5. Overview of the 3D NVD with a site model of the TUB Campus.

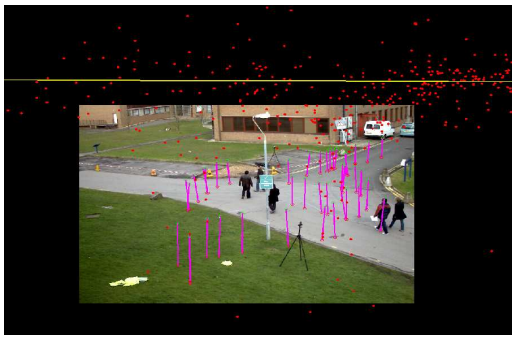

Fig. 6. Estimation of horizon line (yellow) by a set of head and feet detections (pink) of a person walking in the scene which geometrically meet on the horizont.

ized successfully, the buffer sends a synchronization signal to the object and to the event renderer in order to draw the overall scene of the 3D GUI. Note, that the object id is not a globally unique identifier, since this would require a synchronization of all connected NVAs among each other.

The global identifier of an object is defined as the triple of receiver token, object id and video analytics configuration token. Thus, the proposed framework implies the receiver token and the video analytics configuration token as being unique in the network.

C. 3D model of the site

As NVD in our system we use an OpenGL-based 3D model of the CCTV site where we display the contents of the scene in an avatar-based manner. This allows an intuitive and easy understanding of where events occur and how they might be related to each other. The position of the CCTV cameras is plainly visible and relations between camera views are easily comprehensible for the operator who is able to freely navigate like a hummingbird in all directions through the scene.

For a high usability, an event list ensures that incoming analysis events are shown in a prioritized manner and the operator has several tools which allow him to see the event displayed in the 3D environment and query information associated with it. The NVD can be placed independently from the NVT and NVA devices on the site and thus increases privacy protection as well as scalability of the overall system. Only a standard LAN connection is needed in order to connect the NVD to the other devices.

III. Automatic Calibration

In order to calibrate the camera used in our system automatically, we follow a method proposed in [19]. In this algorithm, the camera calibration parameters are estimated by silhouettes of walking persons in the scene, and a full projection matrix is estimated.

This framework helps reducing the deployment effort for our system and gives satisfactorily accurate results, as we do not need absolute precision. The main purpose of the calibration in our system is to show a person's avatar in a virtual 3D-environment in order to hide potentially private information from the viewer.

As mentioned, the calibration algorithm works using silhouettes of walking pedestrians which are extracted by Background Subtraction as described in [20]. In order to obtain these, we use a training phase during which the algorithm detects all foreground blobs in the image. A constraint of the used method is that all blobs should represent only one person. If this demand is not met, is still possible to apply a tracking algorithm in order to identify when people are isolated in the scene. However, for our application, this was not necessary. From the blobs obtained, head and feet points are extracted and stored in memory for later use.

Calibration is then done as follows: Given a projection matrix P the relation between 3D-coordinates $(X, Y, Z, 1)^T$ in the world and 2D-camera coordinates $(u, v, 1)^T$ is $(u, v, 1)^T = P * (X, Y, Z, 1)^T$. As a first step in order to obtain P, the vertical vanishing point (u_z, v_z) is computed. Geometrically, this represents the point where all z-axes from any point in the image meet and can be obtained by connections of all head/feet point pairs extracted in the previous step.

The horizon is geometrically the line formed by all points in which parallel line pairs meet. If we assume a person's feet point coordinate changes only in z-axis compared to the respective head position, the lines between two head positions and their respective feet positions are thus parallel and meet somewhere on the horizon. Doing this for all combinations of detection pairs gives a number of points on the horizon which can then be estimated as the best fitting line for these (see Figure 6 for details).

Given the horizon and the vertical vanishing point, the P-matrix can then be estimated directly as shown in detail in [19].

IV. Event Detection

The automatic detection of targeted events is a crucial part of the system, as it can potentially reduce the workload of the CCTV operators by reducing the amount of visual information that they need to process for their specific surveillance purposes, therefore allowing for a potentially higher privacy-preserving capability on the part of the system.

In the current prototype, trespassing and mugging can be detected automatically, as described below.

A. Trespassing

The trespassing detection system uses a standard tripwire/polygon intersection system to detect whether some entity has entered a restricted area. If the entity fails to leave a restricted area in time, an alarm containing the entity ID is triggered. The tripwires system simply checks for intersections between a set of pre-defined lines and the line from the current object position to the previous one. Based on the sign of the dot product of vectors for the tripwire normal vector and the object displacement, the system can be directionally sensitive so as to trigger only when people have crossing it from a specific direction. This will enable the tripwires to detect people entering a building abnormally, e.g. through exits.

B. Mugging

Detecting muggings is obviously a more complex task. Hidden Markov Models [21], [22] have been widely used for analyzing temporal data such as tracks, indicating the flow of movement of persons, for behavior detection, and are therefore suitable as a basis for a mugging detection system.

Our system relies on HMMs for detecting and classifying the action steps taken by the multiple agents that have been detected in a given scene; such actions steps can be distinguished as follows:

- Intercept: when one entity is approaching another one, or tries to block its way.
- Escape: when one entity is abruptly running away from another one.
- Other: when an entity is standing still, moving normally, etc.

The features for the HMM classifiers are computed from the dynamics of the entities and pairs of entities, and are as follows:

- Distance: the squared distance between the two entities
- Relative speed: the difference in velocity between the two entities
- Absolute long term speed: the weighted average of the last n velocities of the entities of interest (for stability purposes)
- Relative direction: dot product of the direction and the displacement between the current and target entity positions

These features are quantized and the recent history of the quantized features is fed to the individual trained HMMs for classification of the actions step taken by each entity in the given scene (video frame). Then, the current action step taken by a given entity in the scene shall be classified as the action type which is linked to the HMM returning the highest likelihood. The HMMs are trained using the standard Baum-Welch algorithm [23], and the distances are computed using the Viterbi [24] algorithm.

In its current implementation, our system uses a state machine that checks for the occurrence of an interception followed by an escape (with the possibility of an intervening 'other' action type having occurred between them for a small period of time), but replacing that state machine with an extra layer of HMM on the single actions is being considered.

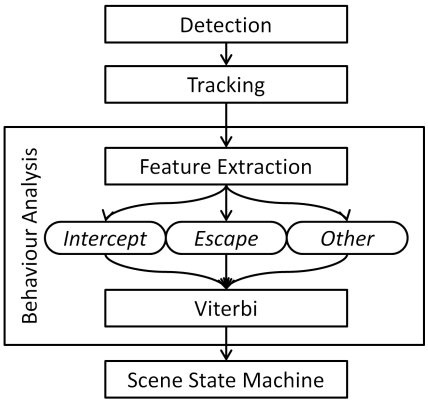

Fig. 7. Overview of the mugging detection architecture: Entities are tracked and features from the tracks are fed into HMM classifiers that send their output to a state machine for event detection.

However, when the video quality is too low to have robust tracks, such a system cannot perform reliably. If the detection of a potential mugging is of such a high priority that the false alarm rate is no longer a concern, an alternative solution would be simply to detect co-occurrences (spatially and temporally) of sudden changes in acceleration of entities in the scene. Although this system cannot be regarded as a mugging detector, it can indeed detect violent muggings. According to our tests, only subtle muggings (that appear to be closer to pickpocketing) could not be detected. Due to its simplicity, this method could be expected to have a high level of false detections. However no false alarms have been triggered during our tests, this being mainly due to the fact that our test set did not feature people running simultaneously in the same area.

Due to the complexity of analyzing mugging events (rising from occlusions, movements, etc.), object tracking algorithms can become very unreliable, and therefore recognizing the muggers from the victims is not possible. In this context, the alarm sent to the rest of the system can store only the 3D position of the location of interest, or store the entity ID of all the individuals involved (i.e. muggers and victims). However, if all the individuals were to be tracked specifically after such an event having been detected, privacy can be a concern, as the victim may not want to be tracked and treated like his aggressors, especially since there are no grounds for tracking him.

Future work for the event detection will involve loitering detection, using behavior analysis algorithms similar to the ones used for detecting muggings. Furthermore, appropriate evaluation will be carried out, especially regarding the impact on precision when using multiple cameras for detecting and locating people in the scene.

Fig. 8. Example of the GUI view for the mugging test sequence. From left to right: 3D model of our campus site with the pedestrian detections of the tracking module, video frame of mugging event with two involved persons and the detected mugging event with labeled pedestrian detections. It is visible that context information of the scenario is affected by the imperfect calibration.

V. Conclusion

The design of Video Surveillance systems will be more and more influenced by privacy related aspects. In this paper we propose a framework to support video operators in their work with respect to privacy protection techniques by design. To ensure a high degree of easy deployment, extendability and conformity to open standards the framework uses a decentralized architecture based on the ONVIF specification. An advantage of our system is the usage of automated camera calibration which allows us to display recognized events and the extracted objects in a 3D model. As seen in the mugging detection example, the accuracy of an automated camera calibration system is sufficient for showing symbolical content of the observed scenarios in the 3D model but for future work we aim to enhance the robustness of the calibration estimation in order to enable overlapping multi-camera settings.

Acknowledgment

This work has received funding under the VideoSense project which is co-funded by the European Commission under the 7th Framework Programme Grant Agreement Number 261743.

References

[1] S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *Computer Vision and Pattern Recognition*, 2007, pp. 1–6.

[2] B. Solmaz, B. E. Moore, and M. Shah, "Identifying behaviors in crowd scenes using stability analysis for dynamical systems," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 2064–2070, 2012.

[3] A. Kuhn, T. Senst, I. Keller, T. Sikora, and H. Theisel, "A lagrangian framework for video analytics," in *International Workshop on Multimedia Signal Processing*, 2012, pp. 387–392.

[4] M. Pätzold, R. Heras Evangelio, and T. Sikora, "Counting people in crowded environments by fusion of shape and motion information," in *International Conference on Advanced Video and Signal Based Surveillance*, 2010, pp. 157–164.

[5] V. Eiselein, D. Arp, M. Pätzold, and T. Sikora, "Real-time multi-human tracking using a probability hypothesis density filter and multiple detectors," in *International Conference on Advanced Video and Signal-Based Surveillance*, 2012, pp. 325–330.

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Robust tracking-by-detection using a detector confidence particle filter," in *International Conference on Computer Vision*, 2009, pp. 1515–1522.

[7] E. Acar, T. Senst, A. Kuhn, I. Keller, H. Theisel, S. Albayrak, and T. Sikora, "Human action recognition using lagrangian descriptors," in *Workshop on Multimedia Signal Processing*, 2012, pp. 360–365.

[8] T. Senst, A. Kuhn, H. Theisel, and T. Sikora, "Detecting people carrying objects utilizing lagrangian dynamics," in *International Conference on Advanced Video and Signal-Based Surveillance*, 2012, pp. 398–403.

[9] I. Haritaoglu, R. Cutler, D. Harwood, and L. S. Davis, "Backpack: Detection of people carrying objects using silhouettes," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 385–397, 2001.

[10] D. Damen and D. Hogg, "Detecting carried objects in short video sequences," in *European Conference on Computer Vision*, 2008, vol. 3, pp. 154–167.

[11] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976 – 990, 2010.

[12] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," in *Conference on Computer Vision and Pattern Recognition*, 1999, pp. 246–252.

[13] R. Heras Evangelio and T. Sikora, "Complementary background models for the detection of static and moving objects in crowded environments," in *International Conference on Advanced Video and Signal Based Surveillance*, 2011, pp. 71–76.

[14] D.N. Serpanos and A. Papalambrou, "Security and privacy in distributed smart cameras," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1678 – 1687, oct. 2008.

[15] T. Winkler and B. Rinner, "A systematic approach towards user-centric privacy and security for smart camera networks," in *International Conference on Distributed Smart Cameras*, 2010, pp. 133–141.

[16] Andrea Cavallaro, "Privacy in video surveillance," *IEEE Signal Processing Magazine*, vol. 24, pp. 168–166, 2007.

[17] "Open network video interface forum, documentation,," http://www.onvif.org/specs/DocMap.html.

[18] T. Senst, M. Pätzold, R. Heras Evangelio, V. Eiselein, I. Keller, and T. Sikora, "On building decentralized wide-area surveillance networks based on onvif," in *Workshop on Multimedia Systems for Surveillance*, 2011, pp. 420–423.

[19] Worapan Kusakunniran, Hongdong Li, and Jian Zhang, "A direct method to self-calibrate a surveillance camera by observing a walking pedestrian," in *DICTA*, 2009, pp. 250–255.

[20] R. Heras Evangelio, M. Pätzold, and T. Sikora, "Splitting gaussians in mixture models," in *International Conference on Advanced Video and Signal-Based Surveillance*, 2012, pp. 300–305.

[21] L. Rabiner and B. Juang, "An introduction to hidden markov models," *ASSP Magazine, IEEE*, vol. 3, no. 1, pp. 4 –16, 1986.

[22] K. Huang, D. Tao, Y. Yuan, X. Li, and T. Tan, "View-independent behavior analysis," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 4, pp. 1028 –1035, 2009.

[23] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B*, vol. 39, pp. 1–38, 1977.

[24] Jr. Forney, G.D., "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.