Machine Vision and Applications (2008) 19:279–290
DOI 10.1007/s00138-008-0152-0

SPECIAL ISSUE PAPER

The evolution of video surveillance: an overview

Niels Haering · Péter L. Venetianer · Alan Lipton

Received: 30 April 2008 / Accepted: 27 May 2008 / Published online: 19 June 2008

Abstract Over the past 10 years, computer vision research has matured significantly. Although some of the core problems, such as object recognition and shape estimation are far from solved, many applications have made considerable progress. Video Surveillance is a thriving example of such an application. On the one hand, worldwide the number of cameras is expected to continue to grow exponentially and security budgets for governments, corporations and the private sector are increasing accordingly. On the other hand, technological advances in target detection, tracking, classification, and behavior analysis improve accuracy and reliability. Simple video surveillance systems that connect cameras via wireless video servers to Home PCs offer simple motion detection capabilities and are on sale at hardware and consumer electronics stores for under $300. The impact of these advances in video surveillance is pervasive. Progress is reported in technical and security publications, abilities are hyped and exaggerated by industry and media, benefits are glamorized and dangers dramatized in movies and politics. This exposure, in turn, enables the expansion of the vocabulary of video surveillance systems paving the way for more general automated video analysis.

Keywords Object recognition · Object-based video segmentation · Video surveillance · Visual tracking · Surveillance system · Scene segmentation · Target detection · Vision system

N. Haering (✉) · P. L. Venetianer · A. Lipton
ObjectVideo, 11600 Sunrise Valley Drive, Reston, VA 20191, USA
e-mail: niels@objectvideo.com; haering@gmail.com

P. L. Venetianer
e-mail: pvenetianer@objectvideo.com

A. Lipton
e-mail: alipton@objectvideo.com

1 Who needs automated video surveillance?

The causes of the need for Video Surveillance are debatable. Regardless of the causes though, governments, corporations, public services and private citizens have spent increasing amounts of money on the protection of borders, critical infrastructure, public transportation, malls, office buildings, parking lots, and homes. And, according to current market research this trend is going to accelerate to about 36.9% annual growth in the security industry by 2009 [13].

As a result more sites deploy more Closed-Circuit TV (CCTV) systems. This has led to a monitoring hazard. The constant flow of video provides data but no actionable information, as illustrated in Fig. 1. Monitoring is tiring, expensive and ineffective. For instance, monitoring 25 cameras 24×7 using human observers costs $150 k per year. Furthermore, experiments run at Sandia National Laboratories for the US Department of Energy study found that: "…such a task, even when assigned to a person who is dedicated and well-intentioned, will not support an effective security system. After only 20 min, human attention to video monitors degenerates to an unacceptable level", see Fig. 1. A practical solution therefore, automates the monitoring process and uses personnel to evaluate and respond to detected events.

This approach and many of the concepts of current video surveillance systems were prototyped under the VSAM (Video Surveillance And Monitoring) part of DARPA's IUBA program [17]. In addition to VSAM, R&D programs in other parts of the world contributed related technologies. In Japan the Cooperative Distributed Vision (CDV) program [CDV1970], and the EU Chromatica and Prismatica programs are such programs. Many of VSAM's design decisions are reflected in ObjectVideo's Video Early Warning (VEW) product, described in Sects. 2 and 3. DARPA's CZTS (Combat Zones That See) program [4] recently combined

Fig. 1 The spread of CCTV cameras leads to video information overload

the VSAM technologies in deployable, reliable, COTS-based multi-sensor systems. Many of the multi-sensor, multi-modal, calibration, robustness and visualization issues CZTS addressed are reflected in ObjectVideo's VEW system (see Sect. 4).

To date, computer vision technologies that deal with automated video surveillance have found significant commercial success in a number of vertical applications. The primary focus for the technology has been in physical security applications – mostly concerning critical infrastructure protection [11]. But there are other application in law enforcement [12]; traffic management; retail loss prevention; market data gathering; and general machine vision.

2 Design issues

With the rapid spread of automated video surveillance it is has become increasingly important to have low maintenance systems. If a system has to be regularly reconfigured, or if highly trained specialist need to spend time with the installation of every single system, the installation and maintenance costs will quickly become prohibitive. To minimize these costs the ultimate intelligent video surveillance system should allow the user to simply provide a video feed to the system, with only a minimal amount of extra information required during setup. Incorrect user configuration can be very costly: false alarms may result in the user ignoring the alerts, though if they are due to incorrect configuration, the false alerts at least

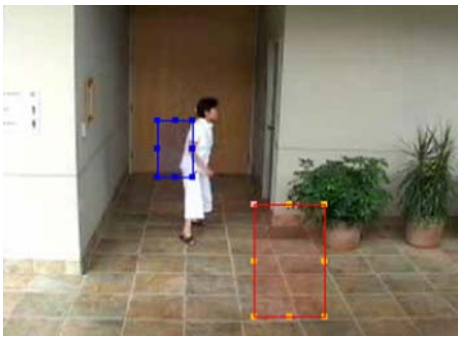

Fig. 2 If calibration is not available, size filters specifying the minimum and maximum allowed size of an object nearfield and farfield help eliminate several false alarms

point out the problem. Missed detections are an even bigger problem, since those may not be discovered till it is too late. For this reason the ideal system should be capable of verifying that the user configuration is sensible. In this section we will describe different approaches to video analysis, and in later section we will discuss some approaches to tackle the problems listed above.

Since this ideal, completely autonomous surveillance system is not yet available, there are several design trade-offs. The following paragraphs will discuss some of these trade-offs, the approach ObjectVideo took, and reasoning behind these decisions.

2.1 Sensor calibration

One of the biggest questions is whether to require calibration or not. Combining calibration information with a site map or site model (even a simple ground-plane model) enables the system to use the absolute size and speed of the detected objects, which helps reduce false alarms. On the other hand, calibrating each camera in the system can be labor intensive, making installation more time consuming and expensive. Incorrect calibration (either caused by user error or by the camera moving after being calibrated) can even hurt system performance.

The ObjectVideo VEW® system operates without requiring calibration. This significantly decreases cost of ownership. To compensate for the lost benefits of a calibrated camera, the system allows the user to define simple size filters: these optional filters tell the system the minimum and maximum size of valid objects, as illustrated in Fig. 2. They are usually applied when the system generates false alerts.

2.2 Systems or components

Video analysis systems are commonly installed as part of a comprehensive security installation integrating video storage, alerting services, different sensors, mapping, camera

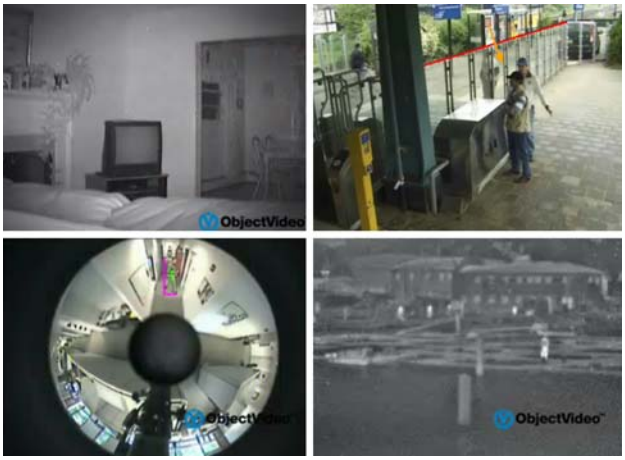

Fig. 3 Common camera types: color, near IR, omni-directional and thermal

Fig. 4 Detecting and eliminating water problems

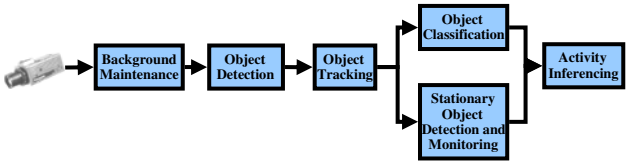

Fig. 5 The major functional components of a video analysis engine

controls, etc. With its background in computer vision processing ObjectVideo decided to focus and excel on the intelligent video surveillance functionality instead of providing a mediocre solution for all components. This approach eases the integration into existing security systems. To support partner integration, the ObjectVideo system includes a complete client SDK that allows partners to define rules, configure the system and receive alerts.

2.3 Supported camera types

A consequence of integrating into existing systems is that video surveillance systems must support a wide range of camera types, see Fig. 3, with no or minimal configuration. The most frequently used camera type is a regular color CCTV camera, but the system can also handle black-and-white, IR, thermal or omni-directional cameras. The resolution is limited only by processing requirements. Most installations still use 320×240 resolution.

Less prevalent omni-directional and IP cameras process up to 1000×1000 pixels.

2.4 Specific versus generic

The ideal automated surveillance system is really generic, being able to detect events of interest in all environments. Certain environments, however, cause special challenges which need to be handled, often in a way applicable only to those environments. One such specific environment is water. The basic characteristics of water are very different from the land, and it has some special problems. For example when monitoring a shoreline, the white waves are often detected as objects, and these waves are frequently tracked for quite a long time, thus fooling salience filters. Specularities also frequently cause problem. For both of these problems it is important to detect them and exclude them from consideration, as illustrated in Fig. 4.

2.5 System functionality

The first generation of automated video surveillance systems were motion detectors, detecting any motion in the camera view. The user may have the option of specifying an area of interest or disinterest. This basic capability already provided significant improvement over manually monitoring the videos, but its usefulness was greatly reduced due to the extremely high false alarm rates caused by such phenomena as shadows, foliage, or small animals. The next step in the evolution was object based video analysis. Detecting and tracking objects allows more sophisticated filtering capabilities, thus reducing false alarm rates and giving more flexibility to the user in terms of defining events of interest. Figure 5 illustrates the major components of a typical object based video analysis system. Such a system builds and maintains a dynamic background model [15,16]. Pixels deviating from the background model statistics are labeled as foreground. These pixels are grouped together into spatial blobs [1],, then tracked [10,2,3,8,9], thus creating spatio-temporal objects. Finally these objects may be classified at various levels, e.g. human, vehicle, animal; moving or stationary; etc. These objects already provide useful information to the security guard, drawing his/her attention to any legitimate moving target.

The usability of the system can be further improved by allowing the user to predefine rules that describe events of interest [6,7]. Such events may include virtual tripwires (alert if an object crosses the tripwire in a predefined direction), an area of interest with an associated action (e.g. an object enters, exits, loiters in, appears in, disappears from, is left behind in the area). These rules may be used in conjunction

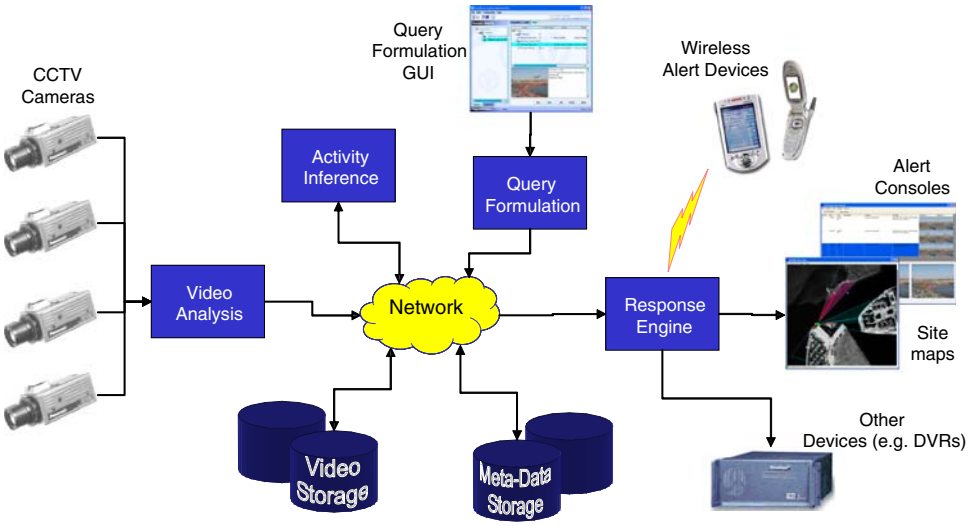

Fig. 6 ObjectVideo VEW® system architecture

with filters like the classification of the object, the size or speed of the object, or a time of interest. More sophisticated rules may analyze actions, e.g. if a person falls, steals something, people are fighting, etc.

As a further step in the evolution of the video analysis system, the system may be able to operate even without a user defining exact rules. A typical surveillance system continuously watches the same view for months or years. This enables the system to deduce what is normal behavior and alert for anything deviating from normal.

The detected events (alerts) are provided the user in any of a number of ways. It can be displayed on a monitor, optionally with a customizable audio signal, it can be sent in an email, it can be forwarded directly to a PDA, it can automatically activate additional security measures, e.g. lock some doors using a dry-contact relay. All these different response types can be customized per rule, so it is possible to have some show up only on the monitor, but others be sent to a PDA. The alerts may also force the guard to acknowledge it, thus eliminating the possibility of the guard not noticing the alert. The alert contains all the information needed for the security personnel to understand the threat and act on it. This includes some snapshots showing the rule that was violated and the object(s) violating it, with all properties relevant to the alert. If the camera location is known in a map, the alert may be illustrated on a map, with the trajectory of the perpetrator. The video from the time of the alert can also be displayed on demand (this is not automatic due to potential bandwidth limitations). The alerts are also stored in a database to allow reviewing and searching them after the fact.

The current version of the ObjectVideo VEW® system requires the user to define events of interest. The related activity inferencing, however, is in a component separate from the object detection. This separation enables additional functionality: efficient forensic analysis. The results of object detection are represented as video description metadata. This metadata can be fed directly to the activity inferencing component, allowing real-time event detection. To support activity inferencing, the metadata should contain all information required by the inferencing process. For a tracked object the metadata can include e.g. the trajectory, the size, the color, the classification, the texture, the rigidity, etc.

In addition, this metadata can also be archived in a forensic storage. Rules can later be applied offline to the metadata, and events can be detected. In forensic mode the time consuming video analysis is performed in real-time, so the forensic analysis relies solely on the stored metadata. This metadata can be analyzed very quickly, and the need for storing high quality video for post-processing is also eliminated. Forensic analysis is a powerful tool for multiple reasons. Its main purpose is to reduce data storage and speed up offline processing. But it can also be used to improve the performance of the real-time surveillance system: the user can experiment with different rules and system settings on the metadata, thus finding the most efficient settings quickly, without having to stage scenarios multiple times. The system architecture is illustrated in Fig. 6.

2.6 Video metadata and event language

The video metadata can be thought of as data stored in a database. To detect events in it, an efficient query language is required. This event specification language is described below.

Traditional relational database querying schemas often follow a Boolean binary tree structure to allow users to create flexible queries on stored data of various types. Leaf nodes are usually of the form “property relationship value,” where a property is some key feature of the data (such as time or name); a relationship is usually a numerical operator (“>”,

"<", "=", etc); and a value is a valid state for that property. Branch nodes usually represent unary or binary Boolean logic operators like "and", "or", and "not".

This can form the basis of an activity query formulation schema. The properties may be features of the object detected in the video stream, such as size, speed, color, or classification. Our system extends this schema in two different ways: (1) the basic leaf nodes are augmented with activity detectors describing spatial activities within a scene; and (2) the Boolean operator branch nodes are augmented with modifiers specifying spatial, temporal and object interrelationships.

Activity detectors correspond to a behavior related to an area of the video scene. They describe how an object might interact with a location in the scene. They detect actions such as crossing a virtual tripwire, entering an area of interest, a person falling. These activity detectors can be combined with property queries using a simple Boolean "and" operator.

Combining queries with modified Boolean operators (combinators) add significant further flexibility. Supported modifiers include spatial, temporal, object, and counter modifiers.

A spatial modifier causes the Boolean operator to operate only on child activities that are proximate/non-proximate within the scene. For example, "and – within 50 pixels of" means that the "and" only applies if the distance between activities is less than 50 pixels.

A temporal modifier causes the Boolean operator to operate only on child activities that occur within a specified period of time of each other, outside of such a time period, or within a range of times. The time ordering of events can also be specified. For example "and – first within 10 seconds of second" means that the "and" only applies if the second child activity occurs not more than 10 seconds after the first child activity.

An object modifier causes the Boolean operator to operate only on child activities that occur involving the same or different objects. For example "and – involving the same object" means that the "and" only applies if the two child activities involve the same specific object.

A counter modifier causes the Boolean operator to be triggered only if the condition(s) is/are met a prescribed number of times. A counter modifier generally includes a numerical relationship, such as "at least n times," "exactly n times," "at most n times," etc. For example, "or – at least twice" means that at least two of the sub-queries of the "or" operator have to be true. Another use of the counter modifier is to implement a rule like "alert if the same person takes at least five items from a shelf."

For example, an illegal left turn can be detected using two directional tripwire crossings, as illustrated in Fig. 7, combined with a target modifier (the same target crossing both tripwires), and a temporal modifier (the horizontal tripwire is crossed first, followed by the vertical tripwire).

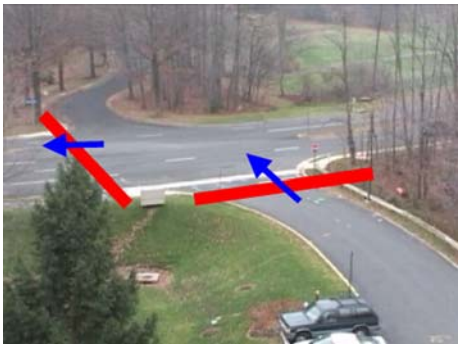

Fig. 7 Detecting a left turn by combining two tripwires

The metadata is not limited to describing only tracked targets and their properties. It can include other information derived from the video source, like lighting changes, camera motion information, scene description (e.g. water regions, foliage, etc.). In addition, the metadata can also include non-video information from other data sources, e.g. RFID tags, card readers, etc. The event specification language can operate on these additional types of metadata, combining them with combinators at multiple levels. For example in an access control application the rule is to detect a person entering without a preceding card swipe within a certain time window.

3 Core capabilities

This section describes the core capabilities of the ObjectVideo VEW® system. The system, as described previously, builds a statistical background model, detects foreground pixels, combines them into blobs, and then tracks those to detect targets. These targets are then classified based on various properties. One of those properties is detecting heads (Fig. 8), greatly increasing the accuracy of human classification. Closely related to detecting a head is detecting the best

Fig. 8 Detected face (*top left*) and best face shot (*top right*)

Fig. 9 Image plane and ground plane areas of interest

face shot (Fig. 8). The detected face can help security personnel identify the perpetrator, or can be fed to a facial recognition system. Or, alternatively, detecting the face enables the system to mask out only the face if privacy is a concern, leaving the rest of the image unchanged.

The ObjectVideo VEW® system supports a large set of rules. A tripwire rule detects if an object crosses the tripwire in a predefined direction (Fig. 10a). The tripwire can be a line segment, or a multi-segment line. For omni-directional cameras arcs and circles are also allowed.

Several rules operate on an area of interest (AOI). The AOI can include the whole frame, or just an arbitrary subset of it. AOI rules can detect objects entering or exiting an AOI (cannot be applied to a full frame AOI), objects appearing or disappearing in the AOI, objects inside an AOI (Fig. 10b) objects stationary in the AOI for a predefined amount of time (e.g. a bag left behind (Fig. 10c), or a parked vehicle), objects taken away from an AOI (e.g. a stolen object), objects loitering in an AOI for a user defined amount of time. Most of these AOI rules can operate either on a groundplane or an image plane. An image plane rule ignores the underlying 3D geometry, just computes the overlap between the AOI and the tracked object. If there is sufficient overlap, the object is deemed to be inside the AOI. In Fig. 9 most of the walking person is inside the AOI (blue area), hence in the image plane mode the person would be detect as inside the AOI. In contrast, if the 3D geometry is taken into account and the blue AOI is assumed to be an area on the ground, then, the person is not inside yet, since his feet are not in the area, so in groundplane mode the AOI inside event would not trigger.

Other rules supported by the Objectvideo VEW® system include detecting when the scene changes (e.g. camera moves), or the lighting changes. The system can detect people moving in the wrong direction, for example when at an airport security area exit somebody tries to enter (Fig. 10e), even if there are big crowds exiting, thus making the detection and tracking of individuals very difficult. The system also allows measuring the crowd density (Fig. 10d), or counting people entering or exiting an area (Fig. 10e).

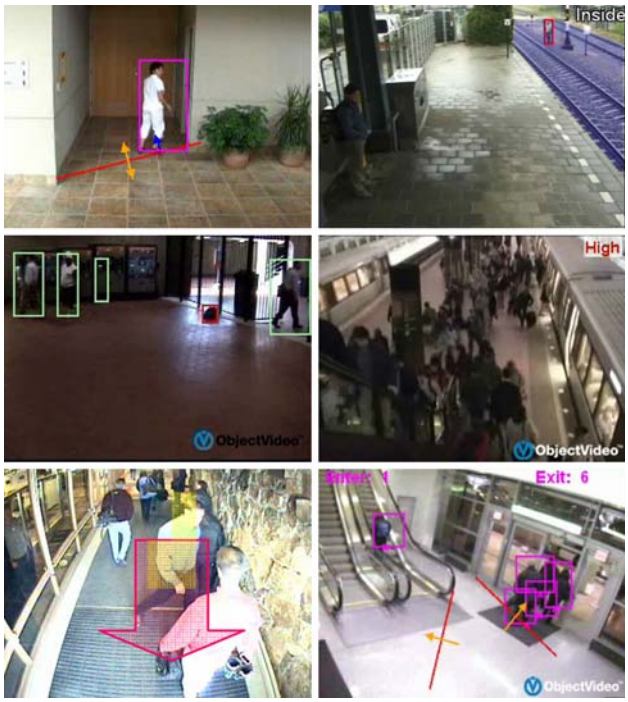

Fig. 10 Some of the capabilities of the ObjectVideo VEW® system: **a** virtual tripwires; **b** object inside AOI; **c** object left behind; **d** crowded density; **e** moving in an illegal direction (flow control); **f** person counting

All this functionality is supported both in real-time and in offline, forensic modes.

4 Extended capabilities

A decade ago the VSAM project advanced the state-of-the-art in robust, real-time video analysis. Since then, about a dozen start-ups and a few established companies have commercialized VSAM's core technology. Now that video surveillance systems are being sold for profit the race is on to sustain a pipeline of innovation. ObjectVideo's research budget for 2006 includes over $5M of government funding and further funding from industry to commercialize, port to DSP, adapt and improve the company's products.

Where is the money going to be spent? Probably not on automatic target recognition, shape-from-x, alignment-based object recognition, or generic sensor fusion methods. There will be incremental progress, toward better detection, tracking, classification and event detection. More functionality will be embedded into cameras, codecs, video routers, and DVRs. Systems will use learning tools to aid the operator, support large sensor networks, and automatically calibrate sensors. Custom solutions will exploit new and special camera types, as well as environmental, temporal, domain and task specific contextual information. The following

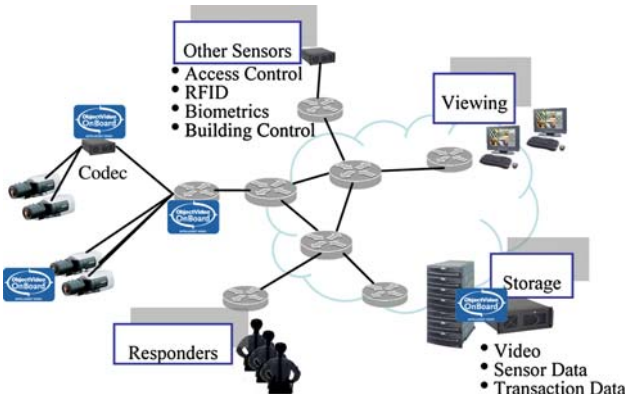

Fig. 11 Components of the video surveillance ecosystem that host ObjectVideo OnBoard functionality

subsections describe ObjectVideo's efforts toward these challenges.

4.1 Embedded systems

Embracing the demand for smaller form factor implementations ObjectVideo ported the core capabilities and many of the solutions discussed in this Section to an embedded platform, sold as OnBoard®. This opened up new parts of the security ecosystem to video surveillance technology. For instance, smart cameras, video codecs, or video routers may keep video data off congested communication channels unless events of interest are unfolding, see Fig. 11. A policy that considers the event class, the threat level and the available bandwidth may decide to stream a video feed or to transmit merely a short alert image sequence. If communication bandwidth is not an issue embedded video surveillance technology can activate or deactivate Digital Video Recorders (DVR), thus significantly increasing their effective capacity.

4.2 Unusual event detection

Jianbo Shi from UPenn and ObjectVideo jointly extended an unusual event detection method [14] to integrate feedback from the operator. Operators specify perceived threats in the usual way, but add an unusual event detection backdrop rule, to detect events that would have gone unnoticed otherwise.

The operator can re-label detected unusual events as 'usual'. Incorrectly labeled events are used to update the unusual event model and to achieve better performance in the future. Unusual events can be further labeled as 'irrelevant', 'suspicious', or 'threatening', and tagged with a text description of the event. The labels allow the ranking of events by priority. Text descriptions provide contextual information for similar events in the future.

An alternative approach to unusual event detection is based on target property maps. These maps gather target properties

Fig. 12 Detection of speeding vehicles using target property maps

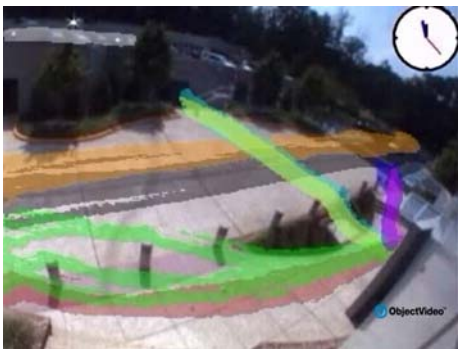

Fig. 13 Target paths detected in an urban scene

for each target type, image location and time of day/week/year. This historical information is then used to enable context sensitive rules, such as the detection of vehicles traveling above 99 percentile of the speeds of targets of the same type at the same location and time of day, see Fig. 12.

Another alternative is based on the extraction of target paths. Target behavior is observed and combined into trajectories. Trajectories, in turn, are grouped by target type, entry and exit point. All trajectories of a given target type originating at the same entry point and terminating at the same exit point are fused into a path, see Fig. 13.

Target path information can be used to detect targets that deviate from paths, see left of Fig. 14, to detect targets that originate from infrequently used entry points, see right of Fig. 14, to detect targets that travel on established paths but whose type differs from that associated with the path, or to detect targets that travel along a path at an unexpectedly fast or slow pace.

4.3 PTZ camera support

Covering large areas and perimeters requires large numbers of static cameras. This incurs high costs for static cameras or leads to poor resolution – often both. PTZ cameras offer an attractive alternative, by offering a high-resolution view of a specified area of interest. Below we describe an approach that

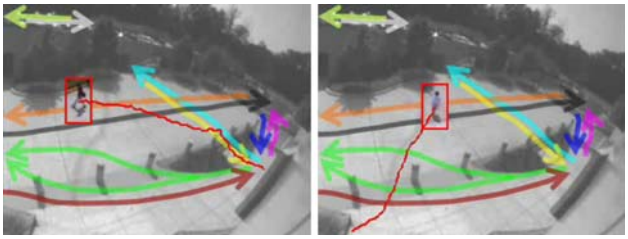

Fig. 14 Detection of targets deviating from normal paths (*left*) and target entering the scene at an unknown entry point (*right*)

Fig. 15 Fields of view of the Leader camera (*left*) and the Follower camera (*right*)

augments existing static camera networks and an approach that replace them.

4.3.1 Leader/follower systems

Static cameras and PTZ cameras can be combined in a leader/follower configuration. Initially both cameras are calibrated with respect to each other. Rule violations are then determined on the leader camera as usual. The follower can be active or passive. When a target of interest has been detected by the leader, it guides the pan, tilt and zoom parameters of the passive follower to provide a high-resolution view of the target to the operator, see Fig. 15. An active follower, on the other hand, autonomously detects and tracks targets designated by the leader.

4.3.2 Scanning cameras

For very large coverage areas the assumption that a leader camera can detect targets reliably is unrealistic. The scanning camera approach depends solely on PTZ cameras. Most PTZ cameras allow the specification of a scan tour that consists of specified waypoints that are visited at specified times, speeds and zoom-levels. Simple scan paths may trace a fence or border, more complex patterns include raster scans of large areas.

For situational awareness it is important that the area viewed by the PTZ camera be geo-registered to enable the specification of target position in Cartesian (latitude/longitude) or polar (bearing/distance) coordinates. Unfortunately, motion estimates from cameras and controlling software are not

Fig. 16 A frame from a scanning camera. While the frame contains discernible texture it does not provide location information to a security operator

standardized and too imprecise to provide accurate position estimates. While successive frames may contain discernible structures they may not be unique enough to provide positional information, see Fig. 16.

Mosaics of the scene are therefore built using visual motion estimates. These mosaics relate the current viewing angle to the entire scan path, enable geo-registration and thus target position estimation. An additional advantage of mosaics is that they facilitate background subtraction for target detection.

4.4 Automatic scene understanding

Environmental, temporal, domain and task specific contextual information can simplify the core video analysis componenents significantly:

- Special detection algorithms can be used at nighttime if we know the time of day
- Tracking targets in the field of view of the camera now, can benefit from the knowledge of historical object paths.
- Classification can benefit from domain specific information: Don't look for watercraft in an airport or urban environment. Distinguish between people and vehicles through knowledge of relative object size. Use water regions to tell land vehicles from vessels: Note that knowledge about the water regions obviates the need for classification
- Event detection can be sensitized through the integration of temporal and historical object information: Even though people may enter a warehouse regularly during business hours, the same event may warrant an alert at night.

Note that the exploitation of task specific information is core to automated video surveillance applications. When an operator specifies that targets shall not, say, cross a given tripwire, there is no need to analyze the behavior of targets far

from the tripwire. Airports, seaports, fuel processing, energy production, perimeters, or urban areas provide valuable domain information. Scene (e.g., water, sky, grass, tree, human-made structure regions) and weather information constrains the possible class types, determines the need to handle shadows, and enable context-sensitive event detection, such as 'Person entered building' as opposed to the context-free alternative: 'Person disappeared'. Temporal information help tell expected from abnormal behavior, or provide weekday/weekend or seasonal information.

4.5 Multi camera networks

Most intelligent video installations include several cameras providing coverage for the area. These cameras are usually treated independently, with rules set separately for each of them, all of them alerting independent from each other. It is also possible, however, that these cameras share their information, allowing cross-camera tracking of moving objects. The cameras of a cross-camera setup need not overlap, though the further away cameras are from each other, the less reliable the cross-camera tracking becomes.

Cross-camera tracking can be implemented either in a distributed manner, with the individual cameras talking to each other, or in a centralized way with a central camera fusion engine combining the information from all the cameras.

The centralized method has all the cameras operating as normal, independent from the other cameras, detecting and tracking targets and even detecting events defined on those individual cameras. At the same time these individual systems send all their object information (basically the video metadata) to a centralized video fusion engine. This engine fuses the information together based on the synchronized time stamps and the camera calibration information, combining the multiple representations of objects which are in the overlapping camera field of view, and building object trajectories even over non-overlapping field of views. This camera fusion engine has its own activity inferencing with a standard set of rules, but operating on a map. This means that for example in a perimeter protection application with dozens of cameras covering the perimeter, the user can set up a single rule on the map marking the whole perimeter instead of setting a separate rule on each camera. The map also allows setting rules that is not possible on the individual cameras, like a rule to detect a vehicle circling suspiciously around the perimeter. Without cross-camera tracking such loitering couldn't be detected. Intergraph, Boeing and others sell Graphical User Interfaces (GUIs) that enable the display and control of video feeds. Many also map the fields of view of the contributing cameras on a satellite image, as shown in Fig. 17.

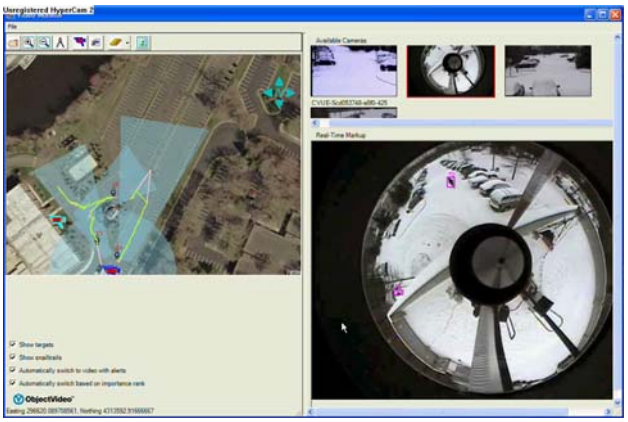

Fig. 17 A graphical user interface that maps the contributing cameras' fields of view to a satellite image of the site and enables the display and control of the cameras

4.6 Self-calibrating sensor networks

In Sect. 2.1 we argued in favor of uncalibrated systems to avoid the high cost associated with manual calibration. Recently, however significant progress has been made on automatic calibration methods. ObjectVideo has developed a set of self-calibration methods for individual static cameras, pairs of static cameras with overlapping FOVs, individual PTZ cameras, and omni-directional cameras. The aim of these methods is to achieve automatic calibration, mensuration and/or geo-registration. The methods exploit knowledge about the domain, the environment, time of day, information about objects with known sizes, locations, or velocities, and calibration information broadcast by objects in the scene.

5 Addressing privacy issues

One of the major objections regularly raised against intelligent video surveillance is its effect on privacy. We are very sensitive so these issues and are working hard to allay fears. Overall we feel that while the proliferation of cameras does raise some legitimate privacy concerns, using automated video surveillance instead of people monitoring the video feeds eliminates most privacy issues. In addition, in some applications it is important to mask out some portions of the image. The whole moving object, or just the face of the person can easily be masked out or pixellated, as illustrated in Fig. 18.

It is also possible to support different alerts for different user types. For example in a home monitoring system all objects may be masked out from the alert sent to a central monitoring location – this still allows the people monitoring the alerts to decide whether it was a person or just a pet; on the other hand, the system can store the original imagery

Fig. 18 Masking out the face of a person to address privacy concerns

locally, so when the responders go to the scene and discover a break-in, they can get access to the images or even video showing the perpetrator.

6 Performance evaluation tools

One of the biggest challenges of developing a commercial video surveillance system is that the system, once deployed, has to operate robustly 24/7 in a completely uncontrolled environment, in a wide range of scenarios. The only way to ensure robust and reliable performance in all these environments is to perform extensive testing. To make this problem tractable, ObjectVideo developed a number of internal tools, enabling the testing of the various system components, and automating as much of the testing as possible.

6.1 Test data

The most crucial component enabling all the testing is the availability of appropriate testing data. Ever since the company started in 1998, it has been collecting videos from all kinds of scenarios, with various camera types, covering a wide range of activities, weather conditions, and sceneries. By now we have compiled a video database of several terabytes. We also have several live cameras mounted around the ObjectVideo headquarters, allowing live testing. In addition, we developed tools that allow us to create artificial videos with arbitrary camera positions. These are particularly useful for initial testing of the algorithms.

To make all this video data really useful for testing, associated groundtruth data is also important. There are several levels of groundtruth data. To test overall system performance, the groundtruth only needs to contain when an event of interest occurs. For example when testing the tripwire detection, all that matters is whether the system alerts when an object crosses a tripwire; whether the objects are properly detected and tracked when not crossing the tripwire is irrelevant.

Fig. 19 Left column: Overlapping views from multiple synchronized cameras observing the same virtual scene. Right column: A scenario at different times of day: midday (*top*), evening (*middle*) and night (*bottom*)

To test the details of the video analysis algorithms, however, the groundtruth needs to be more detailed than just containing the events of interest: it has to contain the detailed spatio-temporal description of all objects, e.g. trajectory, size, classification. The artificial videos have the advantage that they by definition have all associated object level groundtruth, and even the camera calibration information. Event level groundtruth can easily and automatically be generated from the object level data. Throughout the years Objectvideo has compiled a large amount of groundtruth data, both at the event and the object level.

ObjectVideo developed a Synthetic Video Generator that we use to test active and multi-camera setups. The ability to repeat test scenarios is an invaluable asset when testing and evaluating, object handoff between cameras, multi-object interactions, or active tracking of moving objects. The left column of Fig. 19 shows overlapping views from multiple synchronized cameras observing the same virtual scene.

The right column of Fig. 19 shows the same scene and scenario at different times of day: midday (top), evening (middle) and night (bottom).

Figure 20 shows how omni-cameras can be simulated via the tools camera parameters.

Fig. 20 Omni-cameras can be simulated via the tool's camera parameters

6.2 Groundtruth generation tools

We developed tools to generate the different kinds of data required for testing, as described in the previous section. While these tools make these tasks as easy as possible, generating groundtruth data for real videos is still time consuming and mechanical.

To generate event level groundtruth, the user has to select the video and the rule of interest, then play the video. If an event is observed, its time has to be logged by clicking a button. As a result, the groundtruth file will contain the name of the video, the rule, and a list of times when an event is expected. For each event time a time range may also be defined, allowing some flexibility for the detection. The location of the even can also be defined, which is particularly useful in busy scenes to guarantee that an unrelated incorrect detection at the same time doesn't mask missing the real event of interest.

Object level groundtruth is more labor intensive to generate. For that each individual object has to be groundtruthed for all times. Our tool allows marking a target with a bounding box at any given time, basically providing key frames, and interpolates the target the other times. The user can play the video with the target being marked up continuously, and when the markup deviates where interpolated, new key frames can easily be added.

6.3 Data evaluation tools

As described in earlier sections, the two major components of the ObjectVideo VEW® system are the video analysis engine and the activity inferencing. Both of them can run offline, either with a GUI or in batch processing mode.

The offline video analysis engine taking a video stream as an input, with an optional framerate specification, generating metadata as its output, and optionally all kinds of internal data, like background models, foreground masks, blob masks, etc.

The offline activity inferencing takes the metadata and the rules as input, and generates events. At an even more basic level, this same tool can be used even without rules, just playing back the video and the metadata, providing visual verification of the metadata, and allowing the creation of marked up videos for marketing purposes. It can also take event level groundtruth as an additional input, and generate performance statistics in the form of detection and false alarm rates. This same tool is also used to troubleshoot problems encountered in the field by operating on the forensic metadata and the rules defined by the user.

The tool used to create the object level groundtruth can also be used to evaluate it. For that the manually generated object level groundtruth and the automatically generated metadata has to be loaded at the same time and the tool compares them, quantifying differences, in detection, tracking and classification. Detection metrics quantify precision and recall for object detection and groundtruth-to-actual detection overlap. Tracking metrics quantify precision and recall for correct track-to-object association, track length and track deviation. Classification metrics quantify precision and recall for each object type and the corresponding confusion matrix.. In addition, both the groundtruth and the metadata can be stored in Matlab format, allowing easy additional processing and evaluation in Matlab.

6.4 Tools for the field

When designing a system for a customer site, one of the crucial questions is: how many cameras are required to provide the necessary coverage. The camera placement tool helps with the answer, by interactively demonstrating the field of view of the camera for specified parameters, or even suggesting camera parameters to achieve the best coverage.

7 Data

As of 2007, OV has over 223 paid customers with deployments at 318 sites in 29 countries, representing 773,141 committed VEW (server-based) and OnBoard (DSP) licenses. ObjectVideo's direct customers include US Customs, oil refineries, and commercial air and sea ports as well as resellers. Since the release of the DSP based products, indirect sales via resellers have eclipsed direct sales. As a result of site-surveys, field-deployments, and the exposure to customers and their requirements, ObjectVideo has accumulated a significant video data set, see Fig. 21.

Enabling the user to configure powerful algorithms to analyze video and detect specified or unspecified unusual events carries a significant risk of 'operator error'. For instance, allowing the operator to manually geo-register a camera's field of view through the selection of corresponding features in

Fig. 21 A sample of data from ObjectVideo's commercial deployments

the video feed and a satellite image runs the risk of achieving very poor calibration that will prevent the system from functioning properly. Education of the operator, robust automatic algorithms, and automatic system messages that warn the user of common hazards, and the exploitation of contextual information are necessary to ensure that the system works reliably and as intended.

Acknowledgements It is impossible to discuss VSAM, ObjectVideo's VEW product and CZTS without mentioning Tom Strat. Now that he is neither with ObjectVideo nor with DARPA we would like to acknowledge Tom's academic and industrial contributions to automated video surveillance. He managed DARPA's VSAM program in 1996 to explore core video surveillance capabilities. In 1998 Tom co-founded ObjectVideo and served as its CTO until 2001. Back at DARPA he pioneered the deployment of large multi-modal sensor networks under DARPA's CZTS program.

References

1. Boult, T.E., Micheals, R., Gao, X., Lewis, P., Power, C., Yin, W., Erkan, A.: Frame-Rate Omnidirectional Surveillance and Tracking of Camouflaged and Occluded Targets, Proc. Workshop on Visual Surveillance, Fort Collins, CO, June (1999)
2. Cohen, I., Medioni, G.: Detecting and tracking moving objects for video surveillance, in Proc. IEEE Computer Vision and Pattern Recognition, Fort Collins (CO), USA, June (1999)
3. Comaniciu, D., Ramesh, V., Meer, P.: Real-Time Tracking of Non-Rigid Objects using Mean Shift, IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, SC, pp. 142–149, (2000)
4. DARPA program: Combat Zones That See, (2003)
5. Cooperative Distributed Vision. http://vision.kuee.kyotou.ac.jp/CDVPRJ/
6. Qian, R., Haering, N., Sezan, I.: A Computational Approach to Semantic Event Detection. CVPR , pp. 1200–1206 (1999)
7. Haering, N., da V. Lobo, N.: Visual Event Detection, Kluwer, (2001)
8. Isard, M., Blake, A.: CONDENSATION – conditional density propagation for visual tracking. Int. J. Computer Vision. **29**(1), 5–28 (1998)
9. Isard, M., MacCormick, J.: BraMBLe: A Bayesian multiple-blob tracker, Proc. ICCV, (2001)
10. Kalman, R.E.: A New Approach to Linear Filtering and Prediction Problems, Transactions of the ASME–Journal of Basic Engineering. 82, Series D, pp. 35–45 (1960)
11. Lipton, A., Heartwell, C., Haering, N., Madden, D.: Automated Video Protection, Monitoring & Detection. IEEE Aerospace and Electronic Systems Magazine. **18**(5), 3–18 (2003)
12. Lipton, A.: Intelligent Video as a Force Multiplier for Crime Detection and Prevention. IEE International Symposium on Imaging for Crime Detection and Prevention. pp. 151–156. London, (2005)
13. Research And Markets, Report on Closed Circuit TV Industry —A Market Update (2005–2008), 2005
14. Zhong, H., Shi, J. Visontai, M.: Detecting Unusual Activity in Video, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), (2004)
15. Stauffer, C., Grimson, W.: Adaptive Background Mixture Models for Real-Time Tracking, IEEE Conference on Computer Vision and Pattern Recognition, (1999)
16. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: Principles and Practice of Background Maintenance, International Conference on Computer Vision, pp. 255–261, (1999)
17. Video Surveillance And Monitoring, part of DARPA's Image Understanding for Battlefield Awareness (IUBA) program, (1996)