Video Content Analysis with Effective Response

David Abrams
TrueSentry
111 N Chestnut St., Suite 200
Winston Salem, NC 27101
dave.abrams@truesentry.com

Steven McDowall
InterAct Public Safety Systems
111 N Chestnut St., Suite 200
Winston Salem, NC 27101
smcdowall@interact911.com

***Abstract*—Video content analytics effectively identifies threats in video surveillance camera feeds. These behaviors include person/vehicle entering restricted zone, counter-flow detection, loitering, object left behind, and others. In order to provide enhanced security, events need to be integrated with a command and control system capable of effectively responding to hundreds of events per day in a busy, critical infrastructure facility.**

We describe a novel system – *NerveCenter* – that links analytic events with command center data wall camera pop-ups on alarm, extended notification tools that send multi-modal alerts, acknowledgement tracking, 911 computer aided dispatch (CAD), and geo-coded mapping tools that give operators a tactical map. The usability of these tools is discussed along with how to provide situational awareness and a common operating picture to operators and first responders.

***Index Terms*—computer vision, tracking, video surveillance, collaboration, notification, command centers**

INTRODUCTION

VIDEO content analysis enables a small security force to monitor a large number of cameras by focusing attention on probable threats. Vision algorithms identify suspicious behaviors, and then audible and visual cues notify staff in the monitoring station. Alerts of suspicious activity are sent to remote personnel. Intelligent threat detection and response tools provide a force multiplier by enabling a small team to better cover more cameras and sensors.

Video content analysis (VCA) has successfully demonstrated that it does reduce false alarms and provide more actionable alerts than basic video motion detection. In fact, motion detectors have so many false alarms that they are often ignored and are only used for reviewing motion-only archives. Despite these productivity gains, VCA still lacks a comprehensive architecture that puts analytics into the complete incident management life cycle.

An airport, university campus, convention center, or government building with hundreds of cameras will still produce thousands of analytic events in a given week. For example, an airport may have hundreds of object-left-behind analytic events in a given week. VCA by itself does not scale-up to large critical infrastructure facilities, public safety, major sporting events, homeland security, or border monitoring.

NERVECENTER

Homeland security systems need to couple analytics with effective response tools at each stage of the incident management life cycle. These tools more efficiently handle the flow of raw sensor data, filter alarms, and qualify incidents.

The seamless integration of tools at each stage of the incident management life cycle is essential to homeland security and public safety systems. We describe an integrated system called *NerveCenter* that is designed to:

- Lower the unit cost of each incident in the cycle with efficient filtering and remote verification.
- Focus attention on critical incidents, thereby avoiding nuisance alarms.
- Enable faster response time through reliable notification and acknowledgments.
- Provide enhanced collaboration between field personnel and the *NerveCenter* during an incident.
- Create an incident audit trail with centrally archived video for long-term storage and forensic analysis.

The goal is to ensure a common operating picture among all participants. The incident management life cycle is illustrated in Figure 1 and detailed in Table 1.

1. SURVEILLANCE

1.1. Broad Camera Coverage

Broad coverage is essential to a surveillance system. It needs to digitize the existing installed base of analog CCTV cameras to compressed video (e.g., MPEG-4, H.264, M-JPEG) and connect to network cameras. Rate adaptive video is required to handle fluctuations in bandwidth on wireless camera networks, geographically distributed cameras on a wide area

network, and Internet-addressable cameras. Digital video recorders (DVRs) from different manufacturers are integrated into one system so the user has a standard interface across vendors.

Table 1 - Incident Management Life Cycle

1. *Surveillance*—cameras and sensors gather and record data.
2. *Analytics*—filter sensor data, identify behaviors, categorize targets, and potential threats, and create alarm events.
3. *Rules*—rules engine applies conditional logic/scheduling to events, then executes a user-defined action set.
4. *Verification*—remote video verification and escalation of an alarm to report an incident.
5. *Dispatch*—assign response units via geographic mapping and location.
6. *Notification*—multi-modal notification of incident to remote personnel.
7. *Response*—team collaboration, instant conferencing, Web-based situation center with incident video.
8. *Resolution*—audit trail of incident closure and long-term video archive.
9. *Review*—forensics, reports, data mining of incident patterns for process improvement.

1.2. Mobile Units

Mobile units like police cars have dynamic IP addresses. A roaming mode flow control algorithm is used to upload video to the station network video recorder (NVR) over a cellular or wireless network. Mobile units that periodically come into range of a stable Wi-Fi network use video store-and-forward to upload to the NVR.

1.3. Global Camera Registry

A global camera registry is maintained by the system to manage access to cameras. A central server handles single sign-on, maintains session tokens, and enforces access control lists. The administrator assigns user access rights to cameras. Health monitoring tracks the live/down status of cameras, bandwidth use via SNMP, uptime of each network device, and diagnostics and error logs for each camera.

1.4. Sensors

An extensible software development kit (SDK) is used to rapidly integrate new device controller and sensor interfaces. These include access control entry/exit, fire alarms, RFID tracking, motion/infra-red, pressure, biological and chemical agent, radiological, digital I/O, and supervisory command and control (SCADA) systems. Each controller translates sensor state changes into events on the system-wide message bus.

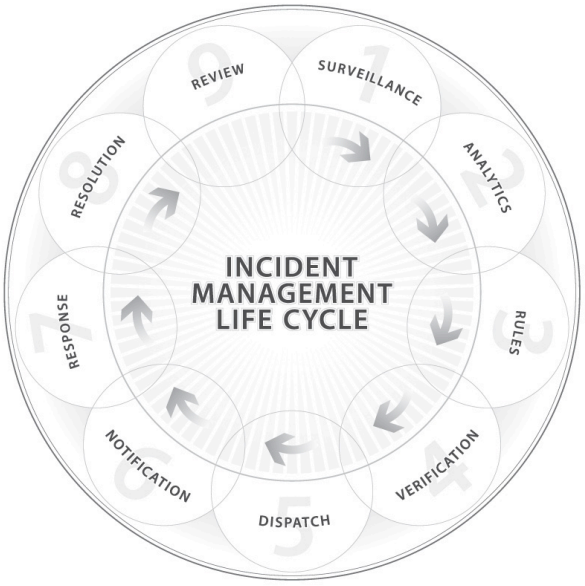

Figure 1- Incident Management Life Cycle

1.5. Incident Archive

When an incident is reported, the video segment from that camera is uploaded to a central incident archive. This ensures long-term archival of the essential video from the start/end times of the incident. Typically, a DVR in the field records all motion-video and has a fixed storage quota of 14-30 days. We ensure long-term, protected access to the incident video by storing a digitally signed copy in the central incident archive.

1.6. Scalable Video Proxy

First responders need temporary access rights to the cameras and archives from the outside network. Typically, the capacity planning of the protected surveillance network and DVRs does not account for a rapid increase in viewers during an incident. A scalable video proxy multiplexes a single stream from the DVR to the first responders outside the firewall. The proxy also manages temporary access rights and transcodes video.

2. ANALYTICS

Video content analysis (VCA) algorithms identify specific behaviors in surveillance video. Outdoor scene algorithms have to handle lighting changes, reflections, shadows, weather (snow and rain), moving clouds, water waves, and flags and trees blowing in the wind. The tracker [1] must re-acquire an object that has been occluded, track multiple objects that merge into one object, and then split. Embedded analytics in the camera can enhance accuracy [2].

2.1. Vision

Multi-modal background modeling [3] can be an effective method for learning repetitive background motion. Foreground segmentation is used to identify candidate objects. Multiple hypothesis tracking [4] is often used to match candidate objects with those currently being tracked.

An object's geometric moments like size, position, velocity, and acceleration are used to classify [5] it as a person or vehicle. Each tracked object is compared to the set of polygonal regions, called zones, associated with analytic behaviors in the field-of-view.

2.2. Behaviors

Tracked objects are classified, and then multiple tracks are analyzed over time. A behavior is then identified in the context of one or more zones including those listed in Table 2.

Table 2 - Analytic Behaviors

Movement
- Entered/exited restricted zone
- Started/stopped moving
- Started/stopped moving in wrong direction
- Merged/split
- Movement between zones

People
- Person started/stopped running
- Person trespassing
- Person loitering
- People talked
- People passed by

Perimeter
- Trespass line breached
- Person on fence line

Crowd
- Counter flow: person/vehicle moving in wrong direction
- Crowd density estimation
- People capacity exceeded

Object
- Object left behind
- Object removed from zone

Traffic
- Car entered/exited lot
- Car counted in lane
- Vehicle speeding
- Car parked in restricted area
- Car needs assistance
- Car pulled off the road
- Car parked in handicapped zone
- Car made an illegal U-turn

Tracking
- Follow moving targets with pan/tilt/zoom
- Object tracked across multiple cameras

2.3. Analytic Events

Analytic events are sent from the smart camera or DVR to the central server. XML is used to describe each analytic event such as hose listed in Table 3.

Table 3 - Analytic Event XML

```
<analytic-event type="[behavior code]" name="[behavior
name]" id="[unique ID]" time="[time code]"
severity="[0 to 100]" alarm="[true/false]"
camera="[camera ID]">
  <tracked-objects>
    <object id="[unique ID]" type="[object, person, car,
    vehicle]" >
      <boundary x="[int]" y="[int]" w="[int]"
      h="[int]" />
      <moments m00="num" m01="num" m10="num"
      m11="num" m20="num" m02="num"/>
    </object>
  </tracked-objects>
  <zones>
    <zone id="[zone ID]" name="[zone name]"
    type="[restricted, exclusion, direction, trespass,
    fence, . . . ]" >
      <boundary x="[int]" y="[int]" w="[int]"
      h="[int]" />
      <points>
        <point x="[int]" y="[int]" />
      </points>
    </zone>
  </zones>
</analytic-event>
```

Each event specifies one behavior, a set of objects that are involved, and a list of zones. For example, two people loitering outside a building would create an event of type "Loitering" and list the object of each person involved and the zone in the scene that contained them.

The events and object tracks are stored in a database. A query might be for all "Object-Left-Behind" events. The object ID is used to get the track history of that target so the user can see what the target did before and after leaving the object at the scene.

2.4. Crowd Analysis

Crowd analysis algorithms use a type of flow control [6] instead of tracking individual objects because in a very busy scene there are too many object merges/splits for object tracking to be accurate. A good example is trying to identify a person entering the exit to the sterile area of an airport. Counter flow detection [7] identifies a concentrated flow of pixels against the main flow.

2.5. Multi-Camera Tracking

Pan/tilt/zoom tracking follows a moving object with a robotic camera, centering that moving object in the field-of-view as it moves. This automatic tracking with optical zoom gives a close-up view of a target in a perimeter monitoring scenario. The pixel image plane <x, y> must be transformed with the physical position of the camera, pan/tilt orientation, and

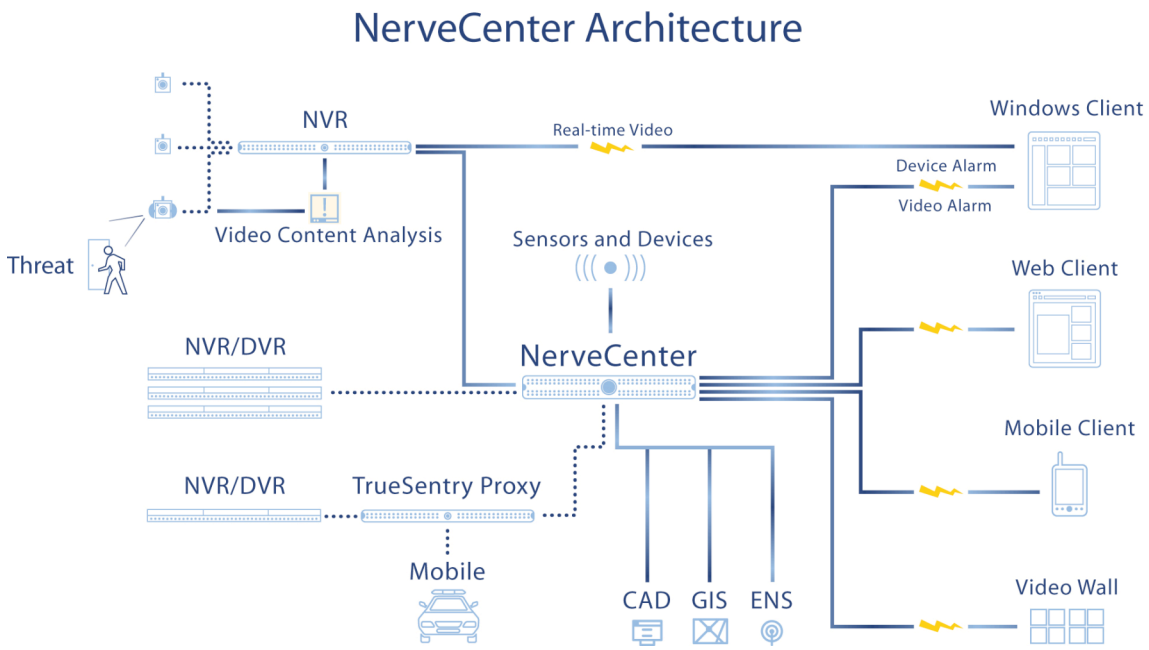

Figure 2 - *NerveCenter* Architecture Diagram

optical zoom field-of-view. Tracking a single object from one fixed camera view to another camera [8] uses a similar transform.

3. RULES

The rules engine performs centralized processing of events and routing to distributed systems. It captures analytic events from cameras, fire safety points, access control entry/exits, RFID, and related sensors. Each event is compared to a user-defined set of conditions using Boolean logic. A schedule can be used to filter out alarms during the day and only consider them at night and on weekends.

Once events are qualified, the rules engine runs an action set. A programmable sequence of actions is used to communicate with other services that handle dispatch, notification, and geographic mapping. The service-oriented architecture is designed to be extended for rapid integration of systems.

4. VERIFICATION

Although the rules engine can directly create incidents, video verification is often used to quickly filter out false positives. Video verification of an alarm event must be fast and easy for a user to do before escalating it to an incident.

4.1. Desktop

Users can configure local alarm settings. Camera pop-up windows are used to augment a dispatcher's workstation. One mouse click is used to remove the video (acknowledging the alarm) or escalate to an incident. Audible sounds describe the alarm event. Map overlays show the real-time status of cameras, sensors, and alarms.

4.2. Mobile Video

Alarms are sent to mobile phones and PDAs. A user can view a snapshot of the event over a mobile Web browser or view live video on the mobile device. A rate-adaptive streaming video algorithm is used to dynamically adjust the bit-rate. When network congestion increases, the bit-rate or frame-rate is reduced.

4.3. Data Wall

Command centers use large data walls to display important camera views, geographic maps, tactical and facility maps, weather, real-time sensor data, traffic statistics, supervisory command and control (SCADA), and teleconferencing. They are also used to display a remote user's desktop. The data wall gives everyone in the command center access to critical information.

A data wall is an integrated multi-monitor display that displays a unified coordinate space. For example, if each cube is 1024x768 then a 5x3 data wall is 5120x2304 pixels.

4.4. Video Rendering

High performance video rendering on the data wall is important. Analog video is routed to an encoder card, which digitizes it into raw 32-bit RGBA. The data wall controller does not send raw video across the host processor PCI bus. Instead, video data is sent directly from the encoder card to the graphics card memory. It sends commands to the encoder and graphics card to direct the data flow, crop windows, and rescale video using hardware acceleration.

4.5. Network Video

Significant performance issues need to be addressed when decoding compressed MPEG-4, H.264, or M-JPEG video on the data wall. Although a multi-processor host system optimized with SIMD instruction set can decode multiple streams, it still has the PCI bus transfer problem. Careful network capacity planning needs be done to ensure that the Gigabit network switch does not suffer from high packet loss.

4.6. Layouts and Pop-ups

A remote operator arranges windows, analog, and network camera sources on the data wall. Layouts are used to memorize arrangements of cameras, maps, and other windows. The data wall can switch to a pre-defined layout for handling a specific type of incident. The rules engine sends camera popup commands to the data wall on alarm/incident to display an analog or network video feed.

5. DISPATCH

The creation of a new incident begins a workflow that includes: additional information gathering, dispatching of resources to handle incident, monitoring of incident progress, and closure of an incident. A Computer Aided Dispatch (CAD) system manages the life cycle of all incidents that require resources to be allocated or dispatched.

5.1. Computer Aided Dispatch

Incident Creation—Incidents may be created by a variety of means including a 9-1-1 or similar system, manual entry, or automated entry via an API. The latter is how the video surveillance sub-system initiates a new incident once the various rules and possible human review takes place. At this point the incident is placed in a queue and assigned a priority for the dispatcher to handle.

Information Gathering—The exact location of an incident should be determined to eliminate any chance of ambiguity or error. Any initial remarks or comments by any persons on the scene could be crucial as well as comments from the dispatcher based on the video images available (both original images and on-going). The event type is an important short code that describes that incident, and it is critical to dispatching resources.

Dispatch Response Units—This is arguably the most difficult step in incident management. Given N active incidents and Y resources, which ones are the best resources to allocate to a given incident? The CAD system continuously monitors the availability status of all units. With the addition of Automatic Vehicle Location (AVL) – a GPS unit attached to a resource – the CAD can also monitor the real-world location of resources. Each event type may require different types of resources to be dispatched to handle the incident. Based on the major factors that may include event type, location of incident, priority, availability of unit, and location of units, the dispatcher dispatches one or more units to the incident using suggested unit recommendations from the CAD system.

Track Progress—The dispatcher can then monitor the incident and show the progress of the response on the data wall. Updates to the incident can happen without dispatcher intervention, but the dispatcher generally updates the incident with progress updates based on radio or text messaging traffic. With an AVL system the dispatcher can also visually see the progress of resource units toward the incident. For safety reasons, a series of timers and alarms are designed in a CAD system to remind the dispatcher to get an update from a resource unit if a status update has not been received within in a certain amount of time.

5.2. Fault Tolerance

The key to success in any critical response center is a high level of fault tolerance. Through a series of novel approaches, *NerveCenter* achieves an unparalleled level of robustness. Dispatchers have the ability to continue basic dispatch functionality even in the event of the central servers being down (DB Server of Application Server). In additional, dispatchers are still be able to coordinate, receive, and view updates from other dispatchers as long as there is any network connectivity between the dispatch stations.

5.3. Distributed Command Centers

Another key ability within the *NerveCenter* is the ability to set up remote command centers. *NerveCenter* allows dispatchers to fully operate on very low bandwidths, such that a mobile remote command center located in a trailer, could fully participate as a member of the dispatch / response team.

In additional to having remote command centers, *NerveCenter* allows multiple zones of responsibility to work together to solve problems. Many times an incident may cross multiple boundaries, requiring the coordinated response from varied and separate dispatch groups. *NerveCenter* allows for the sharing of incident data across multiple separate facilities using an encrypted data subscription mechanism. This allows each response agency to still own their data, but share parts of it in a secure method to other response agencies.

5.4. Message Bus

Each *NerveCenter* component subscribes to one or more message subscriptions on a high-availability message bus. The underlying architecture ensures that each message is delivered, and automatic re-routing ensures that if a central component is down, the messages are still delivered to other clients as if the central component sent the message. This allows centralized components to be down, while maintaining the ability to execute basic functions.

5.5. Persistence Manager and State Manager

The Persistence and State Manager services (PM and SM respectively) act like any other clients that subscribe to messages on the bus. The PM will take data changes and ensure that they are properly recorded on the underlying

persistence store (generally a RDBMS). The state manager gets the same messages and acts as a large cache. This is used across physical network links to coordinate and minimize traffic to the PM (and hence persistence storage access).

5.6. GEO-Coded Mapping

In order for proper incident and vehicle location to be displayed, and possibly used in resource allocation, an accurate GIS map of the incident area is needed. These maps can be either ESRI shape files or stored in a central SDE data store.

6. NOTIFICATION

6.1. Multi-Modal Alerts

Response personnel are notified with multi-modal alerts. The alert is sent to each user based on his/her preferred method of communication, including phone, SMS, e-mail, fax, etc.

Unlike simple SMS and blast dialing systems, the alerts are extremely robust to failures. If the first attempt to reach the responder fails, it retries on the secondary communication device. Each user specifies his/her prioritized list of communication devices and schedules. An alert to ten thousand first responders would be on a mix of devices and modes based on each user's configuration.

6.2. Text-to-Speech

The system uses multi-language text-to-speech to call a responder's mobile phone. The user simply presses 1 on the keypad to acknowledge the alert or 7 to record a voice response. The voice response is then available to all other participants in the incident from a situation center Web site that contains the up-to-date status from responders.

6.3. Acknowledgements

Acknowledgements are particularly useful because the dispatcher can choose to alert a second response team if the first fails to acknowledge within a certain time period. Escalation of alerts from one team to the next is automated with "hunt groups" that follow user-defined rules to send tiered alerts to escalation teams based on the acknowledgements (or lack of response) from the first team.

6.4. Extended Notification

The dispatcher can send mass alerts to a set of response teams that could include security, fire, emergency, police, and state and federal agencies. Network service providers and alert centers on multiple different continents are used to ensure failover and retry. Alerts can scale to ten thousand recipients.

The dispatcher can draw a polygon on a geo-coded map to select all household and business phone numbers within the specified area. Extended notification sends an alert to each number specified in the 911 registry.

7. RESPONSE

7.1. Team Collaboration

Response teams use a situation center Web site to share real-time status information. Tasks are assigned to members of the situation and tracked in the situation center. Team members post text and voice acknowledgements. Secure, real-time chat is used for communication online. New alerts can be sent to team members from the situation center as the incident evolves.

7.2. Live Video

Responders can view live cameras from the site of the incident within the situation center. Video is routed through the video proxy providing scalable streams to first responders outside the surveillance firewall (section I.E) enabling them to view the incident video (section I.F).

7.3. Instant Conferencing

When a group of responders receive an alert on their mobile phones, they simply press 9 on the keypad to initiate a dynamically created conference. The call is recorded as an audio stream accessible from the situation center.

8. RESOLUTION

8.1. Incident Closure

Any responder with access rights can close the incident. All team members can be notified. Rules can be set up to automatically close incidents with no activity after a set time period.

8.2. Incident Archive

A complete audit log of the situation center, dispatch, and notification is securely archived. The video incident archive is digitally signed and stored for long-term archival on a hierarchal storage management system.

9. REVIEW

9.1. Forensics and Reports

Forensic tools are used to query the computer aided dispatch database, situation center audit log, and video incident archive. Data mining tools are used to find threat patterns identified through queries for repeated incident types. All video content analytic events can be queried based on behavior type, object classification, severity, or date/time. The object track history and alarm key frame is available. XML templates are used to configure the report generator, which creates agency-specific reports for the county, state, or federal agency.

9.2. Process Improvement

Incidents are evaluated over time. The VCA false

positive/negative reports are used to tune analytic parameters: learning rate, zones, behaviors, severity, camera angle and position, and low-light settings. Alarm verification settings are adjusted to increase accuracy. The situation center audit log and alert response times are evaluated.

CONCLUSION

NerveCenter addresses each stage of the incident management life cycle. Surveillance provides broad camera and sensor coverage. Video content analytics finds suspicious behaviors, and remote video verification is an efficient way to escalate alarms to incidents. Computer aided dispatch coupled with geographic mapping and real-time unit locations are used to assign the nearest response units. Robust notification and team collaboration in a situation center are used to coordinate a resolution to the incident. *NerveCenter* provides exceptional situational awareness and a common operating picture to all parties throughout the incident life cycle.

ACKNOWLEDGEMENTS

We would like to thank Scott Anderson, Martha Chavez, Peter Dickson, Mark Fetherolf, Mark Koffskey, Ken Matson, Mike Mitchell, Peter Quintas, and Roger Yarrow for their contributions.

REFERENCES

[1] Fatih Porikli, Oncel Tuzel, Peter Meer, "Covariance Tracking using Model Update Based on Lie Algebra," *cvpr*, pp. 728-735, 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06), 2006.

[2] D. Abrams, B. Schulman, “Multi-core DSP Enables Advanced Target Discrimination and Tracking,” IEEE Conference on Technologies for Homeland Security 2006.

[3] Chris Stauffer, Eric Grimson, "Learning Patterns of Activity Using Real-Time Tracking," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 22(8):747-757, 2000.

[4] Ingemar J. Cox, Sunita L. Hingorani, "An Efficient Implementation of Reid's Multiple Hypothesis Tracking Algorithm and Its Evaluation for the Purpose of Visual Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 138-150, Feb., 1996.

[5] E. Rivlin, M. Rudzsky, R. Goldenberg, U. Bogomolov, S. Lepchev, "A Real-Time System for Classification of Moving Objects," *icpr*, p. 30688, 16th International Conference on Pattern Recognition (ICPR'02) - Volume 3, 2002.

[6] E. H. Adelson and J. R. Bergen, "Spatiotemporal energy models for the perception of motion," J. Opt. Soc. Am. A **2**, 284- (1985)

[7] Gabriel J. Brostow, Roberto Cipolla, "Unsupervised Bayesian Detection of Independent Motion in Crowds," *cvpr*, pp. 594-601, 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06), 2006.

[8] Chris Stauffer, Kinh Tieu, "Automated multi-camera planar tracking correspondence modeling," *cvpr*, p. 259, 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03) - Volume 1, 2003.

David Abrams is Chief Technology Officer of TrueSentry for InterAct Public Safety Systems and architect of the TrueSentry surveillance platform that employs analytics to intelligently identify threats from thousands of cameras and sensors to manage incidents with 911 dispatch and collaboration tools. As Director of Technology at divine, Inc. and Perceptual Robotics, he led development of TrueLook – a Web service with over 1.4B transactions to 60M users, deployed in major media events including the NBA Finals, Wimbledon, The Masters, the US Open, and the MLB World Series. He has patented video collaboration tools, and his team developed the first wireless pan/tilt/zoom network camera.

Steven McDowall is Senior Vice President of Product Engineering at InterAct Public Safety Systems. He implements software engineering techniques to maximize the usability and stability of InterAct products. He has managed the engineering efforts of InterAct’s crisis management and communication software, MissionMode Solutions. He was the chief operating officer of illumineX, inc., where he was responsible for the day-to-day management of the company in the areas of profit and loss, development, marketing initiatives, product direction, and corporate strategies.