Event-Driven Visual Sensor Networks: Issues in Reliability

Alexandra Czarlinska
Texas A& M University
College Station, Texas
czlinska@ece.tamu.edu

Deepa Kundur
Texas A& M University
College Station, Texas
deepa@ece.tamu.edu

Abstract

Event-driven visual sensor networks (VSNs) rely on a combination of camera nodes and scalar sensors to determine if a frame contains an event of interest that should be transmitted to the cluster head. The appeal of event-driven VSNs stems from the possibility of eliminating non-relevant frames at the source thus implicitly minimizing the amount of energy required for coding and transmission. The challenges of the event-driven paradigm result from the vulnerability of scalar sensors to attack or error and from the lightweight image processing available to the camera nodes due to resource constraints. In this work we focus on the reliability issues of VSNs in the case of global actuation attacks on the scalar sensors. We study the extent to which various utility functions enable an attacker to increase the average expected number of affected nodes with a relatively small penalty in the loss of stealth. We then discuss trade-offs between different attack detection strategies in terms of the cost of processing and the required information at the cluster head and nodes.

1. Introduction

Visual sensor networks comprised of networked camera nodes are envisioned for a variety of key applications such as distributed surveillance [2]. In the case of cameras dispersed throughout an environment, the acquired images must be transmitted wirelessly from the nodes to a cluster-head or central processing station. Given the energy and computational limitations of such nodes, energy-efficient image processing and transmission emerge as significant issues for visual sensor networks (VSNs) [3]. Indeed recent years have brought many interesting ideas from the growing community of researchers engaged in this area.

One promising approach to address the issue of efficient visual-data handling is based on the *event-driven* paradigm. In this setup, a camera node transmits its captured image frames *only if* an event of interest has been locally detected. The required event detection may be performed by processing the frames at the camera nodes directly [6]. However given the energy and the computational limits of the camera nodes, a lightweight processing algorithm is required and may not always result in a desirable trade-off between the probability of detection P_D and the probability of false alarm P_{FA}. The event-driven paradigm thus aims to improve the detection performance by exploiting other available information, such as scalar-data readings from co-deployed traditional (i.e. non-camera) nodes. Figure 1 depicts one possible scenario where each camera node receives decision support from a scalar node that detects changes in motion, temperature or other environmental readings (depending on the application). In comparison with other VSN techniques, the event-driven approach implicitly minimizes the use of energy and bandwidth via scalar-assisted local image selection. The appeal of the event-driven approach is that non-relevant frames are eliminated at the source instead of undergoing costly coding and transmission to the cluster-head. It should additionally be noted that although more powerful, the cluster-head is generally also resource-limited and benefits from receiving only the relevant frames. A significant challenge of the event-driven approach stems from the need to perform accurate event-detection. Specifically, in addition to the challenges posed by the use of lightweight image processing and vision technologies, it is necessary to consider

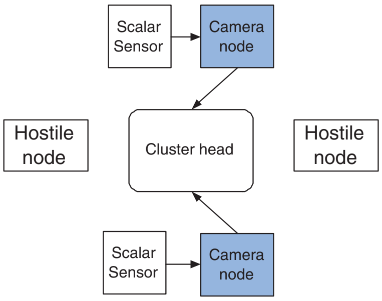

Figure 1. Event-driven VSN with scalar support for each camera node and the presence of a hostile network.

Authorized licensed use limited to: Texas A M University. Downloaded on December 23, 2008 at 02:08 from IEEE Xplore. Restrictions apply.

the reliability of the scalar sensors. The specific type of scalar sensor utilized depends heavily on the application. It is known however that scalar sensors are generally prone to a variety of sensor network attacks, some of which occur at the physical layer of sensing and can result in faulty environmental readings. One such type of attack is the actuation attack where nodes belonging to a hostile network utilize micro-actuators to alter environmental conditions in the physical vicinity of sensor nodes. This type of attack may not be addressed through cryptographic means alone due to its occurrence at the physical level [9], [1]. Furthermore, scalar sensors may suffer from temporary malfunctions due to harsh and varying environmental conditions [4].

Given the vulnerability of the scalar sensors to attack and error, it is important to evaluate the reliability of the event-driven approach. In particular, there is a critical need to detect an attack event which may occur globally throughout a cluster by exploiting the potential of node collaboration and data correlation. In this work we present the results of our study into the severity of a class of attacks known as actuation attacks. The intended focus and contribution are:

1. An investigation of the severity of actuation attacks when the attacker's stealth condition is relaxed.

2. An examination of methodologies for improving attack detection at the cluster-head and at the local nodes.

2. Background & Recent Advances

In the growing literature of energy-efficient VSNs we can identify several general approaches to the efficient handling of rich visual data. One general approach is to exploit node collaboration to distribute the signal and information processing over multiple nodes in the network. Such distributed and collaborative approaches are also well-suited to the inference of higher-level information from the nodes. In the context of reducing the energy required for image processing and transmission, the following methodologies can be identified within this class:

1. Signal processing and geometry-based exploitation of spatial and temporal overlap among cameras to reduce redundant data. This approach is particularly attractive in dense deployments and multi-hop environments where the field-of-view of several cameras is likely to overlap spatially and over-time [14], [12].

2. Information and coding theoretic exploitation of spatial data correlation utilized for distributed image compression, such as via Wyner-Ziv coding. Though fundamentally different in its formulation, this approach also attempts to eliminate data redundancy by allowing the nodes to separately encode the non-redundant data [11].

3. Event-driven approach based on collaboration between (lightweight) image processing nodes and scalar sensor readings. This approach is beneficial in cases where scalar sensors are available in addition to deployed cameras. This approach benefits from several areas such as applied game theory and lightweight image processing. [5].

Other noteworthy approaches include automatic computer vision techniques adapted to the low-energy and low-complexity regime of VSNs [13]. In this work we focus on the event-driven approach and examine its reliability under an actuation attack model.

2.1. Visual Nodes: Lightweight Event-Detection

To motivate the need for scalar-visual collaboration in the event-driven paradigm, we examine the event detection performance of a representative lightweight algorithm detailed in [8]. The algorithm was selected based on its simplicity and shared common features with other algorithms of the lightweight class designed to detect *general events* and is intended for illustration only. It is important to note that the definition of an "event" in event-driven VSNs is application dependent. However this definition is generally based on the detection of "significant" motion. This is in contrast with "pull-based" systems where the cluster-head issues an interest feature to the nodes and where the nodes search their stored frames to determine if a matching feature is discovered [10]. In the event-driven VSN, a node "pushes" a frame to the cluster-head if the algorithm determines that it is a frame of interest. Thus a common lightweight algorithm for the event-driven paradigm is based on the computation of 1- a difference frame, 2- a relevant statistic based on the difference frame and 3- comparison of the statistic to a threshold to determine if an event occurred [8]. Figures 2, 3, 4 and 5 show sample images extracted from four separate sequences acquired under different conditions. In Figure 2, the event of interest is the appearance and passing of an unidentified vehicle while in Figure 5, the event is the appearance of an individual. As can be seen from the figures, the probability of event detection P_D, the probability of false alarm P_{FA} and the resulting total probability of error P_T vary greatly from sequence to sequence. In the context of lightweight algorithms for the detection of general events, P_D is defined as the probability of registering significant motion given the presence of a moving object while P_{FA} is defined as the probability of registering significant motion when *no* moving object is present. The total probability of error $P_T = P_{FA} \cdot P(\mathcal{H}_0) + (1 - P_D) \cdot P(\mathcal{H}_1)$ where $P(\mathcal{H}_0)$ and $P(\mathcal{H}_1)$ are the a priori probabilities of the null hypotheses (no object) and alternative hypothesis (object present) respectively. We note that factors such as the motion of

Authorized licensed use limited to: Texas A M University. Downloaded on December 23, 2008 at 02:08 from IEEE Xplore. Restrictions apply.

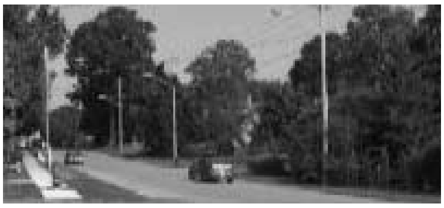

Figure 2. Sequence with vehicle moving away from camera. $P_D = 0.74$, $P_{FA} = 0.25$ and $P_T = 0.26$.

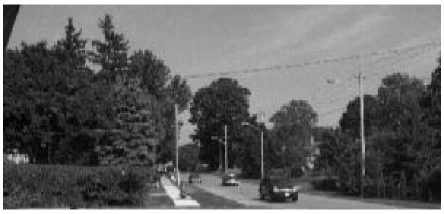

Figure 3. Sequence with two vehicles approaching the camera. $P_D = 0.11$, $P_{FA} = 0.63$ and $P_T = 0.3$.

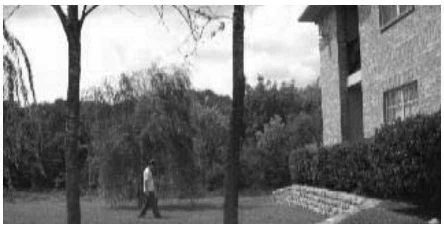

Figure 4. Sequence with individual outdoors. $P_D = 0.05$, $P_{FA} = 0.23$ and $P_T = 0.9$.

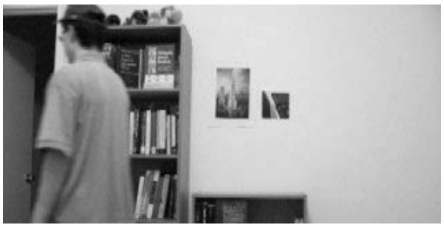

Figure 5. Sequence with individual indoors. $P_D = 1.0$, $P_{FA} = 0.13$ and $P_T = 0.03$.

nearby trees or the relative size of the moving object(s) in the frame affect the detection performance.

3. Scalar Reliability: the Stealth Condition

Scalar sensor nodes deployed in a hostile environment are susceptible to a variety of attacks on their data [4]. Many of the attacks can be averted through the use of cryptographic keys and key management systems. However a type of attack referred to as an actuation attack occurs at the physical sensing layer prior to the encryption process and effectively *perturbs* the collected readings away from their true values [9], [7]. As illustrated in Figure *1*, the attack is perpetrated by a *foreign hostile* network (i.e. a network not under the control of the legitimate scalar nodes). For the purpose of generality and tractability, the attack is modeled by considering its *effect* on the sensors' *decisions* about the presence or absence of an event. As in [9], we assume that under no attack, each sensor i makes a binary yes/no decision x_i where $x_i = 1$ denotes "event present" and has Bernoulli distribution $Bern(p)$. The *effect* of the attack is modeled as flipping a decision from *0* to *1* and vice versa with Bernoulli probability q. Specifically, $y_i = 1$ represents a hostile node i actuating and $y_i = 0$ represents no actuation. The overall effect is that a scalar node i makes decision z_i where $z_i = x_i \oplus y_i$.

Nash game theory analysis is employed to determine the optimal attack parameter q that the hostile nodes should select to remain *stealthy*, that is, undetected. The attack detection may be performed using different methodologies. In [9], the attack detection is performed by having each scalar node transmit its decision not only to its camera node, but also to the cluster-head. The cluster-head compares the received weight of the vector $w(\mathbf{x})$ with prior expectations and raises an alert when the difference exceeds a threshold. In terms of game theory, the scalar sensor network is labeled as player *1* and the hostile network is labeled as player *2*, each comprised of n nodes. The utility or payoff π_2 that player *2* is trying to maximize is shown in Eq. *1* along with the utility that player *1* is trying to maximize.

$$\begin{aligned}\pi_2(p,q) &= Pr\{w(\mathbf{x}) = w(\mathbf{z})\} \\ &= Pr\{w(\mathbf{x}) = w(\mathbf{x} \oplus \mathbf{y})\} \qquad (1) \\ \pi_1(p,q) &= -\pi_2(p,q)\end{aligned}$$

In [9] it is shown that the optimal attack parameter q is *small* in general and that it depends on the network size n and the probability of detecting an event (under no attack) p, i.e. $q(n,p)$. Though such choice of attack parameter avoids detection with large probability, it is not powerful from the hostile network's point of view. This stems from the fact that the average number of affected nodes a is given by $a = n \cdot q$.

In this work we wish to study alternative attacker utility functions $\tilde{\pi}_2$ which may yield a better tradeoff for the attacker between stealth and the number of affected nodes a. This is motivated by a wish to understand if a more severe attack is possible under the stealth condition with the given assumptions. Eqs. 2 to 6 show the various utilities that we wish to consider. The significance of the $n \cdot q$ term is that it represents the attacker's desire to balance stealth and the number of affected nodes. The $\sqrt{nq}$ term expresses a nonlinear preference for the number of affected nodes. In that case, the attacker prefers to affect as many nodes

Authorized licensed use limited to: Texas A M University. Downloaded on December 23, 2008 at 02:08 from IEEE Xplore. Restrictions apply.

as possible but the rate of satisfaction grows more slowly once many nodes have already been affected. The additive form of the utility functions in Eqs. 3, 4 and 6 represents satisfaction if *either* the stealth condition *or* the number of affected nodes is maximized. The multiplicative form in Eqs. 2 and 5 requires that both components be maximized in order for the utility to be maximized. In Eq. 6, the parameter $r \in [0, 1]$ models a preference weighting between the two factors. In each of these cases, player *1* is trying to maximize $-\tilde{\pi}_2(p, q)$.

$$\tilde{\pi}_2(p, q) = \pi_2(p, q) \cdot \sqrt{nq} \tag{2}$$

$$\tilde{\pi}_2(p, q) = \pi_2(p, q) + \sqrt{nq} \tag{3}$$

$$\tilde{\pi}_2(p, q) = \pi_2(p, q) + n \cdot q \tag{4}$$

$$\tilde{\pi}_2(p, q) = \pi_2(p, q) \cdot nq \tag{5}$$

$$\tilde{\pi}_2(p, q) = r \cdot \pi_2(p, q) + (1 - r) \cdot (n \cdot q) \tag{6}$$

4. Stealth Condition Results

Results for the optimal value of the attack parameter q were obtained via simulation of the various payoff curves. These results are shown in Tables *1* through *6* for various cluster sizes n and parameter p. We note that due to the symmetry of the utility function π_2 in parameter p, the results are the same for p and $1 - p$, e.g., for $p = 0.1$ (rare event) and $p = 0.9$ (common event). We also note that results pertaining to Eq. *3* have been omitted from the tables because they yield results resembling those of Eq. *4*.

Figure *6* offers a visual comparison of the effect of the adjusted utility given by Eq.*2* compared to the original utility given by Eq. *1* for the case where $n = 10$ and $p = 0.1$. As shown in the figure, the attacker is able to obtain a seemingly higher utility by migrating to a slightly higher value of q. We therefore conclude that it is indeed beneficial for the attacker to adopt the new payoff given by Eq. *2* and migrate to a higher q value. It is important to note however that while higher than the original utility, the new utility no longer measures the stealth condition alone. If we wish to understand how the higher q value impacts stealth, we need to recompute the resulting probability of attack evasion for the new set of q. This has been done in Tables *1* to *6* to offer a more direct look at the resulting stealth.

In Tables *1* through *6*, q_N refers to the optimal Nash q determined from the original utility given by Eq. *1* and $\tilde{q}_N$ refers to the new optimal Nash q computed for Eqs. *2* through *6*. The last column shows the difference between the achieved stealth S where S_o is the stealth obtained originally from Eq. 1 and S_n is the new stealth obtained using Eqs. *2* through *6*.

$$S = S_n - S_o \tag{5}$$

We observe that S is always negative or zero because by moving to a higher q value, the attacker is giving up some of the stealth in favor of more actuation. We notice however that in many instances the penalty in S is not significant compared to the gain achieved in higher q. Specifically, though the gain in q appears small, it results in a higher expected number of affected nodes for a relatively small loss in stealth. We make some further observations regarding the trends.

Table 1. Rare event ($p = 0.1$) or common event ($p = 0.9$) and small cluster size of $n = 10$ and $\tilde{q}_N = 0.205$.

	q_N	S
Eq. 2	0.255	-0.001
Eq. 4	1.0	-0.059
Eq. 5	0.305	-0.007
Eq. 6	0.255	-0.001

Table 2. Rare event or common event and medium cluster size $n = 35$ and $\tilde{q}_N = 0.063$.

	q_N	S
Eq. 2	0.08	-0.003
Eq. 4	1.0	-0.055
Eq. 5	0.098	-0.01
Eq. 6	0.065	0

Table 3. Rare event or common event and large cluster size $n = 100$ and $\tilde{q}_N = 0.022$.

	q_N	S
Eq. 2	0.028	-0.003
Eq. 4	1.0	-0.054
Eq. 5	0.035	-0.011
Eq. 6	1	-0.054

Table 4. Typical event ($p = 0.5$) and small cluster size $n = 10$ and $\tilde{q}_N = 0.999$.

	q_N	S
Eq. 2	0.999	0.0
Eq. 4	0.999	0.0
Eq. 5	0.999	0.0
Eq. 6	0.999	0.0

Table 5. Typical event and medium cluster size $n = 35$ and $\tilde{q}_N = 0.089$.

	q_N	S
Eq. 2	0.205	-0.047
Eq. 4	1.0	-0.196
Eq. 5	0.934	-0.128
Eq. 6	0.09	0

- For the typical event $p = 0.5$, it is generally harder to achieve an increase in q without a proportional decrease in stealth. We can relate this observation to the findings from [9] where the attacker's best case scenario occurred for $p = 0.5$. The new findings thus signify that the case where $p = 0.5$ is already *efficient*

Authorized licensed use limited to: Texas A M University. Downloaded on December 23, 2008 at 02:08 from IEEE Xplore. Restrictions apply.

Table 6. Typical event and large cluster size $n = 100$

	q_N	$\tilde{q}_N$	S
Eq. 2	0.031	0.999	-0.122
Eq. 4	0.031	1.0	-0.195
Eq. 5	0.031	0.999	-0.122
Eq. 6	0.031	0.032	0

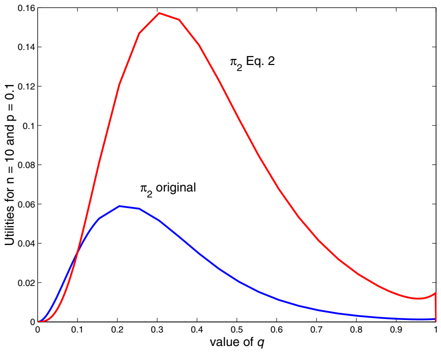

Figure 6. Utility and q comparison between original π_2 and $\tilde{\pi}_2$ from Eq. *2*

for the attacker. Thus no further alteration of the utility is beneficial.

- For the rare event ($p = 0.1$) or the common event ($p = 0.9$), certain modified utility functions do yield an improvement in q at a low penalty in the loss of stealth. This is true for example using Eq. *2* or Eq. *4*.
- For the rare or common event however, the benefits of the modified payoff function start to decrease for increasing cluster size n in terms of the attack becoming more detectable. This is significant since if q were large for large clusters n, the expected number of affected nodes would be large.
- Among the various modified utility functions, Eqs. *2* and *5* appear to offer a somewhat better advantage over the other utilities. This may suggest that a multiplicative form for the utility function is more efficient in moving along the q-π_2 curve. For instance, if we examine the original utility function π_2 shown in Figure 5, we notice that moving to a higher value of q should be possible without decreasing the stealth significantly. Utilizing the modified utility in Eq. *2*, the attacker is able to capture this gain.

5. Examining Attack Detection

Attack detection in the case of an actuation attack is challenging given the globally distributed form of the attack. Specifically, given the deployment of the hostile network throughout the environment, every scalar node inherits the probability q of being affected and thus of giving unreliable readings. In this challenging scenario, the type of attack detection that is performed depends heavily on the type of information available to the cluster head and the nodes. In [9], attack detection was performed by reporting all the scalar sensor decisions not only to the camera nodes, but also to the cluster head. The cluster head in turn compared the received data vector to expected averages for the number of nodes that should report an event for a given physical distribution. Importantly, this solution does not guarantee that the hostile actuation will be detected with certainty and depends on various parameters such as n and p. Secondly, when an attack *is* detected, the cluster head is only able to conclude that an attack occurred and not *which* specific scalar nodes were affected. Furthermore, this solution is somewhat centralized in that the detection is not performed at each local node but rather at the cluster head.

In [8], global attack detection is still performed at the cluster head but the camera nodes also make local decisions regarding the reliability of the scalar sensors. This is achieved by processing the frames at the camera nodes using lightweight event detection as described in Section 2.1. However event detection utilizing the lightweight approach incurs a probability of error q' and under certain environmental and lighting conditions, this error may be larger than the error q due to attack. We wish to briefly examine alternative solutions to this problem, specifying trade-offs between local processing vs. global processing and the information required in each scenario.

Cluster head image processing. In this approach, the cluster head determines not only if an attack occurred but which scalar nodes have been affected. This level of detail comes at the cost of performing higher-quality event detection upon the image frames received from the camera nodes and comparing the results with the scalar sensor decisions. This solution is likely too expensive for energy-limited VSNs in most cases.

Selective Cluster head image processing. In this approach the cluster head performs higher-quality image processing on a chosen number k of the received image frames where $k < n$. In order to minimize energy use and delay (in processing the frames), it is desirable to select an optimal k which achieves a target probability of attack detection. In the case where the probability of natural scalar sensor error (i.e. not due to attack) is known, the minimal k can be determined using the equation $P_D \leq \frac{\binom{nq}{nq'+1}\cdot\binom{n-nq}{k-nq'-1}}{\binom{n}{k}}$ where nq is the expected number of affected nodes (due to attack). Given that the number of nodes that may be reporting unreliable readings due to natural error is nq', in order to conclude that an attack is occurring, we need to find *at least* one more unreliable node than nq'. Figures *7* and *8* illustrate the optimal value of k such that precisely $nq' + 1$ nodes are found and such that *at least* $nq' + 1$ nodes are found. Fig-

Authorized licensed use limited to: Texas A M University. Downloaded on December 23, 2008 at 02:08 from IEEE Xplore. Restrictions apply.

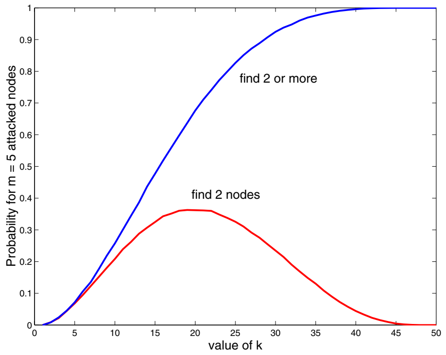

Figure 7. Optimal value of k for finding 2 or more of the 5 affected nodes.

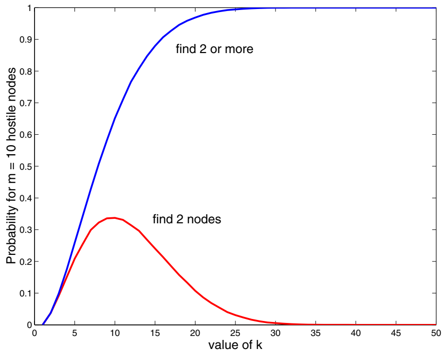

Figure 8. Optimal value of k for finding 2 or more of the 10 affected nodes.

ure *7* shows the results for the case where $n = 50$ nodes, $q' = 0.02$ and $q = 0.1$. Figure *8* shows the results for the case where $n = 50$ nodes, $q' = 0.02$ and $q = 0.2$.

6. Conclusions

In this work we examine issues associated with the reliability of event-driven Visual Sensor Networks (VSNs) that rely on scalar sensors to identify visual events of interest. We focus our study on scalar sensors in the presence of a global actuation attack and examine if the attacking network is able to increase the number of affected nodes while remaining stealthy (undetected by the cluster head). We determine that a number of utility functions exist which enable an attacker to increase the average expected number of affected nodes while incurring a relatively small penalty in the probability of loosing stealth. However the gain of the attacker is still limited and decreases with an increasing cluster size. Finally we examine some alternative strategies for detecting the attack in terms of the trade-offs between required image processing and information known to the cluster head or nodes.

References

[1] I. Akyildiz and I. Kasimoglu. Wireless sensor and actor networks: Research challenges. *Ad Hoc Networks Journal*, 2(4):3351–3677, 2004. 2

[2] I. Akyildiz, T. Melodia, and K. Chowdhury. A survey on wireless multimedia sensor networks. *Computer Networks*, 51(4):921–60, March 2007. 1

[3] A. Basharat, N. Catbas, and M. Shah. A framework for intelligent sensor network with video camera for structural health monitoring of bridges. In *Proceedings. Third IEEE International Conference on Pervasive Computing And Communications Workshops, PerCom 2005 Workshops*, pages 385–9, March 2005. 1

[4] L. Buttyan and J.-P. Hubaux. Report on a working session on security in wireless ad hoc networks. *Mobile Computing and Communications Review*, 6(4):1–17, 2002. 2, 3

[5] V. Cevher, A. C. Sankaranarayanan, J. H. McClellan, and R. Chellappa. Target tracking using a joint acoustic video system. *IEEE Transactions on Multimedia*, 9(4):715–727, 2007. 2

[6] R. Cucchiara. Multimedia surveillance systems. *Proceedings of the third ACM international workshop on Video Surveillance & Sensor Networks (VSSN)*, pages 3–10, November 2005. 1

[7] A. Czarlinska and D. Kundur. Distributed actuation attacks in wireless sensor networks: Implications and countermeasures. *Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems*, pages 3–12, April 2006. 3

[8] A. Czarlinska and D. Kundur. Reliable scalar-visual event-detection in wireless visual sensor networks. In *IEEE CCNC Special Session on Image/Video Processing & Wireless Sensor Networks*, Las Vegas, 10-12 January 2008. Available: www.ece.tamu.edu/˜czlinska/CCNC08/wvsn.pdf. 2, 5

[9] A. Czarlinska, W. Luh, and D. Kundur. Attacks on sensing in hostile wireless sensor-actuator environments. In *IEEE Globecom*, Washington, DC, 26-30 November 2007. Available: www.ece.tamu.edu/˜czlinska/Glob07/AttacksSenAct.pdf. 2, 3, 4, 5

[10] A. Grigorova, F. D. Natale, C. Dagli, and T. S. Huang. Content-based image retrieval by feature adaptation and relevance feedback. *IEEE Transactions on Multimedia*. 2

[11] A. Liveris, Z. Xiong, and C. Georghiades. A distributed source coding technique for correlated images using turbo-codes. *Computer Networks*, 6(9):379–381, 2002. 2

[12] H. Ma and Y. Liu. Correlation based video processing in video sensor networks. In *IEEE International Conference on Wireless Networks, Communications and Mobile Computing*, page 987, Maui, Hawaii, June 2005. 2

[13] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering objects and their location in images. In *IEEE ICCV'05*, pages 370–7, 2005. 2

[14] M. Wu and C. W. Chen. Collaborative image coding and transmission over wireless sensor networks. *EURASIP Journal on Advances in Signal Processing*, 2007(1):9, 2007. 2

Authorized licensed use limited to: Texas A M University. Downloaded on December 23, 2008 at 02:08 from IEEE Xplore. Restrictions apply.