Distributed Video Data Fusion and Mining

Edward Y. Chang[a], Yuan-Fang Wang[b], and Volkan Rodoplu[a]

[a]Department of Electrical & Computer Engineering, University of California, Santa Barbara
[b]Department of Computer Science, University of California, Santa Barbara

ABSTRACT

This paper presents an *event sensing* paradigm for intelligent event-analysis in a wireless, ad hoc, multi-camera, video surveillance system. In particilar, we present statistical methods that we have developed to support three aspects of event sensing: 1) energy-efficient, resource-conserving, and robust sensor data fusion and analysis, 2) intelligent event modeling and recognition, and 3) rapid deployment, dynamic configuration, and continuous operation of the camera networks. We outline our preliminary results, and discuss future directions that research might take.

Keywords: data fusion, event recognition, imbalanced data

1. INTRODUCTION

Video cameras and wireless networks are becoming ubiquitous features of modern life. The confluence of these two technologies now makes it possible to construct wireless ad hoc networks of multiple video cameras that can be rapidly deployed, dynamically configured, and continuously operated to provide highly-available coverage for environment monitoring and security surveillance.

While many extended "eyes" are being installed at an unprecedented pace, the intelligence needed for interpreting video-surveillance events by computers is still rather unsophisticated. In a recent ACM video-surveillance workshop co-chaired by the authors,[1] participating developers and practitioners emphasized the urgent need for an enhanced "brain" to match up with these multiple camera views for video analysis and query answering. Indeed, we need (semi-) automated video-analysis and event-recognition systems that can gather intelligence and provide timely warnings to alert security personnel.

To support (semi-) automatic *event sensing*, we have develop statistical methods to improve the two major phases of a distributed, mobile surveillance system: *data fusion* and *event analysis*.[2,3] The *data-fusion* phase integrates mobile, multi-source data to detect and extract motion trajectories from video sources. The *event-analysis* phase deals with classifying the events as to relevance for the query. The research challenges of the two phases are summarized as follows:

- *Data fusion from mobile cameras.* Data fusion deals with collecting and analyzing data at the cameras, transmitting the data (could be noisy or partial) to the server, and fusing them to extract motion trajectories. Data fusion comes up against three major research challenges: *sensor-network configuration*, *spatio-temporal data fusion*, and *resource management*. Given a query and its precision requirement, the sensor-network may need to move some cameras and reconfigure itself to "see" the event of interest. This reconfiguration must be performed in such a way that useful data can be collected with minimal consumption of power, network bandwidth, and computer resources. Once the network has been reconfigured, observations from multiple cameras should be integrated to build spatio-temporal patterns that can best describe events in the environment. Such integration is necessary to improve surveillance coverage and to deal with transient object-tracking obstacles such as spatial occlusion and scene clutter. In addition, streaming data to the server must observe resource constraints such as network bandwidth or the server's computation and memory capacity.

Further author information: Edward Chang: echang@ece.ucsb.edu; Yuan-Fang Wang: yfwang@cs.ucsb.edu; Volkan Rodoplu: vrodoplu@ece.ucsb.edu

• *Event analysis.* Event analysis deals with mapping motion trajectories (sequence data) to semantics (e.g., benign and suspicious events). Most traditional statistical learning algorithms cannot be directly applied to variable-length sequence data, which may also exhibit temporal ordering. In addition, positive events (i.e., the sought-for hazardous events) are always significantly outnumbered by negative events in the training data. In such an imbalanced set of training data, the class boundary tends to skew toward the minority class and becomes very sensitive to noise.

To answer the above challenges, we have been working on five research tasks to advance fundamental theories and develop statistical methods that can significantly improve the operation of wireless ad hoc camera networks, quality of data fusion, and accuracy of event analysis. The five research tasks are summarized as follows:

1. *Sensor-network resource management* (Section 2). We have developed statistical methods to manage networks for conserving resources, including power at the sensor nodes, as well as network bandwidth and other system resources at the server.

2. *Statistical mobile-sensor data fusion* (Section 3). We have devised algorithms to fuse spatially and temporally overlapped data for reliable event detection. Our research focus is on enhancing the reliability of existing object-tracking algorithms by performing both sensor-to-server data fusion and server-to-sensor information dissemination.

3. *Sequence-data to event mapping* (Section 4). We have developed statistical learning algorithms that consider both the primary and secondary structures of motion patterns in mapping sequence-data to events. In contrast to the widely-used Hidden Markov Models (HMMs), our learning algorithms require a much smaller amount of training data. .

4. *Imbalanced training-data statistical learning* (Section 5). We have designed both algorithmic and data-processing approaches to modify class boundaries for improving event-recognition accuracy when positive training instances are difficult to collect.

5. *Synergistic integration—energy-efficient, distributed topology control* (Section 6). We have developed scalable ad hoc networking protocols that are suitable for video surveillance networks comprising hundreds of video cameras.

2. SENSOR NETWORK RESOURCE MANAGEMENT

In a distributed sensor network, cameras record continuous high-volume video streams. Because of the high data volume and rapid rate, it is infeasible for an untethered, battery-powered sensor node to transmit a large quantity of raw data to a server for processing.[4–6] To conserve resources—network bandwidth, storage, and CPU—many recent papers[7–11] propose methods to reduce the amount of data delivered to the server. In these schemes, provided that the server can answer queries within specified precision constraints, data communication is not enacted.

A major shortcoming of the existing solutions is that they are often ad hoc, as explained in[8] by Widom and Motwani, and are highly application-dependent. No unified solution exists for managing distributed streams. In this task, we treat resource management in a sensor network as fundamentally a filtering problem: an effective stream-filtering algorithm should filter out the maximum amount of data as long as the query-precision constraints are met at the server. We introduce our *Dual Kalman Filter* (DKF) architecture[12] as a general and adaptive solution to the stream-resource-management problem.

Figure 1 depicts the role of our proposed DKF (Dual Kalman Filter) model in a typical sensor-network architecture. A user (on the left-hand side of the figure) issues to the server an event query with certain precision constraints. The server activates a KF, denoted as KF_s, and at the same time, the target sensor activates a mirror KF with the same parameters, denoted as KF_m. The dual filters KF_s and KF_m predict future data values. Only when the filter at sensor KF_m fails to predict future data within the precision constraint (thus preventing KF_s from making an accurate prediction at the server) does the sensor send updates to KF_s. For instance, if no interesting event is taking place at a sensor, no data transmission is made to the server. When multiple events are taking place at a sensor, multiple pairs of KF_s and KF_m will be invoked to track the events. Significant

bandwidth conservation can be achieved if a reliable and accurate data prediction mechanism is employed, and the server resources can be allocated to the sensors where actions are taking place. We plan to use the Kalman Filter as such a mechanism for its simplicity, efficiency, and provable optimality under fairly general conditions. Our preliminary results indicate that DKF shows promise in several scenarios with which we experimented.[12]

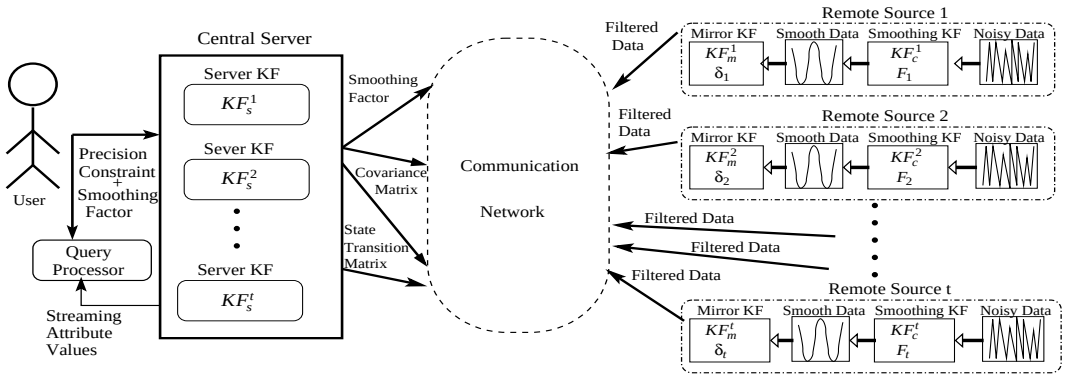


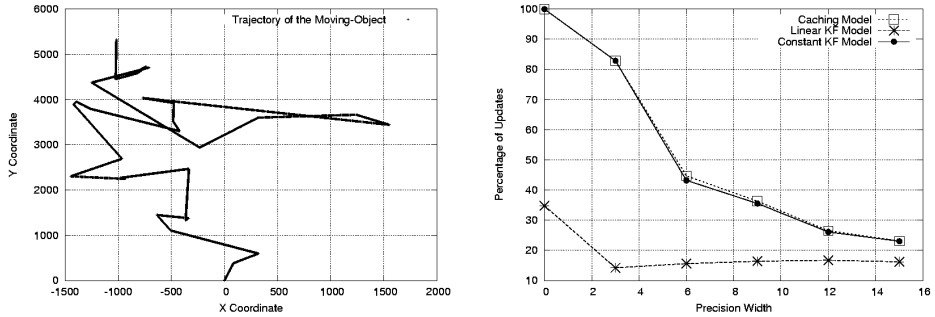
Figure 1. The DKF model

Preliminary Results

We have employed DKF in our multi-camera surveillance prototype for analyzing vehicle/human behavior in a parking lot. One experiment that we have conducted is on moving-vehicle tracking. In our experiment, each moving object had two attributes: namely, *location* (in terms of X and Y coordinates) and *velocity* (in terms *speed* and *angle* of direction). We used a uniform random-number generator to generate different slopes of the velocity vector at random intervals of time. We generated different speeds of the object at random time intervals in a similar manner. Thus the object could randomly change its speed and heading, and then continue on that linear path for a randomly generated length of time. The maximum speed of the object was limited to 500 units, whereas the slope could arbitrarily change by any amount. We constructed a data set shown in Figure 2(a), using the above model containing 4000 data points at a sampling rate of 100 ms.

We tested the performance of the Kalman Filter approach on two different state models:

- *Constant* KF *model:* The system is modeled such that the latest updated value is the best prediction for the future. This model is conceptually similar to the standard cached approximation model. The measurement consists of just the position of the object in the two-dimensional space, i.e., X coordinate and Y coordinate.
- *Linear* KF *model:* Here we take the rate of change of the position into consideration when predicting future values.



(a) Moving-object data set (b) Number of updates

Figure 2. A Resource Conservation Example Using Kalman Filter.

Figure 2(b) shows comparative results of the two Kalman Filter models with the cached approximation scheme. Measurements are taken in the form of position $P(x, y)$. Given a precision constraint δ, point $P(x, y)$ is

updated to the server if an error in either X or Y value is greater than δ. In both KF models, only the position is recorded, not a measurement of the rate of change of coordinate values. As evident from Figure 2(b), the percentage of updates is the same, whether using caching or the constant KF model. This is because the constant model is similar to the caching scheme when the rate of change of values is not considered. However, if we use the linear KF model, we see that utilization of the communication resource is cut down by approximately 75% at a moderate precision width of 3 units. As the precision width increases, the communication resource utilization drops, and all three models show comparable performances. We also observe that the DKF performs at least as well as the caching scheme, even in a worst-case scenario.

3. STATISTICAL MOBILE-SENSOR DATA FUSION

The server receives video streams from distributed cameras, each of which has limited spatial and temporal coverage, is potentially noisy, and is susceptible to occlusion and scene clutter. To achieve wide-area coverage, data from cameras must be fused. Fusing spatially and temporally overlapped data is a challenging task, since cameras may have different sampling rates and resolutions, and some cameras may be mobile.

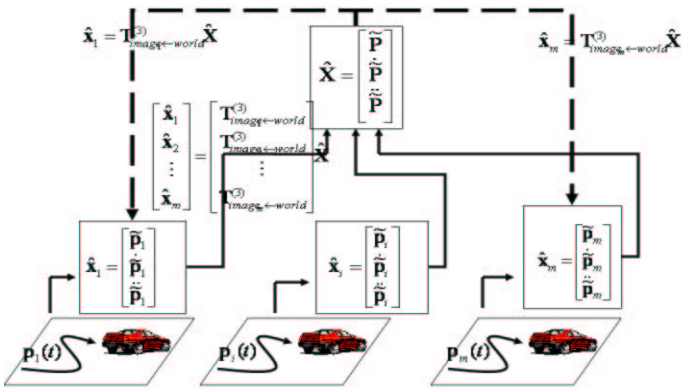


Figure 3. Two-level hierarchical Kalman Filter configuration

We propose here a hierarchical master-slave fusion scheme. Referring to Fig. 3, at the bottom level, each slave station tracks the movements of scene objects semi-independently. The local trajectories are then relayed to a master station for fusing into a consistent, global representation. This represents a "bottom-up" analysis paradigm. Furthermore, as each individual camera has a limited field of view, and occlusion occurs due to scene clutter, we also employ a "top-down" analysis module that disseminates fused information from the master station to slave stations. This top-down information dissemination process assists in tracking, cross validation, and error recovery—if the camera should lose track of an object.

Preliminary Results

We used the Kalman Filter[13, 14] as the tool for fusing information spatially and temporally from multiple cameras to detect motion events. Suppose that a vehicle (or a person) is moving in the parking lot. Its trajectory is described in the global reference system by $\mathbf{P}(t) = [X(t), Y(t), Z(t)]^T$. The trajectory may be observed in camera i, as $\mathbf{p}_i(t) = [x_i(t), y_i(t)]^T$, where $i = 1, \cdots, m$ (m is the number of cameras used). The goal is then to optimally track, correlate, and fuse individual camera trajectories into a consistent, global description.*

We formulate the solution as a two-level hierarchy of the Kalman Filters. Referring to Fig. 3, at the bottom level of the hierarchy, we employ for each camera a Kalman Filter to estimate, independently, the position $\mathbf{p}_i(t)$, velocity $\dot{\mathbf{p}}_i(t)$, and acceleration $\ddot{\mathbf{p}}_i(t)$ of the vehicle, based on the tracked image trajectory of the vehicle in *the*

*There are two issues that need to be addressed here: *registration* and *correspondence*. First, to fuse measurements from multiple sensors into one global estimate, two registration processes are needed: *spatial* registration to establish the transformation among different camera coordinate frames, and *temporal* registration to synchronize multiple local clocks. These techniques are well established in the literature, and we have developed algorithms to accomplish both spatial and temporal registration.[15] Second, it may be difficult to synchronize the activities observed in multiple cameras. The question is how to disambiguate the correspondence of multiple trajectories. Spatial and temporal trajectory correspondence can be established through the camera registration and stereopsis correspondence processes which are well established techniques in photogrammetry and computer vision. For our discussion, we will assume that these problems can be solved and we can achieve spatial and temporal registration of trajectories and disambiguate among multiple trajectories. (Interested readers are referred to our recent paper[15] for more details.

local camera reference frame. Or in the Kalman Filter jargon, the position, velocity, and acceleration vectors establish the "state" of the system while the image trajectory serves as the "observation" of the system state. At the top level of the hierarchy, we use a single Kalman Filter to estimate the vehicle's position $\mathbf{P}(t)$, velocity $\dot{\mathbf{P}}(t)$, and acceleration $\ddot{\mathbf{P}}(t)$ in *the global world reference frame*—this time, using the estimated positions, velocities, and accelerations from multiple cameras ($\mathbf{p}_i(t)$, $\dot{\mathbf{p}}_i(t)$, $\ddot{\mathbf{p}}_i(t)$) as observations (the solid feed-upward lines in Fig. 3). This is possible because camera calibration and registration[16–20] are used for deriving the transform matrices ($\mathbf{T}_{image \leftarrow world}$ and $\mathbf{T}_{world \leftarrow image}$ in Fig. 3). These matrices allow $\mathbf{p}_i$, measured in the reference frame of an individual camera, to be related to $\mathbf{P}$ in the global world system.

An interesting scenario occurs when one (or more) cameras in the sensor network loses track of an object. This can happen because of scene clutter, self- and mutual-occlusion, or the tracked objects exiting the field-of-view of a camera, among many other possibilities. The camera could switch from a "track" mode into a "re-acquire" mode by searching the whole image for telltale signs of the object; however, doing so inevitably slows down event-processing and introduces a high degree of uncertainty in the resulting event description. Instead, we allow the dissemination of fused information to individual cameras (the dashed feed-downward lines in Fig. 3) to help guide the reacquisition process. The Kalman Filter, being a flexible information-fusion algorithm, can readily use the fused information (instead of sensor data) for maintaining and updating state vectors. This hierarchical feed-upward (for sensor data fusion) and feed-downward (for information dissemination) filter structure thus provides a powerful and flexible mechanism for joining sensor data spatially.

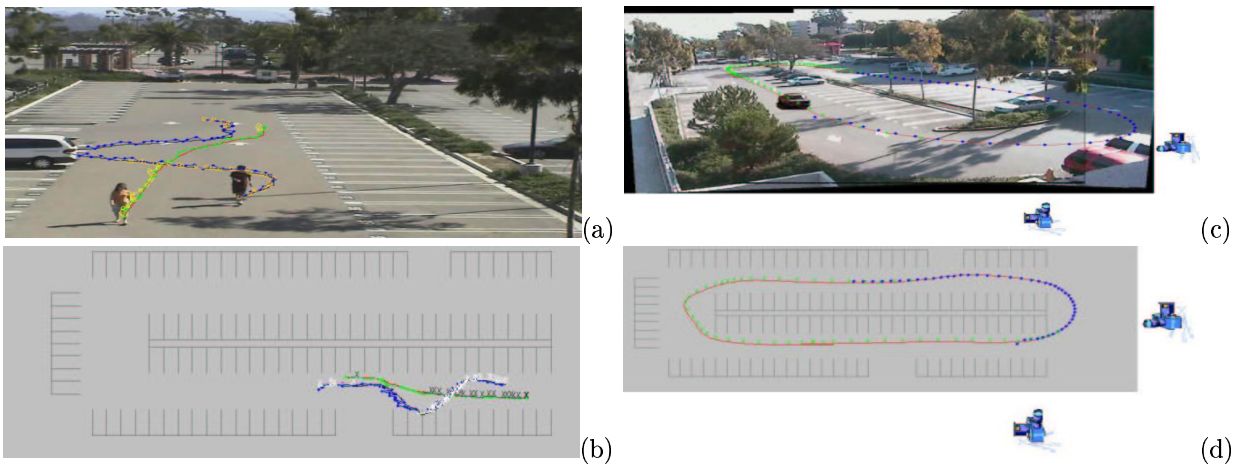


Figure 4. (a) A simulated stalking behavior in a parking lot and (b) trajectories of the sample stalking behavior. (c) and (d): similar data fusion results for vehicular motion. In these figures, the "-" is the fused trajectory; "." is the tracked trajectory from camera 1; "x" is the tracked trajectory from camera 2; and "o" is the tracked trajectory from camera 3.

We have collected hours of video using multiple video cameras in a parking lot. The video frames depicted both human and vehicular motion. The motion patterns for vehicles included entering, exiting, turning, backing up, circling, zig-zag driving, and many more. For human motion, we recorded actions involving both individuals and groups, with patterns such as following, following-and-gaining, stalking, congregating, splitting, and loitering, among many others. Some of these patterns (like zig-zag driving and stalking) were acted out by our group members, while others represented behaviors commonly observed in the parking lot. Due to space limitations, we show only two sample results here. Sample results for tracking the movements of people in a parking lot are shown in Fig. 4(a) and (b). Of the three cameras we used, the views of two were partially occluded by parked cars[†]. The individual camera trajectories could therefore be broken. However, by using our two-level filter structure, we were able to fill in the gap, smooth out sensor noise, and fuse individual trajectories into a complete, global description. Fig. 4(c) and (d) show the analysis of a vehicle's driving pattern when two cameras were used. Note that even with a very small overlap in the fields-of-view of the two cameras and a circling motion covering a large spatial area (hence, each camera observed only a part of the motion trajectory), we were able to fuse the individual camera trajectories to arrive at a complete description.

[†]The camera positions in these figures indicate only the general directions of camera placement. The actual cameras were placed much farther away from the scene and always pointed to the parking lot.

4. SEQUENCE-DATA TO EVENT MAPPING

A sequence data s is defined as an ordered set of items: $a_1 \ldots a_n$. These items are logically contiguous and each item a_i denotes a set of attributes varying according to different applications. Given a set of sequences S, that can be partitioned into a labeled subset L and an unlabeled subset U, the task of sequence-data learning is to learn a discriminative function f from set L using algorithm Φ. Then, using $\hat{y}_i = f(u_i \in U)$, we can predict the label for unlabeled sequence $u_i \in U$.

To conduct supervised learning with a small number of training instances, the discriminant approach has been shown to be much more effective[21] than the generative approach (as in HMMs). In particular, SVMs require only those boundary instances (support vectors) to participate in a class prediction, and hence require a much smaller amount of training data than the other methods. Unfortunately, traditional kernel functions (such as polynomial and RBF functions) that have been employed with SVMs assume a feature space of fixed dimensions. They cannot be applied to sequence data, which are variable-length in nature. We thus design kernel functions that can effectively handle variable-length sequence data.

To conduct supervised learning, we need first to extract useful information (features) from sequence data to form representations.[22] Although many representations have been proposed in the past (see Section C.1 for detailed discussion), we believe that the best representation should be event-dependent. Therefore, our approach first extracts multi-resolution descriptors from sequence-data, and then relies on the algorithms that we subsequently develop to learn the best descriptor-combination for a target semantic. For instance, a motion pattern can be depicted as a sequence of symbolic strings at the coarse level, yet detailed information such as velocity and acceleration is recorded at the refined levels. If an event concerns only the turning pattern of a vehicle, then the coarse-level symbolic representation may be adequate; otherwise, proper secondary structures should be used. To support multi-resolution learning, we have designed kernel functions to characterize *similarity* at individual resolution-levels, and researched kernel-fusion mechanisms to integrate kernels at multiple levels. For both individual kernel design and kernel fusion, we have proven the kernels to be mathematically valid and verify them to be effective.

Preliminary Results

The kernel design task is to find a valid and meaningful kernel for sequence data in two steps. The first step is to design a kernel for each sequence descriptor, and the second step to fuse multiple kernels in an optimal way.

Individual Kernel Design

In this thread, we design new kernels for sequence-data learning. SVMs are the most popular kernel-based methods, but SVMs can be applied only to training data that reside in a vector space. The basic form of an SVM kernel function which classifies an input data $\mathbf{x}$ is expressed as

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}) + b = \sum_{i=1}^{N} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b \tag{1}$$

where ϕ is a mapping function which maps input vectors into the feature space; operator '$\cdot$' denotes the inner product operator; $\mathbf{x}_i$ is the i^{th} training sample; y_i is its class label; and α_i is its Lagrange multiplier. A kernel function is represented by k, and the bias by b.

For sequence data, in particular variable-length sequences, we lack the basis function ϕ for mapping sequences with various lengths to spaces of different dimensions. Fortunately, the embedding of a finite set of points is entirely specified by writing a finite-dimensional *kernel matrix*. Put another way, as long as we have a positive definite kernel matrix K, which characterizes the sequence-data similarity, we can use kernel methods.[23] Hence, the design task is reduced to formulating a kernel matrix satisfying two requirements: the *semantic requirement* and the *mathematical requirement*. Regarding the *semantic requirement*, Kernel matrix K must capture the local and global structure similarity between the sequence data. As to the *mathematical requirement*, a valid kernel matrix K must be symmetric and positive semi-definite[24] to ensure that the projected feature space does exist.

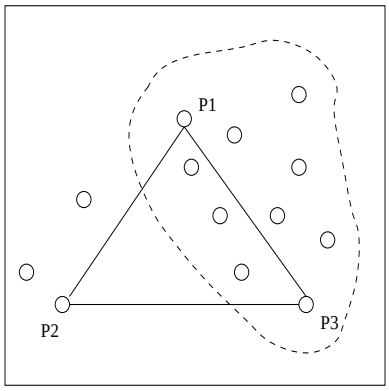


Figure 5. Example of Transitive Similarity

A natural way to define the similarity between two sequences is by using pair-wise string alignment scores.[25] Two sequences with variable lengths can be aligned by matching symbols at corresponding positions and inserting "-" at the unaligned positions. An alignment is a mutual arrangement of two sequences, showing where the two sequences are similar, and where they differ. The more aligned two sequences are, the more similar they are. By performing alignment, given N sequences, we can build a matrix $H_{N\times N}$, in which $H(i,j)$ is the pairwise similarity between sequences $\mathbf{s}_i$ and $\mathbf{s}_j$. However, there is one potential problem with matrix H: though H is symmetric, it might not be positive semi-definite.

To remedy the problem, we initially propose to consider transitive similarity when measuring pair-wise similarity. To motivate our approach, Figure 5 provides an example of considering transitive similarity between data, with each node denoting one data instance in the 2-D space. Assume p_1, p_2 and p_3 form an equilateral triangle, which means that the distances between them are the same. However, if we take data distribution into consideration, we notice that more data are located between p_1 and p_3 than between p_1 and p_2, or p_2 and p_3. More likely, p_1 and p_3 belong to the same class, and p_2 is an outlier. Therefore, we need to define a kernel matrix which can consider both pair-wise similarity and transitive similarity. Intuitively, a transitive relationship is helpful to characterize the similarity between data more accurately by considering data distribution. Furthermore, we have proved the following two important propositions, which show that when a given similarity is symmetric, taking transitive relationship into consideration will result in a legal kernel.

PROPOSITION 4.1. *Denote $w(i,j)$ as the similarity between sequence $\mathbf{s}_i$ and $\mathbf{s}_j$ by using pair-wise string alignment scores. If a matrix H is defined as:*

$$H(i,j) = \begin{cases} w(i,j) & for\ \mathbf{s}_i \neq \mathbf{s}_j \\ 0 & for\ \mathbf{s}_i = \mathbf{s}_j \end{cases} \tag{2}$$

then $K_s = exp(H)$ is a semantically valid kernel, reflecting the similarity relationship between sequences, including transitive similarity.

PROPOSITION 4.2. *$K_s = exp(H)$ is a mathematically valid kernel, which is symmetric and semi-positive definite.*

Kernel Fusion

After formulating individual kernels, the next step is fusing individual kernels. Each individual kernel extracts a specific type of information from given data, thereby providing a partial view of the data. Kernel fusion forms a complete picture of the relationship between different components of the original sequence-data. Assume R is a relation between instances x and y and their parts, i.e., R^{-1} decomposes an instance into a set of D-tuples. Kernel $K_d(x_d, y_d)$ is the similarity between parts x_d and y_d. For different contexts, not all levels' descriptors should be considered as having the same importance. We propose kernel fusion to provide the flexibility to learn which level should be more important according to the target learning task. Possible fusion rules are *weighted sum* and *tensor product*, since kernels have proven to be closed under sum and product. The weighted sum is formulated as

$$K_{fuse} = \mathbf{F}(K'_1, \cdots, K'_D) = \sum_{d=1}^{D} w_d K'_d. \tag{3}$$

Tensor product formulation is defined as

$$K_{fuse} = \mathbf{F}(K'_1, \cdots, K'_D) = K'_1 \otimes \cdots \otimes K'_D. \tag{4}$$

5. IMBALANCED TRAINING-DATA STATISTICAL LEARNING

Skewed class boundary is a subtle but severe problem that arises in using an SVM classifier—in fact in using *any* classifier—for real world problems with imbalanced training data. To understand the nature of the problem, let us consider it in a binary (positive vs. negative) classification setting. Recall that the Bayesian framework estimates the posterior probability using the class conditional and the prior.[26] When the training data are highly imbalanced, it can be inferred that the state naturally favors the majority class much more than the other. Hence, when ambiguity arises in classifying a particular sample because of similar class conditional densities for the two classes, the Bayesian framework will rely on the large prior in favor of the majority class to break the tie. Consequently, the decision boundary will skew toward the minority class and cause a high incidence of false negatives. Why do we care about remedying this problem? False negatives in a surveillance application (i.e., failing to identify a suspicious event) can have catastrophic consequences.

While the Bayesian framework gives the optimal results (in terms of the smallest average error rate) in a theoretical sense, one has to be careful in applying it to real-world applications, such as security surveillance and disease diagnosis. The risk (or consequence) of mispredicting a positive event (a false negative) far outweighs that of mispredicting a negative event (a false positive). It is well known that in a binary classification problem, Bayesian risks are defined as:

$$\begin{aligned} R(\alpha_p|\mathbf{x}) &= \lambda_{pp}P(\omega_p|\mathbf{x}) + \lambda_{pn}P(\omega_n|\mathbf{x}) \\ R(\alpha_n|\mathbf{x}) &= \lambda_{np}P(\omega_p|\mathbf{x}) + \lambda_{nn}P(\omega_n|\mathbf{x}) \end{aligned} \tag{5}$$

where p and n refer to the positive and negative events, respectively, λ_{np} refers to the risk of a false negative, and λ_{pn} is the risk of a false positive. The decision about which action (α_p or α_n) to take—or which has a smaller risk—is affected not just by the event likelihood (which directly influences the misclassification error), but also by the risk of mispredictions (λ_{np} and λ_{pn}).

Preliminary Results

Several attempts have been made to improve class-prediction accuracy of SVMs.[27–31] Given the class prediction function of SVMs,

$$sgn\left(f(\mathbf{x}) = \sum_{i=1}^{n} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b\right), \tag{6}$$

three parameters can affect the decision outcome: b, α_i, and K. Our empirical study[32] shows that the only effective method for improving SVMs, however, is through adaptively modifying K based on the training data distribution.

To adaptively modify K, our prior work[32] proposed using adaptive conformal transformation (ACT). Kernel-based methods, such as SVMs, introduce a mapping function Φ which embeds the $\mathcal{I}$ (input space, or the space formed by the features) into a high-dimensional $\mathcal{F}$ as a curved Riemannian manifold $\mathcal{S}$ where the mapped data reside.[33] A Riemannian metric $g_{ij}(\mathbf{x})$ is then defined for $\mathcal{S}$, which is associated with the kernel function $K(\mathbf{x}, \mathbf{x}')$ by

$$g_{ij}(\mathbf{x}) = \left(\frac{\partial^2 K(\mathbf{x}, \mathbf{x}')}{\partial x_i \partial x'_j}\right)_{\mathbf{x}'=\mathbf{x}}. \tag{7}$$

The metric g_{ij} shows how a local area around $\mathbf{x}$ in $\mathcal{I}$ is magnified in $\mathcal{F}$ under the mapping of Φ. The idea of conformal transformation in SVMs is to enlarge the margin by increasing the magnification factor $g_{ij}(\mathbf{x})$ around the boundary (represented by support vectors) and to decrease it around the other points. This could be implemented by a conformal transformation of the related kernel $K(\mathbf{x}, \mathbf{x}')$ according to Eq. 7, so that the spatial relationship between the data would not be affected too much.[27] Such a conformal transformation can be depicted as

$$\tilde{K}(\mathbf{x}, \mathbf{x}') = D(\mathbf{x})D(\mathbf{x}')K(\mathbf{x}, \mathbf{x}'). \tag{8}$$

In the above equation, $D(\mathbf{x})$ is a properly defined positive conformal function. $D(\mathbf{x})$ should be chosen in such a way that the new Riemannian metric $\tilde{g}_{ij}(\mathbf{x})$, associated with the new kernel function $\tilde{K}(\mathbf{x}, \mathbf{x}')$, has larger values near the decision boundary. Furthermore, to deal with the skew of the class boundary caused by imbalanced

classes, we magnify $\tilde{g}_{ij}(\mathbf{x})$ more in the boundary area close to the minority class. In,[32] we demonstrate that an RBF distance function such as

$$D(\mathbf{x}) = \sum_{k \in \mathbf{SV}} \exp\left(-\frac{|\mathbf{x} - \mathbf{x}_k|}{\tau_k^2}\right) \tag{9}$$

is a good choice for $D(\mathbf{x})$. By carefully adjusting τ_k^2's based on the support vector ratios, our preliminary results show that ACT outperforms traditional methods on several datasets to correct the skewed boundary.

6. SYNERGISTIC INTEGRATION

In this research thread, our aim is to demonstrate the viability of scalable wireless camera networks via system design, simulation and prototype deployment under realistic, representative terrain conditions. By "scalable," we mean that the network protocols must scale easily to hundreds of cameras in terms of both throughput and energy efficiency. We show below that traditional wireless communication protocols are not suitable for camera networks, and we will pursue the joint design of communication protocols with our data-fusion, and event-recognition algorithms, which we have presented in the previous sections.

The task of designing communication protocols that are scalable has been addressed predominantly in terms of scalability of network and transport layer protocols. The past effort has made good progress, but the scalability problem is still far from being solved. One unique challenge in the sensor-network setting is that the performance metrics and data characteristics in such a network are quite different from those of a traditional wireless network. Our research activities to meet the challenge of new performance metrics and data characteristics are as follows:

1. *Energy-efficient protocol design.* While many personal multimedia systems (such as the cellular phone system) as well as wired communication systems fall in the bandwidth-limited regime of network operation, camera networks with severely limited energy supplies fall in the energy-limited regime of network operation. Energy-limited wireless ad hoc networks have constituted a very active field of research in the past few years; however, measures of system performance, such as system lifetime, have not been related to measures of network capacity. In this task, we develop a new measure of capacity called "bits-per-Joule capacity" that measures the maximum number of bits that a wireless camera network can transfer per Joule of energy deployed into the network. We will use this useful metric to compare the performance of different network topologies for wireless camera networks.

2. *Data prioritization.* The information content in camera networks is measured, not by end-to-end data rates, but by semantic information content. In addition, the semantic information content must be relayed to the server with the lowest end-to-end delay possible to alert the master site in a timely manner. This requirement has important consequences in the design and choice of dynamic topology control protocols, which will be a major emphasis of this task.

3. *Topology control and network configuration.* Not only must a video camera network operate with high energy efficiency, but also the underlying network protocols must support camera mobility. Mobility of video cameras is a very desirable feature because a group of cameras can be moved on vehicles to cover a desired area. Camera mobility introduces significant challenges: When the cameras move around, the resulting network must remain connected in order for the Kalman filter algorithms (in Section 3) to send the information reliably to the master for data fusion. Mobile networks of video cameras admit the usage of only localized topology control protocols whose localization increases with the amount of mobility. Nodes that run localized topology control protocols use only the information in their own environment and set of immediate neighbors. As mobility increases, information that can be transferred from remote ends of the network becomes stale (for determining properties such as connectivity).

Preliminary Results

We discuss our preliminary results in two parts: *bits-per-Joule capacity* as a novel metric to measure the capacity of ad hoc wireless networks, and *distributed topology design* for energy-efficient routing and neighbor discovery.

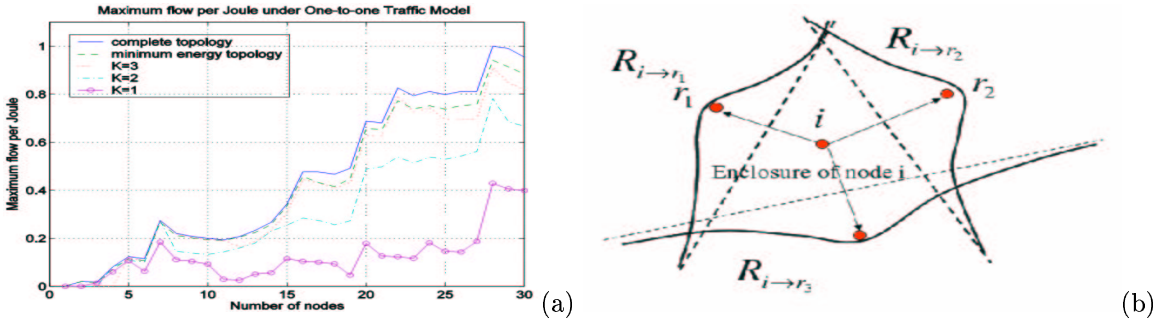


Figure 6. (a) Bits-per-Joule sum capacity under one-to-one traffic model. (b) Enclosure of a node.

Bits-per-Joule Capacity

The bits-per-Joule capacity of a wireless ad hoc network is the maximum number of information bits transferred (under a given traffic pattern) per Joule of energy consumed in the network. We have demonstrated in[34] that in terrestrial wireless environments, the bits-per-Joule capacity of a randomly deployed wireless network grows as more nodes are deployed onto a fixed area, assuming that each node wishes to send traffic to a randomly chosen destination node in the network. The main intuition behind this result is that as more nodes are deployed onto a fixed area, the inter-nodal distances get shorter; hence, the transmit power levels can be reduced and thus conserve energy at the nodes. However, at the same time, the relay traffic that each node has to carry increases as the number of nodes in the network grows. The main result of[34] is that the gains due to transmit power reduction outweigh the losses due to the relay traffic, and thus the bits-per-Joule capacity of a multi-hop, sensor network increases as the number of nodes increases. This is the main promising result encouraging the deployment of wireless networks on large scales for application scenarios in which energy is the limiting resource.

In Figure 6(a), we display the results for a network as we grow the network from 2 nodes to 30 nodes. The graph shows the increasing bits-per-Joule capacity as a function of the number of nodes for 3 different topologies. The first one is the complete topology in which every node can potentially transmit to any other node directly. The performance of the complete topology provides a theoretical benchmark against which the performance of practical topologies can be compared. The complete topology is not practically attainable because the channel gains to all of the other nodes in a topology are typically not available. In ad hoc networks of portable and mobile devices, each device is usually connected to only a few devices in its neighborhood. In the K-best-neighbor topology displayed on this graph, each node has K neighbors where K ranges from 1 to 3. The minimum energy topology (MET) that is shown here is the one in which every node has the global minimum energy path to every other node in the network. We see that both the minimum energy topology and the 3-best-neighbor topologies come appreciably close to the theoretical benchmark performance. Therefore, bits-per-Joule capacity also serves as a useful metric to compare the performance of difference topologies and to determine the minimum number of neighbors required by each node to capture a large percentage of the theoretical performance.

Distributed Topology Control

When cameras are deployed onto a new terrain, they must first establish communication with each other. Such communication will enable in-network data fusion as well as exchange of soft probability estimates that can be used for event recognition, for instance, if a subset of the cameras are observing the same area from different angles. This type of communication with one's neighbors is also required for the entire set of cameras to establish a connected network in which the packets generated and fused among a group of nodes can be transferred in a multi-hop fashion to the ultimate collection site or the headquarters. We developed in[35] an algorithm, called the "enclosure algorithm," for fast, efficient, localized topology control for ad hoc networks. This architecture combines the position estimates from various sources such as GPS receivers, or using different techniques such as time difference of arrival (TDOA), time of arrival (TOA) and angle of arrival (AOA). With the availability of a multitude of position estimation techniques and devices today, the enclosure algorithm[35] has become a very useful tool in neighbor discovery and dynamic link configuration in ad hoc networks.

The main idea behind the enclosure algorithm is illustrated in Figure 6(b). In this figure, we show a "transmit node" (denoted by i) that has found 3 neighbors in its search. Then, transmit node i computes a "relay region"

with each of these 3 neighbors (the boundaries of these regions are shown in the figure). We have shown that in order to discover the global minimum energy links in a network, a node needs to search only its enclosure and can drop the rest of the links from consideration as the remainder of the links are suboptimal from the perspective of energy consumption. In mobile networks, via the use of network synchronization techniques, it is possible to wake up all the nodes at the same time. Upon waking up, a node updates its enclosure by checking whether it still detects presence of its neighbors from the previous cycle period of the network. If its neighbors have shifted due to mobility (of themselves or of the transmit nodes), the enclosure algorithm can dynamically compute the new region of enclosure. Via simulations,[35] we have demonstrated that the enclosure algorithm can be run while incurring minimal overhead in energy consumption to track the neighbors in mobile networks up to a certain displacement where the network can still track the neighbors. When mobility is extremely high, the efficiency of the enclosure algorithm is reduced to that of recomputing the enclosure from scratch upon every iteration. However, even at high mobility, we have demonstrated that the algorithm works albeit with reduced efficiency.

Finally, the enclosure algorithm can also tolerate faults or failures of the nodes in the network. When a node's neighbors have dropped out of the network, the enclosure algorithm can detect such failures in the next network iteration and re-compute the neighbor set. As a result, this algorithm is very suitable for military operations and emergency rescue networks. We will further exploit the full power of the enclusoure algorithm by exploring the information at the upper-layer data-fusion and event-recognition algorithms.

7. CONCLUSION

We have presented our *event sensing* paradigm and its five core research threads. We discussed preliminary results, and briefly sketched our future research directions. We will focus on building a prototype to establish a testbed for our future research and validation.

Acknowledgments

The first author would like to thank the support of two NSF grants: NSF Career IIS-0133802 and NSF ITR IIS-0219885.

REFERENCES

1. E. Chang and Y.-F. Wang, "ACM First International Workshop on Video Surveillance," November 2003.
2. R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson, "A system for video surveillance and monitoring (VSAM project final report)," *CMU Technical Report CMU-RI-TR-00-12* , 2000.
3. G. Wu, Y. Wu, L. Jiao, Y.-F. Wang, and E. Y. Chang, "Multi-camera spatio-temporal fusion biased sequence-data learning for video surveillance," *ACM International Conference on Multimedia* , November 2003.
4. B. Babcock, S. Babu, M. Datar, R. Motwani, , and J. Widom, "Models and issues in data stream systems," in *Principles of Database Systems (PODS)*, June 2002.
5. L. Golab and M. T. Ozsu, "Issues in data stream management," in *Proc.of ACM SIGMOD International Conference on Management of Data*, **32**, (San Diego, California, USA), June 2003.
6. R. Motwani, J. Widom, A. Arasu, B. Babcock, S. Babu, M. Datar, G. Manku, C. Olston, J. Rosenstein, and R. Varma, "Query processing, resource management, and approximation in a data stream management system," in *In Proceedings of the First Biennial Conference on Innovative Data Systems Research (CIDR)*, (Asilomar, California, USA), January 2003.
7. D. Abadiand, D. Carneyand, U. Cetintemel, M. Cherniack, C. Convey, C. Erwin, E. Galvez, M. Hatoun, J. Hwang, A. Maskey, A. Rasin, A. Singer, M. Stonebraker, N. Tatbul, Y. Xing, R.Yan, and S. Zdonik, "Aurora: A data stream management system (demonstration)," in *Proc.of ACM SIGMOD International Conference on Management of Data*, (San Diego, CA), June 2003.
8. A. Arasu, B. Babcock, S. Babu, M. Datar, K. Ito, R. Motwani, I. Nishizawa, U. Srivastava, D. Thomas, R. Varma, and J. Widom, "Stream: The stanford stream data manager," in *IEEE Data Engineering Bulletin*, **26**, March 2003.

9. R. Cheng, D. V. Kalashnikov, and S. Prabhakar, "Evaluating probabilistic queries over imprecise data," in *Proc.of ACM SIGMOD International Conference on Management of Data*, (San Diego, CA, USA), June 9–12 2003.
10. C. Olston, J. Jiang, and J. Widom, "Adaptive filters for continuous queries over distributed data streams," in *Proc.of ACM SIGMOD International Conference on Management of Data*, (San Diego, California, USA), June 2003.
11. N. Tatbul, U. Cetintemel, S. Zdonik, M. Cherniack, and M. Stonebraker, "Load shedding in data streams," in *29th International Conference on Very Large Data Bases (VLDB)*, pp. 309–320, (Berlin, Germany), September 2003.
12. A. Jain, E. Chang, and Y.-F. Wang, "Adaptive stream resource management using kalman filters," *ACM Internation Conference on Management of Data (SIGMOD), to appear* , June 2004.
13. R. G. Brown, *Introduction to Random Signal Analysis and Kalman filtering*, Wiley, New York, NY, 1983.
14. P. S. Maybeck, *Stochastic Models, Estimation, and Control, vol. 1*, Academic Press, New York, NY, 1979.
15. J. Long and G. Wu and Y. Wu and E. Chang and Y. F. Wang, "The Anatomy of a Security Surveillance System," *ACM Multimedia System Journal* , 2004. accepted.
16. E. Church, "Revised Geometry of the Aerial Photograph," *Bulletin of Aerial Photogrammetry* **15**, 1945.
17. O. Faugeras, *Three-Dimensional Computer Vision*, MIT Press, Cambridge, MA, 1993.
18. R. Horaud, F. Dornaika, B. Lamiroy, and S. Christy, "Object pose: The link between weak perspective, paraperspective and full perspective," *Int. J. Comput. Vision* **22**, pp. 173–189, 1997.
19. G. Xu and Z. Zhang, *Epipolar Geometry in Stereo, Motion and Object Recognition*, Kluwer Academic Publishers, The Netherlands, 1996.
20. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Analy. Machine Intell.* **22**, pp. 1330–4, 2000.
21. T. S. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," *In Advances in Neural Information Processing Systems* , 1998.
22. X. Zhu, J. Fan, A. Elmagarmid, and X. Wu, "Hierarchical visual summarization and content description joint semantic and visual similarity," *ACM Multimedia Systems* (1), 2003.
23. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. Jordan, "Learning the kernel matrix with semi-definite programming," *Proc. of Int'l Conference on Machine Learning (ICML)* , 2002.
24. B. Scholkopf and A. Smola, "Learning with kernels," *MIT Press* , 2001.
25. S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of molecular biology* , 1970.
26. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, Boston, MA, 2 ed., 1990.
27. S. Amari and S. Wu, "Improving support vector machine classifiers by modifying kernel functions," *Neural Networks* **12**(6), pp. 783–789, 1999.
28. K. Crammer, J. Keshet, and Y. Singer, "Kernel design using boosting," In *Advances in Neural Information Processing Systems* , 2003.
29. Y. Lin, Y. Lee, and G. Wahba, "Support vector machines for classification in nonstandard situations," *Machine Learning* **46**, pp. 191–202, 2002.
30. C. S. Ong, A. J. Smola, and R. C. Williamson, "Superkernels," In *Advances in Neural Information Processing Systems* , 2003.
31. K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," *Proceedings of the International Joint Conference on Artificial Intelligence* , pp. 55–60, 1999.
32. G. Wu and E. Chang, "Adaptive feature-space conformal transformation for imbalanced data learning," In *Proceedings of the Twentieth International Conference on Machine Learning* , pp. 816–823, August 2003.
33. C. Burges, "Geometry and invariance in kernel based methods. in adv. in kernel methods: Support vector learning," pp. 89–116, 1999.
34. V. Rodoplu and T. H. Meng, "Bits-per-joule capacity of energy-limited wireless ad hoc networks," in *Proc. IEEE GLOBECOM*, **1**, pp. 16–20, 2002.
35. V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks," *IEEE J. Sel. Areas Commun.* **17**, pp. 1333–1344, 1999.