Urban Surveillance Systems: From the Laboratory to the Commercial World

IOANNIS PAVLIDIS, SENIOR MEMBER, IEEE, VASSILIOS MORELLAS, MEMBER, IEEE, PANAGIOTIS TSIAMYRTZIS, AND STEVE HARP

Invited Paper

Research in the surveillance domain was confined for years in the military domain. Recently, as military spending for this kind of research was reduced and the technology matured, the attention of the research and development community turned to commercial applications of surveillance. In this paper we describe a state-of-the-art monitoring system developed by a corporate R&D lab in cooperation with the corresponding security business units. It represents a sizable effort to transfer some of the best results produced by computer vision research into a viable commercial product. Our description spans both practical and technical issues. From the practical point of view we analyze the state of the commercial security market, typical cultural differences between the research team and the business team and the perspective of the potential users of the technology. These are important issues that have to be dealt with or the surveillance technology will remain in the lab for a long time. From the technical point of view we analyze our algorithmic and implementation choices. We describe the improvements we introduced to the original algorithms reported in the literature in response to some problems that arose during field testing. We also provide extensive experimental results that highlight the strong points and some weaknesses of the prototype system.

Keywords*—*Multicamera fusion, object tracking, security market, security system, surveillance system, threat assessment.

I. INTRODUCTION

The current security infrastructure could be summarized as follows. 1) Security systems act locally and they do not cooperate in an effective manner. 2) Very high value assets are inadequately protected by antiquated technology systems. 3) Reliance on intensive human concentration to detect and assess threats.

Manuscript received October 16, 2000; revised June 7, 2001. The prototype DETER system was funded by two major Research Intiative awards from Honeywell Labs.

I. Pavlidis, V. Morellas, and S. Harp are with Honeywell Laboratories, Minneapolis, MN 55418 USA.

P. Tsiamyrtzis is with the School of Statistics, University of Minnesota, Minneapolis, MN 55455 USA.

Publisher Item Identifier S 0018-9219(01)08433-X.

Taking into account the above state of commercial art and the maturation of surveillance research many R&D teams, such as ours, thought that the transfer of surveillance technology to production is not only warranted but also easy. In this context, our team undertook a major effort in coordination with the Honeywell security business units to field one of the first advanced urban surveillance products. In our endeavor we came to learn that good laboratory technology should be supported by deep knowledge of the business, market, and user realities to become a success story. Actually, we can now corroborate that in certain cases technology transfer can be as challenging as the basic research that preceded it.

The result of our endeavor is Detection of Events for Threat Evaluation and Recognition (DETER), a prototype urban surveillance system aimed for the high end of the security market. DETER can be seen as an attempt to bridge the gap between current systems reporting isolated events and an automated cooperating network capable of inferring and reporting threats, a function currently being performed by humans. The prototype DETER system is installed at the parking lot of Honeywell Laboratories (HL) in Minneapolis. The computer vision module of DETER is reliably tracking pedestrians and vehicles and is reporting their annotated trajectories to the threat assessment module for evaluation. DETER features a systematic optical and system design that sets it apart from "toy" surveillance systems.

In Section II of this paper, we analyze the current state of the security market and how it affected our research and development effort. Then, in Section III, we move on to describe the recent technical developments reported in the research literature. Sections IV–VIII describe and analyze the characteristics of our prototype surveillance system. In Section IX, we report extensive experimental results from actual field operations that highlight some strong as well as some weak points of DETER. Finally, in Section X, we conclude our paper by summarizing the business and technical results, drawing conclusions, and outlining our strategic and tactical plan for the future.

II. The Current State of the Security Market

The security business has a surprisingly long history. For example, Pinkerton, one of the premier security services companies, recently celebrated its 150th anniversary. Traditionally, the security industry relies primarily on its human resources. Technology is not always highly regarded and sometimes is viewed with suspicion. The last universally accepted technological change in the security industry is the adoption of radio communication between guarding parties. Many of us may have the impression that analog video recording is another universally adopted technology by the security industry. This is, however, far from true. There are significant portions of the security market that do not use video recording at all and rely exclusively on human labor. A good example is the majority of stake-out operations performed by law enforcement agencies in the United States.

An understanding of the industry's peculiarities and the forces that shape up its current profile is essential for anyone who is interested to perform technology transfer in the security domain. Below, we enumerate what we consider the most important characteristics of the current security market.

Low Profit Margin: The security market is very cost sensitive. One can identify two major segments in the security market: the *Home Security* and the *Building Security*. The competition in the Home Security segment is fierce and the profit margin very low. The average monthly subscription to a home security service in U.S. is about $20 per month in year 2000 valuation. The initial installation cost sometimes is waived as a means to attract customers. The Building Security segment is at the upper end of the market but still cost is a major issue and the volume of this segment is much smaller than the *Home Security* segment. In an era where quarterly profits make or brake corporate giants in areas with much higher profit margin, the security industry always struggles to "make the numbers." Its strategic horizon usually does not extend beyond six months. It is frequently cited in the technical literature that the current low cost of computational power and cameras will open up the way for the automation of surveillance products and services. As it turns out, "low cost" is a relative term. A Pentium II PC box running at 233 MHz and priced around $200 in year 2000 valuation is considered a high-end device with a substantial price tag.

Resistance to Change: Like most traditional industries, the security industry is not an advocate of innovation by nature. Part of the problem is that some of its customers are also resistant to change. A typical example is the failed attempt to introduce GPS receivers into police cruisers in several North American cities. The GPS receivers would enable police departments to know exactly the position of all their cruisers all the time. There are obvious benefits to personnel safety and resource scheduling from this technology. The police unions, however, opposed the plan because they considered it as an invasion of privacy to the lives of the individual officers.

Low-Tech Culture: The security industry is permeated by low-tech culture. The management and the engineers of the security business units are trained and grown within a low-tech environment and are ignorant and suspicious of state-of-the-art developments. Their users and customers are often underpaid and undereducated security guards that also view high technology with skepticism.

Hardware Mentality: The most advanced members of the security industry are probably the camera manufacturers. Even these, although they produce some advanced electronic products, have difficulty outfitting them with the necessary software. An example is Sony, which recently produced some excellent 1394 security cameras like the DFW-VL500 and started selling them in the market without the necessary software drivers. Since these cameras can send their video output only to a computer, without software drivers they were useless.

The problem is compounded by the mentality of the research community that can support the security industry with advanced video surveillance concepts. Computer vision researchers both in academia and corporate labs used to perform research for military surveillance projects where cost was not an issue. Even corporate researchers that performed research and development with a commercial application in mind used to do that in isolation hypothesizing the problems and need to be addressed. In most cases, the development concluded with a demo without addressing system design issues and without performing some rudimentary cost and market analysis. When they tried to sell the idea to a business unit for productization, the result was a predictable failure.

Despite the presence of many negative factors, the future of the security industry can be viewed only in positive light. And although the transformation of the industry and the market will take time to complete, it has already started happening in small steps. As a result of upcoming technology offerings, the Freedonia group is projecting significant growth of the security market during the next several years (see Fig. 1). This growth will fuel further research and development and will hopefully bootstrap the process of incorporating the security industry to the new economy.

Taking into account the practical realities, we decided to cooperate very closely both with the business unit that would ultimately productize our surveillance prototype as well as with potential customers. Out of this cooperation we quickly formed a very specific strategy.

1) The prototype should be developed and tested within an actual environment and not in the lab. This would be the ultimate proof of its fitness.
2) The prototype system should address security needs of buildings and not homes since the profit margin of a potential building product would be far greater and the competition in this market segment less fierce.
3) The prototype system should be geared toward perimeter surveillance and not toward indoor building surveillance. Admittedly, indoor building surveillance correlates more aptly with the notion of "big brother" and would generate bad publicity among the customers' workforce for our perspective product offering. The majority of U.S. business buildings are surrounded by parking lots. So it is the case with the perimeter of suburban malls and other public places. Therefore, we paid some particular attention to the parking lot scenario without overconstraining ourselves.

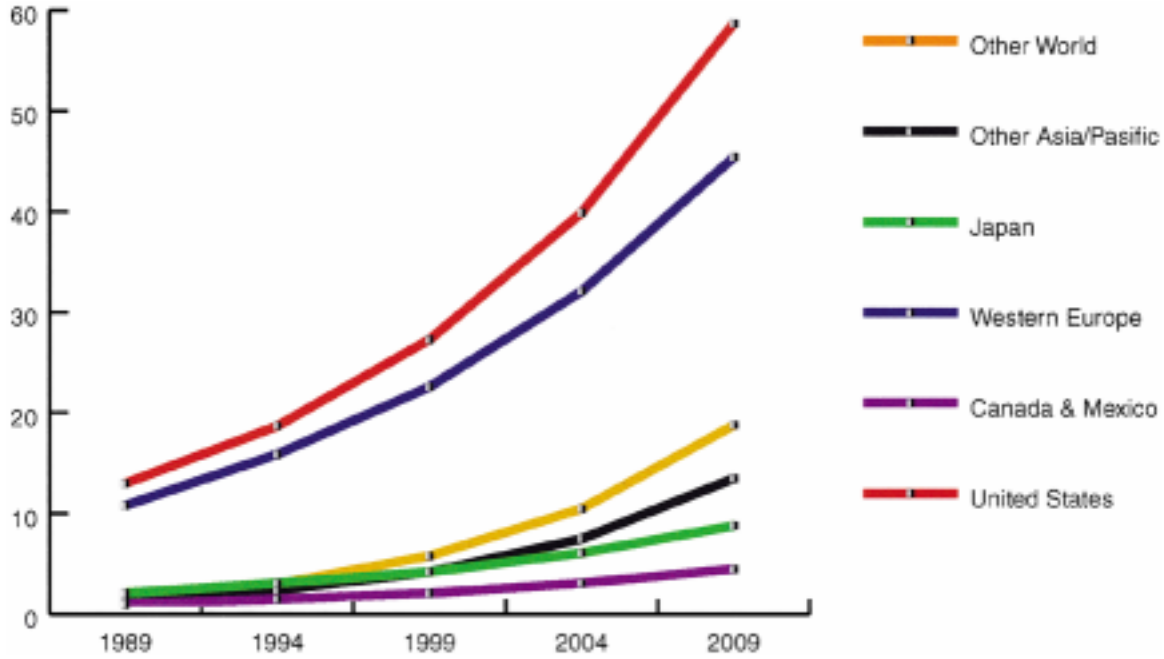

Fig. 1 The security service market by region in U.S. dollars (billions). The numbers after the year 2000 are projections (source: The Freedonia Group).

4) We did not try to invent needs. We paid particular attention to the real needs of our potential customers and tried to address them through state-of-the-art technological solutions. Two potential customers were interviewed extensively and their needs were factored in the design of the system to the degree possible. One customer was the security personnel of our building and the other was the Dade County Sheriff's office in Miami. One could group their feedback into two trends. The security personnel emphasized the necessity for a few simple automated alarms. An example was the capture of any vehicle that enters the parking lot of the building after hours. The sheriff's office showed an interest for the capture of more complicated traffic patterns. An example was a vehicle that enters the parking lot and exits after wandering for a while without ever parking. Also, the sheriff's office was placing a lot of emphasis on the portability of the system. They wanted a system that would be able to set easily and quickly, use it for a period, and then move it to another location. This is consistent with the mode of stakeout operations they perform. We decided to accommodate in the system design the detection of simple as well as somewhat more complicated traffic patterns but leave out any portability considerations. We determined that the portability question would substantially increase the technical risk and the development cost while it is of importance only to a small percentage of the customer base (law enforcement agencies). Our strategy is to address the portability problem in a subsequent stage of development after our baseline product offering generates some revenue first.
5) We decided to design the system in a manner that would be able to perform multiple functions (beyond the security domain). This would increase its appeal to potential customers. We have particularly focused on analyzing traffic statistics for the benefit of building operations. For example, traffic statistics may provide an insight into parking lot utilization during different times and days. This insight can support a functional redesign of the open space to better facilitate transportation and safety needs.
6) On one hand, the cost of the hardware components and installation for the system should be kept at the minimum because of the cost sensitivity of the security industry. On the other hand, the computer vision algorithms require substantial computational power and full coverage of the surveyed area. As a way of compromise, we chose not the low-end processors (233 MHz), currently in wide use by the security industry, but rather mid-end processors (500 MHz) that we project will become the mainstay at about the same time the prototype system will move into production in 2002. We also identified the need for an optimization method that, given the CAD design of the surveyed area, will produce the minimum number of cameras and their locations for full coverage.
7) We decided to choose off-the-self hardware and software development components and adopt an open architecture strategy. For example, we used off-the-shelf PCs, cameras, and nonembedded software tools. This was a radical move in the framework of an industry that is used to produce "proprietary systems." Our rationale is that open systems reduce development and maintenance cost and time. Open systems can also capitalize upon existing assets at the customer's site and make the technology transition more appealing. We also believe that nowadays the best way to outma-

neuver the competition is not by building proprietary systems, but by delivering continuous value to the customers through innovation and streamlined operations.

8) In addressing the technical challenges for our surveillance system, we decided not to start from scratch. The purpose of a corporate R&D effort is not innovation for the sake of innovation but innovation for the sake of results. We performed a careful evaluation of the technical literature to find an appropriate starting point. Then, we filled up the gaps and improved the initial idea in step with our experimental experience and results.

III. Relevant Technical Work

The computer vision community has performed extensive research in the area of video-based surveillance for the past 20 years. Initially, the research was focused almost exclusively to military applications and employed nonvisible band cameras (e.g., thermal, laser, and radar). The emphasis was on the recognition of military targets (automatic target recognition or ATR). An interesting survey of this type of work can be found in [1]. Upon the end of the cold war in the 1990s, attention shifted gradually to surveillance applications in nonmilitary settings using visible band cameras. The research emphasis was also shifted from object recognition to tracking of human and vehicular motion. Even the military participated in this research shift to prepare for the so called "asymmetric threat." Asymmetric threat refers to the possibility of terrorist activities against animate and inanimate government assets (e.g., government officials, embassies, etc.). The Video Surveillance and Monitoring program exemplifies the shift to urban surveillance scenarios. The VSAM program was funded by DARPA in 1997-99 [2], [3] and pushed the state of the art to a point where future commercial application of the technology is not unthinkable anymore. No large-scale research and development effort has been undertaken since then in the area of surveillance. Isolated research efforts, however, continued to push the state of the art in a variety of urban surveillance applications [4], [5]. In these efforts, we witness increased participation by commercial R&D labs [6], [7].

The latest research activities in the area of commercial surveillance applications are ripe as they are aided by improvements in the computational power, the camera technology, and the introduction of robust statistical methods in computer vision. All these research efforts try to address to one degree or the other the fundamental urban surveillance question: *motion detection*. If a system can reliably detect motion, then it can reason about motion patterns, record intrusion, and issue alerts (*reason-record-issue*). It is worth mentioning that existing commercial security systems cannot perform the sequence of the above three functions. They rely exclusively on human attention and labor to close the feedback loop.

Some research groups reach a lot further than the basic reason-record-issue paradigm and perform research on analyzing human motion or modeling human interactions [8], [9]. Although these investigations are scientifically elegant, their value to the security industry in the near- and mid-term is minimal. The industry is preoccupied with a lot more mundane problems at the moment.

A variety of moving object segmenters has been reported in the literature. There are two conventional approaches to moving object segmentation with respect to a static camera: temporal differencing [10] and background subtraction [11]. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting all the relevant object pixels. Background subtraction provides the most complete object data, but is extremely sensitive to dynamic scene changes due to lighting and extraneous events. Most researchers have abandoned nonadaptive methods of backgrounding, which are useful only in highly supervised, short-term tracking applications without significant changes in the scene. More recent adaptive backgrounding methods [12] can cope much better with environmental dynamism. They still, however, cannot handle bimodal backgrounds and have problems in scenes with many moving objects. Stauffer *et al.* [13], [14] have proposed a more advanced object detection method based on a mixture of Normals representation at the pixel level. This method features a far better adaptability and can handle even bimodal backgrounds (e.g., swaying tree branches). The secret is in the powerful representation scheme. Each Normal reflects the expectation that samples of the same scene point are likely to display Gaussian noise distributions. The mixture of Normals reflects the expectation that more than one process may be observed over time.

Elgammal *et al.*, in [15], propose a generalization of the Normal mixture model where density estimation is achieved through a Normal kernel function. Their method features some improved behavior with respect to the method proposed in [13] including the suppression of shadows. In general, the mixture of Normals paradigm is not only theoretically elegant but has also produced promising test results in challenging outdoor conditions. It is for this reason we chose it as the baseline algorithm for our surveillance system.

Clearly, most of the research in urban surveillance system was directed toward moving object segmentation. There is a good reason for that since failures at the segmentation level can seal the fate of the entire surveillance system. Nevertheless, a comprehensive surveillance system involves additional technologies to moving object segmentation. These technologies include tracking, multicamera fusion, and threat assessment. The research community addressed these problems to various degrees. Tracking refers to the association of segmented objects across the timeline. The tracking methods employed in surveillance systems are usually borrowed from research performed for radar applications. The issue of multicamera fusion is an important one for seamless tracking in large open spaces that cannot be covered by a single camera. Researchers have addressed this issue in the surveillance and other contexts [16]–[18]. The interested reader can look at [19] for a thorough presentation of the relevant mathematics. The stage of threat assessment is the least explored. It is the one that interfaces with the human operator, however, and in this respect is very important. In Section VIII, we present our approach to threat assessment, which focuses on the detection of a few threatening motion patterns.

IV. System Architecture

A comprehensive urban video surveillance system, such as DETER, depends primarily on two different technologies: computer vision and threat assessment. The computer vision part consists of the optical and system design, the moving object segmentation and tracking and the multicamera fusion stages. The threat assessment part consists of the feature assembly, the off-line training, and the threat classification stages (see Fig. 2). We will give a brief overview of each stage and compare our solutions to others proposed in the literature.

Our system is probably the only one that features a formal optical and system design stage. Most of the efforts reported in the literature had as their main objective to demonstrate the feasibility of a novel idea and they did not pay any attention to the practical aspects of fielding a surveillance system. There is a number of requirements that a surveillance system needs to fulfill to function properly and be commercially viable. First, it should ensure full coverage of the open space or blind spots may pause the threat of a security breach. It is often argued in the technical literature that video sensors and computational power are getting cheaper and therefore can be employed in mass to provide coverage for any open space [20]. In reality, things are not so rosy. Most of the cheap video sensors still do not have the required resolution to accommodate high-quality object tracking. Both cheap and expensive cameras also need to become weather proof for employment outdoors, which increases their cost substantially. Then, it is the issue of installation cost that includes the provision of power and the transmission of video signals, sometimes at significant distances from the building. The installation cost for each camera is usually a figure many times its original value. Even if there were no cost considerations, cameras cannot be employed arbitrarily in public places. There are restrictions due to the topography of the area (e.g., streets, tree lines) and due to city and building ordinances (e.g., aesthetics). All these considerations severely curtail the allowable number and positions of cameras for an urban surveillance system.

In addition to optical considerations there are also system design considerations including the type of computational resources, the computer network bandwidth, and the display capabilities. Due to the cost sensitivity of the security market, all these become critical issues and should be addressed in an optimal manner.

We achieve motion segmentation through a multi-Normal representation at the pixel level. Our method resembles the method described in [14] with some interesting modifications. The method identifies foreground pixels in each new frame while updating the description of each pixel's mixture model. The labeled foreground pixels can then be assembled into objects using a connected components algorithm [21]. Establishing correspondence of objects between frames (tracking) is accomplished using a linearly predictive multiple hypotheses tracking algorithm which incorporates both position and size.

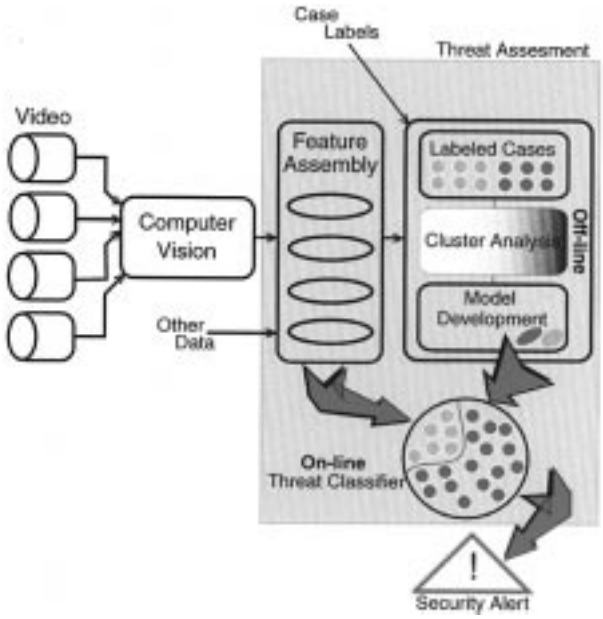

Fig. 2. Architecture of the DETER system.

No single camera is able to cover large open spaces, like parking lots, in their entirety. Therefore, we need to fuse the fields of view (FOV) of the various cameras into a coherent super picture to maintain global awareness. We fuse (calibrate) multiple cameras by computing the respective homography matrices. The computation is based on the identification of several landmark points in the common FOV between camera pairs.

The threat assessment portion of DETER consists of a feature assembly module followed by a threat classifier. Feature assembly extracts various security relevant statistics from object tracks and groups of tracks. The threat classifier decides in real time whetheher a particular point in feature space constitutes a threat. The classifier is assisted by an off-line threat modeling component (see Fig. 2).

V. Optical and System Design

The optical and overall system design for DETER includes the specification of a camera set arrangement that optimally covers the HL Minneapolis parking lot. It also includes the specification of the computational resources necessary to run the DETER algorithms in real-time. Finally, it includes the specification of the display hardware and software.

The optical design effort, in particular, has the following objectives.

1) Specify the camera model.
2) Specify the camera lens.
3) Specify the number of cameras.
4) Specify the camera locations.

We decided to employ dual channel camera systems. These systems utilize a medium-resolution color camera during the day and a high-resolution grayscale camera during the night. Switching from day to night operation is controlled automatically through a photo-sensor. The dual channel technology capitalizes upon the fact that

color information in the low light conditions at night is lost. Therefore, there is no reason for employing color cameras during nighttime conditions. Instead we can employ cheaper and higher resolution grayscale cameras to compensate for the loss of color information. We have selected the camera model to be the DSE DS-5000 dual channel system. The color day camera has a resolution of $H_D = 480$ lines/frame. The grayscale night camera has a resolution of $H_N = 570$ lines/frame. The DSE DS-5000 camera system has a $2.8 - 6$ mm $f/1.4$ vari-focal auto iris lens for both day and night cameras. This permits us to vary the FOV of the cameras from FOV $= 44.4° - 82.4°$.

We seek an optimal solution that provides coverage to the entire parking lot area with the minimum number of cameras and installation cost. There are practical constraints imposed by the topography of the area under surveillance. For example, we cannot place a camera pole in the middle of the road, existing poles and rooftops should be utilized to the extent possible to reduce the installation cost and city codes regarding the aesthetics have to be obeyed. Taking into account all these considerations we can delineate in the computer-aided design (CAD) of the parking lot the possible installation sites. These are usually only a small fraction of the entire open area and, therefore, our search space is drastically reduced.

The installation search space is reduced even further when we consider the constraints imposed by the computer vision algorithms. Specifically:

1) An urban surveillance system such as DETER is monitoring two kind of objects: vehicles and people. In terms of size, people are the smallest objects under surveillance. Therefore, their footprint should drive the requirements for the limiting range of the cameras. In turn, the determination of the limiting range will help us to verify if there is any space in the parking lot that is not covered under any given camera configuration.
2) Each camera should have an overlapping FOV with at least one more camera. The overlapping arrangement should be done in such a way, so that we are able to transition from one camera to the other through indexing of the overlapped areas and manage to visit all the cameras in a unidirectional trip without encountering any discontinuity.
3) The overlapping in the FOVs should be between 25%–50% for the multicamera calibration algorithm to perform reliably. This requirement stems from the need to get several well-spread landmark points in the common field of view for accurate homography. Usually, a portion of the overlapping area cannot be utilized for landmarking because it is covered by nonplanar structures like tree lines. Therefore, at times the common area between two cameras may be required to cover as much as half of the individual FOVs.

As we mentioned earlier, the DSE DS-5000 cameras feature a vari-focal lens with a FOV that can range between 44.4° and 82.4°. We choose the intermediate value of FOV $=$ 60° as the basis of our calculations. To satisfy the overlapping constraints, we may need to increase or decrease the FOV of some of the cameras from this average value. The camera placement algorithm proceeds as follows.

1) In one of the allowed installation sites place a camera in such a way that its FOV borders the outer edge of the parking lot.
2) Continue adding cameras around the initial camera until you reach the next outer edge of the parking lot. Make sure there is at least 25% overlapping in neighboring FOVs.
3) Compute the limiting range of the installed cameras. By knowing the FOV and the limiting range, we know the full useful coverage area for each camera.
4) Continue with the next installation site that is just outside of the already covered area. Make sure that at least one of the new cameras overlaps at least 25% with one of the previous cameras.
5) Repeat the above three steps until the entire parking lot area is covered.
6) Make some post-processing adjustments. These involve usually the increase or reduction of the FOV for some of the cameras. This FOV adjustment is meant to either trim some excessive overlapping or add some extra overlapping in areas where there is little planar space (lots of trees).

Of particular interest is the computation of the camera's limiting range R_c. It is computed from the equation

$$R_c = \frac{P_f}{\tan(\text{IFOV})}$$

where P_f is the smallest acceptable pixel footprint of a human and IFOV is the instantaneous field of view. Based on our experimental experience, the signature of the human body should not become smaller than a $w \times h = 3 \times 9 = 27$ pixel rectangle on the focal plane array (FPA). Clusters with fewer than 27 pixels are likely to be below the noise level. If we assume that the width of an average person is about $W_p = 24$ in then the pixel footprint $P_f = 24/3 = 8$. The IFOV is computed from the following formula:

$$\text{IFOV} = \frac{\text{FOV}}{L_{\text{FPA}}}$$

where L_{FPA} is the resolution for the camera. For FOV $= 60°$ and $L_{\text{FPA}} = 480$ pixels (color day camera), the limiting range is $R_c = 305$ ft. For FOV $= 60°$ and $L_{\text{FPA}} = 570$ pixels (grayscale night camera), the limiting range is $R_c = 362$ ft. In other words, between two cameras with the same FOV, the higher resolution camera has larger useful range. Conversely, if two cameras have the same resolution, then the one with the smaller FOV has larger useful range. During post-processing, we needed to reduce the FOV (FOV $= 52°$) in some of the lower resolution day camera channels to increase their effective range limit. Extended tree lines in the HL parking lot necessitate larger overlapping areas than the anticipated minimum.

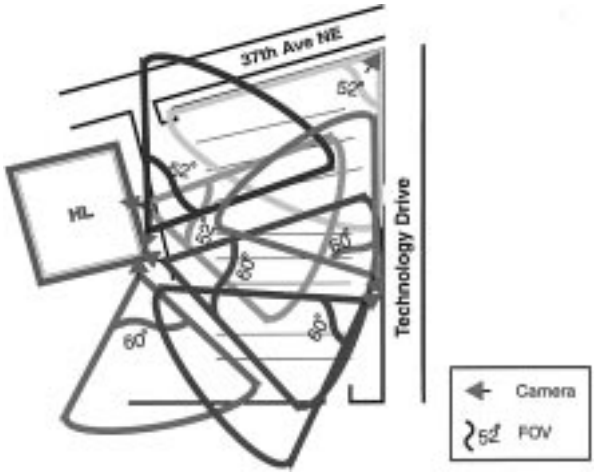

Fig. 3. Camera configuration scheme for DETER in the HL parking lot. This figure depicts the FOVs for the day channels of the cameras.

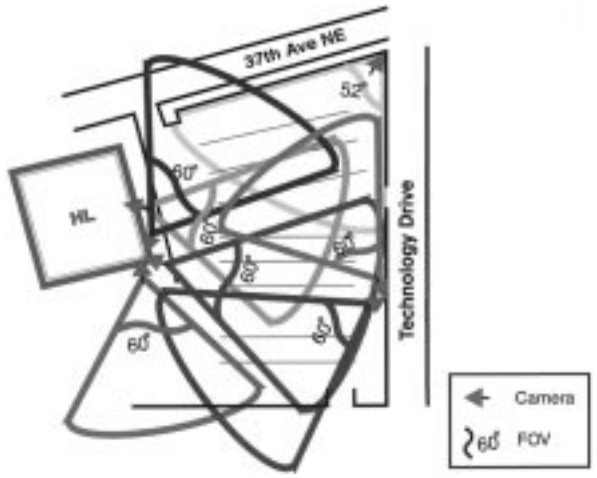

Fig. 4. Camera configuration scheme for DETER in the HL parking lot. This figure depicts the FOVs for the night channels of the cameras.

A good optical design is essential to the success of an urban surveillance system and many computer vision projects often ignore this aspect altogether. The principles, algorithms and computations we used for the DETER optical design can be codified and automate the optical design of future similar security systems in any other parking lot or open area.

Our study concluded that seven cameras in the configuration shown in Figs. 3 and 4 is the recommended arrangement for our parking lot. We have assigned one standard PC (500-MHz Pentium) for the processing requirements of each camera. One of the seven PCs is designated as the server and this is where the fusion of information from all seven cameras takes place. As a way of comparison, see in Fig. 5 the camera arrangement in the parking lot of our building before DETER. The inadequate coverage is the typical outcome of bad design and budgetary restrictions.

The fused video information is displayed in a 44-inch flat panel display along with all the necessary annotation. This comprehensive high-quality picture allows the security operator to maintain instant awareness without the distraction of multiple fragmented views. It also underlines our ultimate goal, which is the enhancement and not the replacement of the role of the security guard.

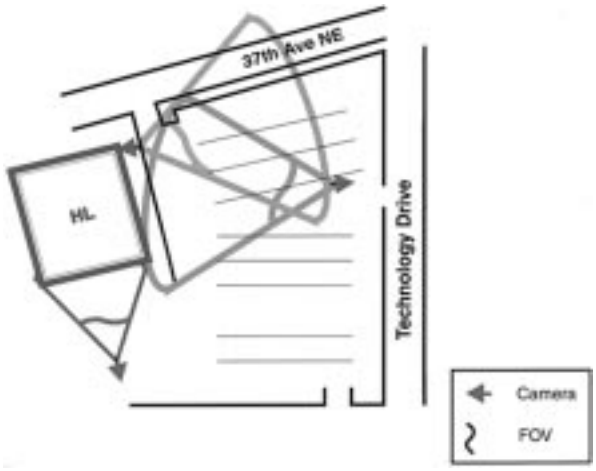

Fig. 5. Camera configuration scheme in the HL parking lot before DETER. The ad-hoc and inadequate coverage is obvious.

Our design philosophy is geared toward open systems. We chose to use standard NTSC cameras that are favored by the security industry. We do not aim at developing smart on-the-chip cameras. We project that the cost of developing special hardware and embedded software is quite substantial. Also, a product based on smart cameras would appeal only to new customers. The management of existing buildings would much rather prefer to upgrade their legacy infrastructure than scrap it altogether. With our design, they can use their old cameras and possibly add a few more to achieve complete coverage. Also, the computational hardware could be found for free. Most corporations renew their PCs every 2–3 years. Since DETER is designed to run on moderate PC hardware, recycled PC units can be used for the processing of the video information. There is no bandwidth problem between the camera and the PC since the standard coaxial cable can accommodate comfortably video transmissions of 30 frames per second. After the information is processed at the PC, either is stored locally or transmitted across the building's intranet on an event basis. Based on the above description, DETER can be sold more as an upgrade service instead of a new security product. We believe that this business model is necessary for the rapid spread of high technology in the security marketplace.

VI. Object Segmentation and Tracking

A. Initialization

The goal of the initialization phase is to provide statistically valid values for the pixels corresponding to the scene. These values are then used as starting points for the dynamic process of foreground and background awareness. Initialization happens only once and there are no strict real-time processing requirements for this phase. We process a certain number of frames N ($N = 70$) on-line or off-line.

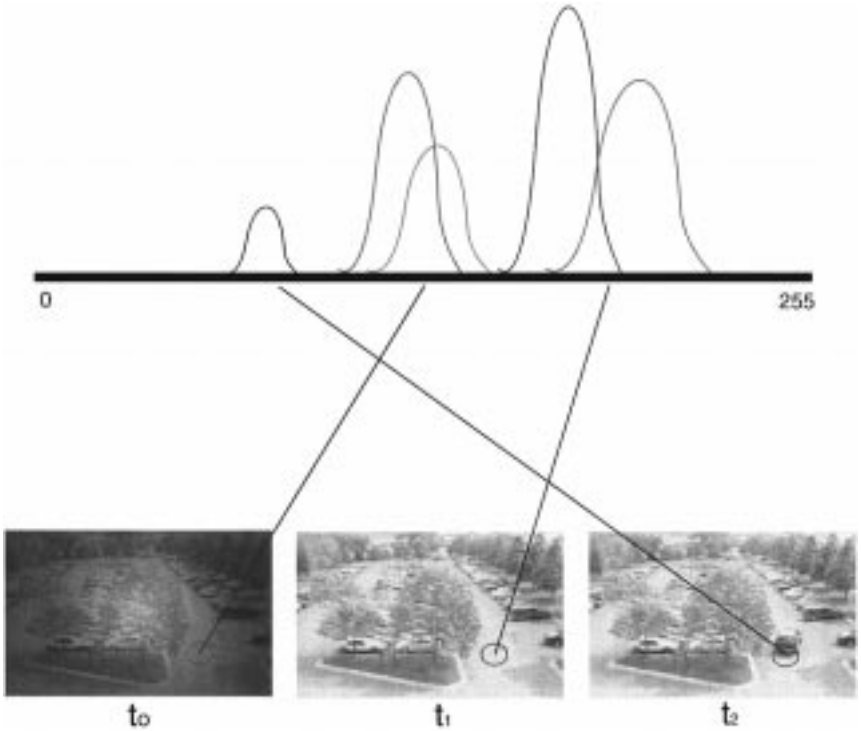

Fig. 6. Visualization of the mixture of normals model at the pixel level. The normals of a gray channel is depicted for simplicity purposes.

Each pixel $\mathbf{x}$ is considered as a mixture of five time-varying trivariate normal distributions

$$\mathbf{x} \sim \sum_{i=1}^{5} \pi_i \mathbf{N}_3 (\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$$

where

$$\pi_i \geq 0, \quad i = 1, \ldots, 5 \qquad \text{and} \qquad \sum_{i=1}^{5} \pi_i = 1$$

are the mixing proportions (weights) and $\mathbf{N}_3 (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a trivariate Normal distribution with vector mean $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$. The distributions are trivariate to account for the three component colors (red, green, and blue) of each pixel in the general case of a color camera. Please note that

$$\mathbf{x} = \begin{pmatrix} x^R \\ x^G \\ x^B \end{pmatrix}$$

where x^R, x^G, and x^B stand for the measurement we received from the red, green, and blue channel of the camera for the specific pixel

For simplification, the variance-covariance matrix is assumed to be diagonal with x^R, x^G, x^B having identical variance within each Normal component, but not across all components (i.e., $\sigma_k^2 \neq \sigma_l^2$ for $k \neq l$ components). Therefore,

$$\mathbf{x} \sim \sum_{i=1}^{5} \pi_i \mathbf{N}_3 \left[\begin{pmatrix} \mu_i^R \\ \mu_i^G \\ \mu_i^B \end{pmatrix}, \sigma_i^2 I \right].$$

Other similar methods reported in the literature initialize the pixel distributions either randomly or with the K-means algorithm. Random initialization results in slow learning during the dynamic mixture model update phase. Sometimes, it even results in instability. Initialization with the K-means or the expectation-maximization (EM) method [22] gives significantly better results. The EM algorithm is computationally intensive and takes the initialization process off-line for about 1 min. In the parking lot application where human and vehicular traffic is small, the short off-line interval is not a problem. Actually, the EM initialization performs a little better particularly if the weather conditions are dynamic (e.g., fast moving clouds). But, if the area under surveillance were a busy plaza (many moving humans and vehicles), the on-line K-means initialization might have been more preferable.

B. Segmentation of Moving Objects

The initial mixture model is updated dynamically thereafter. The update mechanism is based on the incoming evidence (new camera frames). Several things could change during an update cycle.

1) The form of some of the distributions could change (weight π_i, mean $\boldsymbol{\mu}_i$, and variance σ_i^2).
2) Some of the foreground states could revert to background and vice versa.
3) One of the existing distributions could be dropped and replaced with a new distribution.

At every point in time, the distribution with the strongest evidence is considered to represent the pixel's most probable background state. Fig. 6 presents a visualization of the

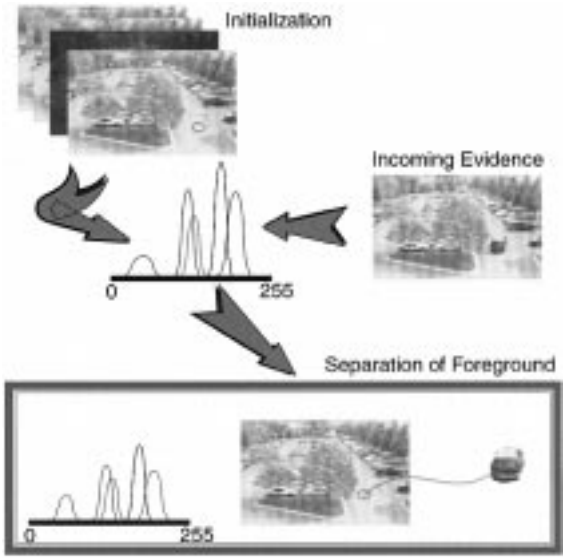

Fig. 7. Visualization of the mixture model update mechanism. The normals of a gray channel is depicted for simplicity purposes. The small ellipse marks the pixel area under monitoring.

mixture of Normal's model while Fig. 7 depicts the update mechanism for the mixture model.

The update cycle for each pixel proceeds as follows:

1) First, the existing distributions are ordered in descending order based on their weight values.
2) Second, the algorithm selects the first B distributions that account for a predefined fraction of the evidence T

$$B = \arg\min_b \left\{ \sum_{i=1}^{b} w_i > T \right\}$$

where w_i, $i = 1, \ldots, b$ are the respective distribution weights. These B distributions are considered as background distributions while the remaining $5 - B$ distributions are considered foreground distributions.
3) Third, the algorithm checks if the incoming pixel value can be ascribed to any of the existing Normal distributions. The matching criterion we use is the Jeffreys (J) divergence measure and is a key differentiator of our approach from other similar approaches.
4) Fourth, the algorithm updates the mixture of distributions and their parameters. The nature of the update depends on the outcome of the matching operation. If a match is found, the update is performed using the method of moments. This is also a key differentiator of our approach. If a match is not found, then the weakest distribution is replaced with a new distribution. The update performed in this case guarantees the inclusion of the new distribution in the foreground set, which is another novelty of our method.

The matching and model update operations are quite involved [23] and are described in detail in the next three subsections.

1) The Matching Operation: We use the Jeffreys divergence measure $J(f, g)$ [24] to determine whether the incoming data point belongs or not to one of the existing five distributions. The Jeffreys number measures how unlikely it is that one distribution (g) was drawn from the population represented by the other (f). For a presentation of the theoretical properties of the Jeffreys divergence measure, see [25]. The five existing Normal distributions are: $f_i \sim \mathbf{N}_3(\boldsymbol{\mu}_i, \sigma_i^2 I)$, $i = 1, \ldots, 5$. Since the $J(f, g)$ relates to distributions and not to data points, we need to associate the incoming data point with a distribution. We construct the incoming distribution as $g \sim \mathbf{N}_3(\boldsymbol{\mu}_g, \sigma_g^2 I)$. We assume that

$$\boldsymbol{\mu}_g = \mathbf{x}_t, \qquad \text{and} \qquad \sigma_g^2 = 25$$

where $\mathbf{x}_t$ is the incoming data point. The choice of $\sigma_g^2 = 25$ is the result of experimental observation about the typical spread of successive pixel values in small time windows. The five divergence measures between g and f_i, $i = 1, \ldots, 5$ are computed by the following formula:

$$J(f_i, g) = \frac{3}{2}\left(\frac{\sigma_i}{\sigma_g} - \frac{\sigma_g}{\sigma_i}\right)^2 + \frac{1}{2}\left(\frac{1}{\sigma_i^2} + \frac{1}{\sigma_g^2}\right)(\boldsymbol{\mu}_g - \boldsymbol{\mu}_i)'(\boldsymbol{\mu}_g - \boldsymbol{\mu}_i).$$

Once the five divergence measures have been calculated, we find the distribution f_j ($1 \le j \le 5$) for which

$$J(f_j, g) = \min_{1 \le i \le 5} \{J(f_i, g)\}$$

and we have a match between f_j and g if and only if

$$J(f_j, g) \le K^*$$

where K^* is a prespecified cutoff value. In the case where $J(f_j, g) > K^*$, then the incoming distribution g cannot be matched to any of the existing distributions.

The key point here is that we measure dissimilarity against all the available distributions. Other approaches, like [13], measure dissimilarity against the existing distributions in a certain order. Depending on the satisfaction of a certain condition the process may stop before all five measurements are taken and compared. We will see in Section VI-B4 how this "preferential" treatment can weaken the performance of the segmenter under certain weather scenarios.

2) Model Update When a Match is Found: If the incoming distribution matches to one of the existing distributions, we pool them together to a new Normal distribution. This new Normal distribution is considered to represent the current state of the pixel. The state is labeled either background or foreground depending on the position of the matched distribution in the ordered list of distributions. The next issue needed to be clarified is how we update the parameters of the mixture. We use the *Method of Moments*. First, we introduce some learning parameter α, which weighs on the weights of the existing distributions. So we subtract $100\alpha\%$ weight from each of the five existing weights and

we assign it to the incoming distribution's weight. In other words, the incoming distribution has weight α since

$$\sum_{i=1}^{5} \alpha\pi_i = \alpha \sum_{i=1}^{5} \pi_i = \alpha$$

and the five existing distributions have weights: $\pi_i(1-\alpha)$, $i = 1, \ldots, 5$.

Obviously, for α we need to have $0 < \alpha < 1$. The choice of α depends mainly on the choice of K^*. The two quantities are inversely related. The smaller the value of K^*, the higher the value of α and vice versa. The values of K^* and α are also affected by how much noise we have in the monitoring area. So if, for example, we were monitoring an outside region and had a lot of noise due to environmental conditions (rain, snow, etc.), then we would need a "high" value of K^* and thus a "small" value of α, since non-match to one of the distributions is very likely to be caused by background noise. On the other hand, if we were recording indoors where the noise is almost non existent then we would prefer a "small" value of K^* and thus a "higher" value of α, because any time that we do not get a match to one of the existing five distributions, that is very likely to occur due to some foreground movement (since the background has almost no noise at all).

Let us assume that we have a match between the new distribution g and one of the existing distributions f_j where $\mathbf{1} \leq j \leq 5$. Then, we update the weights of the mixture model as follows:

$$\begin{aligned} \pi_{i,t} &= (1-\alpha)\pi_{i,t-1} \qquad i = 1, \ldots, 5 \quad \text{and} \quad i \neq j \\ \pi_{j,t} &= (1-\alpha)\pi_{j,t-1} + \alpha. \end{aligned}$$

We also update the mean vectors and the variances. If we call w_1 as: $(1-\alpha)\pi_{j,t-1}$, i.e., w_1 is the weight of the jth component (which is the winner in the match) before pooling it with the new distribution g and if we call $w_2 = \alpha$, i.e., the weight of the new observation, then define

$$\rho = \frac{w_2}{w_1 + w_2} = \frac{\alpha}{(1-\alpha)\pi_{j,t-1} + \alpha}.$$

Using the method of moments [26], we get

$$\begin{aligned} \boldsymbol{\mu}_{j,t} &= (1-\rho)\boldsymbol{\mu}_{j,t-1} + \rho\boldsymbol{\mu}_g \\ \sigma^2_{j,t} &= (1-\rho)\sigma^2_{j,t-1} + \rho\sigma^2_g \\ &\quad + \rho(1-\rho)(\mathbf{x}_t - \boldsymbol{\mu}_{j,t-1})'(\mathbf{x}_t - \boldsymbol{\mu}_{j,t-1}) \end{aligned}$$

while the other four (unmatched) distributions keep the **same** mean and variance that they had at time $t-1$.

3) Model Update When a Match is Not Found: In the case where a match is not found (i.e., $\min_{1 \leq i \leq 5} K(f_i, g) > K^*$), then we commit the current pixel state to be foreground and we replace the last distribution in the ordered list with a new one. The parameters of the new distribution are computed as follows.

1) The mean vector $\boldsymbol{\mu}_5$ is replaced with the incoming pixel value.
2) The variance σ^2_5 is replaced with the minimum variance from the list of distributions.
3) The weight of the new distribution is computed as follows:

$$w_{5,t+1} = \frac{1-T}{2}$$

where $\boldsymbol{T}$ is the background threshold index. This formula guarantees the classification of the current pixel state as foreground. The weights of the remaining four distributions are updated according to the following formula:

$$w_{i,t+1} = w_{i,t} + \frac{w_{5,t} - \frac{(1-T)}{2}}{4}.$$

4) Justification of the Modifications Introduced to Normal Mixture Modeling: We initially implemented the Normal Mixture Modeling reported in [13] . The performance of the moving object segmenter under that scheme was satisfactory in the experimental trials and we did not plan on modifying the approach in any way. During late spring and early summer of 2000, however, weather phenomena in Minneapolis revealed some weak points of the method. During this time of year, the weather in Minneapolis features broken clouds, due to increased evaporation from the lakes and brisk Canadian winds. Small clouds of various density pass rapidly across the camera's field of view in high frequency. This type of weather substantially affected the performance of the segmenter and either increased dramatically the false alarms or reduced the detection sensitivity depending on how we set the algorithmic parameters.

In [13], the distributions of the mixture model are always kept in a descending order according to w/σ, where w is the weight and σ the variance of each distribution. Then, incoming pixels are matched against the ordered distributions in turn from the top toward the bottom of the list. If the incoming pixel value is found to be within 2.5 standard deviations of a distribution, then a match is declared and the process stops. This method is vulnerable to the following scenario: An incoming pixel value is more likely to belong, for example, to distribution 4 but still satisfies the 2.5 standard deviation criterion for a distribution earlier in the queue (e.g., 2). Then, the process stops before it reaches the right distribution and a match is declared early (see Fig. 8). The match is followed with a model update that unjustly favors the wrong distribution. These cumulative errors can affect the performance of the system after a certain point. They can even have an immediate and serious effect if one distribution (e.g., 2) happens to be background and the other (e.g., 4) foreground.

The above scenario can be put into motion by fast moving clouds. In [13], when a new distribution is introduced into the system it is centered around the incoming pixel value and is given an initially high variance and small weight. As more evidence accumulates, the variance of the distribution drops and its weight increases. Consequently, the distribution advances in the ordered list of distributions. Because, however, the weather pattern is very active, the variance of the distribution remains relatively high since supporting ev-

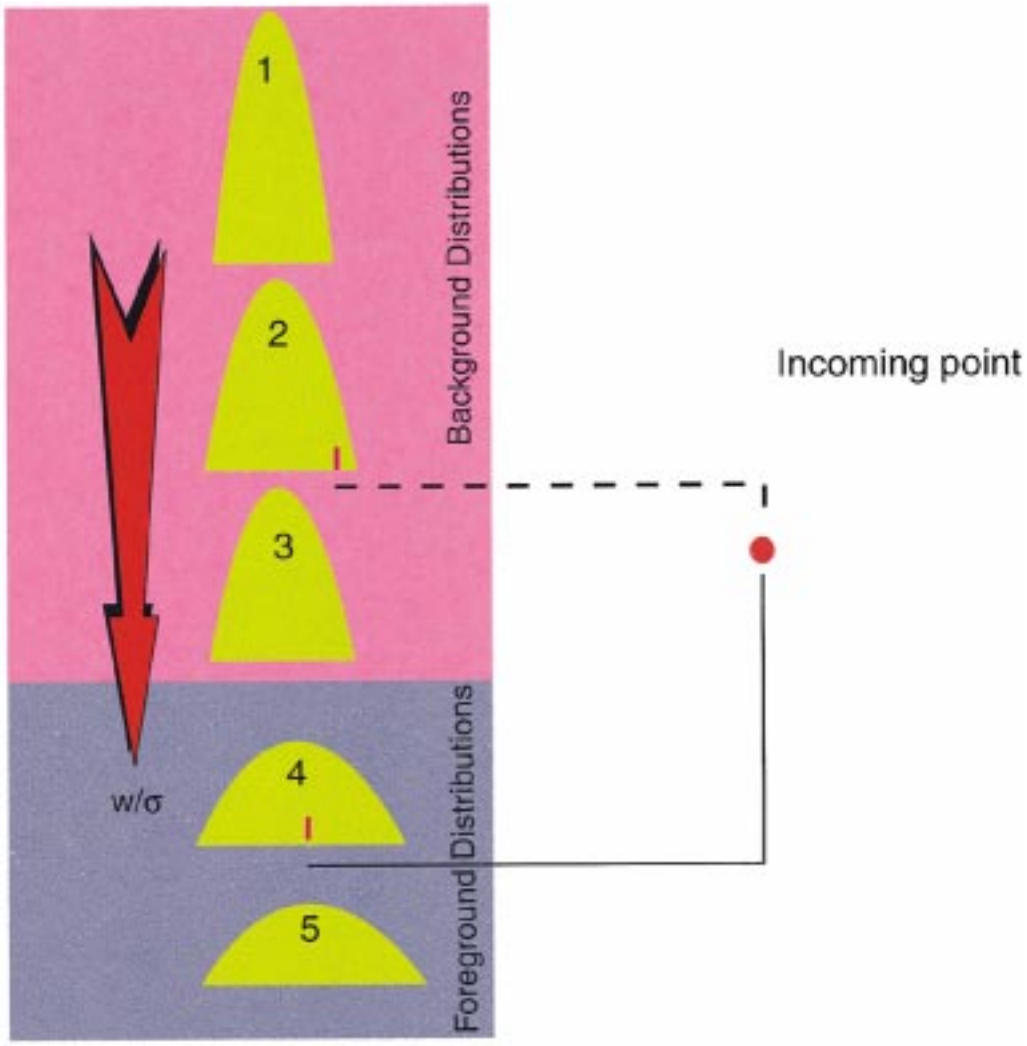

Fig. 8. Visualization of the failure mode of the method described in [13].

idence is switched on and off at high frequency. This results in a mixture model with distributions that are relatively spread out. If an object of a certain color happens to move in the scene during this time, it generates incoming pixel values that may marginally match distributions at the top of the queue and therefore interpreted as background. Since the moving clouds affect wide areas of the camera's field of view post-processing cannot save the day.

In contrast, our method does not try to match the incoming pixel value from the top to the bottom of the ordered distribution list. It rather creates a narrow distribution that represents the incoming data point. Then, it performs the match by finding the minimum divergence value between the incoming distribution and **all** the distributions of the mixture model (see Fig. 9). In this manner, the incoming data point has a much better chance of being matched to the right distribution than in [13].

C. Multiple Hypotheses Predictive Tracking

In the previous section we described a statistical procedure to perform on-line segmentation of *foreground pixels* corresponding to moving objects of interest, i.e., people and vehicles. In this section, we describe how to form trajectories traced by the various moving objects. Fig. 10 shows a snapshot of the output from the various computer vision modules of DETER. The basic requirement for forming object trajectories is the calculation of blob centroids (corresponding to moving objects). Blobs are formed after we apply a standard 8-connected component analysis algorithm to the foreground pixels. The connected component algorithm filters out blobs with area less than $A = 3{\times}9 = 27$ pixels as noise. According to our optical computation in Section V, this is the minimal pixel footprint of the smallest object of interest (human) in the camera's FOV.

A *Multiple Hypotheses Tracking (MHT)* algorithm is then employed that groups the blob centroids of foreground objects into distinct trajectories. MHT is considered to be the best approach to multitarget tracking applications. It is a recursive Bayesian probabilistic procedure that maximizes the probability of correctly associating input data with tracks. Its superiority against other tracking algorithms stems from the fact that it does not commit early to a trajectory. Early commitment usually leads to mistakes. MHT groups the input

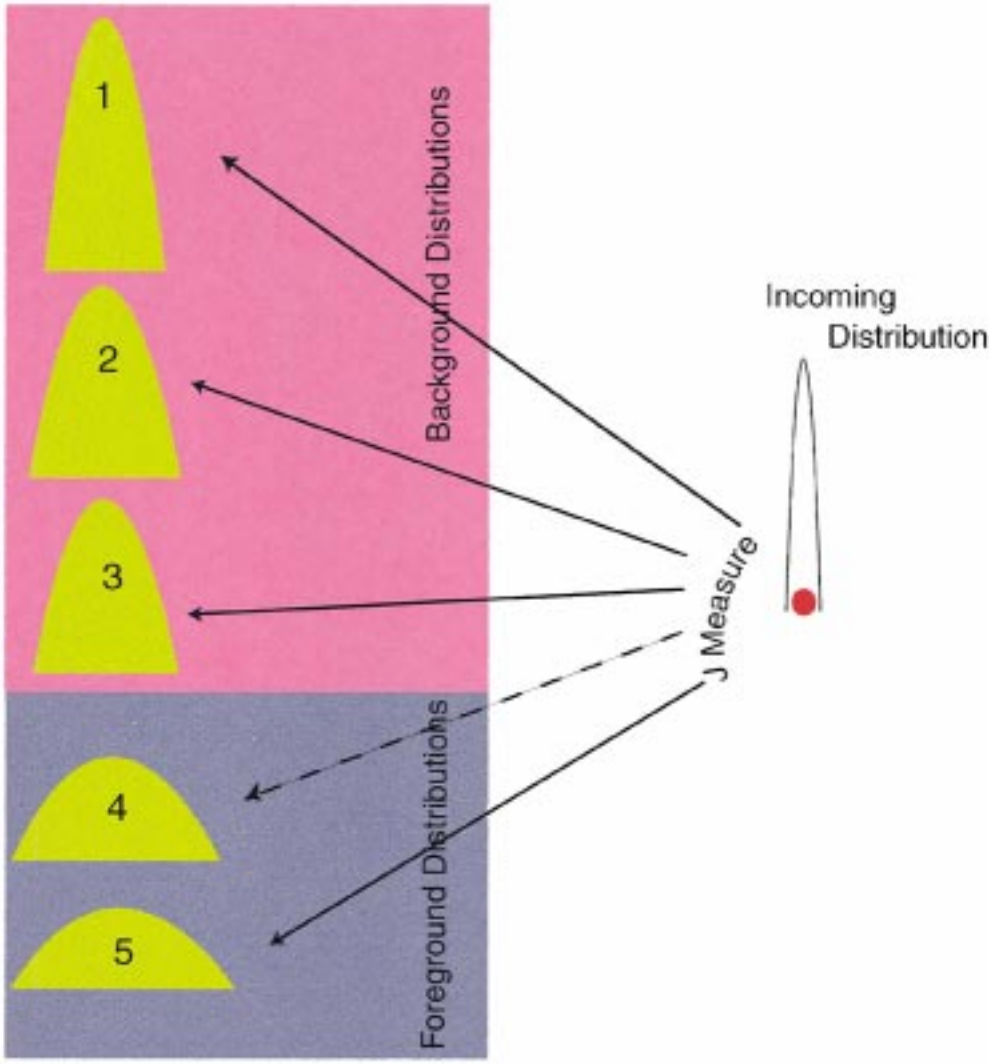

Fig. 9. Visual representation of the way our method matches incoming data points to existing distributions.

data into trajectories only after enough information has been collected and processed. In this context, it forms a number of candidate hypotheses regarding the association of input data with existing trajectories. MHT has shown to be the method of choice for applications with heavy *clutter* and dense *traffic*. In difficult multitarget tracking problems with crossed trajectories, MHT performs very well as opposed to other tracking procedures such as the *Nearest Neighbor (NN) correlation* and the *Joint Probabilistic Data Association (JPDA)* [27].

Fig. 11 depicts the architecture of our MHT algorithm. An integral part of any tracking system is the prediction module. Prediction provides estimates of moving objects' states and in the DETER system is implemented as a Kalman filter. Kalman filter predictions are made based on a priori models for target dynamics and measurement noise. Validation is a process which precedes the generation of hypotheses regarding associations between input data (blob centroids) and the current set of trajectories (tracks). Its function is to exclude, early on, associations that are unlikely to happen thus limiting the number of possible hypotheses to be generated.

Central to the implementation of the MHT algorithm is the generation and representation of track hypotheses. Tracks are generated based on the assumption that a new measurement may:

1) belong to an existing track;
2) be the start of a new track;
3) be a false alarm.

Assumptions are validated through the validation process before they incorporated into the hypothesis structure. The complete set of track hypotheses can be represented by a hypothesis matrix as shown in Table 1. The hypothetical situation in Table 1 corresponds to a set of two scans of 2 and 1 measurements made respectively on frame $k = 1$ and $k + 1 = 2$. Some notation clarification is in order. A measurement $z_j(k)$ is the jth observation (blob centroid) made on frame k. In addition, a false alarm is denoted by 0 while the formation of a new track (T_{newID}) generated from an old track (T_{oldID}) is shown as $T_{\text{newID}}(T_{\text{oldID}})$. The first column in this table is the Hypothesis index. In our example case we have a total of four hypotheses generated during scan 1, and eight more are generated during scan 2. The last column lists the tracks that the

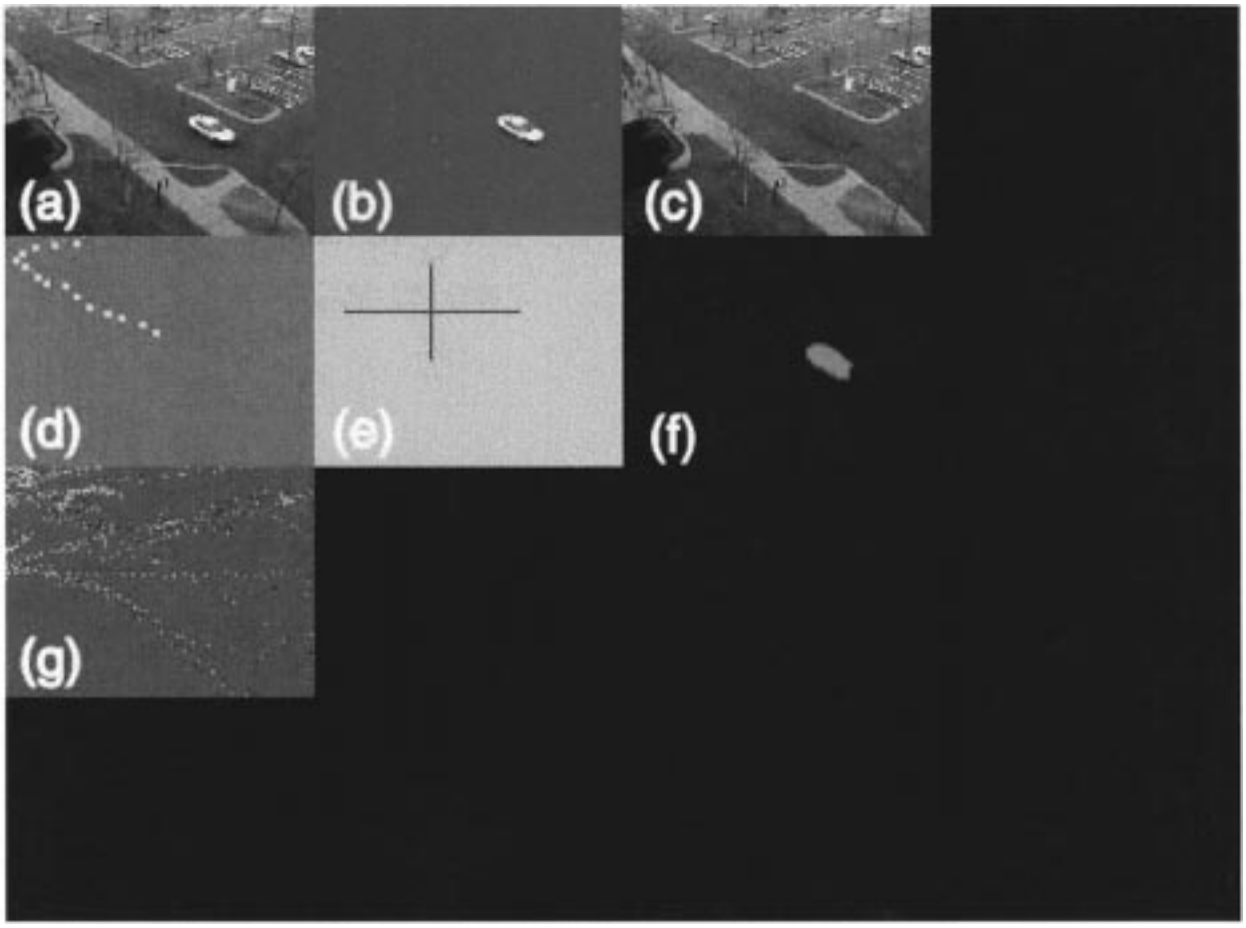

Fig. 10. Visualization of the computer vision operation of DETER. The snapshot was taken "live" on March 3, 2000. (a) Live video feed. (b) Segmented moving object. (c) Dynamically updated backgroud. (d) Trajectories of the current moving objects. (e) Centroids of the moving objects. (f) Results of the blob analysis. (g) Cumulative trajectory visualization of human and vehicle traffic for the past hour.

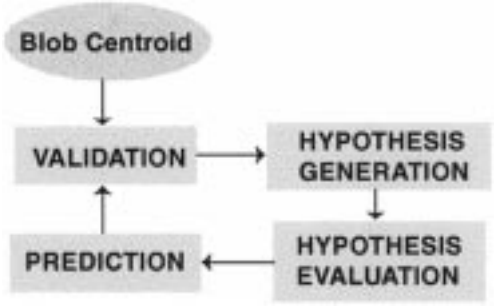

Fig. 11. Architecture of the MHT algorithm.

particular hypothesis contains (e.g., hypothesis H_8 contains tracks 1 and 4). The row cells in the hypothesis table denote the tracks to which the particular measurement $z_j(k)$ belongs (e.g., under hypothesis H_{10} the measurement $z_1(2)$ belongs to track 5). A hypothesis matrix is represented computationally by a tree structure as it is schematically shown in Fig. 12. The branches of the tree are in essence the hypotheses about measurements-track associations.

As it is evident from the above example, the hypothesis tree can grow exponentially with the number of measurements. We apply two measures to reduce the number of

Table 1
Complete Set of Track Hypotheses with the Associated Sets of Tracks

Hypothesis	$z_1(1)$	$z_2(1)$	$z_1(2)$	*Track No.*
H_1	0	0	-	0
H_2	1	0	-	1
H_3	0	2	-	2
H_4	1	2	-	1,2
H_5	1	0	3(1)	3
H_6	1	2	3(1)	2,3
H_7	0	2	4(2)	4
H_8	1	2	4(2)	1,4
H_9	0	0	5	5
H_{10}	1	0	5	1,5
H_{11}	0	2	5	2,5
H_{12}	1	2	5	1,2,5

hypotheses. Our first measure is to cluster the hypotheses into disjoint sets [28]. In this sense, tracks that do not compete for the same measurements compose disjoint sets which in turn are associated with disjoint hypothesis trees. Our second measure is to assign probabilities on every branch of hypothesis trees. The set of branches with the N_{hypo} highest probabilities are only considered. The derivation of hypothesis probabilities is out of the scope of this paper. However, the

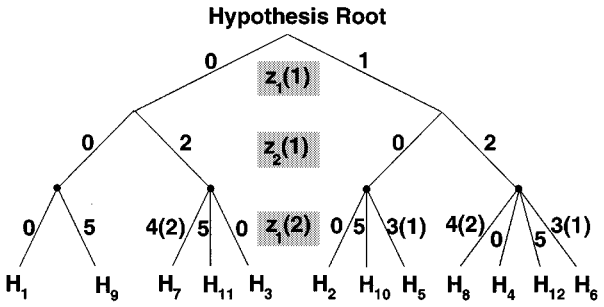

Fig. 12. Formation of a hypothesis tree.

interested reader is referred to [28] and [29]. It only suffices to say that a recursive Bayesian methodology is followed for calculating hypothesis probabilities from frame to frame.

VII. Multicamera Fusion

Monitoring of large sites (such as parking lots) can be accomplished only through the coordinated use of multiple cameras. In DETER, we need to have seamless tracking of humans and vehicles across the whole geographical area covered by all cameras. We produce a panoramic view of the HL parking lot by fusing the individual camera FOVs. Then, object motion is registered against a global coordinate system. We achieve multicamera registration (fusion) by computing the *Homography* transformation between pairs of cameras (see Fig. 13). Our homography computation procedure takes advantage of the overlapping that exists between pairs of camera FOVs. We use the pixel coordinates of more than four points to calculate the homography transformation matrix. These points are projections of physical ground plane points that fall in the overlapping area between the two camera FOVs. We select and physically mark these points on the ground with paint during the installation phase. We then sample the corresponding projected image points through the DETER graphical user interface (GUI). This is a process that happens only in the beginning and once the camera cross-registration is complete is never repeated.

A. Homography Computation

The homography computation is challenging primarily for two reasons.

- It is an underconstrained problem that is usually based on a small number of matching points.
- It introduces inaccuracies in specialized transformations (e.g., pure rotation or translation).

A very popular and relatively simple method for the computation of the homography matrices is the so-called *least squares* method [16]. This method may provide a poor solution to the underconstrained system of equations due to biased estimation. It also cannot effectively specialize the general homography computation when special cases are at hand.

We have adopted the algorithm by Kanatani [17] to compute the homography matrices. The algorithm is based on a statistical optimization theory for geometric computer vision [18] and cures the deficiencies exhibited by the least squares method. The basic premise is that the *epipolar constraint* may be violated by various noise sources due to the statistical nature of the imaging problem (see Fig. 14).

VIII. Threat Assessment

Automation is clearly necessary to allow limited and fallible human attention to monitor a large protected space. The primary objective of DETER is to alert security personnel to just those activities that require their scrutiny, while ignoring innocuous use. DETER achieves its objective by processing the computer vision information through its threat assessment module. All of the threat assessment analysis is done after converting the pixel coordinates of the object tracks into a world coordinate system set by the CAD drawing of the facility. Thus, we can use well-known landmarks to provide content for evaluating intent. Such landmarks include individual parking spots, lot perimeter, power poles, and tree lines. The coordinate transformation is achieved through the use of the optical computation package *CODE V*.

The feature assembly uses the trajectory information provided by the computer vision module to compute relevant higher level features on a per-vehicle/pedestrian basis. The features are designed to capture "common sense" beliefs about innocuous, law abiding trajectories, and the known or supposed patterns of intruders. In the current prototype, the features calculated include the following:

- number of sample points;
- starting position (x, y);
- ending position (x, y);
- path length;
- distance covered (straight line);
- distance ratio (path length/distance covered);
- start time (local wall clock);
- end time (local wall clock);
- duration;
- average speed;
- maximum speed;
- speed ratio (average/maximum);
- total turn angles (radians);
- average turn angles;
- number of "M" crossings.

Most of these are self explanatory, but a few are not so obvious. The wall clock is relevant since activities on some paths are automatically suspect at certain times of day—particularly late night and early morning.

The turn angles and distance ratio features capture aspects of how circuitous was the path followed. The legitimate users of the facility tend to follow the most direct paths permitted by the lanes. "Browsers" may take a more serpentine course.

The "M" crossings feature attempts to monitor a well-known tendency of car thieves to systematically check multiple parking stalls along a lane, looping repeatedly back to the car doors for a good look or lock check (two loops yielding a letter "M" profile). This can be monitored by keeping reference lines for the parking stalls and counting the number of traversals into stalls. An "M" type pedestrian crossing captured by DETER is illustrated in Fig. 15.

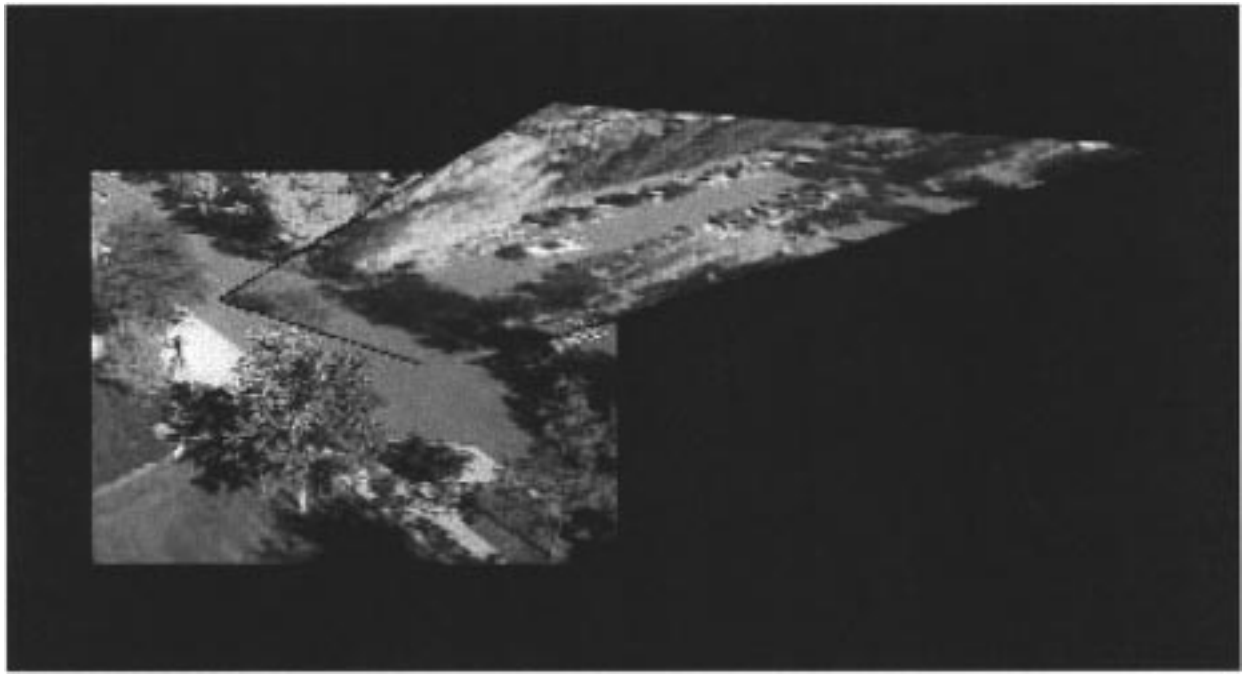

Fig. 13. Fused view from two DETER cameras. Because we compute a near optimal camera configuration scheme (coverage versus cost), the cameras are far apart and their optical axes form angles that vary wildly. As a result, one can notice the substantial image skewing produced by the highly nonlinear homography transformation. Despite the nonlinearity we achieve smooth image display thanks to a proprietary Honeywell warping algorithm.

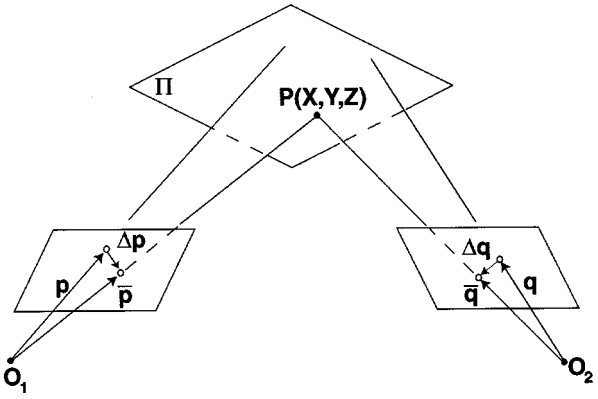

Fig. 14. The statistical nature of the imaging problem affects the epipolar constraint. O_1 and O_2 are the optical centers of the corresponding cameras. P(X, Y, Z) is a point in the scene that falls in the common area between the two camera FOVs. Ideally, the vectors $\overrightarrow{O_1p},\ \overrightarrow{O_2q},\ \overrightarrow{O_1O_2}$ are coplanar. Due to the noisy imaging process, however, the actual vectors $\overrightarrow{O_1p},\ \overrightarrow{O_2q},\ \overrightarrow{O_1O_2}$ may not be coplanar.

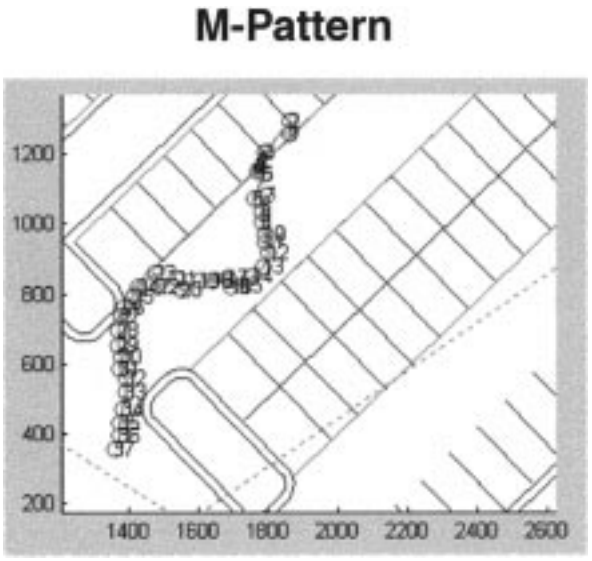

Fig. 15. An M-pattern traced by DETER. The centroids constituting the track are superimposed on the parking lot's CAD drawing. The M-pattern is a stroll mode favored by potential car thieves and it was one of the events staged during the benchmark recording.

The output of the feature assembly module for trajectories recorded from the site over some period of time is fed into the off-line training module. The goal of off-line training is to produce threat models based on a database of features. In the current system, we have gathered data by running DETER over a period of several hours. During this period, we staged several suspicious events (like "M" type strolls) to enrich our data collection. We then manually labeled the individual object trajectories as either innocuous (*OK*) or suspicious (*THREAT*). In the future, a clustering algorithm (see Fig. 2) will assist in the production of more parsimonious descriptions of object behavior. The complete training data consist of the labeled trajectories and the corresponding feature vectors. They are all processed together by a classification tree induction algorithm based on CART [30]. The trained classifier is then used on-line to classify incoming live data as either innocuous or suspicious.

IX. Experimental Results

At the time of the writing, DETER has been operating for almost a year. During this time there have been incremental improvements at the algorithmic and software level. We use the experience of the building's guards as the primary feedback mechanism. This feedback is primarily qualitative but is very important since this is the way products are evaluated in the security market place. The fundamental criteria they use in their evaluation are as follows.

- Is the system trustworthy? In other words, does it produce a lot of false alarms or does it miss important events?
- How does it compare with the legacy system?
- Does it add value to their function?
- Is it easy to learn and operate?

In the matter of trustworthiness, the guards were the first to pinpoint the faulty behavior of the system when the weather featured broken clouds and brisk winds. This prompted the investigation by the R&D team that led to the modification of the moving object segmenter. After the modified computer vision subsystem was put into use in August 2000, the problem was fixed and no other major complaints came into being.

The guards were also very excited with some functions of DETER that did not relate directly to automated surveillance. An example was the fusion of the multiple camera field of views into a super picture and its projection on a big flat panel display. This gives the guards a comprehensive view of the entire perimeter of the building and does not fragment their attention. This attitude is a testament to the anthropocentric character of the security market.

The only persistent complaint that still stands regards the user interface portion of DETER. Ultimately, the guards would like to add functions, like the detection of overspeeding, by clicking and pointing away. Right now they need the help of a member of the R&D team whenever they want to set a new function for the threat assessment module.

In addition to the qualitative testing performed by the actual users, we also performed quantitative testing for benchmarking purposes. Since August 11, 2000, we measured the tracking performance of DETER in the HL parking lot for 8 h. The testing was done in 1 h increments spread over different days, times of day, and seasons. Meticulous ground-truthing was performed by two R&D engineers and their results were compared and reconciled for accuracy. We selected this data set to fulfill certain requirements.

1) Sizeable duration (several hours).
2) Scenarios with significant traffic and others predominantly inactive. Typical busy times that were captured were in mid-afternoon during a workday when people leaving for their homes. Typical inactive times were late night hours.
3) Inclusion of some unusual events. We have induced these events ourselves in the absence of criminal activity.
4) Challenging weather conditions. We have included a partly cloudy day with strong winds (1 h). We have also included a snowy day (1 h) and a rainy day (1 h).

Table 2
Experimental Results for the 8-Hour-Long Data Set

Perfect Tracks	Split Tracks	Joint Tracks	False Alarms	Missed Tracks
554	77	16	5	3

Table 2 shows the results of the DETER performance in the field tests . The ground truth was done by indexing back the actual events on the video clip to the annotated output of DETER on the CAD design of our lot (see Fig. 16). Parking lot activity included walking and running of a single individual, simultaneous walking of a number of individuals (following crossing or parallel paths), driving of a single and multiple cars, and finally a combination of cars and humans in motion. As we explained earlier, staged events included geometrically interesting walking patterns such as the ones we call *M-Patterns* (see Fig. 15) and dangerous driving. These events were identified as suspicious by the Threat Assessment classifier.

DETER detected and tracked perfectly 554 objects out of 666. In 77 instances, DETER has lost momentarily track of the object but regained it very quickly. The result was a split track. That was typically the case with pedestrians as they ventured momentarily under the tree lines (summer and early fall trials). Tracking was correctly resumed once the pedestrians were again out of the tree line and in clear view. We do not consider the split tracks of pedestrians as a sign of algorithmic weakness. DETER employs a relatively small number of cameras because it is a cost-sensitive application. Therefore, during summer time when the trees are fully bloomed, coverage under the tree lines is not perfect. The problem can be solved by employing additional cameras if split tracks prove to be a serious security loophole (cost versus risk analysis). Alternatively, DETER can maintain the same number of cameras and recognize objects that appear and disappear from the FOV within short time intervals. To perform this recognition function, DETER needs cameras with higher resolution to capture detailed features of cars and especially humans. A solution would be to have the DETER cameras equipped with automated zoom mechanisms. Then they will be able to zoom in momentarily on every detected object and capture a detailed object signature. This capability will increase exponentially the algorithmic and software complexity of DETER.

Another type of event that was prone to split tracks was the unparking of vehicles in the parking lot. As the vehicles back up to get out of the parking stall, they stop temporarily before they start moving forward. This results in the loss of track association. This is a predictive tracking problem and not an object segmentation problem. For all practical purposes, it does not have any substantial effect on the intended use of the system and we have decided to ignore it.

In a few occasions (16) where pedestrians were moving next to each other (party of two), DETER correctly detected

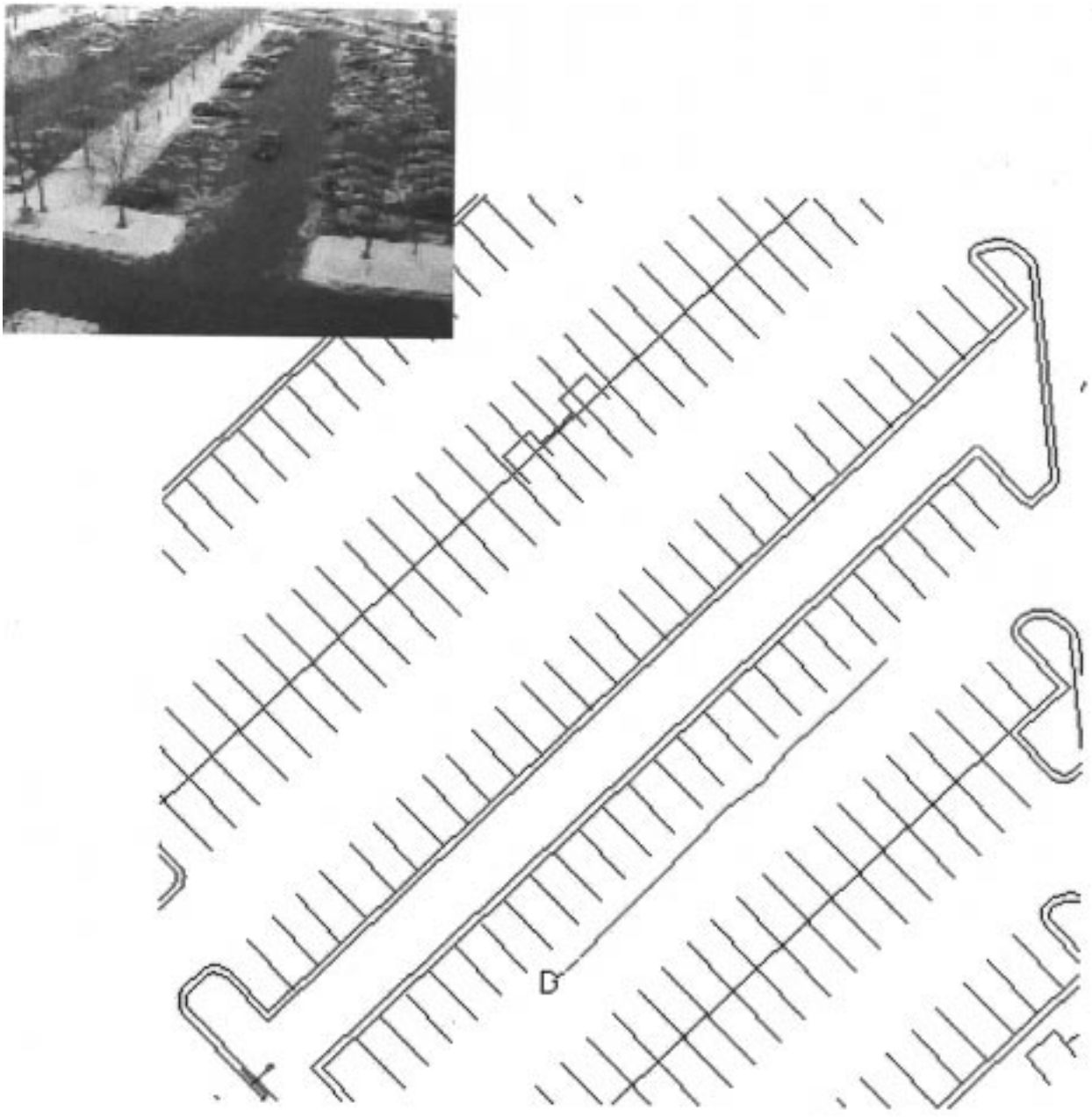

Fig. 16. Live video snapshot of a car moving out of the parking lot and its itinerary (line marked by the letter D) as it is recorded by DETER at the CAD design level.

and tracked the motion but as a single object. This is a camera resolution problem. If we covered less area with each camera the resolution would have been better and the segmentation of closely spaced moving objects more accurate. This loss of information would have been important only if we were interested in monitoring human interaction.

DETER produced a small number of false alarms. Four of the five false alarms were produced in a snowy day as accumulated iced snow was hovering from the top cover of one of the cameras.

Finally, DETER missed altogether three objects—all pedestrians. The puzzling thing is that all three cases were recorded on a clear day and the objects were in clear view of the cameras. The issue is under study. Although the number of missed objects is small, it is clearly a concern since it relates to DETER's most important requirement—to function as a sophisticated motion detector.

In general, the computer vision part of DETER and particularly the moving object segmenter performed very well for the purposes of its intended use.

We have also set up a laboratory experiment to quantify the performance of our latest moving object segmenter with regard to the old moving segmenter modeled after [13]. The experiment was geared to gauge the performance of the two systems under frequent global illumination changes. The experiment took place in our lab where we had a model train that was running up and down a fixed track. During the experiment, we were switching on and off some of the over-

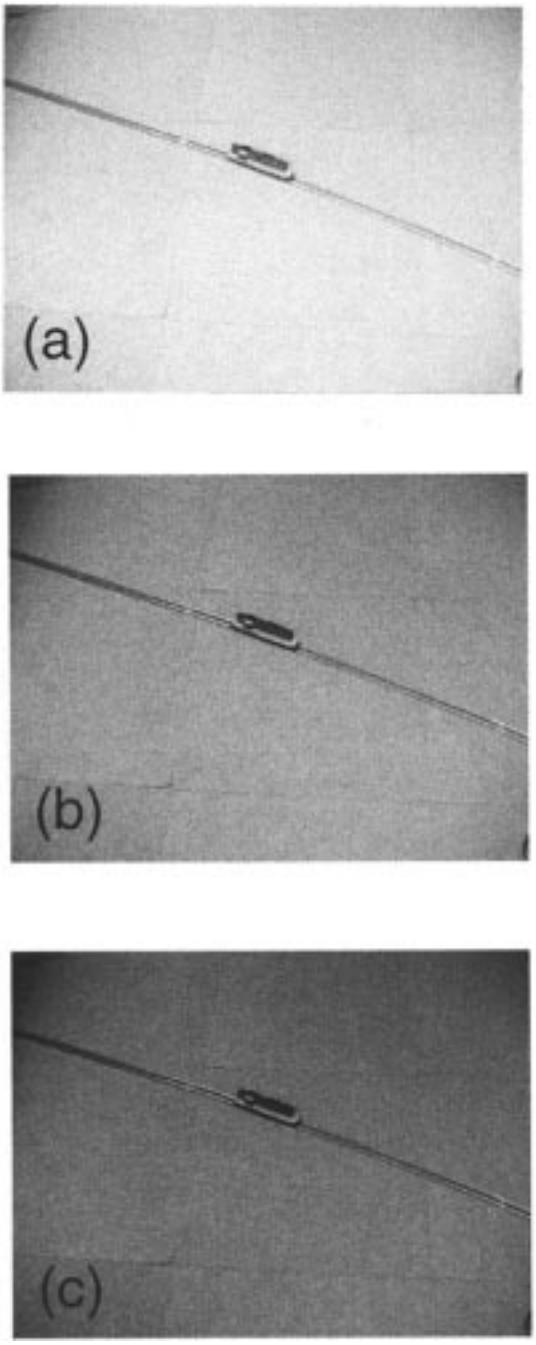

Fig. 17. Three different snapshots from the lab experimental setup. The scene appears in three different lighting conditions. One can notice the proximity in tones of the train and the floor background.

head lights randomly from time to time to emulate the effect of passing clouds (see Fig. 17). The experiment run for 15 min. During this period, the model train made 30 passes through the camera's field of view and, thus, a perfect detection and tracking performance would have produced 30 tracks. Table 3 shows the results of the experiment.

The older system modeled after [13] produced a rather high number of instances of split and missed tracks verifying the field test indications and our theoretical analysis (see Section VI-B4). This behavior can be rectified if one lowers substantially the background threshold B that defines how many

Table 3
Experimental Results for the Comparative Experiment in the Laboratory

	Perfect Tracks	Split Tracks	False Alarms	Missed Tracks
Old Segmenter	17	4	1	9
New Segmenter	26	4	3	0

of the distributions can be considered background at each point. Of course, the system then performs at a high false alarm rate, which is worse because it affects performance during normal weather conditions. Our modified system exhibited substantially better detecting power at only a slightly higher false alarm rate.

X. Conclusion and Future Work

We have presented DETER, a prototype urban surveillance system for monitoring large open spaces. We have provided the context of the current state of the security market and how it affected the design of DETER. DETER reliably tracks humans and vehicles both day and night. It consists of a computer vision module and a threat assessment module. The two primary components of the computer vision module is the moving object segmenter and the associated tracker. We have adopted the general approach described in [13] . We have introduced, however, some modifications that improve the performance of the system when there is high frequency of global illumination changes. Based on the object segmentation results, tracks are formed using a MHT algorithm and external multicamera calibration is achieved through the computation of homographies. The calibrated scene is mapped into the CAD design of the area under surveillance to facilitate higher level reasoning. The threat assessment module reports suspicious patterns detected in the annotated trajectory data at the CAD level. The threat assessor also uses the information produced by the computer vision module to perform some nonsecurity functions, like monitoring the capacity of the parking lot.

DETER is the result of compromise among lofty research and development ideals and the business and market realities. It is characteristic that the information produced by the computer vision module is used only for a small number of relatively simple functions (e.g., motion detection, recognition of a few specific motion patterns, and detection of overspeeding). The current experimental users of the prototype find these features nearly overwhelming. Our ongoing work focuses on the development of a more sophisticated user interface that will allow naive users of the system to introduce new behaviors at the CAD level by pointing and clicking away. Additionally, we are working toward the improvement of the threat assessment module with the inclusion of a clustering algorithm. The clustering algorithm will help in the partial automation of the off-line training, currently performed manually.

DETER is scheduled for productization in 2002, after the above mentioned improvements get incorporated into the prototype. It is characteristic of the global nature of

the security industry that the software maintenance of the product (or service) has been assigned to the Honeywell division in Bangalor India to keep the price competitive and the marketing to the Honeywell Australian security division.

ACKNOWLEDGMENT

We would like to thank a number of individuals for contributing to the success of this project, including K. Haigh, M. Bazakos, J. Droesller, R. Van Riper, P. Reutiman, and T. Faltesek.

REFERENCES

[1] J. A. Ratches, "Aided and automatic target recognition based upon sensory inputs from image forming systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, pp. 1004–1019, Sept. 1997.

[2] Vsam home page [Online]. Available: www.cs.cmu.edu/vsam/vsamhome.html

[3] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsim, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson, "A system for video surveillance and monitoring: Vsam final report," Robotics Institute, Carnegie Mellon Univ., Pittsburgh , PA , Tech. Rep. CMU-RI-TR-00–12, 2000.

[4] E. Stringa and C. S. Regazzoni, "Real-time video-shot detection for scene surveillance applications," *IEEE Trans. Image Processing*, vol. 9, pp. 69–79, Jan. 2000.

[5] C. Sacchi and C. S. Regazzoni, "A distributed surveillance system for detection of abandoned objects in unmanned railway environments," *IEEE Trans. Veh. Technol.*, vol. 49, pp. 2013–2026, Sept. 2000.

[6] X. Gao, T. E. Boult, F. Coetzee, and V. Ramesh, "Error analysis of background adaptation," in *Proc. 2000 IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, Hilton Head Island, SC, June 2000, pp. 503–510.

[7] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. 2000 IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, Hilton Head Island, SC, June 2000, pp. 142–149.

[8] D. Ormoneit, H. Sidenbladh, M. J. Black, T. Hastie, and D. J. Fleet, "Learning and tracking human motion using functional analysis," in *Proc. 2000 IEEE Workshop Human Modeling, Analysis and Synthesis*, Hilton Head Island, SC, June 2000, pp. 2–9.

[9] N. M. Oliver, B. Rosario, and A. P. Pentland, "A bayesian computer vision system for modeling human interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, pp. 1004–1019, Sept. 1997.

[10] C. H. Anderson, P. J. Burt, and G. S. V. D. Wal, "Change detection and tracking using pyramid transform techniques," in *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 579, Cambridge, MA, Sept. 16–20, 1985, pp. 72–78.

[11] I. Haritaoglu, D. Harwood, and L. S. Davis, "W/sup 4/s: A real-time system for detecting and tracking people in 2 1/2d," in *Proc. 5th Eur. Conf. Computer Vision*, vol. 1, Freiburg, Germany, June 2–6, 1998, pp. 877–892.

[12] T. Kanade, R. T. Collins, A. J. Lipton, P. Burt, and L. Wixson, "Advances in cooperative multi-sensor video surveillance," in *Proc. DARPA Image Understanding Workshop*, Monterey, CA, Nov. 1998, pp. 3–24.

[13] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. 1999 IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, Fort Collins , CO, June 23–25, 1999, pp. 246–252.

[14] ——, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 747–767, Aug. 2000.

[15] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *Proceedings IEEE FRAME-RATE Workshop*, Corfu, Greece , Sept. 2000, www.eecs.lehigh.edu/FRAME.

[16] L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: Establishing a common coordinate frame," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, pp. 758–767, Aug. 2000.

[17] K. Kanatani, "Optimal homography computation with a reliability measure," in *Proc. IAPR Workshop Machine Vision Applications*, Makuhari, Chiba , Japan, Nov. 1998, pp. 426–429.

[18] ——, *Statistical Optimization for Geometric Computer Vision: Theory and Practice*. Amsterdam, The Netherlands: Elsevier, 1996.

[19] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[20] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," in *Proc. 1998 IEEE Conf. Computer Vision and Pattern Recognition*, Santa Barbara, CA, June 23–25, 1998, pp. 22–29.

[21] B. K. P. Horn, *Robot Vision*. Cambridge, MA: MIT Press, 1986, pp. 66–69.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm (with discussion)," *J. Roy. Stat. Soc. B*, vol. 39, pp. 1–38, 1977.

[23] P. Tsiamyrtzis, "A Bayesian approach to quality control problems," Ph.D. dissertation, School of Statistics, Minneapolis, MN, 2000.

[24] H. Jeffreys, *Theory of Probability*. London, U.K.: Oxford Univ. Press, 1948.

[25] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. Inform. Theory*, vol. 37, pp. 145–151, Jan. 1991.

[26] G. J. McLachlan and K. E. Basford, *Mixture Models Inference and Applications to Clustering*. New York : Marcel Dekker , 1988.

[27] S. S. Blackman, *Multiple-Target Tracking with Radar Applications*. Norwood, MA: Artech House , 1986.

[28] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automat. Contr.*, vol. 24, pp. 843–854, 1979.

[29] I. J. Cox and S. L. Hingorani, "An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, pp. 138–150, Feb. 1996.

[30] W. Buntine, "Learning classification trees," *Stat. Comput.*, vol. 2, no. 2, pp. 63–73, 1992.

[31] "World security services to 2004," The Freedonia Group , Tech. Rep. 1348, 2000.

Dr. Ioannis Pavlidis (Senior Member, IEEE) received the B.S. degree in electrical engineering from the Democritus University, Greece, the M.S. degree in robotics from the Imperial College of the University of London, and the M.S. and Ph.D. degrees in computer science from the University of Minnesota.

He joined the Honeywell Laboratories, Minneapolis, MN, immediately upon his graduation in January 1997. His expertise is in the areas of computer vision beyond the visible spectrum and pattern recognition of highly variable patterns. He published extensively in these areas in major journals and refereed conference proceedings over the past several years. He is the co-chair of the IEEE series of Workshops in Computer Vision Beyond the Visible Spectrum and serves as a Program Committee member in several other major conferences.

Dr. Pavlidis is a Fulbright Fellow and a Member of ACM.

Vassilios Morellas (Member, IEEE) received the B.S. degree in mechanical engineering from the National Technical University of Athens, Greece, the M.S. degree in mechanical engineering from Columbia University, and the Ph.D. degree in mechanical engineering from the University of Minnesota.

He has been with the Honeywell Laboratories, Minneapolis, MN, since 1998. His expertise is in the areas of computer vision, sensor integration, and learning theories as they apply to enhancing robot autonomy and advancing machine intelligence. Prior to his current position, he pioneered the SAFETRUCK research project while working at the University of Minnesota as a Research Associate. SAFETRUCK successfully demonstrated the use of differential GPS (global positioning system) and radar sensing technologies to enhance safety of semi-tractor-trailers by developing lane departure detection and collision avoidance systems. SAFETRUCK won the second prize in the 1997 ITS World GPS Showcase competition.

Panagiot Tsiamyrtzis received the B.S. degree in mathematics from the Aristotle University, Greece, and the Ph.D. degree in statistics from the University of Minnesota.

He served as a faculty member in the Department of Statistics of the University of Minnesota in Fall 2000. He is currently with the Greek Army. His expertise is in the area of quality control.

Dr. Tsiamyrtzis was the recipient of the best student paper award in 2000 from the American Statistical Association.

Steve Harp received the Ph.D. degree in psychology (program in perception) from Northwestern University in 1986, where his research was on the perception of visual motion and camouflage, and the M.S. degree in statistics from the University of Minnesota in 1994.

He has been with the Honeywell Laboratories, Minneapolis, MN, since 1985, when he was first employed as an intern. Since then, he has worked on a wide range of projects involving artificial intelligence, statistical analysis, communications networks, and user interfaces. He has delivered numerous public talks and papers on these topics.

Dr. Harp is the recipient of two technical achievement awards and the Honeywell Sweatt award. He is a Member of the American Statistical Association and the American Association of Artificial Intelligence.