# Towards Detection of Suspicious Behavior from Multiple Observations

**Boštjan Kaluža**
Jozef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
bostjan.kaluza@ijs.si

**Gal Kaminka**
Bar Ilan University
Ramat Gan 52900, Israel
galk@cs.biu.ac.il

**Milind Tambe**
University of Southern California
3737 Watt Way, Los Angeles, CA 90089
tambe@usc.edu

## Abstract

This paper addresses the problem of detecting suspicious behavior from a collection of individuals events, where no single event is enough to decide whether his/her behavior is suspicious, but the combination of multiple events enables reasoning. We establish a Bayesian framework for evaluating multiple events and show that the current approaches lack modeling behavior history included in the estimation whether a trace of events is generated by a suspicious agent. We propose a heuristic for evaluating events according to the behavior of the agent in the past. The proposed approach, tested on an airport domain, outperforms the current approaches.

## Introduction

Identification of suspicious activities arises in many domains where an adversary has a motivating goal and exhibits behavior that deviates from behavior of normal agents. The goal is to augment traditional security measures by scrutinizing behavior of all subjects in the environment. This can be applied, for example, to detect a passenger at the airport that plans to smuggle drugs while keeping contacts with authorities at minimum, to detect a pirate vessel that plans to capture a transport vessel and therefore avoids security patrols, to identify a user that misuses access to the server, to catch a reckless driver, a shoplifter, etc. There are two approaches to detect suspicious behavior: suspicious detection models which depend on suspicious behavior definitions and anomaly detection models which measure deviations from defined normal behavior. The basic unit of such analysis is *behavior trace* that provides characterized agent's actions over a period of time. However, given increasingly longer behavior traces it becomes inefficient to encapsulate the entire spectrum of either suspicious or normal behavior.

An important step in such analysis is therefore to utilize domain knowledge to identify interesting parts characterizing behavior trace. We denote them as *trigger events*. Trigger events can be defined with the help of domain experts and present either positive or negative belief about the motivating goal. In many cases no single action or event is sufficient to reveal adversary intentions, but a collection of events enables the observer to infer the underlying intentions. The main question we are addressing is how to decide whether an event trace corresponds to behavior of normal or suspicious agent.

We establish a Bayesian framework for evaluating event traces and show that evaluation (i.e., whether a trace is produced by a suspicious agent) does take into account interactions between events. More precisely, evaluation of an event depends not only on the current time step but also on the events in prior time steps. We discuss approaches that probabilistically estimate prior history as well as approaches that relay on other frameworks such as plan recognition. The main problems that arise are over-simplification of the models and insufficient modeling of behavior history included in the estimation.

We propose a heuristic approach that defines a family of *well-behaved* scoring functions that interpret an event trace to produce a score presenting overall suspicion that the trace corresponds to the behavior of a suspicious agent. The key component is that the events are evaluated according to the behavior of the agent in the past. We present a set of scoring functions that satisfy that conditions.

Experimental evaluation on an airport domain first proposes an approach for detecting trigger events, which is based on interactive behavior among agents. Next, it compares discussed approaches for evaluating whether an event trace is generated by normal or suspicious agent against the heuristic approach. The experiments in a simulated multiagent environment show that the proposed approach outperforms other discussed solutions.

## Motivating Domain

A drastic increase in air travel in recent years has made every airport a potential terror target, hence intense security is a necessary requirement. This global transportation system is no longer considered as safe but rather as a potential liability exposed to terrorist attacks and other criminal activities such as drug and contraband trafficking. Airports require vast security solutions including identification of suspicious activities amongst passengers and staff in and surrounding areas.

To ensure the safety of people at the airport, several surveillance systems monitor inside the terminals and outer perimeters. Various systems were introduced to automatically detect some of the threats such as leaving objects behind (Hongeng and Nevatia 2003), suspicious trajectory paths (Vaswani, Chowdhury, and Chellappa 2005), suspicious transportations (Arsić, Schuller, and Rigoll 2007), thefts (Hongeng and Nevatia 2003) and even vandalism acts and fights (Naylor and Attwood 2003). There is also a commercially available system (Feris et al. 2009) able to detect events such as running passengers, climbing over fence, etc. However, these approaches mainly deal with detection of single incidents or monitor only a part of the premises. An approach for monitoring behavior of passengers over longer periods of time relies upon security personnel such as behavior detection officers (BDOs) that patrol airport to identify passengers who display *involuntary physical and physiological actions* (US Transportation Security Administration (www.tsa.gov) trained and deployed BDO officers at 161 US airports). In this context we strive to observe passengers for the whole time they spend at the airport. We are focused on trigger events in terms of actions, events and incidents that can be potentially suspicious in order to identify individuals who exhibit behaviors that indicate high levels of stress, fear or deception.

## Detection Objectives

We leverage Bayesian framework for intrusion detection (Helman, Liepins, and Richards 1992) for problem definition. *Event trace* $\mathbf{x}^{(k)}$ is a sequence of $k$ events $\mathbf{x}^{(k)} = (x_1, x_2, ..., x_k)$ from a set of traces $D$. At each time step $t$ an event $x_t$ is generated by a hidden stochastic process $H$ that is a mixture of two auxiliary stochastic processes, namely the normal process $N$ and the suspicious process $S$. In real-world there can be many subprocesses contributing to each of them, i.e., many normal users with different behavior patterns, however, here we assume only a single $N$ and a single $S$ that capture all variability. Random variable $y_t = 0$ if $x_t$ is generated by $N$ and $y_t = 1$ if $x_t$ is generated by $S$. The event $x_t$ may depend on the current step $t$ as well as on the pattern of events generated at time steps prior $t$. This allows that $N$ and $S$ are non-stationary, where their distribution depends both on actual time step $t$ and events previously generated by both process. The non-stationary nature might reflect that: (i) agent behavior depends on his/her prior actions; (ii) behavior changes over time (different population of agents); (iii) the nature of motivating goals changes over time; and (iv) the environment changes over time.

We assume a prior probability $\lambda = Pr\{S\} = Pr\{y = 1\}$. In most cases $\lambda$ is close to 0, since in real-world applications suspicious activities are sparse. The stochastic processes $N$ and $S$ induce measures $n(x_t) = Pr\{N(t) = x_t\}$ and $s(x_t) = Pr\{S(t) = x_t\}$, respectively. The mixture distribution of an event $x_t$ is

$$\begin{aligned} &Pr\{H(t) = x_t, H(t-1) = x_{t-1}, ..., H(1) = x_1\} = \\ &\qquad \lambda n(x_t, x_{t-1}, ..., x_1) + (1-\lambda)s(x_t, x_{t-1}, ..., x_1). \end{aligned} \tag{1}$$

The objective of suspicious behavior detection is to identify those traces $\mathbf{x}^{(k)} = (x_1, x_2, ..., x_k)$ that are likely to be suspicious activities, that is traces $\mathbf{x}$ for which

$$Pr\{S|H(t) = x_t, t = 1, ..., k\} > \tau, \tag{2}$$

is above some threshold $\tau$ or is large relative to the probability for other traces.

## Detectors

Several approaches have been proposed to tackle the problem of suspicious behavior detection and the literature is vast. Much of it is only superficially related, in the sense that the overall goals may be the same, but the application domains and the applied methods differ. For instance, detecting suspicious behavior from video surveillance cameras pursue the same goal, but the focus is on video analytics (Visontai 2004). Similarly, we will not address here related work on suspicious behavior detection from video features, e.g., (Arsić, Schuller, and Rigoll 2007; Bak et al. 2009; Barbará et al. 2008) or anomalous trajectory shapes (Nguyen et al. 2005; Piciarelli et al. 2008; Sillito and Fisher 2008; Tung 2010). We focus instead on the observable actions of agents that reveal their intentions. We thus limit ourselves to related research within recognition of multiple events giving a special focus to Hidden Markov Models (HMMs) (Rabiner 1989) and Utility-based Plan Recognition (UPR) (Avrahami-Zilberbrand and Kaminka 2007).

In this section we discuss approaches that decide whether a trace is generated by suspicious process. First, we present an optimal detector derived from Eq. (2) and show that solving it is infeasible. Next, we discuss approaches that directly attack the problem of estimating likelihood that the trace was generated by suspicious process in terms of the parameters in which the problem is formulated. They estimate conditional probabilities either by simplifying the assumptions or by modeling the conditional probabilities with another process. Finally, as an alternative, we discuss approaches that do not explicitly estimate the probability. Instead, they use heuristics, statistical measures, and plan recognition framework to provide an evaluation that the trace is generated by suspicious agent.

### Bayes-Optimal Detector

Using Bayes theorem we can derive from Eq. (2)

$$\begin{aligned} &Pr\{S|H(t) = x_t, t = 1, ..., k\} = \\ &\quad = \frac{\lambda \cdot Pr\{H(t) = x_t|S\}}{\lambda \cdot Pr\{H(t) = x_t|S\} + (1-\lambda) \cdot Pr\{H(t) = x_t|N\}} \\ &\quad = \frac{\lambda \cdot s(x_k, ..., x_1)}{\lambda \cdot s(x_k, ..., x_1) + (1-\lambda) \cdot n(x_k, ..., x_1)}. \end{aligned} \tag{3}$$

To this point, we implicitly assumed that distributions $\lambda$, $n$ and $s$ are reliably estimable. The degree to which this assumption is valid depends on our detection capability.

Suppose we have a dataset $D_l$ of labeled event traces. Suppose the $D_l$ is sufficiently large, we can estimate prior probability $\lambda$ from the $D_l$ using relative frequency presenting the number of traces generated by suspicious agent divided by the total number of traces[1]. Note, that in order to compute $Pr\{H(t) = x_t, t = 1, ..., k|S\}$ one has to evaluate

$$s(x_1) \cdot s(x_2|x_1) \cdot ... \cdot (x_k|x_{k-1}, ..., x_1) \tag{4}$$

[1] Since traces can be of different length, the quotient is normalized by traces length.

While some first terms, i.e., $s(x_t), s(x_t|x_{t-1})$ can still be estimated, the latter terms including increasingly more history become intractable. In real-world applications we have no direct knowledge of values of the conditional probabilities, that is, we are unable to specify probability of an event given all possible combinations of history (the same applies for $Pr\{H(t) = x_t, t = 1, ..., k|N\}$). For this reason, we must approximate Bayes optimality in general. In particular, we will be concerned with estimating $Pr\{S|H(t) = x_t, t = 1, ..., k\}$ using approximate approaches.

## Naive Bayes Detector

A naive approach assumes that (i) events are independent and (ii) processes $\hat{N}$ and $\hat{S}$ are stationary, which means that the current event depends only on the current time step $t$ and not on time steps prior $t$. Evaluation of the Eq. (3) is simplified using naive assumption:

$$Pr\{S|H(t) = x_t, t = 1, ..., n\} = \frac{\lambda \cdot \prod_{t=1}^{k} \hat{s}(x_t)}{\lambda \cdot \prod_{i=1}^{k} \hat{s}(x_t) + (1-\lambda) \cdot \prod_{i=1}^{k} \hat{n}(x_t)} \quad (5)$$

We have to evaluate probability $Pr\{H(t) = x_t|y_t\}$ that an event is generated by normal stationary process $\hat{n}(x_t)$ and suspicious stationary process $\hat{s}(x_t)$, which is tractable in terms of evaluation. Approaches for estimating $\hat{n}$ and $\hat{s}$ may include frequentist estimator, Hidden Markov Models, k-nearest neighbor, neural networks, etc. The paper does not explicitly address the problem of deciding whether an event is suspicious or not, however, we show an approach using Coupled HMM in the section with experiments.

In practice, the assumptions may over-simplify the model; however, we will use it as a baseline in our experiments.

## Hidden Markov Models

Estimation of conditional probabilities including history can be encoded with Hidden Markov Models (HMMs) (Rabiner 1989). HMM is a temporal probabilistic model with two embedded stochastic processes: an unobservable (hidden) process $Q$, which can be observed only through another (visible) stochastic process $O$. Each state in $Q$ has state transition probabilities (which are visible) and probability distribution over the possible values of $O$. The key assumption is that the current hidden state of the agent is affected only by its previous state. Now suppose we create an HMM to estimate $Pr\{H(t) = x_t, t = 1, ..., k|S\}$, more precisely, it models probability that a trace of events is generated by a suspicious agent. The hidden states of process $G$ may be referred to as internal states presenting intentions of the suspicious agent. For example, assume only two hidden states, normal intention and suspicious intention emitting normal and suspicious events, respectively. Transitions between the hidden states can be explained as probabilities that the agent will either follow or change its current intention. Although the information about the history is now partially encoded in the transition probabilities (i.e., given the agent's intention at time step $t$ is suspicious it is more likely that the intention at $t+1$ will be suspicious as well), the model still uses Markov assumption, that is, the next agent's intention depends only on it's current intention.

We construct two HMM models, normal model $\bar{N}$ and suspicious model $\bar{S}$. We split all labeled traces $\mathbf{x} \in D_l$ to traces generated by normal and suspicious agents, and use them to adjust the parameters of the models $\bar{N}$ and $\bar{S}$, respectively. Model parameters can be locally optimized using iterative procedure such as Baum-Welch method (Rabiner 1989). Given a new event trace $\mathbf{x}^{(k)} = (x_1, x_2, ..., x_k)$ we compute probability that the trace was generated by both models $Pr\{x_1, x_2, ..., x_k|\bar{N}\}$ and $Pr\{x_1, x_2, ..., x_k|\bar{S}\}$ using forward-backward procedure (Rabiner 1989). Given the prior probability $\bar{\lambda}$ we compute an estimate the trace $\mathbf{x}$ was generated by suspicious process $S$:

$$Pr\{S|H(t) = x_t\} = \frac{\bar{\lambda} \cdot Pr\{x_1, x_2, ..., x_k|\bar{S}\}}{\bar{\lambda} \cdot Pr\{x_1, x_2, ..., x_k|\bar{S}\} + (1-\bar{\lambda}) \cdot Pr\{x_1, x_2, ..., x_k|\bar{N}\}}. \quad (6)$$

Although widely used, HMMs may became inadequate when events have long-term temporal dependencies. Brand et al. (1997) introduced Coupled HMMs as an extension with multiple hidden interacting chains that are able to model interactive behavior. Moreover, Layered HMMs (Oliver, Garg, and Horvitz 2004) and Hierarchical HMMs (Fine, Singer, and Tishby 1998) can handle activities that have hierarchical structure, e.g., activity recognition from trajectories (Nguyen et al. 2005). Dueong et al. (2005) focused on duration of activities and introduced Switching Hidden Semi-Markov Models that provide probabilistic constraints over the duration of plans as well the ability to detect anomalies. Vaswani et al. (2005) introduced Continuous State HMMs for modeling trajectories in order to detect anomalous activities.

## Utility-based Plan Recognition

Another approach for deciding whether a trace is suspicious or not originates from plan recognition. Avrahami-Zilberbrand and Kaminka (2007; 2009) presented Utility-based Plan Recognition (UPR) that introduces utility to the observer. The main strength of UPR is that it can incorporate observer's bias to events with low likelihood, for example, a-priori probability for planting a bomb is very low, but detecting it has high expected utility. The recognition process utilizes a plan library, which encodes behaviors of the observed agents in a form of directed graph. Low-likelihood behaviors, which may be significantly costly to the observer, might be overlooked. Hence, the observer can select such a behaviors by assigning them high utility (or cost in risk-averse case). Behavior matching is performed with a symbolic plan recognizer (Avrahami-Zilberbrand 2009) that returns a set of behaviors (hypotheses) that the observed agent might have executed by the time of the last observation with corresponding posterior probabilities. In the next step, utilities are assigned to the transitions in the plan library and behaviors are then ranked according to their utility to the observer.

Inspired by the approach for catching a dangerous driver (Avrahami-Zilberbrand 2009), we propose a single plan-step encoded in the plan library as showed in Figure 1.

An agent can generate a suspicious event with probability $\hat{s}(x_t)$ and fixed cost $c_s < 0$ or a normal event with probability $\hat{n}(x_t)$ and fixed cost $c_n > 0$, followed by the end of the plan. All other costs are zero.

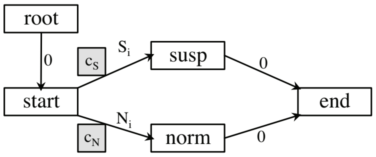

Figure 1: A single-step plan in UPR plan library.

The accumulated cost of the event trace over time can be simplified and computed as

$$\begin{aligned} U_{UPR}(\mathbf{x}^{(k)}) &= \sum_{t=1}^{k} u(x_t), \qquad (7) \\ u(x_t) &= \begin{cases} c_n\hat{n}(x_t); & \text{if } |c_n\hat{n}(x_t)| > |c_s\hat{s}(x_t)| \\ c_s\hat{s}(x_t); & \text{if } |c_n\hat{n}(x_t)| \leq |c_s\hat{s}(x_t)| \end{cases}. \end{aligned}$$

If the accumulated cost exceeds a given threshold, the trace is classified as generated by suspicious agent.

## Scoring Functions

All of the previous approaches (except HMMs) share the property that events are evaluated according to the probability of being generated by the suspicious process. However, neither accounts for information contained in the prior behavior of the agent. In this subsection, we define the scoring function, which is capable of estimating event in a trace according to the complete prior history.

The detection system can employ a *scoring function $f$* that interprets events to produce a score characterizing the overall suspicion that is to be contributed to the trace. Given a threshold value $\tau$ and a trace $\mathbf{x}^{(k)}$ we can classify as generated by a suspicious process if function value $f(\mathbf{x}^{(k)}) \geq \tau$.

**Definition 1.** *A scoring function $f$ over a trace of events $\mathbf{x}$ is a function*

$$f : D \to \mathbb{R}$$

The function $f$ assigns a real value to any trace $\mathbf{x}^{(k)}$ of length $k = 1, ..., K$.

**Definition 2.** *A class of* well-behaved *functions consists of scoring functions for any $\mathbf{x}^{(k)}$, $x_{k+1}$*

$$\begin{aligned} f(\mathbf{x}^{(k)}, x_{k+1}) \geq f(\mathbf{x}^{(k)}) \qquad & \textit{if } \Delta(x_{k+1}) = 1 \\ f(\mathbf{x}^{(k)}, x_{k+1}) \leq f(\mathbf{x}^{(k)}) \qquad & \textit{if } \Delta(x_{k+1}) = 0 \end{aligned}$$

The conditions imply that: (i) scoring function $f$'s evaluation increases when a new suspicious event is added to the trace and (ii) decreases when a normal event is added to the trace. The well-behaved scoring functions are motivated by the key observation that a suspicious event $x_{k+1}$ ($\Delta(x_{k+1}) = 1$) is more likely to be generated by a suspicious process $S$ than a normal process $N$ regardless of the history $\mathbf{x}^{(k)}$, i.e.,

$$\begin{aligned} s(x_{k+1}|\mathbf{x}^{(k)}) \geq n(x_{k+1}|\mathbf{x}^{(k)}) \qquad & \text{if } \Delta(x_{k+1}) = 1 \text{ and} \\ s(x_{k+1}|\mathbf{x}^{(k)}) \leq n(x_{k+1}|\mathbf{x}^{(k)}) \qquad & \text{if } \Delta(x_{k+1}) = 0. \end{aligned}$$

Given such assumptions the likelihood that a trace is emitted by a suspicious process as given by Eq. (3) is a well-behaved function.

The true likelihood function is difficult to obtain. Therefore, we defined the following well-behaved heuristic function to approximate it. Let $\mathbf{x}^{(k)} = (x_1, x_2, ..., x_k)$. We define the number of suspicious events as

$$\eta_s(\mathbf{x}^{(k)}) = \sum_{t=1}^{k} \Delta(x_t), \qquad (8)$$

where $\Delta(x_t)$ decides whether an event is suspicious or not

$$\Delta(x_t) = \begin{cases} 1; & \text{if } \tilde{s}(x_t) \geq \tilde{\tau} \\ 0; & \text{else} \end{cases}, \qquad (9)$$

$$\tilde{s}(x_t) = \frac{\lambda_\eta \cdot \hat{s}(x_t)}{\lambda_\eta \cdot \hat{s}(x_t) + (1 - \lambda_\eta) \cdot \hat{n}(x_t)}. \qquad (10)$$

$\lambda_\eta$ is prior probability for detecting suspicious event (if we have no prior knowledge, we can assume $\lambda_\eta = 0.5$), $\hat{s}$ and $\hat{n}$ can be the same procedures as discussed previously, and $\tilde{\tau}$ is a threshold value (if $\lambda_\eta = \tilde{\tau} = 0.5$ the condition in (9) can be simplified to $\hat{s}(x_t) > \hat{n}(x_t)$). Similarly, $\eta_n(\mathbf{x}^{(k)}) = k - \eta_s(\mathbf{x}^{(k)})$ presents the number of normal events. Suppose we observed a trace $\mathbf{x}^{(k)}$ of all suspicious events, i.e., $\tilde{s}(x_t) > \tilde{\tau}$ for $t = 1, ..., k$. Intuitively, terms in Eq. (4) suggest that probability that event $x_t$ was indeed generated by suspicious process should increase exponentially according to the number terms, i.e., to the number of events. On the other hand, if events in $\mathbf{x}$ were $\tilde{s}(x_t) < \tilde{\tau}, t = 1, ..., k$, the probability would exponentially decrease as the number of events increases. We define an exponential scoring function $f_e$ recursively as follows:

$$\begin{aligned} f_e(x_t, \mathbf{x}^{(t-1)}) &= a_t \cdot (f_e(\mathbf{x}^{(t-1)}) + b_t), \\ f_e(\mathbf{x}^{(0)}) &= 0, \\ b_t &= \beta \cdot \eta_s(\mathbf{x}^{(t)})^{\alpha(\tilde{s}(x_t) - \tilde{\tau})}, \qquad (11) \\ a_t &= e^{-(\delta + \eta_n^*(\mathbf{x}^{(t)}))/(\gamma \cdot \eta_s(\mathbf{x}^{(t)}))}. \end{aligned}$$

The $b_t$ term models the above mentioned observation with an exponential function using $\eta_s$ as the base and likelihood that the event was generated by suspicious agent $\tilde{s}$ as argument. Parameters $\alpha > 0$ and $\beta > 0$ can be estimated from $D_l$. Additionally, the $a_t$ term employs a *forgetting mechanism*, an exponential time decay function that discounts overall evaluation at time $t$ in respect to agent's behavior prior $t$. Parameters $\gamma > 0$ and $\delta \geq 0$ are also estimated from $D_l$. The modified $\eta_n^*$ presents *the time elapsed* since the last event $\tilde{s}(x_t) > \tilde{\tau}$, that is, the number of normal events since the last suspicious event; the higher the number of normal events the faster the forgetting rate. Finally, we use a threshold value to decide whether a trace is generated by suspicious agent or not $f_e(\mathbf{x}) > \tau_{f_e}$. The function $f_e$ is a well-behaved function by definition.

Consider the following example. Suppose we have a trace $\mathbf{x}^{(18)} = (x_1, ..., x_{18})$, where events $x_i > \tilde{\tau}$, for $t = \{1, 6, 12\}$ (most likely suspicious) and $x_t < \tilde{\tau}$ elsewhere (most likely not suspicious). The evaluation of trace $f_e(\mathbf{x}^{(k)})$ is showed in Figure 2 for each time step, i.e., after each event. The score is much higher for each subsequent suspicious event, while it decreases at slower rate.

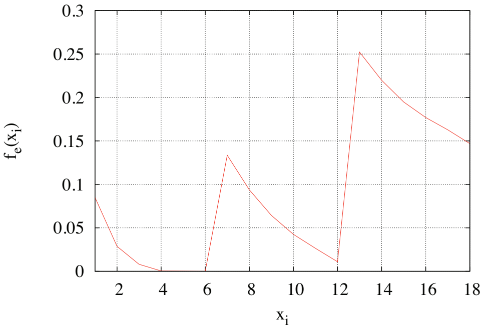

Figure 2: Evaluation of a trace $f_e(\mathbf{x})$ over time.

# Experimental Evaluation

## Experimental Setup

We demonstrate and evaluate the proposed approach on the airport domain. To run proof-of-concept tests we first consider a simulated environment due to several reasons. First, a simulation is controllable and repeatable in terms of ensuring statistical relevance. Second, obtaining real-world data with annotated suspicious behavior might present a difficulty in terms of cost and amount of data required to create a statistically representative dataset. In practice there are tens of hours of several hundred people with only a few instances of suspicious behaviors. Third, a simulator enables the control of the amount of noise that is otherwise introduced by various vision systems (occlusions, false detections, etc.). And last, real-world data in security domain present a difficulty due to privacy issues, confidentiality and national security concerns.

Tsai et al. (2011) developed ESCAPES, a multi-agent simulator for airport evacuations with several types of agents exhibiting behaviors of regular travelers, authorities, and families, which do not necessarily follow often-assumed pedestrian behaviors. The agents' behavior incorporates emotional, informational and behavioral interactions such as emotional contagion, spread of knowledge/fear, social comparison, etc. Therefore an agent is affected by the behavior of other agents and their emotional states, and faced with uncertainty as to what happened and where the nearest exits are. ESCAPES consists of two parts, a 2D environment based on the open-source project OpenSteer (OpenSteer 2004), outputting agents' physical and behavioral information into files, and a 3D visualization component using the Massive Software (Regelous 2011). In one scenario, they modeled the Tom Bradley International Terminal at Los Angeles International Airport including terminals and shops as a realistic simulation environment. This served as our playground for introducing suspicious behaviors prior to an evacuation occurring.

In cooperation with security officials we defined a basic scenario where a suspicious passenger goes from point $A$ to point $B$ while trying to avoid security personnel at the airport. One may argue that an adversary that plans to do something malicious would behave normally in a presence of authorities, which might be true for a highly trained individual. An average person exposed to a high level of stress produces behavior that indicates fear, anxiety, pressure, tension, deception etc. Hence, it is rational for the suspicious agent to minimize contacts with the authorities. Implementation details are explained in Appendix.

A simulation is run with a given airport map, authority agents, regular passengers and a suspicious agent going from point $A$ to $B$, outputting traces with 2D coordinates for all agents. An example is visualized in Figure 3, where the trace of the suspicious agent is marked with red (going from left to right), traces of authorities are green and regular passengers are blue and grey. We initialized the simulator with 100 agents including 10 authorities and a suspicious person with randomly chosen initial and final points. We ran 20 simulations, each consisting of $1500 - 3000$ time steps. In total there was 2000 traces and 4316 interactions between authorities and passengers.

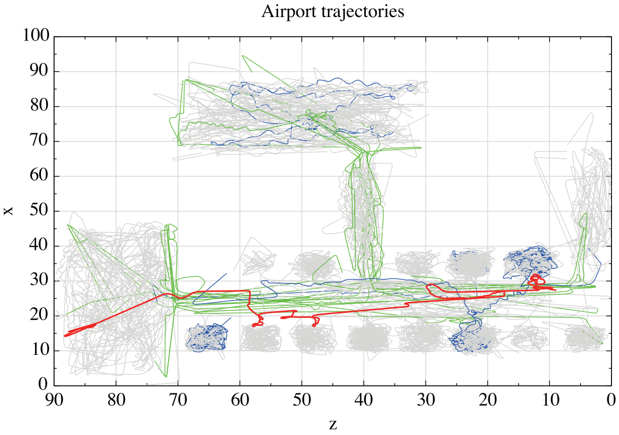

Figure 3: Traces of all agents at the end of a simulation: authorities (green), suspicious (red) and usual passengers (grey, blue).

For evaluation we use *precision*, *recall* and *F-measure*. Precision is defined as a number of true positives (all suspicious cases correctly classified as suspicious) divided by the number of all cases marked as suspicious (true and false positives): $pr = TP/(TP + FP)$. A perfect score 1 means that all cases marked as suspicious were indeed suspicious. Hence score $1 - pr$ presents the amount of *false alarms*. Recall is defined as a number of true positives divided by the number of all suspicious cases: $re = TP/(TP + FN)$. A perfect score 1 means that all suspicious cases were detected (but says nothing about falsely marked normal cases). Since the objectives of this measures are contradictory, we use F-measure, which is harmonic mean of precision and recall, to compare methods: $fm = 2 \cdot pr \cdot re/(pr + re)$.

## Detection Based on Complete Behavior Trace

The first baseline approach evaluates the whole behavior trace consisting of actions. We used HMMs, since they are considered as a baseline for modeling a sequence of actions. Not, that it is not based on trigger events but rather takes the whole trace of 2D coordinates presented as actions. The goal is to differentiate between a sequence of actions produced by a suspicious and a regular passenger. We expect this approach to not perform well, since it too general and unable to precisely model interactive behavior.

We consider three transformations of 2D traces to actions. The first one is based on work by Avrahami-Zilberbrand and Kaminka (2009) where the airport map is divided with a square-based grid with numbered squares. Each trace with Cartesian coordinates is transformed into a sequence of squares. We denote this as *fixed presentation*. The second presentation transforms 2D traces to actions taken in each time step. The action are defined as moving North, South, East, West and their combinations. In total there are nine actions (including staying at the same spot). Compared to the fixed presentation, this presentation can also describe the shape of a trajectory but discards the location information (which can lead to better generalization). We call this a *relative presentation*. The last presentation denoted as *relative position and orientation* defines actions as moving Forward, Backward, Left, Right and their combinations. Compared to the previous presentation it also discards the information about orientation.

Suspicious behavior detector consists of two ergodic HMMs; $\check{S}$ trained on suspicious and $\check{N}$ trained on regular traces. A new trace is first transformed in one of the presentations and then matched against both HMMs. Each one returns likelihood that it produced the given trace $\mathbf{x}$. If the likelihood

$$\frac{\check{\lambda} Pr\{\mathbf{x}|\check{S}\}}{(\check{\lambda} Pr\{\mathbf{x}|\check{S}\} + (1-\check{\lambda}) Pr\{\mathbf{x}|\check{N}\})} > \check{\tau},$$

the sequence is classified as suspicious.

Table 1 shows recall, precision and F-measure for trajectories presented with absolute position (column 2), relative position (column 3), and relative position and orientation (column 4). The second row shows results with acceptable discovery rate (recall), but extremely low precision around 10% in the third row. This means that only 1 out of 10 passengers marked as suspicious was indeed suspicious. States presented with relative position outperform presentations with absolute position, which was expected. Absolute position requires a large data set to cover the complete state space, and second, it is prone to over-fitting. The presentations with relative position and orientation achieved lower recall than other presentations and slightly better precision. The reason can be in over-generalization. However, states represented with relative position were used in further experiments. The overall performance was consistent with our expectations.

Table 1: Evaluation results for HMMs applied to the whole sequence of 2D coordinates presented as actions.

| [%] | Abs. pos. | Rel. pos. | Rel. pos.&ori. |
|---|---|---|---|
| Recall | 62.63 | 66.23 | 40.86 |
| Precision | 7.04 | 10.42 | 11.54 |
| F-measure | 13.24 | 18.01 | 18.00 |

## Detection Based on Trigger Events

The next approaches are based on a two-level architecture. The first level is focused on detection of trigger events, while the second level takes a trace of events and decides whether it was generated by suspicious agent. We first present the detection of trigger events based on Coupled HMMs and trajectory curvature, and then we the discussed approaches at the second level.

**Trigger Events** Trigger events can be of any kind of partial observations we are able to extract from the domain. In the airport domain one can focus on leaving bags unattended, access to restricted area, walking with increased speed, inappropriate clothing for the weather, etc. Limited with our simulator capabilities and focused on the scenario where a suspicious person tries to access some point, we focus on a rather novel descriptor we were unable to find in the literature on suspicious behavior detection at the airport. We are observing interactions between agents at the airport, more precisely, we are interesting how a passenger behaves in a presence of an authority figure. This results in a set of partial observations describing interactive behavior of a passenger. The recognition process first extracts all interactions between passengers and authority figures inside a given radius, producing a set of trajectory pairs that are transformed to relative presentation.

To estimate the probability that an interaction was generated by suspiciousness passenger we approached with Coupled HMMs. CHMM architecture comprises two HMMs chains where hidden states from one chain directly impact the hidden states from the other chain. The current state $Q_t^A$ of agent $A$ is affected by both its previous state $Q_{t-1}^A$ and previous state $Q_{t-1}^B$ of agent $B$ (similar $Q_t^B$ is affected by $Q_{t-1}^B$ and $Q_{t-1}^A$). Each state $Q_i$ also impacts the corresponding observation state $Y_i$. We create and train two CHMMs, $\hat{N}_I$ modeling interactions produced by regular passengers and $\hat{S}_I$ modeling interactions produced by suspicious passengers. An event (interaction) $x$ is then classified with both models yielding $\hat{n}_I(x)$ and $\hat{s}_I(x)$, respectively. We also experimented with more complex CHMM structures including other features such as relative speed and distance, but the results were comparable or even worse.

As a trigger event we also extract all turns in absence of authority when the trajectory curvature exceeds a threshold value. Probabilities that a turn event was generated by suspicious $\hat{n}_T(x)$ or regular passenger $\hat{s}_T(x)$ is acquired with

Table 2: Evaluation results for all approaches comparing recall, precision and F-measure.

| Trace | | Relative presentation | | |
|---|---|---|---|---|
| Actions | Events | Recall | Precision | F-measure |
| HMMs | | 66.23 | 10.42 | 18.01 |
| CHMM | If $\exists k$ | 70.00 | 43.75 | 53.85 |
| | Naive | 90.00 | 40.91 | 56.25 |
| | HMMs | 80.00 | 53.33 | 64.00 |
| | UPR | 80.00 | 66.67 | 72.73 |
| | Scoring $f_e$ | 90.00 | 90.00 | 90.00 |

frequentist estimator from the learning set $D_l$.

**Detectors** The input to the detectors is an event trace produced by the previous level. We instantiated Naive Bayes, HMMs, UPR, and Scoring function detectors. Additionally, we consider another baseline detector using a simple rule saying that if a trace $\mathbf{x}$ contains more than $k$ events most likely generated by a suspicious passenger, then classify it a suspicious. The Naive Bayes detector used probabilities $\hat{s}$ and $\hat{n}$ as returned by previous level. For HMMs we considered two ergodic HMMs, one modeling intentions of regular passenger and other modeling intentions of suspicious passenger. We used two observations, normal and suspicious event, and varied the number of hidden states. The best results were achieved with three hidden states. Note, that HMMs applied atop of CHMMs basically present the Layered HMMs structure (Oliver, Garg, and Horvitz 2004). For all models (including UPR and Scoring function detectors) we estimated the parameters on a part of learning data.

The results were obtained with 10-fold-cross validation. Table 2 summarizes the results. The first two columns show the combination of approaches applied to on the first level (the whole trace of actions) and the second level (trace of trigger events). The third, fourth and fifth columns show recall, precision and f-measure, respectively. The second row shows the best results for HMMs applied in a single layer (the best result from Table 1). We assume that the reason for poor performance lies in inability to model interactive behavior. The third row shows a simple rule saying if there are $k$ events in the trace most likely generated by a suspicious agent, then mark this agent as suspicious. Performance of Naive Bayes detector is showed in the fourth row, outperforming rule baseline. Fifth row shows results for HMMs detector: recall is lower compared to the Naive Bayes baseline but precision is better. According to F-measure HMMs detector performs better. The performance of UPR detector is showed in the sixth row. High recall is accompanied with high precision, outperforming all previous approaches. The last row shows results for proposed Scorning function detector $f_e$ achieving the highest recall and precision. F-measure indicates that it performs better any other approach.

## Summary

We have modeled a trace of multiple trigger events as a mixture of two stochastic processes, normal and suspicious. We have defined detection of suspicious behavior as identifying those traces generated by the suspicious process. We have shown that optimal detection must take into account the complete history in the trace, which present a challenge in real-world applications. We discuss approaches that simplify detection by estimating conditional probabilities as well as approaches that relay on other frameworks such as plan recognition. We proposed a heuristic approach based on a family of *well-behaved* scoring functions that interpret event to produce a score presenting overall suspicion that the trace corresponds to behavior of the suspicious agent using the complete behavior of the agent in the past. The approach was compared on the airport domain against both trajectory-based detection using HMMs and event-based detection using CHMM for detecting events and Naive Bayes, HMMs, UPR, and rule baseline for evaluating event traces. All of them were outperformed by the proposed approach.

## Acknowledgments

The authors would like to thank Zhengyu Yin for valuable discussion and suggestions, and Jason Tsai and Matthew Brown for their help with implementation challenges in the ESCAPES simulator. The work of Boštjan Kaluža was supported partly by the Slovene Human Resources Development and Scholarship Fund and partly by the Slovenian Research Agency.

## References

Arsić, D.; Schuller, B.; and Rigoll, G. 2007. Suspicious behavior detection in public transport by fusion of low-level video descriptors. In *Proceedings of the 8th ICME*, 218–221.

Avrahami-Zilberbrand, D., and Kaminka, G. A. 2007. Incorporating observer biases in keyhole plan recognition (efficiently!). In *Proceedings of AAAI-07*.

Avrahami-Zilberbrand, D. 2009. *Efficient Hybrid Algorithms for Plan Recognition and Detection of Suspicious and Anomalous Behavior*. Ph.D. Dissertation, Bar-Ilan University.

Bak, P.; Rohrdantz, C.; Leifert, S.; Granacher, C.; Koch, S.; Butscher, S.; Jungk, P.; and Keim, D. A. 2009. Integrative visual analytics for suspicious behavior detection. In *IEEE Symposium on Visual Analytics Science and Technology*, 253–254.

Barbará, D.; Domeniconi, C.; Duric, Z.; Filippone, M.; Mansfield, R.; and Lawson, E. 2008. Detecting Suspicious Behavior in Surveillance Images. In *IEEE International Conference on Data Mining Workshops*, 891–900.

Brand, M.; Oliver, N.; and Pentland, A. 1997. Coupled Hidden markov models for complex action recognition. In *CVPR'97*, 994 – 999.

Duong, T. V.; Bui, H. H.; Phung, D. Q.; and Venkatesh, S. 2005. Activity Recognition and Abnormality Detection with the Switching Hidden Semi-Markov Model. In *CVPR'05*, 838–845.

Feris, R. S.; Hampapur, A.; Zhai, Y.; Bobbitt, R.; Brown, L.; Vaquero, D. A.; li Tian, Y.; Liu, H.; and Sun, M.-T. 2009. *Case Study: IBM Smart Surveillance System*. Taylor & Francis Group.

Fine, S.; Singer, Y.; and Tishby, N. 1998. The Hierarchical Hidden Markov Model: Analysis and Applications. *Machine Learning* 32(1):41–62.

Helman, P.; Liepins, G.; and Richards, W. 1992. Foundations of intrusion detection. In *The IEEE Computer Security Foundations Workshop V*.

Hongeng, S., and Nevatia, R. 2003. Large-scale event detection using semi-hidden Markov models. In *IEEE International Conference on Computer Vision*, 1455–1462.

Naylor, M., and Attwood, C. I. 2003. Advisor: Annotated digital video for intelligent surveillance and optimised retrieval. Final report.

Nguyen, N. T.; Phung, D. Q.; Venkatesh, S.; and Bui, H. 2005. Learning and detecting activities from movement trajectories using the hierarchical hidden Markov models. In *CVPR'05*, 955–960.

Oliver, N.; Garg, A.; and Horvitz, E. 2004. Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision and Image Understanding* 96:163–180.

OpenSteer. 2004. Steering behaviors for autonomous characters. http://opensteer.sourceforge.net.

Piciarelli, C.; Micheloni, C.; Foresti, G. L.; and Member, S. 2008. Trajectory-Based Anomalous Event Detection. In *EEE Trans. on Circuits and Systems for Video Technology*, 1544 – 1554.

Rabiner, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, 257–286.

Regelous, S. 2011. MASSIVE: Multiple Agent Simulation System in Virtual Environment. http://www.massivesoftware.com.

Sillito, R. R., and Fisher, R. B. 2008. Semi-supervised learning for anomalous trajectory detection. In *Proc. of BMVC*, 1035–1044. In Proc. BMVC.

Tsai, J.; Kaminka, G.; Epstein, S.; Zilka, A.; Rika, I.; Wang, X.; Ogden, A.; Brown, M.; Fridman, N.; Taylor, M.; Bowring, E.; Marsella, S.; Tambe, M.; and Sheel, A. 2011. ESCAPES - Evacuation Simulation with Children, Authorities, Parents, Emotions, and Social comparison. In *AAMAS-2011*.

Tung, F. 2010. *Goal-based trajectory analysis for unusual behaviour detection in intelligent surveillance*. Ph.D. Dissertation, University of Waterloo.

Vaswani, N.; Chowdhury, A. R.; and Chellappa, R. 2005. ”Shape Activity”: A Continuous State HMM for Moving/Deforming Shapes with Application to Abnormal Activity Detection. In *IEEE Transactions on Image Processing*, 1603 – 1616.

Visontai, M. 2004. Detecting unusual activity in video. In *Proceedings of CVPR 2004*, 819–826.

## Appendix: Generating Suspicious Behavior

To simulate behavior of a suspicious passenger, which tries to get from point $A$ to point $B$ unnoticed, within the ESCAPES simulator we defined a new agent type as follows. A state of the suspicious agent $s$ contains the current position $Q_s(x,y)$. At each time step the agent $s$ computes the probability for being seen by any authority figure $a \in S$, where $S$ is a set of authorities in a certain range. Similarly, a state of an authority agent $a$ is defined by position $Q_a(x,y)$ and direction $\vec{d_a}$. Probability that the authority agents $a$ sees another agent at distance $r$ with an offset angle $\theta$ from the current direction $\vec{d_a}$ is defined as a bivariate normal distribution $N_a(r,\theta)$.

Points $A$ and $B$ are randomly chosen for each independent simulation. When the agent $s$ reaches the point $B$ the simulation ends. The behavior of the suspicious agent follows a few simple rules:

1. Compute $p$ as a sum of probabilities for being seen by any authority figure $a \in A$ in the current position $Q_s$ (and nearby $\pm\epsilon$ region)
$$p = \sum_{a \in A} \iint_{Q_s-\epsilon}^{Q_s+\epsilon} N_a(r,\theta) \tag{12}$$
2. If $p$ exceeds a threshold value then compute eight random points $c_i \in C$ in radius $r$, else restore the original final point $B$ and go to step 4.
3. Select a point with the lowest price such that the sum of probabilities among the current point $Q_s$ and the end point $c_i$ is the smallest
$$\arg\min_{c_i \in C} \sum_{a \in A} \int_{Q_s}^{c_i} N_a(r,\theta)$$
and define it as a new final point $B'$.
4. Move towards the final point. If the distance $d(Q_s,B) < \epsilon$ end, else go to step 1.

The resulting behavior is quite convincing and complex; ability to take into account several authorities and find the best solution in the given situation results in avoiding authorities in a half circle, making u-turns and continuing in the opposite direction, and even hiding in nearby stores.