Real-time event recognition from video via a "bag-of-activities"

Rolf H. Baxter **Neil M. Robertson** **David M. Lane**

Heriot-Watt University
Edinburgh, Scotland, EH14 4AS

Abstract

In this paper we present a new method for high-level event recognition, demonstrated in real-time on video. Human behaviours have underlying activities that can be used as salient features. We do not assume that the exact temporal ordering of such features is necessary, so can represent behaviours using an unordered "bag-of-activities". A weak temporal ordering is imposed during inference, so fewer training exemplars are necessary compared to other methods. Our three-tier architecture comprises low-level tracking, event analysis and high-level recognition. High-level inference is performed using a new extension of the Rao-Blackwellised Particle Filter. We validate our approach using the PETS 2006 video surveillance dataset and our own sequences. Further, we simulate temporal disruption and increased levels of sensor noise.

1 INTRODUCTION

Considerable attention has been given to the detection of events in video. These can be considered low-level events and include agents entering and exiting areas (Fusier et al., 2007), and object abandonment (Grabner et al., 2006). High-level goals have been recognised from none-visual data sources with reasonable success (Liao et al., 2007). However, there has been far less progress towards recognising high level goals from low-level video.

Detecting events from surveillance video is particularly challenging due to occlusions and lighting changes. False detections are frequent, leading to a high degree of noise for high-level inference. Although complex events can be specified using semantic models, they are largely deterministic and treat events as facts (e.g. (Robertson et al., 2008)). Mechanisms for dealing with observation uncertainty are unavailable in these models (Lavee et al., 2009). On the other hand, probabilistic models are very successful in noisy environments, and are at the core of our approach.

Plan recognition researchers such as (Bui and Venkatesh, 2002; Nguyen et al., 2005) used hierarchical structures to model human behaviour. By decomposing a goal into states at different levels of abstraction (e.g. sub-goals, actions), a training corpus can be used to learn the probability of transitioning between the states. Although this work does consider video, a major shortfall is the necessity for training data, which is often unavailable in surveillance domains.

A common way to avoid this issue is to model "normal" behaviours for which training data is easier to obtain (Boiman and Irani, 2007; Xiang and Gong, 2008). Activities with a low probability can then be identified as abnormal. Because semantic meanings cannot be attached to the abnormal activities, they cannot be automatically reasoned about at a higher level, nor explained to an operator.

Another alternative to learning temporal structure is to have it defined by an expert. For simple events this is trivial, but increases at least proportionally with the complexity of the event. In (Laxton et al., 2007) the Dynamic Belief Network for making French Toast was manually specified. Their approach only considers a single goal.

Dee and Hogg showed that interesting behaviour can be identified using motion trajectories (Dee and Hogg, 2004). Their model identified regions of the scene that were visible or obstructed from the agent's location, and produced a set of goal locations that were consistent with the agent's direction of travel. Goal transitions were penalised so irregular behaviours were identified via their high-cost.

In (Baxter et al., 2010) a simulated proof of concept suggested behaviours could be identified using temporally unordered features. This has the advantage that training exemplars are not required. Our work furthers the idea that complex behaviour can be semantically recognised using a feature-based approach. We present methods for representing behaviours, performing efficient inference, and demonstrate validity and scalability on real, multi-person video.

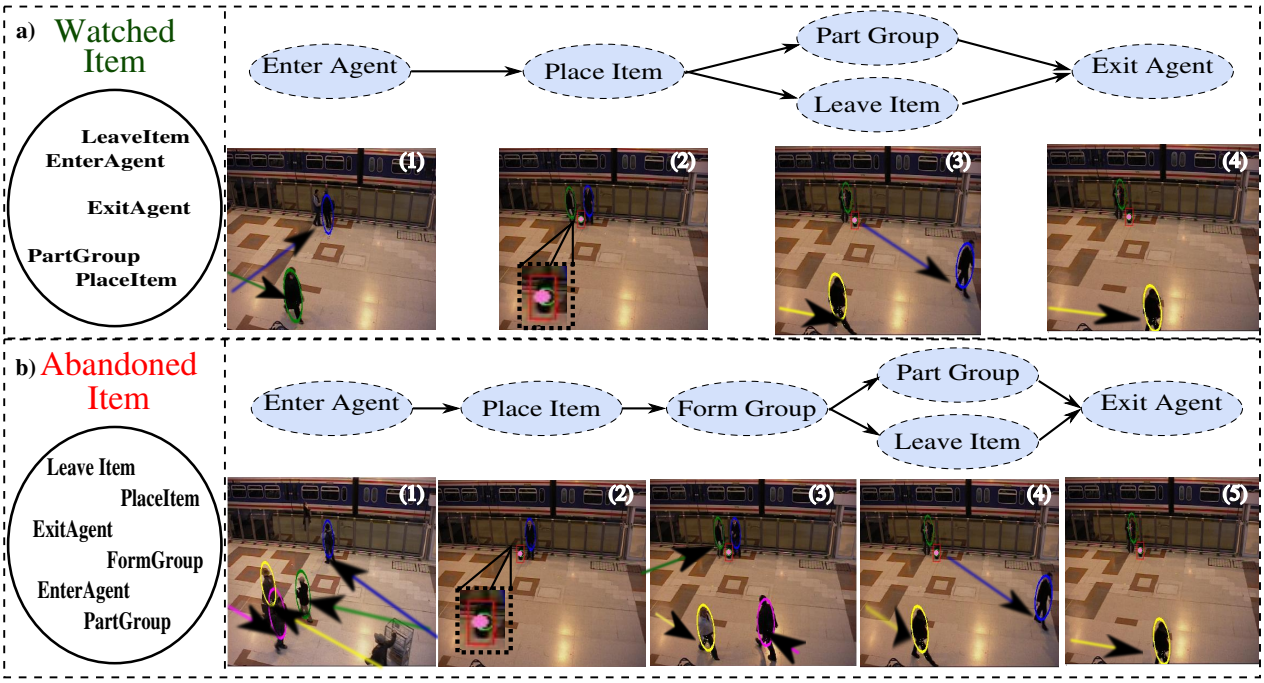

Figure 1: (a) When two agents enter together, an item left by one agent is not a threat when the second agent remains close. (b) When two agents enter separately, it cannot be assumed that the item is the responsibility of the remaining agent

This paper presents a framework with three major components : (1) low-level object detection and tracking from video; (2) detecting and labelling simple visual events (e.g. object placed on floor), and (3) detecting and labelling high-level, complex events, typically including multiple people/objects and lasting several minutes in duration. Our high-level inference algorithm is based upon the Rao-Blackwellised Particle Filter (Doucet et al., 2000a), and can recognise both concatenated and switched behaviour. Our entire framework is capable of real-time inference.

We validate our approach chiefly on real, benchmarked surveillance data: the PETS 2006 video surveillance dataset. We report classification accuracy and speed on four of the original scenarios, and one additional scenario. The fifth scenario was acquired by merging frames from different videos to provide a complex, yet commonly observed behaviour. Further evaluation is conducted by simulating sensor noise and temporal disruption, and on additional video recorded in our own vision laboratory.

Throughout this paper the term activity is used to refer to a specific short-term behaviour that achieves a purpose. An activity is comprised of any number of atomic actions. Activities are recognised as simple events. These terms are interchanged depending upon context. Similarly, collections of activities construct goals, and will be referred to as features of that goal. Goals are detected as complex events.

2 RECOGNITION FRAMEWORK

Figure 1 illustrates two complex behaviours: *Watched Item* and *Abandoned Item*. *Watched Item* involves two persons who enter the scene together. One person places an item of luggage on the floor and leaves, while the other person remains in close proximity to the luggage. This scenario is representative of a person being helped with their bags. *Abandoned Item* is subtly different: the two people do not enter the scene together (Frames 1 and 3 in Figure 1b).

Traditionally, the proximity of people to their luggage is used to detect abandonment. This would generate an alert for both of the above scenarios. To distinguish between them, we integrate low-level image processing with high-level reasoning (Figure 2). We use a hierarchical, modular framework to provide an extendible system that can be easily updated with new techniques. Video data is provided as the source of observations and is processed at three different levels: Object Detection and Tracking, Simple Event Recognition, and Complex Event Recognition. Image processing techniques provide information about objects in the scene, allowing simple semantic events to be detected. These then form observations for high-level recognition.

2.1 OBJECT DETECTION AND TRACKING

Static cameras allow foreground pixels to be identified using background subtraction. This technique compares the current frame with a known background frame. Pixels that

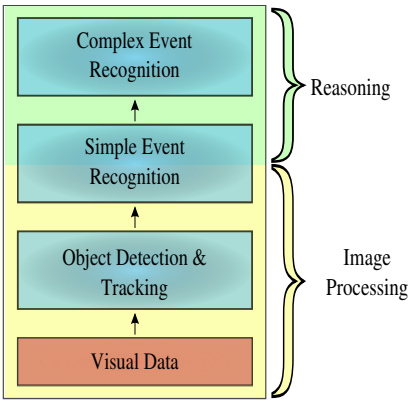

Figure 2: Our architecture for complex event recognition

are different are classed as the foreground. Connected foreground pixels give foreground blobs, and are collectively referred to as B_t. The size/location of each blob can be projected onto real-world coordinates using the camera calibration information. Two trackers operate on B_t.

Person Tracker: Our person tracker consists of a set of SIR filters (Gordon et al., 1993). SIR filters are similar to Hidden Markov Models (HMMs) in that they determine the probability of a set of latent variables given a sequence of observations (Rabiner, 1989). However, when latent variables are continuous, exact approaches to inference become intractable. The SIR filter is an approximation technique that uses random sampling to reduce the state space.

Our filters consist of one hundred particles representing the person's position on the ground plane, velocity, and direction of travel (Limprasert, 2010). For each video frame, the blobs (groups of foreground pixels) that contain people are quickly identified from B_t using ellipsoid detection. We denote these blobs E_t. For each ellipsoid that cannot be explained by an existing filter, a new filter is instantiated to track the person.

In order to address the temporary occlusion of a person (e.g. people crossing paths), particles also contain a visibility variable (0/1) to indicate the person's disappearance. This variable applies to all particles in the filter. By combing this variable with a time limit, the filter continues to predict the person's location for short occlusions, while longer occlusions will cause the track to be terminated.

Object Tracker: Our second tracking component consists of an object detector. In the video sequences this detects luggage and is similarly heuristic to other successful approaches (Lv et al., 2006). To remove person blobs and counteract the effect of lighting changes, which spuriously create small foreground blobs, the tracker eliminates blobs that are not within the heuristically defined range: $0.3 \leq width/height \leq 1m$. Each remaining blob is classified as a stationary luggage item if the blob centroid remains within 30cm of its original position, and is present for at least 2 continuous seconds. The red rectangle identifies a tracked luggage item in Figures 1a&b, frame 2. Inversely, if the blob matching a tracked luggage object cannot be identified for 1 second, the luggage is classed as "removed". To prevent incorrect object removal (e.g. when a person is occluding the object), the maximum object size constraint is suspended once an object is recognised.

2.2 SIMPLE EVENT RECOGNITION

Simple events can be generated by combining foreground detection/tracking with basic rules. Table 1 specifies the set of heuristic modules used in our architecture to encode these rules. It should be highlighted that the *GroupTracker* only uses proximity rules to determine group membership (we suggest improvements in Future Work). *Group Formed* events are trigged when two people approach, and remain within close proximity of each other. Inversely, *GroupSplit* events are triggered when two previously "grouped" people cease being in close proximity.

Although these naive modules achieve reasonable accuracy on the PETS dataset, it is important to acknowledge that they would be insufficient for more complex video. The focus of our work is high-level inference and thus state-of-the-art video processing techniques may not have been used. The modularity of our framework allows any component to be swapped, and thus readily supports the adoption of improved video processing algorithms. Furthermore, we demonstrate via simulation that high-level inference remains robust to increased noise.

2.3 COMPLEX EVENT RECOGNITION

Human behaviour involves sequential activities, so it is natural to model them using directed graphs as in Figure 1. Dynamic Bayesian Networks (Figure 3) are frequently chosen for this task, where nodes represent an agent's state, solid edges denote dependence, and dashed edges denote state transitions between time steps (Murphy, 2002). Each edge has an associated probability which can be used to model the inherent variability of human behaviour [1].

Like many others, (Bui and Venkatesh, 2002) learnt model probabilities from a large dataset. However, annotated libraries of video surveillance do not exist for many interesting behaviours, making there no clear path for training high-level probabilistic models. Similar problems occur when dealing with military or counter-terrorism applications, where data is restricted by operational factors. Alternative approaches include manually specifying the probabilities, and using a distribution that determines when transitions are likely to occur (Laxton et al., 2007).

We hypothesise that many human behaviours can be recog-

[1] Figure 3 will be fully explained in section 3

nised without modelling the exact temporal order of activities. This means that model parameters do not need to be defined by either an expert, or training exemplar. We consider activities as salient features that characterise a behaviour. Goals can be recognised by combining a collection (bag) of activities with a weak temporal ordering.

Feature based recognition algorithms have primarily been developed for object detection applications. To identify features that are invariant to scale and rotation, object images are often transformed into the frequency or scale domains, where invariant salient features can be more readily identified (Lowe, 1999). The similarities between recognising objects and human behaviours has previously been noted (Baxter et al., 2010; Patron et al., 2008), and it is this similarity upon which we draw our inspiration.

Figure 1 helps visualise a behaviour as a set of features. Each ellipse represents a complex event as a bag of activities (cardinality: one). We formally denote a bag by T, the **T**arget event, where each element is drawn from the set of detectable simple events α. Each simple-event is a feature.

The agents progress towards a target event can be monitored by tracking the simple events generated. Fundamentally, the simple events should be consistent with T if T correctly represents the agent's behaviour. For instance, if simple event α^i is observed but $\alpha^i \ni T$, then α^i must be a false detection, or T is not the agent's true behaviour.

As time increases more events from T should be generated. If we make the assumption that each element of a behaviour is only performed once, then the set of expected simple events reduces to the elements of T not yet observed. If $T = \langle \gamma, \delta, \epsilon \rangle$ and γ has already been observed, then the set of expected events is $\langle \delta, \epsilon \rangle$. In this way a weak temporal ordering can be applied to the elements of T without learning their absolute ordering from exemplar.

If C is defined as the set of currently observed simple events, $T \backslash C$ is the set of expected events. At each time step, events in $T \backslash C$ have equal probability, while all other events have 0 probability. This probability distribution encapsulates the assumption that each simple event is only truthfully generated once per behaviour, and is consistent with other work in the field (Laxton et al., 2007). We discuss the implications and limitations of this assumption in section 5.

Worked Example: Using Figure 1's *Watched Item* behaviour as an example, at time step t=0 each of the 5 events (LeaveItem, EnterAgent, ExitAgent, PartGroup, PlaceItem) has equal probability. In frame 1 ($t = 1$), $p\left(EnterAgent\right) = 0.2$. At $t = 2$, $p(EnterAgent) = 0$, while $\forall i \in T \backslash C : p(i) = 0.25$. Note that $p(FormGroup|T = WatchedItem) = 0$ at all time steps, because *FormGroup* $\ni$ *WatchedItem*.

Table 1: The simple event modules used by our architecture

Module	Description
Agent Tracker	Detects the entry/departure of people from the scene.
Object Tracker	Upon luggage detection, associates that luggage with the closest person.
Group Tracker	Identifies when people are in close proximity, and split from a single location.
Abandoned Object Detector	Detects when luggage is ≥ 3 metres from its owner.

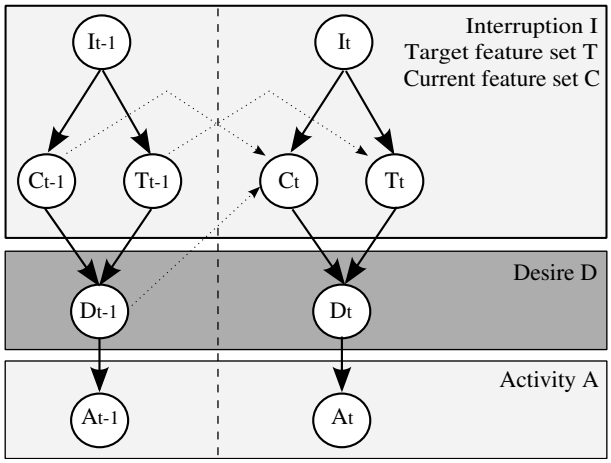

Figure 3: The top two layers of the Dynamic Bayesian Network predict low-level events for a complex event

3 DYNAMIC BAYESIAN NETWORK

This approach can be captured by the Dynamic Bayesian Network (DBN) in Figure 3. Nodes within the top two layers represent elements of the person's state and can be collectively referred to as x. The bottom layer represents the simple event that is observed. The vertical dashed line distinguishes the boundary between time slices, $t-1$ and t.

Activity observations: Recognition commences at the bottom of the DBN using the simple-event detection modules. Ours are described in section 3. Each detection must be attributed to a tracked object or person.

Desire: Moving up the DBN hierarchy the middle layer represents the agent's current desire. A desire is instantiated with a simple-event (activity) that supports the complex-event (goal). Given the previous definitions of T and C the conditional probability for D (desire) is:

$$p\left(d^i\right) = p\left(d^j\right) \forall_{i,j} : d^i, d^j \in C \backslash T \quad (1)$$

$$p\left(d^k\right) = 0 \, \forall_k : d^k \ni C \backslash T \quad (2)$$

Define $TP\left(\alpha^i\right)$ as the true positive detection probability of simple event α^i. Having now defined A and D the the emission probabilities can also be defined by the function $E\left(A_t, D_t\right)$:

$$E\left(A_t, D_t\right) = p\left(A_t = \alpha^i | D_t = \alpha^j\right) \quad (3)$$

$$= TP\left(\alpha^i\right) \quad : \text{i = j} \quad (4)$$

$$= 1 - TP\left(\alpha^i\right) \quad : \text{i} \neq \text{j} \quad (5)$$

Goal Representation: The top layer in the DBN represents the agent's top-level goal and tracks the features that have been observed. The final node; I, removes an important limitation in (Baxter et al., 2010). I represents behaviour interruption, which indicates that observation A_t cannot be explained by the state x_t (the top two layers of the DBN). It implies one of two conditions. 1) A person has switched their complex behaviour (e.g. goal) and thus $T_{t-1} \neq T_t$. Although humans frequently switch between behaviours, this condition breaks the assumptions made by (Baxter et al., 2010), causing catastrophic failure. 2) A_t is a false detection. In this case, the elements of *T* and *C* are temporarily ignored.

3.1 MODEL PARAMETERS

Given the model description above, the DBN parameters can be summarised as follows.

Variables: α is the set of detectable simple events. T represents a single behaviour (complex event) and $\forall t \in T : t \in \alpha$. C represents the elements of T that have been observed and thus $\forall c \in C : c \in T$. D is a prediction of the next simple-event and is drawn from $T \backslash C$. Finally, A is the observed simple event and is drawn from α.

Probabilities: Define $Beh\left(\beta\right)$ as the target feature set for behaviour β, and $Pr\left(\beta\right)$ as the prior probability of β. The transition probabilities for latent variables C and T can then be defined as per Table 2.

The distribution on values of D is defined by equations 1 and 2, and the emission probabilities by equations 3 to 5.

It should be noted that of all these parameters, only functions $Beh\left(\beta\right)$ and $E\left(A_t, D_t\right)$ need to be defined by the user. It is expected that $Beh\left(\beta\right)$ (the set of features representing behaviour β) can be easily defined by an expert, while $E\left(A_t, D_t\right)$ may be readily obtained by evaluating the simple-event detectors on a sample dataset. All other parameters are calculated at run-time, eliminating learning.

3.2 RAO-BLACKWELLISED INFERENCE

The DBN in Figure 3 is a finite state Markov chain and could be computed analytically. However, given our target application of visual surveillance, which has the requirement of near real-time processing, we adopt a particle filtering approach to reduce the execution time. In Particle Filtering the aim is to recursively estimate $p(x_{0:t}|y_{0:t})$, in which a state sequence $\{x_0, ..., x_t\}$ is assumed to be a hidden Markov process and each element in the observation sequence $\{y_0, ..., y_t\}$ is assumed to be independent given the state (i.e. $p(y_t|x_t)$) (Doucet et al., 2000b).

We utilise a Rao-Blackwellised Particle Filter (RBPF) so that the inherent structure of a DBN can be utilised. We wish to recursively estimate $p(x_t|y_{1:t-1})$, for which the RBPF partitions x_t into two components $x_t : (x_t^1,\ x_t^2)$ Doucet et al. (2000a). This paper will denote the sampled component by the variable r_t, and the marginalised component as z_t. In the DBN in Figure 3, $r_t : \langle\ C_t,\ T_t,\ I_t\ \rangle$ and $z_t : D_t$. This leads to the following factorisations:

$$p(x_t|y_{1:t-1}) = p(z_t|r_t, y_{1:t-1})p(r_t|y_{1:t-1}) \quad (6)$$

$$= p(D_t|C_t, T_t, I_t, y_{1:t-1})p(C_t, T_t, I_t|y_{1:t-1}) \quad (7)$$

The factorisation in 7 utilises the inherent structure of the Bayesian network to perform exact inference on D, which can be efficiently performed once $\langle\ C_t,\ T_t,\ I_t\ \rangle$ has been sampled. Each particle i in the RBPF represents a posterior estimate (hypothesis) of the form $h_t^i : \langle\ C_t^i,\ T_t^i,\ I_t^i,\ D_t^i,\ W_t^i \rangle$, where W_t is the weight of the particle calculated as $p(y_t^i|x_t^i)$.

For brevity we will focus on the application of the RBPF to our work, but refer the interested reader to (Bui and Venkatesh, 2002; Doucet et al., 2000a) for a generic introduction to the approach.

3.2.1 Algorithm

At time-step 0, T is sampled from the prior and $C = \emptyset$ for all N particles. For all other time steps, N particles are sampled from the weighted distribution from $t-1$ and each particle predicts the new state $\langle C_t^i, T_t^i, I_t^i \rangle$ using the transition probabilities in Table 2.

After sampling is complete, the particle set is partitioned into those where $p(y_t|C_t, T_t, I_t)$ is non-zero, and zero. The first partition is termed the *Eligible* set because the particle states are consistent with the new observation, while the second partition is termed the *Rebirth* set. Particles in the *Rebirth* set represent those where an interruption has occurred. For each particle in this set, T and C are re-initialised according to the prior distribution with a probability of $p(TP)$, indicating the true positive rate of the observation. With a probability of $1 - p(TP)$, particles are flagged as "FP" (False Positive), and are not re-initialised.

At the next step, the *Eligible* and *Rebirth* sets are recombined and the Rao-Blackwellised posterior is calculated: $p(z_t^i|r_t^i, y_{1:t-1}) = p(D_t^i|C_t^i, T_t^i, I_t^i, y_{1:t-1})$. The value of D_t^i (the agent's next desire) is then predicted according to the Rao-Blackwellised posterior. At this point each particle has a complete state estimate x_t^i, and can be weighted according to equation 8. It is important to note that particles

Table 2: DBN transition probabilities between time steps $t-1$ and t

$p\left(C_t = C_{t-1} \cup \{D_{t-1}\} | I_t = 0\right)$	$= TP\left(A_{t-1}\right)$	when $D_{t-1} = A_{t-1}$
$p\left(C_t = C_{t-1} \cup \{D_{t-1}\} | I_t = 0\right)$	$= 0$	when $D_{t-1} \neq A_{t-1}$
$p\left(C_t = \emptyset | I_t = 1\right)$	$= 1$	
$p\left(T_t \neq T_{t-1} | I_t = 0\right)$	$= 0$	
$p\left(T_t = Beh\left(\beta\right) | I_t = 1\right)$	$= pr\left(\beta\right)$	if A_{t-1} not assumed false positive
$p\left(T_t = T_{t-1} | I_t = 1\right)$	$= 1$	if A_{t-1} assumed false positive

flagged as "FP" are weighted with $1 - p(TP)$.

$$p(y_t|x_t^i) = p(A_t|C_t^i, T_t^i, I_t^i, D_t^i) \quad (8)$$

The final step in the algorithm is to calculate the transition probabilities. This step ensures that the algorithm is robust to activity recognition errors. The transition probability encapsulates the probability that the agent really has performed the predicted feature D_t^i, observed via A_t.

4 RESULTS AND DISCUSSION

Two datasets were used to evaluate our framework. Five complex behaviours were extracted from four PETS 2006 scenarios, and our own video dataset contains the same behaviours but encompasses more variability than PETS in terms of luggage items and the ordering of events. Experiments were run on a Dual Core 2.4Ghz PC with 4GB RAM.

Figure 4 shows the average F-Scores for the low-level detectors (trackers, event modules). An F-score is a weighted average of a classifiers accuracy and recall with range [0:1], where 1 is optimal. Our person tracker performs well (F-Score ≥ 0.92), but occasionally misclassified non-persons (e.g. trolley), instantiates multiple trackers for a single person, or does not detect all persons entering in close proximity. The object tracker has an F-Score ≥ 0.83, and is limited by partial obstructions from the body and shadows.

The naivety of our simple event modules makes them reliant on good tracker performance. Although the average score is 0.83, the "Group Formed" module is particularly unreliable (F-Score: 0.6).

4.1 COMPLEX EVENT RECOGNITION

The five complex behaviours used in our evaluation are: *Passing Through 1 (PT-1):* Person enters and leaves, *Passing Through 2 (PT-2):* Person enters, places luggage, picks it up and leaves, *Abandoned Object 1 (AO-1):* Person meets with a second person, places luggage and leaves, *Abandoned Object 2 (AO-2):* Person enters, places luggage and leaves, and *Watched Item (WI):* Two people enter together, one places luggage and leaves, one remains. This last behaviour was synthesized for the PETS dataset by merging track data from scenarios six and four.

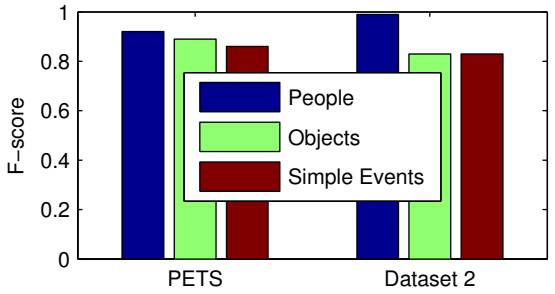

Figure 4: The Low-level F-scores for objects and people tracking, and simple events

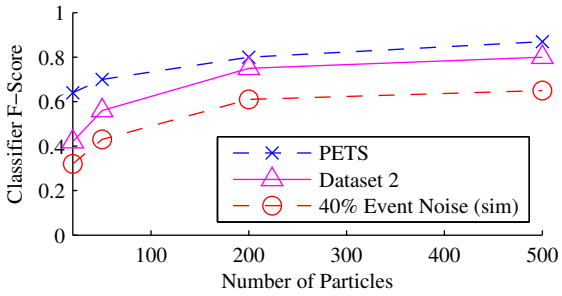

Figure 5: Classifier F-Score as the number of particles is increased (reducing speed).

Figure 5 compares the average classifier F-Scores as the number of particles is increased. Classifications are made after all simple events have been observed by selecting the most likely complex event. A minimum likelihood of 0.3 was imposed to remove extremely weak classifications. As the number of particles increases accuracy/recall is improved. The algorithm remains very efficient with 500 particles, and is capable of processing in excess of 38,000 simple events per second. The classifiers achieve 0.8 F-Score on Dataset 2, and 0.87 on PETS.

In section 3 we highlighted that our naive simple-event

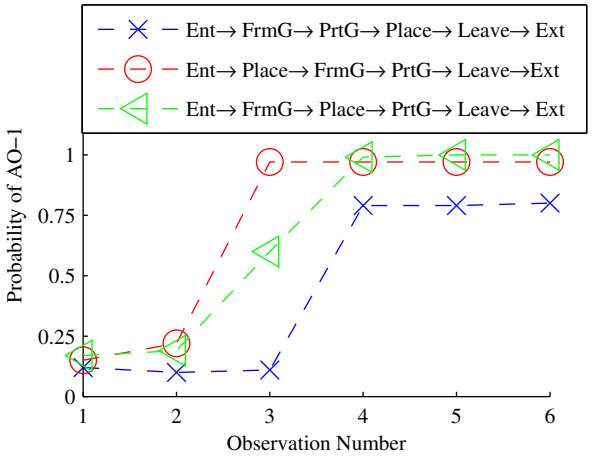

Figure 6: Observations arriving in different orders still match the correct goal (AO-1). Nomenclature: Enter (Ent), Form Group (FrmG), Part Group (PrtG), Place Item (Place), Leave Item (Leave), Exit (Ext)

modules would perform poorly on more complex video. To simulate these conditions, we artificially inserted noise into the observation stream to lower the true positive rate to 60%. Figure 5 shows that even with this high degree of noise, complex events can be detected with 0.65 F-Score.

Table 3 shows classifier confusion across both datasets. Missing object detections cause confusion between *PT-1* and *PT-2*. Behaviours *AO-1* and *WI* differ by only one event (Form Group), and thus absent group detections lead to confusion here.

Table 3: Confusion Matrix for the combined video datasets

	Scenario				
	PT-1	PT-2	WI	AO-1	AO-2
PT-1	0.92	0	0	0.08	0
PT-2	0.33	0.58	0	0	0.08
WI	0	0	0.9	0.1	0
AO-1	0	0	0.2	0.8	0
AO-2	0	0	0	0	1

4.2 TEMPORAL ORDER

We proposed that the exact temporal order of observations does not need to be modelled to recognise human behaviour. Figure 6 supports this thesis by showing complex-event likelihood for three different activity permutations of the *AO-1* behaviour. In all three cases *AO1* is highly probable, although there are differences in probability. These differences are because some activity subsequences are shared between multiple behaviours. For instance, $\langle PlaceItem, LeaveItem, Exit\rangle$ matches both

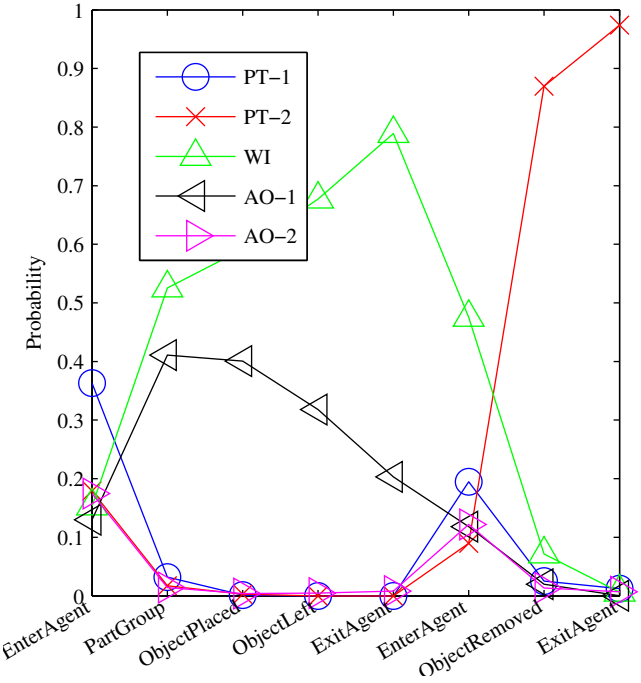

Figure 7: Probability of each behaviour as observations are made. Behaviour switches from WI to PT-2 at observation 6, causing a similar shift in behaviour probability.

AO-1 and *AO-2*. There is a low probability that observations $\langle FormGroup, PartGroup\rangle$ were false detections, and thus some probability is removed from AO1 in support of AO2, which can also explain the subsequence $\langle PlaceItem, LeaveItem, Exit\rangle$. Although observation order can have an impact on goal probability, it is clear that our thesis holds for these behaviours.

4.3 BEHAVIOUR SWITCHING

Our inference algorithm contains components to detect behaviour switching, which occurs when an agent concatenates or otherwise changes their behaviour (see Section 2.3). To demonstrate the effectiveness of these components Figure 7 plots the probability of each behaviour as observations are received from two concatenated behaviours. The behaviours are *WI*, followed by *PT-2*.

In observation 1 the agent enters. The distributions on the features within each behaviour causes *PT-1* to be most probable because it has the least features. The second observation can only be explained by two behaviours and is reflected in the figure. At observation six "EnterAgent" cannot be explained by any of the behaviours, triggering behaviour interruption. Observation seven can only be explained by *PT-2* and this is reflected in the figure. As a result, the behaviours that best explain the observations are *WI* and *PT-2*, which matches the ground truth.

5 CONCLUSION AND FUTURE WORK

This paper has argued that data scarcity prevents the advancement of high-level automated visual surveillance using probabilistic techniques, and that anomaly detection side-steps the issue for low-level events. We proposed that simple visual events can be considered as salient features and used to recognise more complex events by imposing a weak temporal ordering. We developed a framework for end-to-end recognition of complex events from surveillance video, and demonstrated that our "bag-of-activities" approach is robust and scalable.

Section 2.3 made the assumption that for a set of features defining a behaviour, each feature is only performed once. This assumption limits our approach but is not as strong as it may at first appear. An agent who enters and exits the scene can still re-enter, as this is simply the concatenation of two behaviours. Each individual behaviour has only involved one 'EnterAgent' event so the assumption is not in conflict. Furthermore, it is also possible to consider actions that are opposites. For instance, placing and removing a bag, or entering and exiting the scene, can both be considered action pairs that 'roll-back' the state. Although not implemented in this paper, further work has shown that this is an effective means of allowing some action repetition. The only behaviours prevented by the assumption are those that require performing action A twice (e.g. placing two individual bags).

Clearly, improving the sophistication of the simple event detection modules is a priority in extending our approach to more complicated data. The *Group Tracker* module could be improved by estimating each person's velocity and direction using a Kalman filter. These attributes could then be merged with the proximity based approach to more accurately detect the forming and splitting of groups.

Acknowledgements

This work was partially funded by the UK Ministry of Defence under the Competition of Ideas initiative.

References

Rolf H. Baxter, Neil M. Robertson, and David M. Lane. Probabilistic behaviour signatures: Feature-based behaviour recognition in data-scarce domains. In *Proceedings of the 13th International Conference on Information Fusion*, 2010.

Oren Boiman and Michal Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision*, 74(1): 17–31, 2007.

Hung H. Bui and Svetha Venkatesh. Policy recognition in the abstract hidden markov model. *Journal of Artificial Intelligence Research*, 17:451–499, 2002.

Hannah Dee and David Hogg. Detecting inexplicable behaviour. In *British Machine Vision Conference*, volume 477, page 486, 2004.

Arnaud Doucet, Nando de Freitas, Kevin Murphy, and Stuart Russell. Rao-blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 176–183, 2000a.

Arnaud Doucet, Simon J. Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000b.

Florent Fusier, Valry Valentin, Franois Brmond, Monique Thonnat, Mark Borg, David Thirde, and James Ferryman. Video understanding for complex activity recognition. *Machine Vision and Applications*, 18:167–188, 2007. ISSN 0932-8092.

Neil J. Gordon, David J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107 –113, April 1993. ISSN 0956-375X.

Helmut Grabner, Peter M. Roth, Michael Grabner, and Horst Bischof. Autonomous learning of a robust background model for change detection. In *Workshop on Performance Evaluation of Tracking and Surveillance*, pages 39–46, 2006.

Gal Lavee, Ehud Rivlin, and Michael Rudzsky. Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 39(5):489–504, 2009. ISSN 1094-6977.

Benjamin Laxton, Jongwoo Lim, and David Kriegman. Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –8, 2007.

Lin Liao, Donald J. Patterson, Dieter Fox, and Henry Kautz. Learning and inferring transportation routines. *Artificial Intelligence*, 171(5-6):311–331, 2007. ISSN 0004-3702.

Wasit Limprasert. People detection and tracking with a static camera. Technical report, School of Mathematical and Computer Sciences, Heriot-Watt University, 2010.

David G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, volume 2, pages 1150–1157, 1999.

Fengjun Lv, Xuefeng Song, Bo Wu, Vivek Kumar, and Singh Ramakant Nevatia. Left luggage detection using bayesian inference. In *Proceedings of PETS*, 2006.

Kevin P. Murphy. *Dynamic Bayesian networks: representation, inference and learning*. PhD thesis, 2002.

Nam T. Nguyen, Dinh Q. Phung, Svetha Venkatesh, and Hung Bui. Learning and detecting activities from movement trajectories using the hierarchical hidden markov models. In *Computer Vision and Pattern Recognition*, volume 2, pages 955–960, 2005. ISBN 0-7695-2372-2.

Alonso Patron, Eric Sommerlade, and Ian Reid. Action recognition using shared motion parts. In *Proceedings of the 8th International Workshop on Visual Surveillance*, October 2008.

Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, San Francisco, CA, USA, 1989.

Neil Robertson, Ian Reid, and Michael Brady. Automatic human behaviour recognition and explanation for CCTV video surveillance. *Security Journal*, 21(3):173–188, 2008.

Tao Xiang and Shaogang Gong. Video behavior profiling for anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):893, 2008.