Workflow Activity Monitoring using Dynamics of Pair-wise Qualitative Spatial Relations

Ardhendu Behera, Anthony G Cohn, and David C Hogg

School of Computing, University of Leeds,
Woodhouse Lane, Leeds, LS6 4JZ, UK
{A.Behera,A.G.Cohn,D.C.Hogg}@leeds.ac.uk

Abstract. We present a method for real-time monitoring of workflows in a constrained environment. The monitoring system should not only be able to recognise the current step but also provide instructions about the possible next steps in an ongoing workflow. In this paper, we address this issue by using a robust approach (HMM-pLSA) which relies on a Hidden Markov Model (HMM) and generative model such as probabilistic Latent Semantic Analysis (pLSA). The proposed method exploits the dynamics of the qualitative spatial relation between pairs of objects involved in a workflow. The novel view-invariant relational feature is based on distance and its rate of change in 3D space. The multiple pair-wise relational features are represented in a multi-dimensional relational state space using an HMM. The workflow monitoring task is inferred from the relational state space using pLSA on datasets, which consist of workflow activities such as 'hammering nails' and 'driving screws'. The proposed approach is evaluated for both 'off-line' (complete observation) and 'on-line' (partial observation). The evaluation of the novel approach justifies the robustness of the technique in overcoming issues of noise evolving from object tracking and occlusions.

Keywords: Qualitative Spatio-temporal Relations, Workers Instructions, Activity Recognition, Hidden Markov Model (HMM), Probabilistic Latent Scemantic Analysis (pLSA)

1 Introduction

A *workflow* is a temporally ordered set of procedural steps for accomplishing a task in which people and tools are involved in each step of the process. In an industrial environment, the aim of workflow monitoring is to assist operators unfamiliar with a workflow by providing *on-the-fly* instructions from an automatic system. This enables continual interaction between operators and the system while performing a workflow. In an on-going workflow, the proposed monitoring system should be able to anticipate the next possible tasks and recognize the deviations from the correct workflows which may lead to quality and/or health and safety problems. In our case, the operators' instructions will be provided via augmented reality, video clips and/or text using a see-through Head Mounted

Display (HMD)[27]. Therefore, the monitoring system requires a general ability to learn, analyze and model workflow patterns. This associates to a problem of activity recognition.

The more general problem of activity recognition is widely studied within Computer Vision. Much of this work has focused on the development of probabilistic models over object configuration spaces and estimated from training data. Examples include Hidden Markov Models [5, 25, 17, 6], stochastic context free grammars (SCFG) [15, 1], echo state networks (ESN) [22], propagation networks (P-nets) [21], Past-Now-Future networks (PNF-networks) [19] and Bayesian networks [12, 11]. Very often the configuration space is confined to the location and motion of objects within a scene based frame of reference [14, 9]. Most of these models consider only the behaviour of an individual object, such as location and speed in the image plane. Though an activity recognition using a trajectories-based model is powerful, the model complexity increases quadratically with an increase in interactions between multiple objects participating in a task. Furthermore, the tracking algorithm often fails due to occlusion and inability to distinguish between foreground and background.

In this paper, we explore the activity recognition problem in the context of workflow by using qualitative spatio-temporal pair-wise relations between human body parts, tools and objects in a workspace. These relations are established using a relational feature vector representing distance and the rate of change of distance between pairs of objects in 3D space. The motivation for using relational features is to enable the model to follow the ongoing workflow, even though an object is missing due to occlusion or scene complexity. This is possible by considering the spatio-temporal configurations of other observed objects. For example, during the task of hammering, if the individual's hand moves towards the nail box and back to the work bench, it is most likely that the he/she has picked up a nail, by considering the spatio-temporal configurations between nail box and hand during the 'retrieve-nail' subtask. Similarly, if the participant's hand moves towards the screw box, the system should then assert a violation of workflow since the ongoing task is *hammering of nails*.

In the present study, we consider all possible pair-wise relations among objects in a given workspace. These relations are then represented in a relational state space. We propose a novel method to model workflow from this relational state space by using probabilistic *Latent Semantic Analysis* (pLSA) [10]. We evaluate our proposed technique with the workflows of *hammering nails* and *driving screws*. In this model, each workflow sequence consists of multiple subsequences of *primitive events*.

2 Related Work

Activity recognition in the context of workflow is still an active field of research. In this section, a brief description of related work on workflow monitoring and computer vision-based activity recognitions most associated to the context of workflow, is presented.

Veres *et al.* [22] proposed a method for monitoring workflows in a car assembly line. The method uses a global motion descriptor by sampling an input image sequences by a fixed overlapping spatial grids over whole image. Each grid is represented by local motion descriptor based on pixel intensity. The global motion descriptor for an image at a given timestamp is the concatenation of these local motion descriptors. Eco state networks (ESN) [13] are used as a time series predictor for workflow monitoring. Pody *et al.* [18] uses a hierarchical-HMM with observation of 3D optical flow-features for monitoring a hospital's operating rooms. The 3D flow-features are extracted by quantisig the optical flow of pixels inside a spatio-temporal cell of fixed volume. The top-level topology of the hierarchical-HMM is temporally constrained and the bottom level sub-HMM is trained independently with labelled sub-sequences. Pinhanez and Bobick introduced the Past-Now-Future networks (PNF-networks) [19] using Allen's temporal relations [2] to express parallelism and mutual exclusion between different sub-events. In order to gain a detection of actions and sub-actions, Allen's interval algebra network is mapped into a simpler three-valued PNF-network representing temporal ordering constrained between the start and end timing of event instances. Shi *et al.* [21] presented propagation networks (P-nets) to model and detect primitive actions from videos by tracking individual objects. P-nets explicitly model parallel streams of events and are used for classification. The detailed topology is handcrafted and trained from partially annotated data. Moore and Essa [15] use stochastic context-free grammars (SCFG) to recognize separable multi-tasked activities from a video illustrating a card game. All relations between the tracked events are described using manually-defined production rules.

In another context, event recognition in meetings using layered-HMMs is proposed by Oliver *et al.* [17]. The HMMs operate in parallel at different levels of data granularity which allow event classification using multi-modal features. An integrated system for modelling and detecting both high- and low-level behaviours is demonstrated by Nguyen *et al.* [16]. The system uses the trajectories of occupants in a room consisting of pre-defined multiple cells in a given zone. The goal is to recognize behaviours that differ in the occupied cells and in the sequence of their occupation.

In most of the above-mentioned models: 1) object trajectories in the image plane are used as a feature descriptor. However, tracking algorithms often fail to detect and track objects efficiently due to variations in workspace settings, occlusions as well as dynamic or cluttered background. We partially address this issue by using spatio-temporal relational configurations of the objects involved. 2) The models take into consideration a limited number of objects at a given time. The complexity of the learning algorithm increases with the involvement of more objects or interactions, thereby hindering 'real-time' monitoring. This is overcome by the proposed probabilistic *Latent Semantic Analysis* (pLSA). 3) Additionally, we employ view-invariant relational feature for our model whereas view-dependent features are used in most models [22, 18, 17].

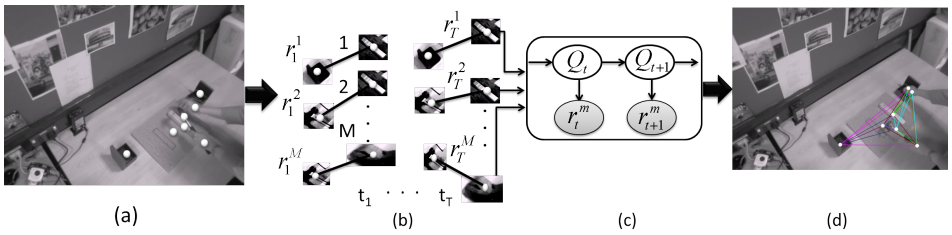

Fig. 1: Workflow monitoring model overview: a) tracked objects in a workspace, b) pair-wise relational feature, c) state space representation of each pair-wise relations, and d) reflections of pair-wise relations (state space) in the workspace.

3 Qualitative Relations to Workflow Patterns

The proposed model for workflow activity monitoring comprises of four steps. The systematic procedure for this is shown in Fig. 1. In the first step, the relevant objects in a given workspace are tracked. The tracking system provides instantaneous 3D positions of objects of interest at each time frame (Fig. 1a). Secondly, a view-invariant relational feature vector for each pair of objects for each time point, is computed (Fig. 1b). In the third step, these relations are quantised into a finite number of states using an HMM (Fig. 1c and Fig. 1d). In the final step, the framework uses a generative process of pLSA for monitoring and recovering workflow activity from the relational configuration of quantised pair-wise relations as shown in Fig. 1d.

3.1 Feature for Qualitative Spatial Relations

Our model is based on the joint motion of a collection of N *key objects* relevant to the task at hand. Let $(\mathbf{x}_t^1, \mathbf{x}_t^2, ..., \mathbf{x}_t^N)$ be the respective 3D positions of these objects at time t, where $\mathbf{x}_t^i = (x, y, z)_t^i$. The joint motion is described in a view-invariant fashion as the set of spatial and kinematic relations between every pair of *key objects*. At each time step, the relation between a pair of objects i and j is represented by a real valued vector composed of the separation and the first derivative of separation with respect time *i.e.* $\mathbf{r}_t^{i,j} = (d_t^{i,j}, \dot{d}_t^{i,j}) \in \Re^2$, where $d_t^{i,j} = \|\mathbf{x}_t^i - \mathbf{x}_t^j\|$ for $\forall i < j$. For convenience, we order the set of pair-wise relations $\{\mathbf{r}_t^{i,j}, i < j\}$ and express as $R = [\mathbf{r}_t^m]_{M \times T \times 2}$, where $m = 1 \ldots M$ and $M = N(N-1)/2$ and T is the number of time steps. We now discretise the pair-wise feature vectors using an HMM to capture the temporal dependencies, and after discretisation it will be represented by corresponding HMM states $S = [s_t^m]_{M \times T}$.

3.2 State Space Representation of Spatial Relations

The state space $S = [s_t^m]$ representation of the corresponding relational feature set $R = [\mathbf{r}_t^m]$ is carried out using an HMM (Fig. 1c). This is defined as a quintuple (Q, R, π, A, B), where Q is a finite non-empty set of 'relational' states, R is the

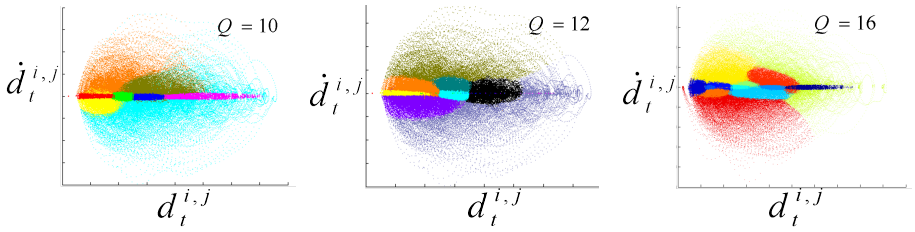

Fig. 2: State space (Viterbi path) representation of pair-wise relations using an HMM of 10, 12 and 16 states respectively (from left). Each colour represents a particular state in the HMM.

input relational feature, $\pi = \{\pi_q\}$ is the starting probability for an element $q \in Q$, $A = \{a_{q,q'}\}$ are the state transition probabilities from the state q to state q' and $B = \{b_q(\mathbf{r}) = N(\mathbf{r}, \mu_q, \sum_q)\}$ is the output function, which is represented as a Gaussian density with mean vector μ_q and covariance matrix $\sum_q$ for the state q emitting feature $\mathbf{r}$. The optimal parameter $\lambda^* = (\pi^*, A^*, B^*)$ of the HMM is estimated using Baum-Welch forward-backward algorithm [3] from a training dataset consisting of $\mathcal{W}$ workflow sequences, where each workflow sequence is represented by M parallel sequences of pair-wise relational features:

$$\begin{aligned} \lambda^* &= \underset{\lambda}{\operatorname{argmax}} \prod_{w=1}^{\mathcal{W}} \prod_{m=1}^{M} P(\mathbf{r}^{m,w}|\lambda) \\ P(\mathbf{r}^{m,w}|\lambda) &= \sum_{\text{all } Q} P(\mathbf{r}^{m,w}|Q,\lambda)P(Q|\lambda) = \sum_{\text{all } Q} \prod_{t=2}^{T} \pi_q b_{q_t}(\mathbf{r}_t^{m,w}) a_{q_{t-1},q_t} \end{aligned} \tag{1}$$

where $\mathbf{r}^{m,w}$ denotes the m^{th} series of pair-wise relational features from the w^{th} workflow sequence and consists of T time steps. The Viterbi algorithm [24] is used to find the most likely hidden states sequence from a given observed sequence of relational feature using the optimal parameter λ^*. Fig. 2 demonstrates the pair-wise relations with the varying number of states Q in the HMM.

3.3 pLSA for Modeling Spatial Relations

A workflow sequence can be decomposed into multiple sub-sequences. The decomposition granularity often depends on type of the workflows and the methods used for its realisation. In the case of 'hammering nail' and 'driving screws' workflows, we use a set of *primitive events* (Table 1) which have been manually annotated. The generative model of pLSA is used for this multi-class classification problem instead of discriminative classification techniques such as support vector machine *i.e.* SVM.

Probabilistic latent space models [10, 4] were initially proposed to automatically discover the recurrent themes or topics from a corpus of text documents. They are used to analyze topic distributions of documents and word distributions in a topic. The model is estimated from the co-occurrence of words and

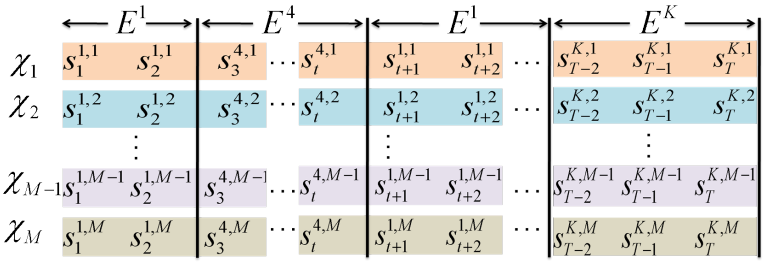

Fig. 3: Partition of a workflow sequence into *primitive events* E^p, pair-wise relations χ_m and spatial relations $s_t^{p,m}$. In [4], these are equivalent to 'corpus', 'document' and 'word' respectively.

documents. In our work, we extract this by dividing a workfow into subsequences of *primitive events* ($E^{p=1\ldots K}$ in Fig. 3). Each pair-wise relation in a *primitive event* is represented by 'document' ($\chi_{m=1\ldots M}$ in Fig. 3). Every quantised relation ($s_t^{p,m}$) in a pair-wise relation is characterised by 'word'. In our framework, each *primitive event* symbolises a corpus and is modelled separately using a pLSA, namely a 'corpus-model'.

The spatio-relational structure of key objects changes over time with the progress of a *primitive event*. In our 'corpus-model', those underneath relational structures for that *primitive event*, are captured by the distribution of latent variable, which is known as 'topics' in the pLSA models. The latent variable itself characterised by probability distribution over the relational states in each pair-wise relation belongs to the *primitive event*. Therefore, we use all the instances of a given *primitive event* from all training sequences to train the corresponding 'corpus-model'. During evaluation of an unseen workflow sequence, the model uses a sliding window of duration $\mathcal{T}$ and decides its association with the seen *primitive events* based on maximum posterior probability.

We begin with some notations for our 'corpus-model'. A workflow sequence is a collection of K *primitive events* represented by $E = \{E^{p=1\ldots K}\}$. A *primitive event* is a group of M parallel pair-wise relations indicated by $\chi = \{\chi_{m=1\ldots M}\}$. The m^{th} pair-wise relation in the p^{th} *primitive event* $\chi_m^p = \{s_{t=1\ldots\tau}^{p,m}\}$ is a sequence of τ quantised spatial relations, where $s_t^{p,m} \in Q$ (Fig. 3). In fact, in a given *primitive event*, all pair-wise relations will have the same number of quantised spatial relations *i.e.* the same τ for $\forall m$. The pLSA-model parameter for each *primitive event* is learned from the training examples. This is done by considering all instances of the same *primitive event* appearing in all the training sequences (Fig. 3). For convenience, from here onwards χ^p represents the collection of M pair-wise relations and the corresponding quantised spatial relations s^p for all instances of *primitive event* E^p appeared in the training sequences.

For each *primitive event* E^p, our aim is to find the joint distribution $P_p(\chi^p, s^p)$ between the pair-wise relations χ^p and spatial relations s^p belonging to the E^p ($p = 1\ldots K$). This is done by using a latent variable model for general co-occurrence of χ^p and s^p which associates an unobserved class variable $z^p = \{z_1^p, z_2^p, \ldots, z_Z^p\}$ [10]. The model assumes the conditional independence of χ^p

and s^p given a latent variable z^p. The graphical representation of our pLSA is shown in Fig. 4. The joint probability $P_p(\chi^p, s^p)$ can be expressed as:

$$P_p(\chi^p, s^p) = P_p(\chi^p)P_p(s^p|\chi^p) \tag{2}$$

$$\text{where, } P_p(s^p|\chi^p) = \sum_{k=1}^{Z} P_p(s^p|z_k^p)P_p(z_k^p|\chi^p) \tag{3}$$

The conditional probabilities $P_p(s^p|z_k^p)$ and $P_p(z_k^p|\chi^p)$ are learned using the EM algorithm [8] by maximizing the following log-likelihood function:

$$L_p = \sum_{\chi^p} \sum_{s^p} n(\chi^p, s^p)\log(P_p(\chi^p, s^p)) \tag{4}$$

where the E-step is shown as:

$$P_p(z^p|\chi^p, s^p) = \frac{P_p(s^p|z^p)P(z^p|\chi^p)}{\sum_{z^{p'}} P_p(s^p|z^{p'})P(z^{p'}|\chi^p)} \tag{5}$$

and the M-step is:

$$P_p(s^p|z^p) = \frac{\sum_{\chi^p} n(\chi^p, s^p)P_p(z^p|\chi^p, s^p)}{\sum_{r^p}\sum_{s^{p'}} n(\chi^p, s^{p'})P_p(z^p|\chi^p, s^{p'})} \tag{6}$$

$$P_p(z^p|\chi^p) = \frac{\sum_{s^p} n(\chi^p, s^p)P_p(z^p|\chi^p, s^p)}{n(\chi^p)} \tag{7}$$

where $n(\chi^p, s^p)$ is the number of co-occurrences of the spatial relation s^p and the pair-wise relations χ^p in the *primitive events* E^p. The proposed 'corpus-model' computes the joint distribution $P_p(\chi^p, s^p)$ for each E^p $(p = 1 \ldots K)$ by considering the temporally segmented subsequences representing the corresponding *primitive events* in the training *dataset* of workflow sequences (Fig. 3). During recognition of an unknown workflow sequence, the co-occurrences matrix of $n(\hat{\chi}, \hat{s})$ is computed by using a sliding window of duration $\mathcal{T}$ over it. At each time step, the likelihood of co-occurrences matrix $n(\hat{\chi}, \hat{s})$ with respect to each *primitive event* E^p is computed using the joint-distribution $P_p(\chi^p, s^p)$ of E^p via Eqn. 4. The unknown sliding window at each time step is assigned a *primitive event* $e^* = \text{argmax}(L)$, where $L = \{L_1, L_2, \ldots, L_K\}$ is the measured likelihood from all *primitive events*.

3.4 Activity Monitoring

For workflow activity monitoring, the model is not only for the recognition of ongoing activity but also for advising the agent on the next possible tasks. In order to achieve this, a top-level workflow topology is required. Often, this top-level topology is provided manually for a well-defined structured workflow [21, 15]. We achieve this by modelling event spaces with an HMM. The graphical structure is shown in Fig. 4. The monitoring-HMM consists of K hidden states

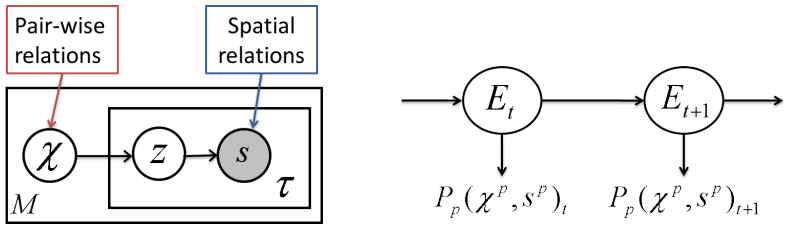

Fig. 4: The Generative pLSA model (left) and workflow monitoring-HMM (right)

denoting K *primitive events*. The observation likelihood for each hidden state E_t at time t is computed from the respective *primitive event's* likelihood via co-occurrence matrix $n(\chi^p, s^p)_t$ through a sliding window of duration $\mathcal{T}$.

$$P_p(n(\chi^p, s^p)_t | E_t) = \prod_{\chi^p} \prod_{s^p} P_p(\chi^p, s^p)^{n(\chi^p, s^p)_t} \tag{8}$$

We are interested in the transition probabilities from state E_t to state E_{t+1}; these are estimated via the Baum-Welch forward-backward algorithm [3] from the training sequences.

Our model can also be readily used for abnormal behaviour detection while monitoring a workflow. This can be achieved via examining the observation likelihood (Eqn. 8) of the ongoing activities. A lower score of this likelihood indicates higher abnormality of ongoing activities.

3.5 Handling of Occlusions

In general, the conventional HMM-based model faces difficulties in finding the most likely state sequences for missing observations. Therefore, a continuous most likely state sequences is not re-established once the observations reappear after a certain duration. A bottom-level HMM is used for the quantisation of pair-wise relations (section 3.2) and another top-level HMM is for monitoring workflows.

The quantisation HMM successfully handles the occlusions by treating the reappeared pair-wise relational observations $\mathbf{r}_t^m$ as a new sequence with a new starting point from the time it reappeared. For these reappeared sub-sequences, the model enforces the uniform starting probability $\pi = \{\pi_q\}$ of the HMM parameter $\lambda = (\pi, A, B)$ (section 3.2). As mentioned earlier, each pair-wise relational feature sequence belonging to a workflow sequence is treated separately for the quantisation. Therefore, the state space representation of pair-wise relation sequences corresponding to the observed objects are not affected by other occluded objects.

The monitoring HMM tackles occlusions by taking advantage of the pLSA, which uses the co-occurrence matrix $n(\chi^p, q^p)$ to consider the occurrence frequency of quantised spatial relations in a pair-wise relation. In the event of an occlusion, pLSA masks off spatial relations corresponding to the occluded object.

1. Grab nail baton	2. Place nail baton within marked region	3. Release nail baton	4. Grab hammer
5. Retrieve nail	6. Insert nail	7. Place hammer	8. Hammering nail
9. Release nail	10. Put down hammer	11. Grab screws baton	12. Placed screw baton within marked region
13. Release screw baton	14. Pick screwdriver	15. Retrieve screw	16. Insert screw
17. Release screw	18. Move screwdriver	19. Switch on screwdriver	20. Push down screwdriver
21. Turn off screwdriver	22. Put down screwdriver	23. Unknown	

Table 1: *Primitive events* for 'hammering nails' and 'driving screws' workflow sequences

4 Experiments

Our experimental datasets consist of two type of workflow sequence, 1) hammering 3 nails and 2) driving 3 screws. Two individuals are used to carry out the workflows on a bench. The sequences are captured using the vicon motion capture system [23]. Vicon markers are placed on all *key objects* utilized in the workflow including both wrists of the participants. This dataset consists of 9 objects (hammer, electric screwdriver, nail box, screw box, nail baton, screw baton, left wrist, right wrist and a piece of wood). The workflows are carried out on the workflow bench. Given the tools above, the user is asked to hammer 3 nails and drive 3 screws into the respective nail and screw batons. Using the setup above, a total of 16 (4 per participant per workflow) sequences are obtained. The vicon system provides the output at 50 Hz and 6 *DoF* (3D positions and orientations) for each tracked object while performing a task.

4.1 Evaluations

A total of 23 *primitive events* (Table 1) are identified for the 'hammering nails' and 'driving screws' workflows including an 'Unknown' event for time steps those are not labeled. We evaluated our approach for both off-line and on-line recognition. The off-line evaluation considers the whole workflow sequence for the recognition. The on-line evaluation takes into account the samples from the beginning until time step t, where $t = \{2, 3, \ldots, T\}$ and T is the total duration of the workflow sequence.

The frame-wise recognition rate is compared with the baseline approaches. The baseline evaluations use input as the 3D motion vectors $\mathbf{v}_t^o = (\dot{x}, \dot{y}, \dot{z})_t^o$ for individual object $o = 1, \ldots, N$ at each time step t. The final motion vector $\boldsymbol{v}_t = (\mathbf{v}_t^1, \mathbf{v}_t^2, \ldots, \mathbf{v}_t^N)$ at a given time t is a single vector by stacking the individual motion vector. In this experiment, the length of $\boldsymbol{v}_t$ is 27 for 9 objects. We compare our approach with HTK-PaHMM [28, 25], SVM-Multiclass [20, 7] and pLSA 'topic-model' [26, 10].

In the HTK-PaHMM model, there are 23 parallel-HMM representing 23 *primitive events* in workflow sequences. Each HMM is trained separately with subsequences of corresponding *primitive events* from training workflow sequences. We use the HTK-toolkit [28] for this model.

Methods	Off-line	On-line
HTK-PaHMM	77.40%	12.20%
SVM-Multiclass	24.90%	24.90%
pLSA	36.84%	36.84%
M-HMM-pLSA	61.51%	61.10%

Table 2: Performance comparison for leave-one-out experiment

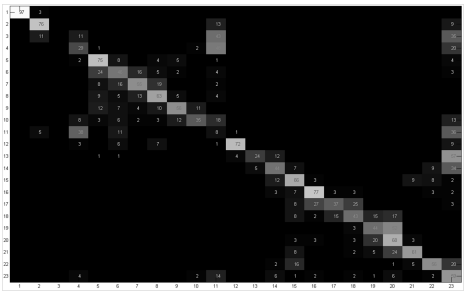

Fig. 5: Confusion matrix for the frame-wise evaluation of 23 *primitive events* for the leave-one-out experiment (off-line)

For the SVM-Multiclass representation, each *primitive event* is treated as a class. A normalized $[-1, 1]$ 3D motion vector $\boldsymbol{v}_t$ at each time step t is used as a input feature. As in HTK-PaHMM, the model is trained on the training dataset comprising subsequences of *primitive events* using RBF-kernel. However, the temporal dependency of $\boldsymbol{v}_t$ is not considered. During testing of a workflow sequence, the class label of an unknown $\boldsymbol{v}_t$ at time t is inferred from the learnt model.

For the pLSA 'topic-Model', the input motion vectors $\boldsymbol{v}_t$ are represented as a word $w = \{w_1, w_2, \ldots, w_K\}$ by quantising it using k-means clustering algorithm. Each *primitive event* symbolises a topic $z = \{z_1, z_2, \ldots, z_{23}\}$. In [26], 'topic-model' is used for finding topics or themes corresponding to activities those are frequently occurring in a scene. In our model, we know these topics (*primitive events*) from the labelled workflow sequences. For each topic p, we compute $P(w|z_p)$, $p = 1 \ldots 23$ by counting the occurrence frequency of w. For an unknown document d, we assign a topic $z^* = argmax_z(P(z|d))$, where $P(z|d)$ is estimated using the procedure described in [10] without changing $P(w|z)$. For this model, document d is represented as a sequence of words w taken from a sliding window of duration $\hat{\mathcal{T}}$ (1 sec in this evaluation). This model gave better performance on our dataset for 100 clusters.

The performance of frame-wise comparison for the leave-one-out experiment on 16 workflow sequences is shown in Table 2. The HTK-PaHMM model performed better for the off-line evaluation. However, it gave very poor outcome for the on-line. SVM-Multiclass and 'topic-model' do not consider temporal dependency and performed reasonably well. Our HMM-pLSA 'corpus-model' gave the best performance over all. The confusion matrix of our model for 23 *primitive events* is shown in Fig. 5. The confusion matrix reveals that some frames in the current *primitive event* are misclassified as either next or previous *primitive events*. This is typical synchronisation error as ground-truth for the evaluation is manually annotated.

Object trajectories captured in our motion capture system are reasonably clean in comparison to vision-based tracking. In order to validate the robustness of our approach, we injected random Gaussian noise of zero mean with varying

Noise level	Inserted noise during training and testing		Inserted noise during testing only
	Off-line	On-line	Off-line
No noise $\sigma = 0$	61.51%	61.10%	61.51%
$\sigma = 4$	49.97%	48.90%	40.93%
$\sigma = 10$	52.00%	51.20%	34.56%
$\sigma = 15$	51.68%	50.40%	32.44%
$\sigma = 20$	50.95%	49.57%	28.44%

Table 3: Performance comparison of our model for leave-one-out experiment with the insertion of random noise to 1) both training and testing workflow sequences, 2) only testing sequences

Noise level	Inserted noise during training and testing		Inserted noise during testing only	
	Test on P_1	Test on P_2	Test on P_1	Test on P_2
No noise $\sigma = 0$	53.21%	59.31%	53.21%	59.31%
$\sigma = 4$	52.24%	48.86%	42.92%	52.10%
$\sigma = 10$	51.59%	53.39%	46.59%	08.63%
$\sigma = 15$	53.53%	54.13%	39.80%	12.42%
$\sigma = 20$	48.77%	52.23%	27.68%	12.15%

Table 4: Inter participants off-line performance comparison with random noise inserted in 1) both training and testing workflow sequences, 2) only testing sequences

standard deviation $\sigma = \{4, 10, 15, 20\}$ in centimeters to the 3D positions of objects in our workflow sequences. The frame-wise evaluations for both on-line and off-line is presented in Table 3. The declining performance is less than 12% for $\sigma = 20$ centimeters in both off-line and on-line experiments, when noise is inserted into both training and testing sequences.

In our dataset, two participants P_1 and P_2 carried out an equal number of workflows. We evaluated our method with workflows carried out by one participant in training and the rest for testing, and vice versa. The performance of frame-wise evaluation is shown in Table 4. Surprisingly, performance is comparable in most cases, although there is a large deterioration in performances for higher added noise levels in test data only and with training on a single participant.

4.2 Evaluation of Occlusions

The Vicon motion capture system [23] provides relatively clean data w.r.t. visual analysis and is not enough to validate our hypothesis about handling occlusions. Therefore, we evaluated our approach by removing one or more objects from the testing workflow sequences, whereas the model was trained on sequences by considering all objects. The average performance of complete removal of an individual object in testing sequences and a leave-one-out experiment is shown in Table 5. Removing static objects such as 'wood piece', 'nail box' and 'screw box', the drop off in performance is less than 1%. However, the model gave encouraging performance to the occlusion of actively involved objects such as 'hammer', 'screwdriver', 'left wrist' and 'right wrist' (Table 5). We, then evaluated our model by removing two or more objects from the testing sequences. In this evaluation, while removing two or more objects all possible combinations of objects are considered and the average performance is shown in Fig. 6. The method gave accuracy $> 50\%$ for the complete occlusion up to two objects.

Occluded objects	Off-line (Average)
screwdriver	58.61%
wood piece	60.98%
nail baton	56.03%
hammer	48.52%
nail box	59.90%
screw baton	55.11%
screw box	61.49%
left wrist	51.66%
right wrist	55.27%

Table 5: Leave-one-out experiment with complete occlusion of an object in the testing sequences. The performance is 61.51% without occlusion

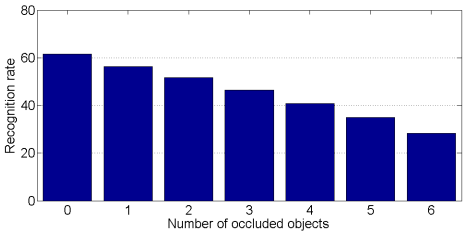

Fig. 6: Recognition performance (off-line) for the leave-one-out experiment with complete occlusion of 0-6 objects.

5 Conclusion

In this work, we proposed an innovative approach for real-time monitoring of workflows. The proposed method uses a novel view-invariant qualitative spatial feature, which is extracted by considering distance and rate of change of distance between a pair of objects in 3D space. The dynamics of this pair-wise relational feature is captured using an HMM. Realisation of workflows from the relational state space is carried out using a 'corpus-model', which is derived from probabilistic Latent Semantic Analysis (pLSA). Each *primitive event* in a workflow is modeled separately using our 'corpus-model'. In order to predict the next possible *primitive event*, the approach uses a monitoring-HMM.

Acknowledgments. We would like to thank Dima Damen and Andrew Gee (Department of Computer Science, University of Bristol) for providing the Vicon dataset. The work is supported by EU grant COGNITO (www.ict-cognito.org, ICT-248290). We thank Elizabeth Carvalho (Center for Computer Graphics, Portugal) for supplying ground-truth for this dataset.

References

1. A.Ivanov, Y., Bobick, A.F.: Recognition of visual acticities and interactions by stochastic parsing. IEEE Trans. on PAMI 22(8), 852–872 (2000)
2. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM 26(11), 832 – 843 (1983)
3. Baum, L.E., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics 41(1), 164–171 (1970)
4. Blei, D.M., Ng, A., Jordan, M.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (March 2003)
5. Brand, M., Oliver, N., Pentland, A.: Coupled Hidden Markov models for complex action recognition. In: Proc. of IEEE CVPR. pp. 994–999 (1997)

6. Bui, H., Venkatesh, S., West, G.: Policy recognition in the abstract hidden markov model. Journal of Artificial Intelligence Research 17, 451–499 (2002)
7. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), http://www.csie.ntu.edu.tw/~cjlin/libsvm
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Royal Statistical Society. 39(1), 1–38 (1977)
9. Grimson, W.E.L., Stauffer, C., Romano, R., Lee, L.: Using adaptive tracking to classify and monitor activities in a site. In: Proc. of IEEE CVPR. pp. 22–29 (1998)
10. Hofmann, T.: Probabilistic latent semantic analysis. In: Proc. of Uncertainty in Artificial Intelligence, UAI99. pp. 289–296 (1999)
11. Hongeng, S., Nevatia, R.: Multi-agent event recognition. In: Proc. of ICCV. vol. 2, pp. 84–91 (2001)
12. Intille, S.S., Bobick, A.F.: A framework for recognizing multi-agent action from visual evidence. In: AAAI-99. pp. 518–525 (1999)
13. Jaeger, H.: The "echo state" approach to analysing and training recurrent neural networks. Tech. rep., Fraunhofer Institute for Autonomous Intelligent Systems (December 2001)
14. Johnson, N., Hogg, D.C.: Learning the distribution of object trajectories for event recognition. Image Vision Comput. 14(8), 609–615 (1996)
15. Moore, D., Essa, I.: Recognizing multitasked activities from video using stochastic context-free grammar. In: Proc. AAAI National Conf. on AI. pp. 770–776 (2002)
16. Nguyen, N., Phung, D., Venkatesh, S., Bui, H.: Learning and detecting activities from movement trajectories using the hierarchical hidden Markov model. In: Proc. of IEEE CVPR. vol. 2, pp. 955 – 960 (2005)
17. Oliver, N., Garg, A., Horvitz, E.: Layered representations for learning and inferring office activity from multiple sensory channels. Computer Vision and Image Understanding 96(2), 163–180 (2004)
18. Padoy, N., Weinl, D.M.D., Berger, M.O., Navab, N.: Workflow monitoring based on 3D motion features. In: Proc. of ICCV Workshop on Video-oriented Object and Event Classification (2009)
19. Pinhanez, C., Bobick, A.: Human action detection using pnf propagation of temporal constraints. In: Proc. of IEEE CVPR (1998)
20. Schldt, C., Laptev, I., Caputo, B.: Recognizing human actions: A local svm approach. In: Proc. ICPR. pp. 32–36 (2004)
21. Shi, Y., Huang, Y., Minnen, D., Bobick, A., Essa, I.: Propagation networks for recognition of partially ordered sequential action. In: Proc. of CVPR. vol. 2, pp. 862–869 (2004)
22. Veres, G., Grabner, H., Middleton, L., Gool, L.V.: Automatic workflow monitoring in industrial environments. In: Proc. of ACCV (2010)
23. Vicon Systems: www.vicon.com
24. Viterbi, A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. on Information Theory 13(2), 260–269 (1967)
25. Vogler, C., Metaxas, D.: Parallel Hidden Markov Models for American sign language recognition. In: Proc. of IEEE ICCV. vol. 1, pp. 116–122 (1999)
26. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. IEEE Trans. PAMI 31, 539–555 (2009)
27. Wanstall, B.: HUD on the Head for Combat Pilots. In: Interavia 44. pp. 334–338 (April 1989)
28. Young, S.J.: The htk hidden Markov model toolkit: Design and philosophy. Tech. rep., Cambridge University Engineering Department (September 1994)