Temporal Relations in Videos for Unsupervised Activity Analysis

Fabian Nater[1]
fnater@vision.ee.ethz.ch
Helmut Grabner[1]
grabner@vision.ee.ethz.ch
Luc Van Gool[1,2]
vangool@vision.ee.ethz.ch

[1]Computer Vision Laboratory
ETH Zurich, Switzerland
[2]ESAT-PSI / IBBT
K.U. Leuven, Belgium

Figure 1: In videos, each frame strongly correlates with its neighbors. Our approach exploits this fact and enables the segmentation of the video and the interpretation of unseen sequences.

Abstract

Temporal consistency is a strong cue in continuous data streams and especially in videos. We exploit this concept and encode temporal relations between consecutive frames using discriminative slow feature analysis. Activities are automatically segmented and represented in a hierarchical coarse to fine structure. Simultaneously, they are modeled in a generative manner, in order to analyze unseen data. This analysis supports the detection of previously learned activities and of abnormal, novel patterns. Our technique is purely data-driven and feature-independent. Experiments validate the approach in several contexts, such as traffic flow analysis and the monitoring of human behavior. The results are competitive with the state-of-the-art in all cases.

1 Introduction

The analysis of activities from videos is important to solvemay and diverse tasks (see [6, 10] for surveys). In most systems expert knowledge is required to train specific models with labeled data. Arguably, a one-time training process cannot anticipate all the possible activities, and the monitored setups may vary considerably. Hence, recent research tries to build or adapt such models automatically and in an unsupervised manner.

In previous works, human actions [17] or surveillance scenes [4, 8, 18] are analyzed automatically for the extraction of *topics* from spatio-temporal words. Their goal is to find correlated motion in order to segment behavior in space and time. Other approaches to video summarization [15, 22] cluster video streams into repeated activities. Trained models

can further be used to analyze unseen behavior. In such approaches, abnormal events are often detected as outliers. This has been successfully applied to traffic monitoring [4, 6], the surveillance of public places [1], assisted living [11] or the analysis of motion patterns [14]. However, these methods often suffer from either (i) strong constraints which limit their use to specific applications, (ii) the need for prior knowledge (*e.g.*, the number of activities) and/or, (iii) being too abstract for easy interpretation.

In order to overcome these limitations, we seek for an 'invariant characteristic' that can underpin generic model building and reasoning. Observing the different sequences in Fig. 1, increments between frames are quite small compared to the changes throughout the whole sequence. For instance, the behavior of a tracked person (2^{nd} row) is composed of a certain repertoire of activities with transitions in between that are typically short in comparison. This can also be observed at larger scales, like day-night changes or seasonal changes (3^{rd} and 4^{th} row) and already suggests a hierarchical structure.

The contributions of this paper are twofold:

- We propose an unsupervised technique to segment the data into compact and meaningful activities. To this end, we explore the strong temporal relations in the video (Sec. 2). The automatically discovered activities are efficiently represented and continuously refined in a hierarchical manner (Sec. 3).
- Analysis and interpretation of unseen data is demonstrated as a result of the coarse to fine representation in the hierarchy that enables abnormal event detection (Sec. 4). Anomalies can be spotted, such as the big tent in a street festival (3^{rd} row in Fig. 1).

Experimental results, presented in Sec. 5 for different video surveillance scenarios, show the usefulness and generality of the technique. We demonstrate activity segmentation, the surveillance of public places, as well as the detection of abnormalities in indoor scenarios.

2 Activities in data streams

Due to the large variety of observations in a data stream, it often is difficult to build a single model which describes the data and its dynamic behavior precisely. In this work, we automatically split the data stream into meaningful subsequences. We call these subsequences *activities*. If they are consistent and have low complexity, they can be represented more easily and precisely. This principle is exploited by arranging the video data in a hierarchical manner as outlined in Fig. 2. In a long data-stream, some activities may be very distinct

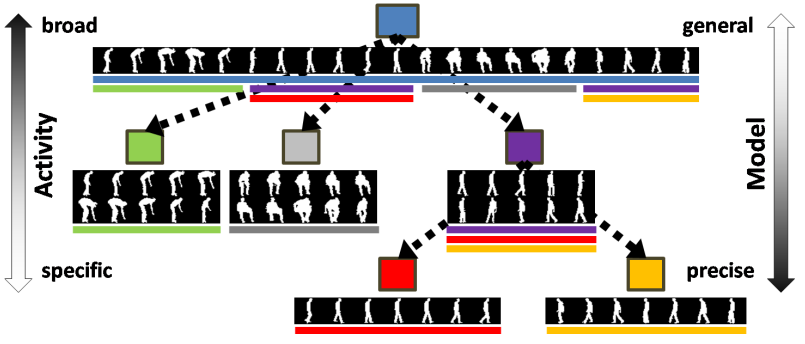


Figure 2: Overview of the proposed hierarchical model that splits and represents the data in a coarse to fine manner. As an example, we consider indoor actions. At the top node, the entire video stream is taken into account, while at lower levels, more specific concepts, like picking up, or walking leftwards are found.

and can be segmented high up, while more subtle differences only appear deeper down. The concept is similar to *motion segments* in [13] or *micro-actions* in [11], but we do not restrict ourselves to human actions.

In order to build up such a hierarchy, we exploit the strong link between temporally adjacent observations in videos. Hence, activities are characterized to have a certain duration, to be observed frequently, and to be interconnected by shorter transitions. In other words, *with high probability, neighboring frames share their activity label.* The advantages of our approach are:

Definition of activities. Activities are automatically explored from their temporal characteristics based on discriminative modeling techniques. No prior knowledge on the boundaries or the total number or activities is required.

General vs. specific. The dilemma between generalization capacity and precision of the model is naturally handled in the hierarchy. Nodes higher up in our hierarchical model are general and represent a broad variety of activities (*e.g.*, 'an object is moving'), whereas lower nodes only incorporate very specific activity patterns (*e.g.*, 'a person walking to the right').

Interpretation. If the model is applied to new, unseen data at runtime, the search through the hierarchy is not only more efficient, it also allows conclusions about the nature of the unseen data. In particular, a new observation can either be assigned to a known activity or is recognized as outlier at a certain level in the hierarchy.

In the following section, we show how we establish such a hierarchical activity model.

3 Activity summarization

Our approach is inspired by the principle of invariant or slowly varying features. Wiskott and Sejnowski [21] have proposed Slow Feature Analysis (SFA) as an unsupervised learning technique for continuous data streams, inspired by human learning capacities. Recently, Klampfl and Maass [7] have shown that SFA yields the classification capacities of Fisher's Linear Discriminant, if temporally adjacent samples in the data stream are likely to belong to the same class. This requirement is fulfilled in our setting, as we analyze continuous streams of images and assume that activities therein are performed over a certain time span.

Given an image stream, $S = \{I_1, I_2, \ldots, I_T\}$ of T images, $I_t \in \mathbb{R}^{n \times m}$, each image I_t is represented by a D-dimensional feature vector $\boldsymbol{f}_t \in \mathbb{R}^D$. As our experiments will show, various feature representations can be used.

3.1 Data segmentation

In the segmentation step, the goal is to split the data stream into its composing activities. A broader set of activities is partitioned into subsets.

Slow Feature Analysis. The output signal $\boldsymbol{z}_t$ of the Slow Feature Analysis represents the slowest components in $\boldsymbol{f}_t$, *i.e.*, it minimizes the average temporal variation:

$$\min J_{SFA} = \min \mathbb{E}_t(\Delta \boldsymbol{z}_t), \text{ where } \Delta \boldsymbol{z}_t = ||\boldsymbol{z}_t - \boldsymbol{z}_{t-1}||^2. \tag{1}$$

To avoid the trivial solution $\boldsymbol{z} \equiv 0$, additional constraints for zero mean and unit variance are introduced. Multiple slow features need to be decorrelated and they are ordered by decreasing *slowness*.

Let $\boldsymbol{y}_t = \boldsymbol{f}_t - \mathbb{E}_t(\boldsymbol{f}_t)$ be the zero-mean feature vector. Considering only linear functions of the form $\boldsymbol{z} = \boldsymbol{w}^\top \boldsymbol{y}$, it can be shown [20] that the objective becomes

$$\min J_{SFA}(\boldsymbol{w}) := \frac{\boldsymbol{w}^\top \dot{\boldsymbol{D}} \boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{D} \boldsymbol{w}}, \tag{2}$$

where $\boldsymbol{D} = \mathbb{E}_t(\boldsymbol{y_t}\boldsymbol{y_t}^\top)$ is the covariance matrix of the data and $\dot{\boldsymbol{D}} = \mathbb{E}_t\big((\boldsymbol{y_t} - \boldsymbol{y}_{t-1})(\boldsymbol{y_t} - \boldsymbol{y}_{t-1})^\top\big)$ the covariance matrix of the temporal differences. The weight vectors $\boldsymbol{w}$ which minimize Eq. (2) are the solutions to the generalized eigenvalue problem $\dot{\boldsymbol{D}}\boldsymbol{w} = \lambda \boldsymbol{D}\boldsymbol{w}$. The slowest varying components in $\boldsymbol{y}$ are their projections onto the eigenvectors $\boldsymbol{w}$ associated to the smallest eigenvalues λ [20].

Clustering. In the SFA subspace, distinct activities are discriminatively mapped to distinct high density regions with sparse transitions [7]. Hence, we apply Gaussian Mixture Model (GMM) clustering to separate the activities. By means of expectation maximization, the regions where the data is densely scattered are found. The cluster index assigned to a data point corresponds to the cluster number with maximal posterior probability [7]. Initialization is done with *k-means*. Since the desired number of clusters is not known a priori, a sweep over k is performed and the sum of posterior probabilities over all datapoints is calculated. The second derivative of this sum characterizes the curvature and we select its maximum as the desired number of clusters. A postprocessing step ensures temporal smoothness and discards very short sequences.

3.2 Building the activity hierarchy

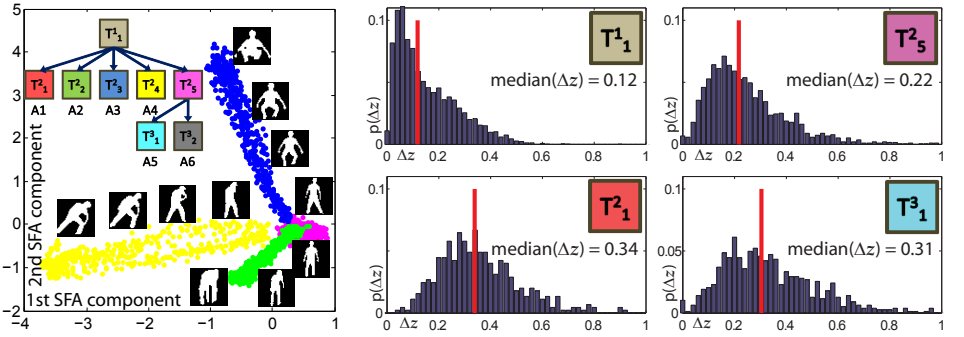
The segmentation is applied recursively on the data. In the first step, we split according to the most dominant (slowest) cues in the entire datastream. In order to create a hierarchy, the segmentation process is repeated for each obtained subset and other discriminative components may now appear. This is encouraged since we keep the dimensionality $d_{SFA} \ll D$ of the SFA subspace fixed. At high levels, the established nodes T_i^j (node i on level j) contain very broad activity concepts while at lower levels in the hierarchy, specific actions are found.

Basic activities. The decision whether or not a node is further refined is based on the representation in the SFA space. The data is projected so that the average distance between consecutive samples is minimized, *c.f.* Eq. (1). If the distances are approximately equal across the whole sequence, the data is well described by its slowest components [21]. In this case, we define a *basic activity* A and the data is not split any further. This corresponds to a leaf node in the hierarchy. On the other hand, if major parts of the data are connected with short distances in the subspace, there must be a few consecutive samples which lie far apart, such that the unit variance constraint is fulfilled. This case is consistent with the assumption of [7], hence, splitting the data is stimulated.

As a simple measure of data compactness, we use the median of distances between consecutive samples in the SFA space. It turns out to be robust against outliers, and reflects well the concept above. If we measure a small median value, the data is further segmented. For a larger median, a basic activity A is detected.

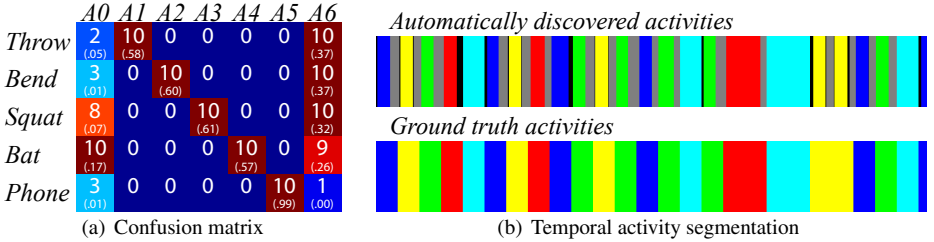
Illustration. To get an intuition, we now discuss our activity detection technique with respect to the dataset from Turaga *et al.* [15] and show how our results compare. We use silhouette data from two views as provided by the authors, apply a distance transform and concatenate the rows to one feature vector. The data exhibits five actions (*throw, bend, squat, bat, pick phone*). Each of them is repeated ten times with different execution speeds. We randomly permutate the actions and the repetitions in order to form the input video.

In Fig. 3(a), the first two dimensions of the clustered SFA subspace ($d_{SFA} = 3$) are displayed. It is obtained at the root node, where all five actions are included. The sketched hierarchy shows that four basic activities are extracted at the first split. The pink node is subdivided further, yielding two more basic activities. In Fig. 3(b) the stopping criterion is verified. The empirical distributions of distances $\Delta\boldsymbol{z}_t$ and their medians are shown. For nodes T_1^1 and T_5^2, the shift of the mode towards the origin suggests to further split these nodes.



(a) Clustering (b) Median criterion

Figure 3: (a) GMM clustering in the SFA subspace, viewed in two dimensions. The resulting activity hierarchy is sketched. (b) Splitting criterion based on the distribution of distances between consecutive samples in the SFA space. For basic activities, the median is higher, and they are not further segmented.



(a) Confusion matrix (b) Temporal activity segmentation

Figure 4: (a) Automatically discovered basic activities ($A0-A6$) *vs.* the ground truth, (b) Color coded labeling for the discovered and the ground truth actions (see text for details).

We automatically discovered six basic activities ($A1-A6$). The samples that were filtered out during clustering (short sequences and outliers) are collected in $A0$. From the results reported in Fig. 4, one can notice that activities $A1-A5$ match the five ground-truth actions as defined by Turaga *et al.* [15]. $A6$ corresponds to standing still, as observed at the beginning and the end of each action, but not annotated in the ground-truth. The confusion matrix in Fig. 4(a) is obtained from the compositions of the ground truth snippets as in [15] and we outperform their results. The proportion of the discovered activities with respect to the total number of frames is reported in brackets. Since standing still is not included in the ground-truth annotation, this difference obviously lowers the values. In Fig. 4(b) the temporal evolution of discovered and ground truth activities are depicted for half of the sequence.

3.3 Data modeling

As we want to use the hierarchy to classify the activities in previously unseen videos, the data underlying each of its nodes is additionally modeled with respect to shape and dynamics. Biological studies on human motion perception suggest that motion analysis is performed from sequences of appearance snapshots [9]. Similarly, we create an extended feature vector $\boldsymbol{v}_t = (\boldsymbol{f}_t, \boldsymbol{f}_{t-1}, \ldots \boldsymbol{f}_{t-n})^\mathsf{T}$ as the concatenation of the last n feature representations, like in [16]. We model the zero-mean feature vector $\boldsymbol{x}_t = \boldsymbol{v}_t - \mathbb{E}_t(\boldsymbol{v}_t)$ by means of PCA.

The data in each node T_i^j in the activity hierarchy is represented with the model M_i^j in a PCA space with a fixed number of dimensions $d_{PCA} \ll D$. When moving down in the hierarchy, the data in each node describes more specific activity concepts. Likewise, the models naturally are more general at the top of the hierarchy and more precise at leaf nodes, as sketched in Fig. 2. At the leaf nodes, each basic activity A is described by model M_A.

4 Analysis of unseen data

We now show how the hierarchical model efficiently detects known activities and signals anomalies that may occur in unseen data.

4.1 Activity detection

Given a new sequence $\boldsymbol{x}'$ and a set of basic activities $\mathscr{A}$, the task is to identify $A \in \mathscr{A}$, which best explains the data. To this end, $\boldsymbol{x}'$ is projected into the PCA subspaces and the reconstruction errors are calculated. The leaf node model M_A with be lowest reconstruction error e_{M_A} determines the discovered activity $A^\star = \arg\min_A e_{M_A}(\boldsymbol{x}')$. The hierarchical arrangement of nodes makes sure that not all PCA models need to be tested, as discussed in the next section.

Simultaneous target localization and activity detection. In certain applications, only a sub-region of the entire scene might be considered. For example, if the actions of a person are analyzed, the features will only describe this person but not the surroundings. In order to correctly detect the performed activity, this sub-region needs to be localized correctly. We opt to integrate the search for an optimal location in the previous formulation for activity detection. At various image locations ρ (including scale), the reconstruction error $e_{M_A}(\boldsymbol{x}'|\rho)$ is determined for the activity A. If we evaluate multiple activities, the optimal location and activity are found simultaneously, *i.e.*, $(\rho^\star, A^\star) = \arg\min_{A,\rho} e_{M_A}(\boldsymbol{x}'|\rho)$. For efficiency reasons and since temporal consistency is assumed, only the local neighborhood of ρ^*_{t-1} (the location at the previous timestep) is scanned. This is usually referred to as *tracking*.

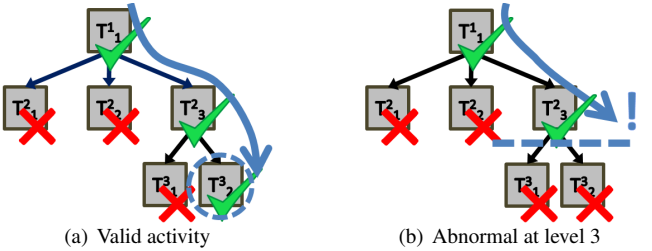
4.2 Exploiting the hierarchy

We now show how the hierarchical model paves the way for a more sophisticated and efficient analysis. Since the hierarchy consists of a set of more general and more specific models, we can apply the anomaly reasoning as proposed in [19]. To this end, we first need to determine if an observation is well described by a certain node in the hierarchy. A node T_i^j with model M_i^j is considered *active* for an observation $\boldsymbol{x}'$ based on its normalized reconstruction error:

$$\text{active}(T_i^j) = \begin{cases} 1 & \text{if } \dfrac{e_{M_i^j}(\boldsymbol{x}') - \mu_{M_i^j}}{\sigma_{M_i^j}} < \theta \\ 0 & \text{otherwise} \end{cases} , \tag{3}$$

where $\mu_{M_i^j}$ and $\sigma_{M_i^j}$ are respectively the mean and the standard deviation of the reconstruction error for model M_i^j, obtained from the training data. θ is a user-defined threshold.

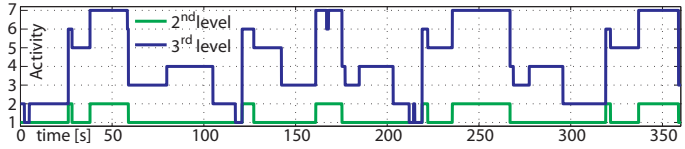
To respect the hierarchy, each observation is propagated from the root node to the leaves as sketched in Fig. 5(a). Only subnodes of active nodes need to be considered, which in-



(a) Valid activity (b) Abnormal at level 3

Figure 5: Use of the hierarchical model for the interpretation of unseen data. (a) A known activity is detected for an active leaf node. (b) A reasoning on abnormal conditions in the hierarchy is deduced from active and inactive nodes on different levels.

(a) Obtained basic activities: Driving left to right, right to left, straight, straight and turn right



(b) Activities over time at the first two levels in the hierarchy

(c) Anomalies: Ambulance, collision course, wrong direction, unseen configuration)

Figure 6: Activities, their temporal variation and detected anomalies for the QMU junction [4]. Regions with high reconstruction error are shaded in red (best viewed in color).

creases the efficiency. As long as the observations are according to expectations, there is always a leaf node (*i.e.* basic activity) which is able to explain the data.

If a more general node validates the observation, but none of its more specific sub-nodes does, then this signals an abnormal activity (Fig. 5(b)). Such abnormality can occur at any level. From the location in the hierarchy where this happens, interpretations about the nature of the abnormality can be made.

5 Experiments

Parameters. At training, an initial noise reduction step is applied to keep 95% of the data variance in each node. Subsequently, SFA and PCA subspaces are modeled with $d_{SFA} = 3$ and $d_{PCA} = 3$ dimensions. For motion encoding, $n = 5$ last frames are used. At test, the threshold $\theta = 3$ is applied for hierarchical reasoning.

Runtime. Due to its low complexity, the analysis is practicable in real-time. On a standard PC, our current matlab implementation runs at more than 12 frames per second. The exhaustive search in the case of target localization slows down the evaluation by approximately a factor of 10. Model building takes in the order of a few minutes for our cases.

5.1 Surveillance of public places

We show how our technique performs on two different visual surveillance datasets using holistic scene descriptors.

QMU junction [4]. (360×288 pixels, 25 fps, 1 hour). This data has previously been used for learning spatio-temporal scene topics [4]. As car and pedestrian flow patterns are of importance in this scene, we apply the motion features proposed in [17] and extract the motion in 18×18 pixel patches. To additionally encode the motion direction, a forgetting rate of 0.95 is applied. Training is done on 50,000 images, the runtime evaluation takes into account all 90,000 frames.

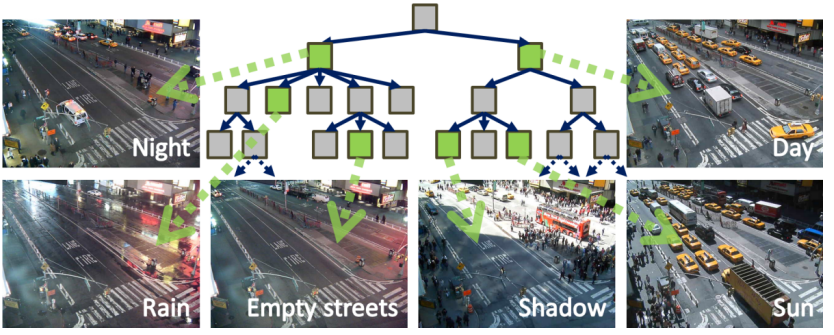


Figure 7: Part of the obtained tree with interpreted activities for the Times Square dataset [3].

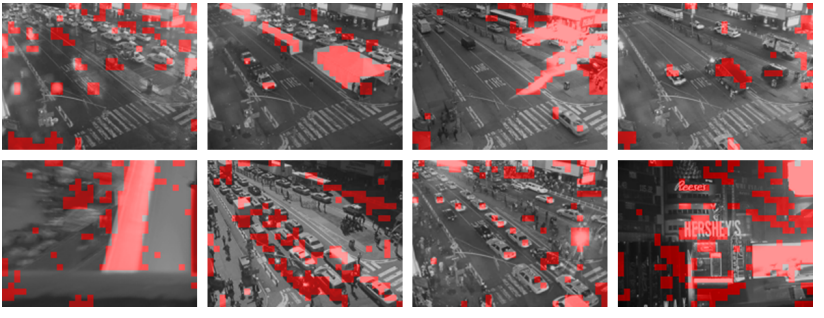


Figure 8: Automatically detected anomalies: Heavy rain, festival tent, shadow shape, parked trucks; camera failure, jam with reflections, strong light, camera moved.

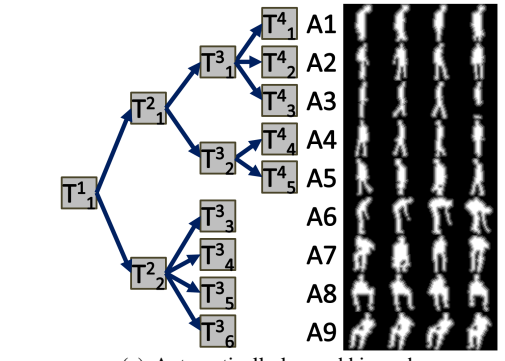
The learning procedure extracts 48 nodes of which 19 are leafs. Some of these discovered basic activities are depicted in Fig. 6(a). The hierarchical analysis nicely groups co-occurring traffic patterns in leaf nodes. Further basic activities summarize streets with only pedestrian motion, cars accelerating, and different turn configurations. In Fig. 6(b) we show the obtained activity segmentation over time, for the second and third level in the hierarchy. Without enforcing any larger scale temporal relations, we discover pseudo-repeated patterns in the data that correspond to different phases in traffic light cycles.

Four examples of abnormal situations are presented in Fig. 6(c), the ambulance and the wrong driving direction have also been reported in [4]. Since we use a holistic scene descriptor, unseen configurations, like the collision course, are also reported as abnormal. Among all the detected abnormal events, there are hardly any that have no plausible interpretation.

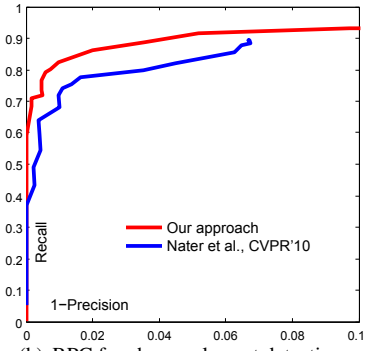
Times Square [3]. (640×480 pixels, approximately 0.3 fps, 2 months). In this dataset, images from a webcam overseeing Times Square in New York are taken at low frame rate over a long period. Hence, the relation between adjacent frames is on a larger time scale. We downsample the original color images to 24×32 pixel grayscale images and concatenate the rows to a vector. Due to the low frame rate, no motion is included ($n = 1$).

The hierarchical model was constructed with data from 17 days (150,000 images, every 3^{rd} image). The obtained hierarchy has 65 nodes, thereof 26 basic activities. In Fig. 7, we display the tree-like structure for the first four levels, and show typical instances of some nodes, together with their interpretation. Day-night changes turn out to be the most dominant cues, which are separated in the first step.

In Fig. 8 we show eight illustrative abnormal events that are detected among more than 250,000 evaluated frames. We detect similar anomalies as reported in [3], such as the ones

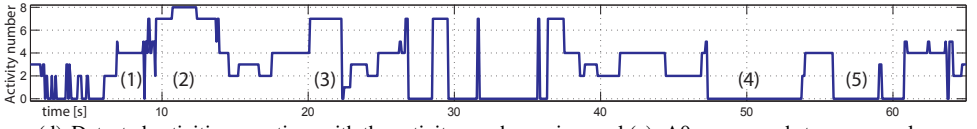


(a) Automatically learned hierarchy



(b) RPC for abnormal event detection

(c) Selected frames: 1. Walk leftwards, 2. Sit, 3. Walk occluded, 4. Fall, 5. Wave heavily.



(d) Detected activities over time with the activity number as in panel (a). A0 corresponds to an anomaly

Figure 9: In-house dataset [▭]: (a) Nine basic activities emerged from training, (b) Applied to a test video that contains anomalies, our approach outperforms previous state-of-the-art. (c) Sample frames which illustrate detected familiar activities and abnormal events. (d) The detected activities are reported over time and the location of frames $1-5$ is indicated.

in the first line of Fig. 8. In addition, the system also reported many cases of incomplete frames, camera failures, water on the lens and other salient situations.

5.2 Human behavior analysis

In this task a person is being tracked throughout the video and simultaneously his behavior is analyzed. We use the data from our previous work [▭], motivated for monitoring of elderly people. The same silhouette features are used. Images of 640×480 pixels are recorded at 15 fps. The training sequence *seq1* ($7,100$ frames) contains normal daily activities. The evaluation is carried out on the test sequence *seq2* ($1,030$ frames) that also contains abnormal events such as a fall.

The hierarchy obtained form *seq1* is visualized in Fig. 9(a), and for each leaf node activity, some silhouettes are shown. The hierarchy nicely encodes the different aspects of behavior observed in this video. At higher levels, it distinguishes between upright and other poses, at low levels, sitting, picking up, walking leftwards or rightwards are isolated. Hence, meaningful human activities are discovered automatically.

The model is applied to *seq2*. In Fig. 9(c) some selected frames of are displayed, they show three normal situations and two detected anomalies. The observed person is tracked and the matching activity is determined simultaneously. The plot in Fig. 9(d) characterizes the evolution of the detected basic activity over time. *A*0 groups the outliers.

We quantitatively compare the overall performance of our technique to the results of [▭]. The recall-precision curve is obtained by sweeping the parameter θ (see Eq. 3) and displayed in Fig. 9(b). We outperform the previous state-of-the-art. Due to the integration of temporal

relations, the discovered activity models turn out to be more accurate compared to *k-means* clustering in [11]. In particular the recall is increased from 68% to approximately 83% at 99% precision.

6 Conclusion

In this paper, we presented a data-driven approach to activity segmentation that exploits the temporal relations in video sequences. The small changes from frame to frame are examined with slow feature analysis, in order to automatically represent the data in a meaningful hierarchy. We have shown how this model is applied to unseen videos and that the hierarchy can be used to explain the observations. Due to two linear techniques of low computational complexity, we are able to efficiently detect normal and abnormal activities. Finally, qualitative and quantitative results demonstrate the validity of our technique.

Acknowledgements This work was supported by EU integrated projects DIRAC (6FP IST-027787) and IURO (7FP EC STREP IURO).

References

[1] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *PAMI*, 30(3):555–560, 2008.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

[3] M. D. Breitenstein, H. Grabner, and L. Van Gool. Hunting Nessie – Real-Time Abnormality Detection from Webcams. In *ICCV WS on Visual Surveillance*, 2009.

[4] T. Hospedales, S. Gong, and T. Xiang. A Markov Clustering Topic Model for mining behaviour in video. In *Proc. ICCV*, 2009.

[5] W. Hu, T. Tan, L. Wang, and S Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 34(3):334–352, 2004.

[6] W. Hu, X. Xiao, Z. Fu, D. Xie, F.-T. Tan, and S. Maybank. A System for Learning Statistical Motion Patterns. *PAMI*, 28(9):1450–1464, 2006.

[7] S. Klampfl and W. Maass. Replacing supervised classification learning by Slow Feature Analysis in spiking neural networks. In *NIPS*, 2009.

[8] D. Kuettel, M. D. Breitenstein, L. Van Gool, and V. Ferrari. What's going on? Discovering spatio-temporal dependencies in dynamic scenes. In *Proc. CVPR*, 2010.

[9] J. Lange and M. Lappe. A model of biological motion perception from configural form cues. *Journal of Neurosciences*, 26:2894–2906, 2006.

[10] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *CVIU*, 104(2):90–126, 2006. ISSN 1077-3142.

[11] F. Nater, H. Grabner, and L. Van Gool. Exploiting Simple Hierarchies for Unsupervised Human Behavior Analysis. In *Proc. CVPR*, 2010.

[12] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words. *IJCV*, 79(3):299–318, 2008.

[13] J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling Temporal Structure of Decomposable Motion Segments for Activity Classification. In *Proc. ECCV*, 2010.

[14] C. Stauffer and W. E. L. Grimson. Learning Patterns of Activity Using Real-Time Tracking. *PAMI*, 22(8):747–757, 2000.

[15] P. Turaga, A. Veeraraghavan, and R. Chellappa. Unsupervised view and rate invariant clustering of video sequences. *CVIU*, 113(3):353–371, 2009.

[16] R. Urtasun, D. J. Fleet, and P. Fua. Temporal motion models for monocular and multiview 3D human body tracking. *CVIU*, 104(2):157–177, 2006. ISSN 1077-3142.

[17] G. Veres, H. Grabner, L. Middleton, and L. Van Gool. Automatic Workflow Monitoring in Industrial Environments. In *Proc. ACCV*, 2010.

[18] X. Wang, X. Ma, and W. E. L. Grimson. Unsupervised Activity Perception in Crowded and Complicated Scenes Using Hierarchical Bayesian Models. *PAMI*, 31(3):539–555, 2009.

[19] Daphna Weinshall, Hynek Hermansky, Alon Zweig, Jie Luo, Holly Jimison, Frank Ohl, and Misha Pavel. Beyond novelty detection: Incongruent events, when general and specific classifiers disagree. In *NIPS*, 2008.

[20] L. Wiskott. Slow Feature Analysis: A Theoretical Analysis of Optimal Free Responses. *Neural Computation*, 15(9):2147–2177, 2003.

[21] L. Wiskott and T. Sejnowski. Slow Feature Analysis: Unsupervised Learning of Invariances. *Neural Computation*, 14(4):715–770, 2002.

[22] F. Zhou, F. De la Torre, and J. F. Cohn. Unsupervised Discovery of Facial Events. In *Proc. CVPR*, 2010.