DIRAC

Detection and Identification of Rare Audiovisual Cues

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than
things you do expect) Plautus (ca 200 (B.C.))

Project no: 027787

DIRAC

Detection and Identification of Rare Audio-visual Cues

Integrated Project
IST – Priority 2

DELIVERABLE NO: D3.12
Fully Functional Implementation of a Pre-learned Tracker Tree

Date of deliverable: 31.12.2010
Actual submission date: 07.02.2011

Start date of project: 01.01.2006 Duration: 60 months

Organization name of lead contractor for this deliverable: ETHZ

Revision [0]

Project co-funded by the European Commission within the Sixth Framework Program (2002-2006)		
Dissemination Level		
PU	Public	
PP	Restricted to other program participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	X
CO	Confidential, only for members of the consortium (including the Commission Services)	

DIRAC
Detection and Identification of Rare Audiovisual Cues

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than
things you do expect) Plautus (ca 200 (B.C.))

D3.12 – Fully Functional Implementation of a Pre-learned Tracker Tree

Eidgenoessische Technische Hochschule Zuerich (ETHZ)

Abstract:
We present a tracker tree as a framework for the detection of abnormal events in in-house scenarios. The current implementation of the tree includes multiple trackers that all incorporate different information about the human behavior that is to be expected. The tree now also incorporates action detection based on spatio-temporal interest points, which enables the discrimination of different sitting actions. Each tracker can validate the observations independently, and from the simultaneous output of the trackers at different hierarchical levels in the tree, anomaly reasoning is performed. We show qualitative and quantitative results for in-house video sequences that are recorded at two different locations within the DIRAC project.

Table of Content

1. Introduction

1.1. Context

Figure 1. Illustration of the target application. Abnormal events at peoples homes are detected, automatically triggering an alarm.

In video surveillance, novelty detection is an important task for various applications. Here we are interested in monitoring elderly people in their homes and show how the notion of incongruence helps greatly in the detection of surprising or unusual events. If these events are detected, an automatic alarm can be generated that triggers a certain predefined action, as illustrated in Fig. 1.

In the DIRAC project, tracker trees have been successfully implemented and illustrated for the task of abnormal event detection in peoples homes. A previous deliverable (D3.9) already focused on the technical implementation of the tracker tree. However, in the course project developments, the tracker tree was continuously refined, new trackers were introduced, and the argumentation was adapted accordingly. This deliverable gives an overview of the latest tracker tree version.

1.2. Approach

Visual trackers in general incorporate a certain amount of information about the normal situations they are applied to. For example, an articulated body motion tracker is highly tuned to a walking person and exploits strong priors for successful tracking, whereas a simple blob tracker relies on very weak assumptions. We propose to arrange multiple different trackers a tree-like hierarchy, the location of each tracker is based on the information it relies on. Trackers further up in the tree have been trained for quite a narrow set of actions - e.g. specific to the walking style

of one person - whereas trackers closer to the root node are able to track a broad variety of motions.

Active trackers signal that the observed information corresponds to their underlying assumptions and the according action or concept can thus be detected. An active walking tracker for example signals that the person in the scene is walking. On the other hand and in line with the DIRAC reasoning on novel events, if no tracker at a certain level in the tree can cope with the observations, but more general trackers can, this signals an abnormality. In Sec. 2, we give a more detailed explication of this paradigm and show its application in the elderly surveillance task. Sec. 3 briefly reviews underlying techniques that are used in the specific trackers, and Sec. 4 presents results.

1.3. Related work

Detecting events in image streams is a classical task in computer vision. From a surveillance point of view, it is particularly important to detect unusual events and thus a wide variety of different approaches have been proposed during the last couple of years, covering diverse applications. Abnormal events are often recognized as outliers to previously trained models of normality (e.g. [Johnson96], [Makris05]). The approaches vary from using basic (temporal) features in a statistical analysis (e.g. [Adam08], [Stauffer00]) to high level specific object tracking [Hu06], sometimes incorporating information on the object's behavior at certain scene regions [Li08] or modeling the tracked object's normal actions and action-transitions in a Markovian framework [Veeraraghavan08].

Developed for the task of fall detection, vision systems (e.g. [Anderson09], [Nasution07], [Rougier07]), usually focus on precisely modeling the behavior to be detected i.e. the activity of falling, for example by means of measuring the speed of the foreground blob transformation, or by including the assumed immobility of the person after the fall. In a different approach, [Cucchiara05] use a posture classification system for a more detailed human-behavior analysis that permits the detection of a fallen person.

2. Tracker tree

The currently implemented tracker tree is visualized in Fig. 2. Every node in the tree depicts a separate tracker, all trackers are run simultaneously on the video data, and

reasoning is performed from the confidence scores of the trackers. The trackers shaded in grey have been added to the tree since the last deliverable. Also, the layout of the body part trackers has evolved, we now reason within the black box. In the following, we explain the tree in more detail.

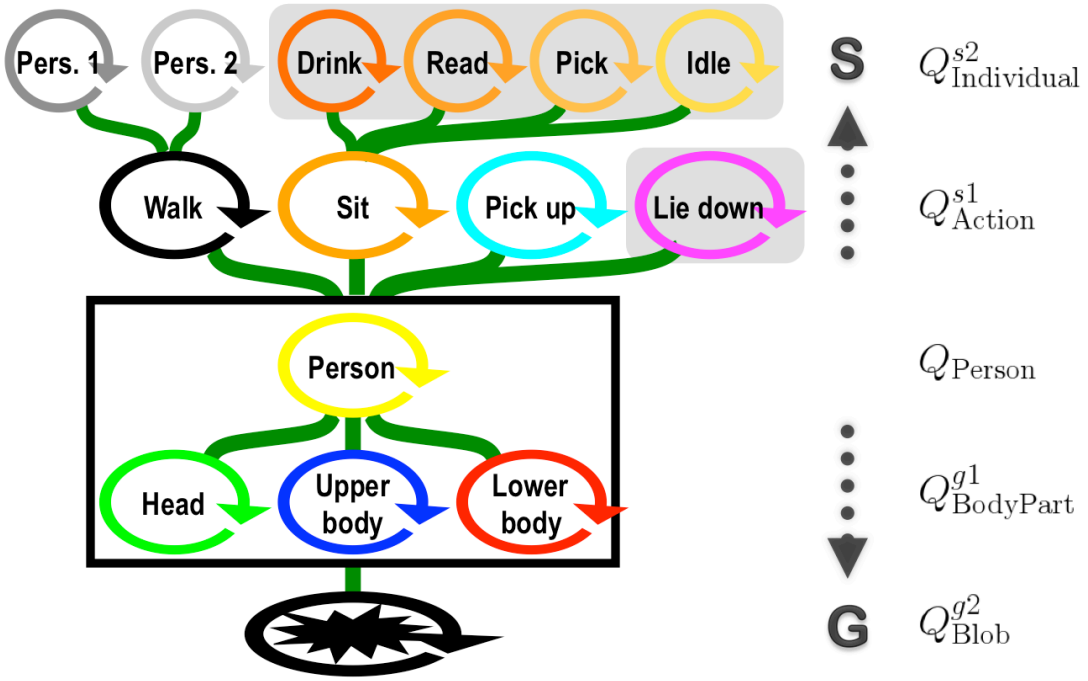

Figure 2. Current version of the implemented tracker tree. More general tracker concepts are towards the root of the tree (**G**), more specific trackers that contain information about specific actions or persons are placed at the leaves (**S**). The evolution of the tracker tree since the last reporting period is shaded in grey.

2.1. Tracking

Each tracker in the tree tries to follow a target over time. It does this with help of some prior information, that tells the tracker what it has to expect. The information that each tracker incorporates is different for each tracker and determines the trackers location within the tree.

At the root, the most general tracker is located, which is a simple foreground blob tracker. As we are dealing with persons inside the scene, the second level of trackers already goes after human targets. These trackers are located inside the black box of Fig. 2. Different body part trackers track partial aspects of the human body, whereas the yellow tracker identifies persons in a tracking-by-detection approach. These four

trackers constitute a conjunctive hierarchy, where the (more general) parts are joined in order to form a (more specific) person. One level up, specific human actions are modeled. Each of the trackers on this level is familiar with a certain action that humans perform in indoor scenes, such as walking, sitting, picking up an item or lying on the couch. At an even higher level, two different specialization aspects are depicted. On the left, connected to the general walking tracker, we have person-specific walking tracker instances. These two trackers have learned the gait and other attributes, which determine the walking style of a person. On the right side, we show that it is possible to refine basic actions (sitting in this case), to be more precise on what kind of arm action the person is performing while sitting. In our configuration, this can be drinking, reading a book or a magazine, picking an item from the table or being idle.

If the observations behave according to expectations, there is always a specific leaf node tracker that best explains the situation, and thus a path exists that connects active trackers from the root node to a leaf node.

As to the employed techniques, they will be explained in Sec. 3 in more detail. We make use of publicly available state-of-the-art trackers for the root node tracker and for the person detector. All other trackers in the tree rely on underlying representations using silhouettes (body part trackers and action trackers) or spatio-temporal features (different sitting trackers at the top level). These techniques were developed in the DIRAC project, and respectively published in [Nater09] and [Willems08].

2.2. Abnormal event detection

An abnormal event is detected when and where a more general tracker can still explain the situation, whereas all connected more specific tracker instances show a low confidence. A tracker that uses less specific knowledge about the world is active for a wider variety of situations, whereas trackers that are specifically tuned to some aspect tend to fail as soon as this aspect is not present. Indeed, using more specific and precise information is only advantageous as long as this information is correct.

The situation is different inside the black box of Fig. 2: As the person detector, the more specific concept, is composed of its more general body parts, the abnormal event is signaled if all the body parts are present, but no person is detected.

One advantage of a tree with multiple hierarchical levels is the possibility of semantic reasoning. From the location in the tree where the novel pattern appears, we can deduce an interpretation on the nature of the incongruence. For instance, if none of the person trackers can explain the data well, but the blob tracker follows an object, we may have a pet entering the home of a person not having one. If none of the normal action specific trackers does well, but tracking by full body detection still works, this might be an indication of an unusual event like limping. If none of the person-specific walking trackers gives a strong output, but the generic walking tracker does, an intruder can be reported.

Occlusion handling: Partial occlusions occur frequently in in-house surveillance scenarios, *e.g.* furniture partially interferes the visual pathway between the person and the camera. In the tracker tree, this means that a person is detected (person tracker is active) but at least one the body part trackers fails. At the same time, all the action trackers are likely to fail, as they were trained when the entire persons were visible. To prevent the tracker tree from signaling an abnormal event in this situation, we propose a different interpretation and reason inside the black box in the tree of Fig. 2. The body part trackers act as conditioners for the action trackers: No active action tracker is expected if any of the body part trackers fails. When for example a sofa occludes the person, the lower body part is missing and therefore no action is expected to be valid. Hence, occlusions are not detected as incongruent activity patterns.

3. Methods

For all the trackers in the tree, we use tracking methods that are state-of-the-art and employed as provided by the authors, as well as two specially developed techniques.

In the current implementation of the tracker tree in Fig. 2, the trackers for a foreground object (root) and the person tracker (yellow) are adopted, the trackers that observe specific sitting actions (drink, pick, read, idle) are due to the extraction of spatio-temporal interest points, whereas all the other trackers in the tree rely on a manifold-based silhouette tracking technique.

3.1. State-of-the-art trackers

3.1.1. Cam shift tracker

In order to track the foreground object in the scene, which can be of any nature, we use the Cam-Shift tracking technique as proposed by [Bradski98]. We use the implementation available in OpenCV2.0 and have integrated it in a matlab mex-interface. Once initialized, the tracker uses color histogram of the foreground image to track the target position and size.

3.1.1. Person tracking-by-detection

Person tracking is performed in a tracking-by-detection paradigm. We use the part-based detector of [Felzenszwalb08], which is available on the author's web page. The detections are connected to form tracks of the person that is moving trough the scene. In the newest cascaded version [Felzenszwalb10], this detector is optimized for execution speed. We integrated this latest version[1] in the tracker tree.

3.2. Manifold-based tracking (ETHZ)

The trackers going after the different actions (walk, sit down, pick up, lie down), the person specific walking trackers and the body part trackers all rely on trackers that model the appearances in underlying manifolds. As features, we use silhouettes obtained from background subtraction. Additionally, a signed distance transform (background: negative values, foreground: positive values) is applied, such that pixels close to the edges have lower weights than pixels in the center of the foreground blobs.

3.2.1. Training

In the training procedure, the information that each tracker incorporates is specified and modeled. We choose to encode the silhouettes in a low dimensional manifold. Isomap [Tenenbaum00] is taken as tool to reduce the dimensionality of the input silhouettes. This nonlinear technique produces smooth manifolds, which preserve local distances. For the generative association between this latent space and the

[1] Downloaded from http://people.cs.uchicago.edu/~pff/latent/, 15. Dec. 2010.

original silhouettes, we learn a regression function by using Gaussian Process [Lawrence03]. Each point in the three dimensional manifold can be mapped to its representation in silhouette space.

In particular, we recently included a 'lie down' tracker that was trained from data recorded at OHSU. To this end, the actors were told to perform a normal lying down and standing up action repeatedly. In the training video, start and end frames of this action were annotated manually, and the silhouette features were extracted for the selected frames. From four of these action snippets, where the silhouettes were nicely segmented (1790 frames), a 'lie down' tracker was trained. The same was done for the 'sit down' tracker. It was also adapted to the OHSU setting and the particular way of each actor to sit down. The remaining trackers were kept as trained previously at ETHZ, since the models performed well in the OHSU setting.

3.2.1. Tracking

The application of the tracker on unseen data is based on a probabilistic tracking approach. After background subtraction, the observation is analyzed by a particle filter that simultaneously scans the location and size of the target in the foreground image and the manifold model. A probabilistic likelihood function determines for every sample how well the observation fits the underlying model. Maximizing the likelihood over all the samples provides a measure on how well the tracker can explain the current observation and sets the target's location and scale. Finally a temporal smoothing step is applied.

We refer to DIRAC deliverable D3.9 and [Nater09] for a detailed description of the manifold-based tracking technique.

3.3. Spatio-temporal interest point tracking (KUL)

As mentioned before, the tree has been extended even further beyond what is possible with the silhouette-based trackers by extracting additional information using local interest points. In order to obtain a description of the content in a video sequence, we opt for the recently proposed spatio-temporal interest points based on the determinant of the Hessian-matrix [Willems08]. A video is first converted into a spatio-temporal volume from which a dense set of scale-invariant (both spatially and temporally) features are extracted. The features have been previously used for action classification, content-based video copy detection and action localization.

In this case, we aim to differentiate between several actions that are preformed while a person is sitting down on the couch. From the DIRAC reasoning on novel events, an abnormal sitting activity is detected if it cannot be matched to any of the known concepts.

We refer to DIRAC deliverable D3.6 and [Willems08] for more detailed information on the spatio-temporal features.

3.3.1. Technique

Four additional leaf nodes were added to the tree. These nodes allow to distinguish between the following actions: idling (where the person is just sitting down, doing nothing), drinking, reading a magazine and picking up an item from the table. Finally, we also want to be alerted when the person is performing another unexpected action (such as coughing heavily).
The input to this level is given by the bounding boxes that were found by the 'person sitting' tracker. Each spatial bounding box is converted into a bounding volume by extending it temporally over a fixed number of frames (2 frames in each direction, in this case).
Features are extracted in parallel and assigned to a cluster. Next, bag-of-words are used to describe each bounding volume by combining all features that reside inside the volume. An SVM is finally applied to compute the probabilities of each bounding volume to one of the 4 actions.

3.3.1. Training and detection procedure

Five video sequences were recorded to perform the training of the system. In a first step, scale-invariant features were extracted over all the videos. Next, the feature's descriptors were clustered and each feature was assigned to one of the 4096 clusters. In a third step, we computed a bag-of-words (BoW) for each of the bounding volumes, derived from the 'person sitting' tracker. For each of the bounding volumes, the ground-truth was manually annotated. Four of the five videos were used to train 4 one-versus-rest SVM (one for each of the actions) using cross-validation and a Chi-squared kernel. In each training step, we also include some BoW to the 'rest' class that contains random behavior (and thus not belonging to one of the 4 actions). As the trained system is SVM based, it returns binary decision values. In order to obtain a probabilistic output, we followed the technique of [Platt00] and used the fifth sequence to compute the required sigmoid functions to provide this conversion. The obtained SVMs and sigmoids can next be used to obtain

4 probabilities (one for each action) of each of the bounding volumes in subsequent test videos.

In order to obtain one final label per bounding box, which is also temporally consistent, we apply a sliding window approach. We therefore define two thresholds: a high threshold Th and a low threshold Tl. Within the sliding window (of length L), we add a vote for one of the actions when its probability is above Th. If on the other hand, no actions have a probability above Tl, we add a vote for unexpected behavior. The action with the most votes is assigned as the label if it has at least L/2 votes. For 'unexpected behavior', we add an additional constraint that several sequential frames all need to vote for unexpected behavior. If the conditions are not met, the label of the previous frame is kept unaltered. This scheme allows for some temporal consistency between consequent frames and removes jitter in the labeling.

3.4. Usability

We have put together a tracker-tree toolbox that integrates all the trackers depicted in the tree of Fig. 2, except the ones on the top level trackers (person specific walking, sitting actions). With the help of other DIRAC partners, especially CTU, a launch function was written, which automatically scans the images in a target directory, initializes all the trackers when a person is detected and autonomously generates the tracking output. Once the person leaves the scene, the tracking is stopped. This toolbox (current version as of 29.01.2011: tracker_trees_v12) is available for the entire DIRAC consortium and is extensively used in WP6 (see deliverable D6.13). We also plan to make the toolbox publicly available on the DIRAC web page in the near future.

4. Evaluation

4.1. Data and setup

We evaluate the tracker tree on two datasets. We recorded the first dataset ourselves in a living-room environment at the ETHZ lab. In fact, as we deal with new developments here, it is important to have data available that is useful for training and can show the improvements and the crucial findings. We extensively evaluate and comment on the tracker tree performance in the ETHZ setup.

Additionally and for completeness, we will show shorter evaluations on data recorded at OHSU living lab.

For all the experiments, we omit the person specific walking trackers (leaf node trackers on the top level at the left in Fig. 2), as their development is not new and their performance has been previously reported in DIRAC deliverable D3.9.

4.1.1. ETHZ data

For the DIRAC project, we installed a mockup living room at the Computer Vision Lab at ETHZ. The scene consists of a couch, a chair, a television, a shelf, cupboards, a carpet on the floor and a lamp in the background. This installation has already been used previously to record training data to train and test previous parts of the tracker tree. Video recording is done with an AVT Marlin firewire camera with a resolution of 640x480 pixels at a frame rate of 15 frames per second.

We recorded several video sequences in order to train and test the trackers in the tree. The training sequences are especially concerned with training the newly added sitting actions. To this end, five training videos were recorded, the actor walks into the scene, sits down, performs all the actions (idling, reading, drinking and picking from the table) in random order and walks out of the scene again. All training sequences contain between 1600 and 2000 frames.

The test sequence shall show all the different benefits of the augmented tracker tree. In this video, the person walks into the living room and performs various actions: walks around, walks behind the couch, sits down, reads, drinks, coughs heavily, stands up, falls, lies down, etc. The test video contains 2380 frames.

4.1.2. OHSU data

OHSU data was recorded at the Living Lab facilities of partner OHSU. For a more detailed description of the recorded sequences and techniques used, we refer to DIRAC deliverables D6.10 and D6.13.

We used the provided training data to train the lying down tracker and adapt the sitting tracker to the OHSU scene layout. In the training sequences, two different actors performed these actions at multiple repetitions. In particular, we used the

sequences s31t2 and s33t2 for training the lying down action, and s32t3 in order to adapt the sitting tracker.

We randomly chose two sequences among all the recorded test sequences in order to illustrate the tracker tree performances on the OHSU data. These are s43t1 and s43t6 (cf. DIRAC deliverable D6.13).

In general, the OHSU data arrived very late in the project and therefore time was short include extensive evaluations in this deliverable. Also, the sitting action trackers at leaf nodes require according training and test data, which was not available. Focus of the OHSU recordings was much more the audio-visual integration, therefore we only show here how the lower part of the tracker tree (Levels 1-4 in Fig. 2) performs when applied to this data. Performances will be reported in D6.13, here we only show some qualitative results including failure cases.

4.2. Results and discussion

We illustrate the performance of the tracker tree depicted in Fig. 2 with the recordings taken at the two localities.

4.2.1. ETHZ sequence - qualitatively

In Fig. 3, 4 and 5 we show a number of selected frames for the ETHZ sequence. The person is tracked throughout the video sequence, and the active trackers are displayed. For sake of visibility, not all active trackers are shown, but always the most informed (highest in the tree of Fig. 2) is depicted, using the color-coding of Fig. 2. For example, if walking is observed, only the bounding box output of the walking tracker is displayed in black, even though other less informed trackers, such as the foreground blob tracker, or the different body part trackers, are also active. The same holds for example if the person is sitting on the couch and reading: the orange bounding box of the sitting tracker is displayed, and the discovered activity of the person is tagged 'reading'.

The person is tracked precisely in terms of location and size of the bounding box in the entire sequence, different trackers appear at different instances in time, according to the activities performed by the protagonist. We now run through a number or interesting cases:

- When the person moves behind the couch, the lower body not visible from the camera location, hence the person is half occluded and not all body parts are

tracked. This can be observed in Fig. 3, frame 3. As said previously, this corresponds to a missing part inside the black box of Fig. 2. Since this situation might happen often in our scenario, we do not signal an anomaly, but rather observe an 'incomplete' person. In this case we do not expect any higher-level tracker to be active, since they have been trained on entirely visible persons. Not detecting all body parts however may also have other reasons. If for example the person has some unknown upper or lower body pose (as in Fig. 4, frame 4), the head might still be tracked (green), and the person is still detected by the person detector (yellow). This actually happens often during transitions from one action to another. These transitions might not be modeled by the action trackers, but body parts are observed. An example is the transition from lying to sitting, as depicted in Fig. 5, frame 7.

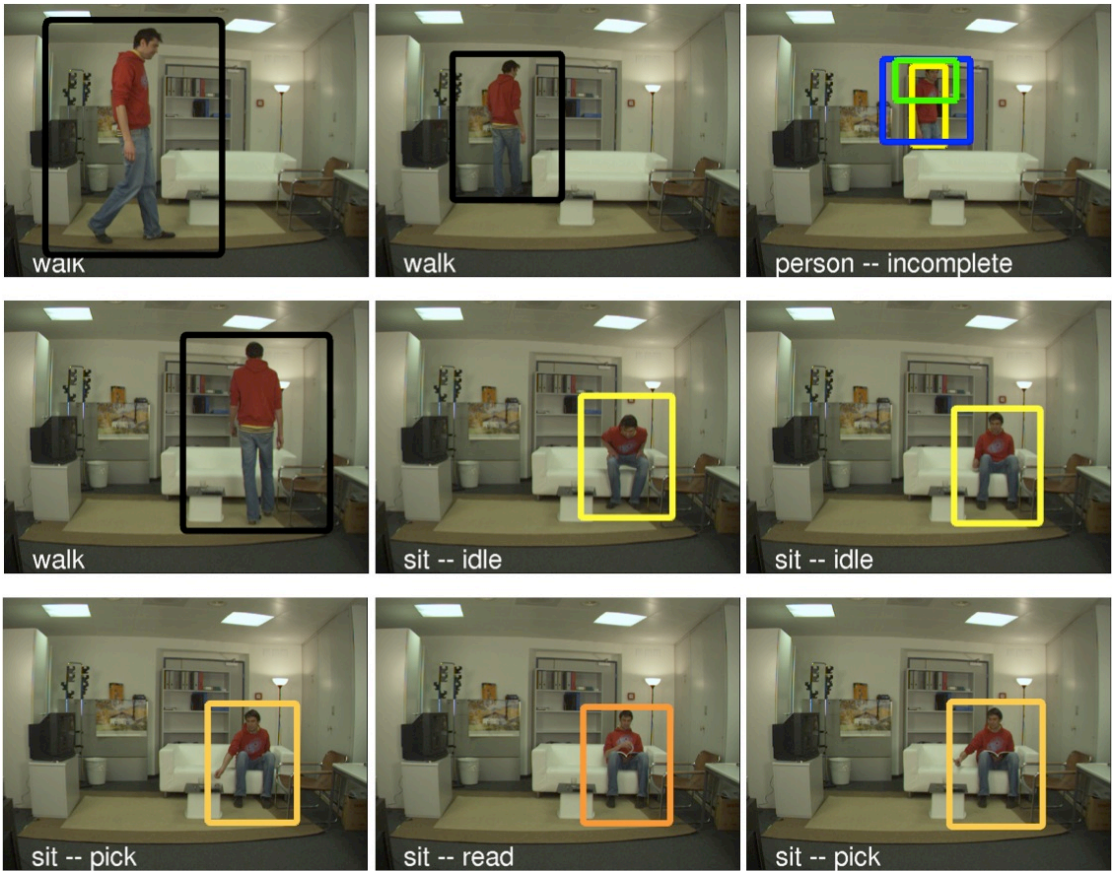

Figure 3. ETHZ test sequence part 1: A person enters the scene, walks behind the couch, then sits down, picks a book and reads. The location and size of the person is tracked throughout the scene. For visibility reasons, only the active leaf node action trackers are displayed. If no action can be tracked (as in frame 3), the detected body parts are shown and the person is marked 'incomplete'. The color code of Fig. 2 is employed.

- When the person is in the sitting position, the spatio-temporal tracking jumps in and evaluates the bounding boxes delivered by the sitting tracker. The person is observed to be either idle, picking something from the small table, reading a book or a magazine or drinking. If none of these trackers validates the situation, an anomaly is detected. This is for example the case if the person needs to cough heavily. He takes his hand to the mouth and makes movements with the upper body. This abnormal event is spotted, as shown in Fig. 4, frames 1 and 3.

Figure 4. ETHZ test sequence part 2: While sitting, the actor suddenly needs to cough heavily twice. The performed hand and upper body movements do not correspond to any of the known sitting action concepts, but sitting as such is still recognized. Thus an abnormal action is detected (frame 1 and 3). Transitions are not always well modeled in the actions themselves, but body parts are discovered as in frame 4. Picking up a pen from the table is also recognized.

- Other actions, such as picking up an object from the floor, or in our case picking a pen from the table are recognized as such (Fig. 4, frames 6 and 7). Also, lying down on the couch is successfully tracked (Fig. 5, frames 5 and 6).

- If the person falls and lies immobile on the floor, none of the trackers that go after normal human motion or body parts in normal configuration remain active, and only a foreground object is tracked (white bounding box in Fig. 5, frame 1). In this case, an abnormal event is reported.

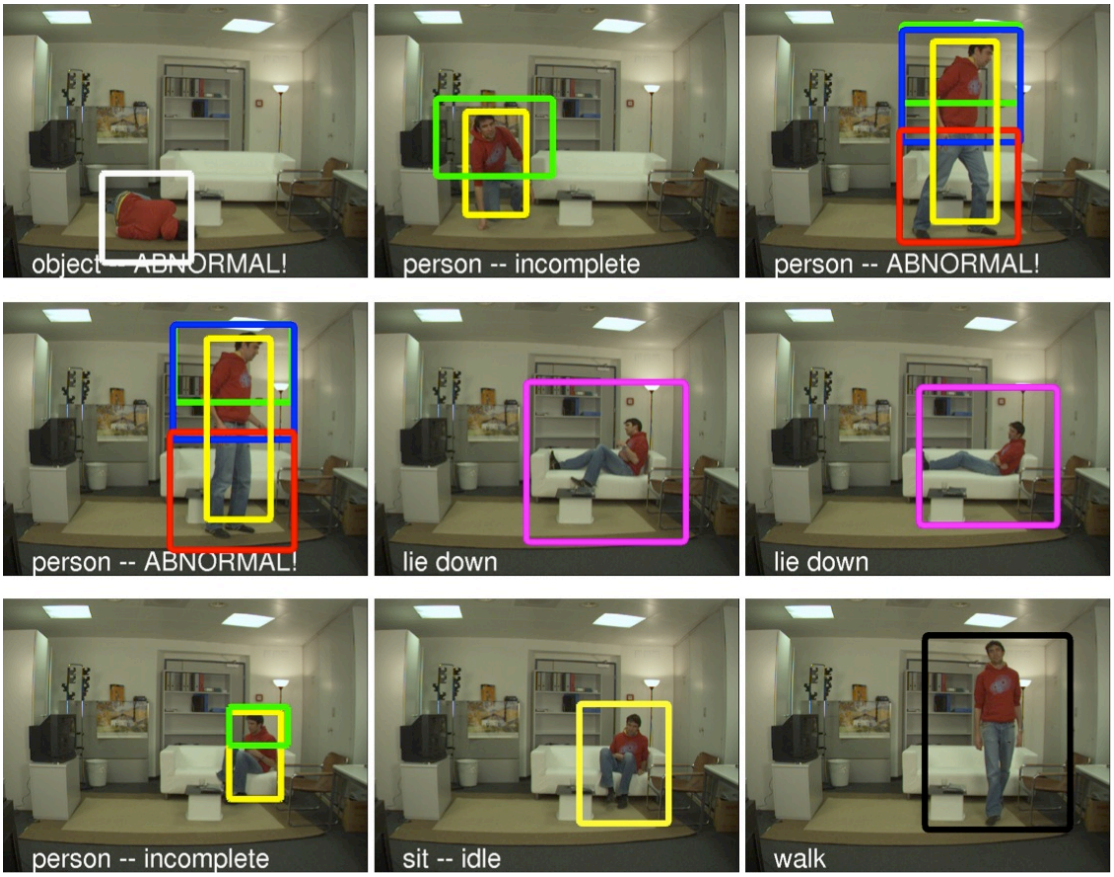

Figure 5. ETHZ sequence part 3: When moving to the front of the scene, the person suddenly stumbles (Fig. 4 last frame) and falls (first frame). Only the foreground blob tracker in white remains active, this signals an abnormal event. After standing up, the person starts to limp (frames 3 and 4). In this case, all the body parts are visible, but no known action is recognized, thus the frames are classified as abnormal. Lying on the couch is subsequently recognized as such, then the person stands up and leaves the scene.

- After standing up, the person might have hurt himself and starts to limp (Fig. 5, frames 3 and 4). In this case, the walking tracker, trained on normal walking motion will lose confidence, while all the body part trackers still validate the observation. In this case, an anomaly is detected. As all the body parts are visible, this must be an unknown action.

4.2.2. ETHZ sequence - numerically

In order to quantitatively evaluate the performance of the tracker tree, we manually annotated the ETHZ test sequence. We choose to evaluate the manifold-based action trackers (4th level in the tree) as well as the interest-point-based sitting action trackers (5th level in the tree) individually and do this by means of Recall-Precision-Curves (RPC).

The performances of the individual trackers can be measured independently, or they can be evaluated jointly. We do both in the following.

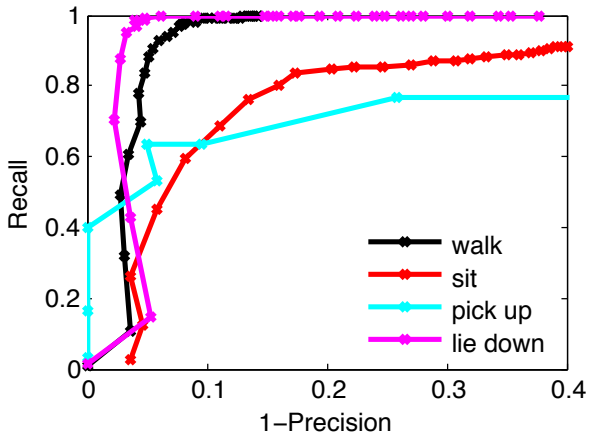

Figure 6. RPC curve of the individual action trackers (trackers on level 4 in Fig. 2: walk, sit, pick up and lie down). Each tracker is individually evaluated. The threshold that classifies the tracker as active or not, is swept, and experimental results are compared to the manually annotated ground truth.

Fig. 6 depicts the four RPCs for the action trackers, (walk, sit, pickup and lie down, respectively). The curves are obtained individually, which means that each tracker judges the observation independent of the output of the other trackers. The involved parameter to sweep is the threshold applied to the probabilistic tracker output. This threshold determines whether the tracker is said to validate the observation or not.

From the curves in Fig. 6, one can notice that the walking and lying trackers appear to be more precise and robust than sitting and picking up trackers.

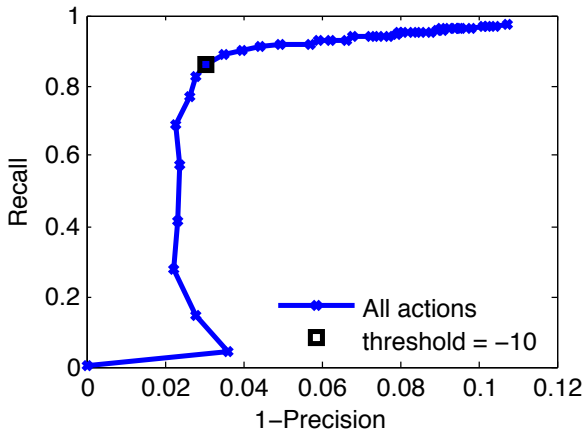

Figure 7. RPC curve for the action level (level 4 in Fig. 2). First, the tracker with the maximal likelihood for each frame is chosen, then the threshold is applied (swept). The threshold used for generating the per frame results is displayed.

The RPC depicted in Fig. 7 combines the outputs of all the trackers. In fact, from the conjunctive tracker hierarchy, we know that only one tracker at this level in the hierarchy can possibly be active at the time. Therefore a max-pooling operation is performed before applying the classification threshold. In black we visualize the threshold that is chosen for the per frame results in Fig. 3, 4, 5. At this threshold, the action trackers show a recall of 86% at a precision of 97%.

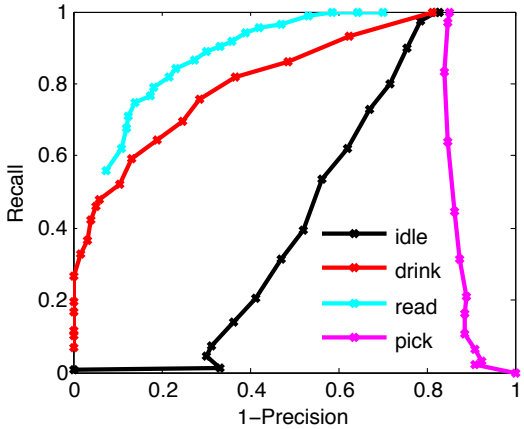

Figure 8. RPC curve for the sitting action trackers (level 5 in Fig. 2: idle, drink, read and pick). A threshold is applied to the probabilistic output of the interest point trackers, and the output is compared to the ground truth.

The RPCs for the individual sitting action trackers (idle, drink, read and pick) are depicted in Fig. 8. Again, the threshold is swept over a range of possible values, and the recall and precision numbers are computed from based on the manually annotated ground truth labels. On can notice that especially the picking tracker, but also the idling tracker perform quite badly if they are run independently. This is not very surprising, as they are trained in an SVM framework, where usually one is evaluated against all the others (cf. Sec. 3.3). In that sense, the probabilistic interpretation due to [Platt00] does not seem to perform very accurately.

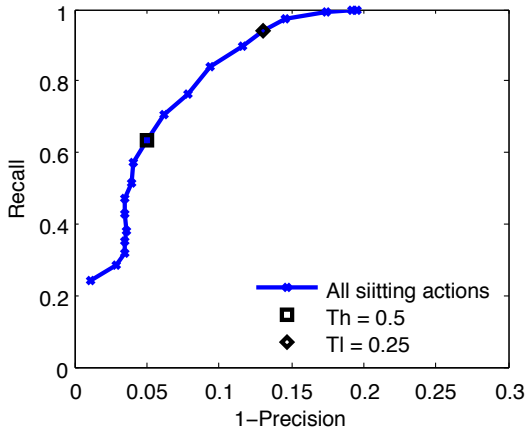

Figure 9. RPC curve for the combination of all sitting action trackers. Before applying the threshold, the tracker with the maximal probability is picked. Note the considerable increase in accuracy compared to the individual trackers, that results from this max-pooling.

The story however is a different one, if a max-pooling operation is applied before thresholding the probabilistic output values. In fact, the most confident tracker appears to be the correct one in most of the cases. The corresponding RPC is visualized in Fig. 9 and manifests an increased performance compared to all individual sitting trackers (note the scale on the x-axis). The curve is obtained by classifiying the probabilsitic per-frame output and comparing the outcome aginst the ground truth labels. This per-frame evaluation only requires one threshold that is swept. The two thresholds which are used in order to assure the temporal consistency (*cf.* Sec. 3.3) are plotted in black (Th=0.5 with 63% recall at 95% precision, and Tl = 0.25 with 94% recall at 87% precision).

To evaluate the abnormal event detection capacity of the tracker tree, we use the thresholds as depicted in Fig. 7 and Fig. 9. After the abnormality reasoning that is performed in the tree, we compare the per-frame output to the manually annotated

abnormality labels. The overall system has a recall of 75% at a precision of 68% for abnormality detection task on the ETHZ sequence.

4.2.3. OHSU sequences

The processing pipeline of OHSU recordings has multiple stages which involve debayering, trimming, fisheye distortion correction, audio and video processing. We refer to D6.13 for a more detailed evaluation of the procedure. Here we simply show how the tracker tree qualitatively performes on these sequences and show some intresting cases including failures.

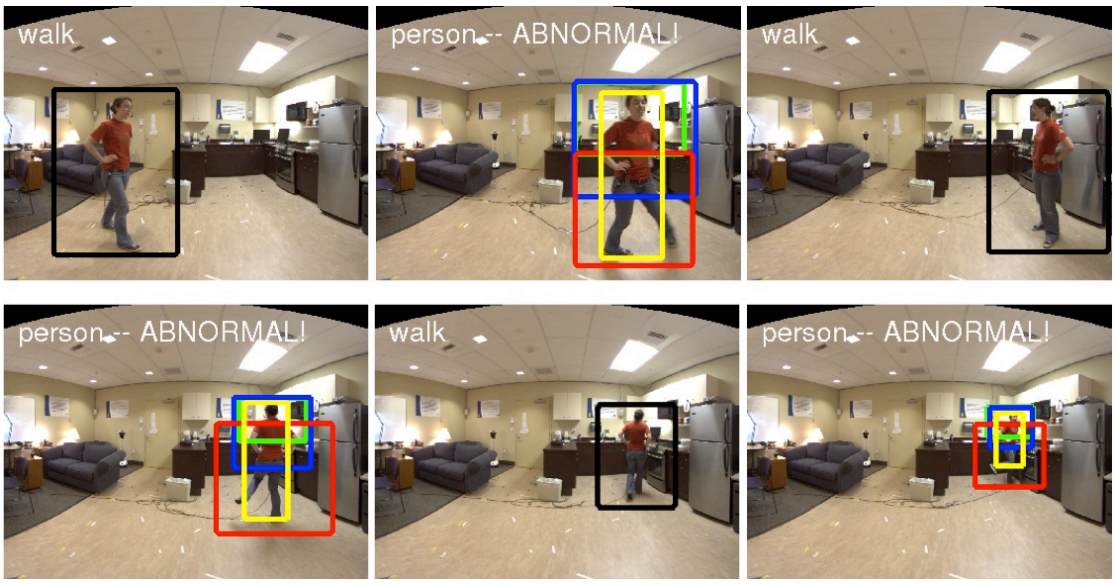

Figure 10. Results for OHSU sequence 1: The actress walks into the scene, makes a strange step to turn around (frame 2), then has to step over the black microphone cable twice (frames 4 and 6). In these three cases, the tracker tree correctly signals an anomaly, because all the body parts are correctly recognized, but no normal walking (or other familiar action) is detected.

Fig. 10 shows three interesting abnormal cases that occurred within a short interval in the video sequence. The actress performed some strange steps at three repetitions. In frame 2, she turns around in a bizarre manner, in frames 4 and 6, she has to move her leg over the cable in order not to stumble. In all cases, all body part trackers correctly validated the observation, but the walking tracker did not cope with the situation, thus signaling the abnormal events.

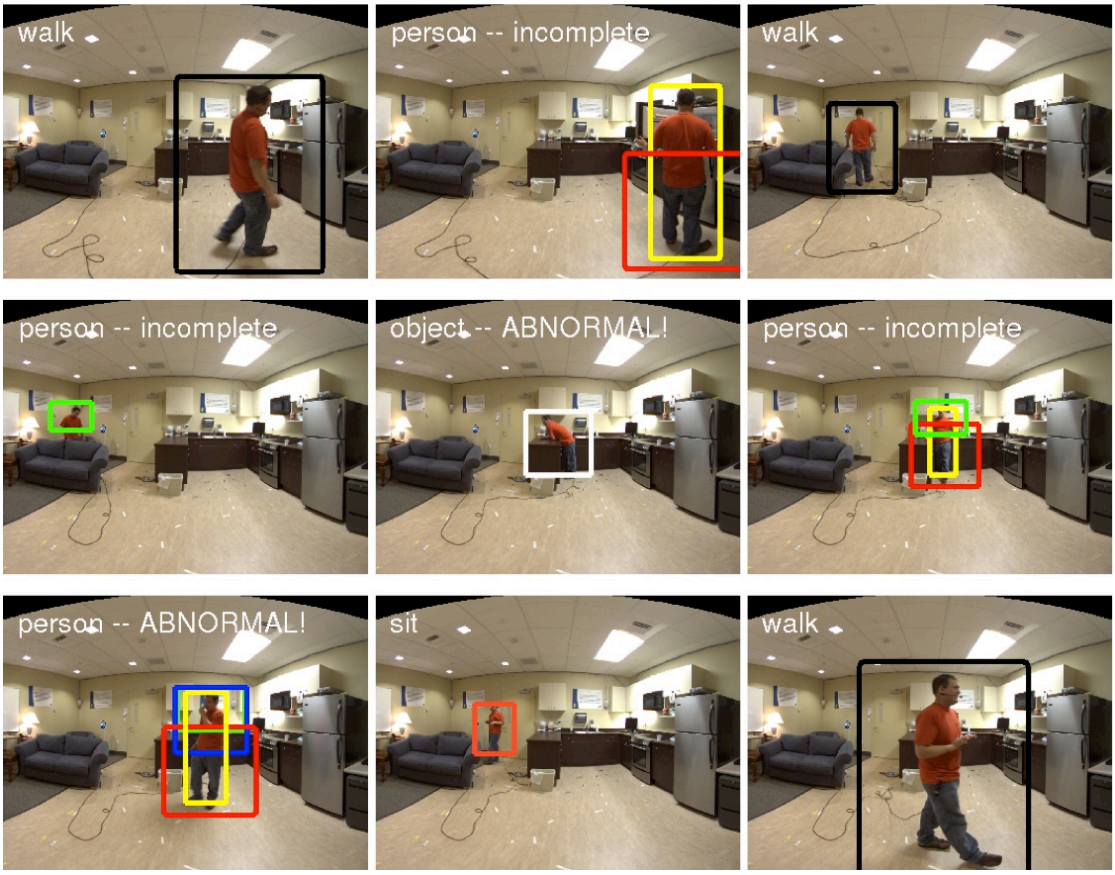

Figure 11. Results for OHSU sequence 2: The person is tracked throughout the scene, here we display some particular failure cases: Often failures are related to wrong foreground estimation (open refrigerator in frame 2). No human body part is observed in frame 5, thus the abnormal event is signaled. In frame 8, sitting is wrongly detected (see text for more details).

Fig. 11 shows some selected frames of the second OHSU sequence. The tracker tree correctly estimates some frames; other cases however can be judged as failures. An example is the distortion of the trackers due to erroneous foreground extraction (opening the refrigerator in frame 2). This new foreground part perturbs the upper body and the head tracker, which are silhouette based. In frame 4, only the head is detected, even though the person with it's upper body is visible. In frame 5, the person leans over the kitchen desk. This motion is apparently unfamiliar to all trained concepts. Since the blob tracker still tracks the person, an abnormal event is signaled in the same way as for a fall. In frame 7, the walking tracker is not active, due to shadows on the floor (the person is standing underneath the light source).

Since all the part trackers are valid, this creates a false alert. Finally, in frame 8, due to unknown reasons, the sitting tracker has higher likelihood than the walking tracker.

As we have seen, the tracker tree performs well in many situations. We have shown the benefit that comes from the detection of pre-learned actions and the discovery of abnormal events due to the hierarchical arrangement of the trackers. However, in some cases the trackers tend to fail or to be mislead by some unfamiliar observations, which causes the reasoning to break down. On the positive side it should be mentioned that in all the observed failure cases, the tree never completely broke down, but the trackers recovered automatically, for example when the refrigerator in Fig. 11, frame 2 was closed again.

5. Conclusion

In this deliverable we have shown how the tracker tree concept can be further extended and an increasing number of trackers can be integrated. In particular, compared to the reporting in DIRAC deliverable D3.9, we have added a tracker that recognizes person who is lying down on the couch, and four trackers that observe different actions that a person performs while sitting. These actions are idling, reading, drinking and picking. We have successfully brought together two tracking techniques (manifold-based silhouette tracking and spatio-temporal interest points) and integrated them in order to have one hierarchy of trackers. Action detection and anomaly reasoning in the DIRAC terminology is then feasible. We have shown quantitative performance of the trackers, and demonstrated their use in the OHSU setup. Finally, we have also pointed out where and in what cases the trackers might fail, which leads to future work considerations.

6. References

[Adam08] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. "Robust real-time unusual event detection using multiple fixed location monitors." In *IEEE Trans. PAMI*, 30(3):555–560, 2008.

[Anderson09] D. Anderson, R. Luke, J. Keller, M. Skubic, M. Rantz, and M. Aud. "Linguistic summarization of video for fall detection using voxel person and fuzzy logic." In *CVIU*, 113(1):80–89, 2009.

[Bradski98] G. R. Bradski. "Computer vision face tracking for use in a perceptual user interface". *Intel Technology Journal*, (Q2), 1998.

[Cucchiara05] R. Cucchiara, C. Grana, A. Prati, and R. Vezzani. "Probabilistic posture classification for human-behavior analysis." In *IEEE Trans. on Systems, Man, and Cybernetics*, 35(1):42–54, 2005.

[Felzenszwalb08] P. Felzenszwalb, D. Mcallester, and D. Ramanan. "A discriminatively trained, multiscale, deformable part model." In *Proc. CVPR*, 2008.

[Felzenszwalb10] P. Felzenszwalb, R Girshick, and D. Mcallester. "Cascade object detection with deformable part models." In *Proc. CVPR*, 2010.

[Hu06] W. Hu, X. Xiao, Z. Fu, D. Xie, F.-T. Tan, and S. Maybank. "A system for learning statistical motion patterns." In *IEEE Trans. PAMI*, 28(9):1450–1464, 2006.

[Johnson96] N. Johnson and D. Hogg. "Learning the distribution of object trajectories for event recognition." In *Proc. BMVC*, 1996.

[Lawrence03] N. Lawrence. "Gaussian process latent variable models for visualisation of high dimensional data." In *NIPS*, 2003.

[Li08] J. Li, S. Gong, and T. Xiang. Scene segmentation for behavior correlation." In *Proc. ECCV*, 2008.

[Makris05] D. Makris and T. Ellis. "Learning semantic scene models from observing activity in visual surveillance." In *IEEE Trans. on Systems, Man, and Cybernetics*, 35(3):397–408, 2005.

[Nasution2007] A. Nasution and S. Emmanuel. "Intelligent video surveillance for monitoring elderly in home environments." In *IEEE Workshop on Multimedia Signal Processing*, 2007.

[Nater09] F. Nater, H. Grabner, T. Jaeggli, L. Van Gool, "Tracker trees for unusual event detection", *ICCV Workshop on Visual Surveillance*, 2009.

[Nater10] F. Nater, H. Grabner, L. Van Gool, "Visual abnormal event detection for prolonged independent living", *mHealth Workshop at IEEE Healthcom*, 2010.

[Platt00] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods", *Advances in Large Margin Classifiers*, 61-74, 2000.

[Rougier07] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau. "Fall detection from human shape and motion history using video surveillance." In *Advanced Information Networking and Applications Workshop*, 2007.

[Stauffer00] C. Stauffer and W. E. L. Grimson. "Learning patterns of activity using real-time tracking." In *IEEE Trans. PAMI*, 22(8):747–757, 2000.

[Tenenbaum00] J. Tenenbaum, V. de Silva, and J. Langford. "A global geometric framework for nonlinear dimensionality reduction." *Science*, 290(5500):2319–2323, 2000.

[Veeraraghavan08] A. Veeraraghavan, R. Chellappa, and M. Srinivasan. "Shape-and-behavior encoded tracking of bee dances." In *IEEE Trans. PAMI*, 30(3):463–476, 2008.

[Willems08] G. Willems, T.Tuytelaars, L. Van Gool. "An efficient dense and scale-invariant spatio-temporal interest point detector." *ECCV*, Marseille, 2008