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: D 3.9

Incongruences detected between trackers working with weaker and stronger assumptions about the world

Date of deliverable: 31.12.2009
Actual submission date: 29.01.2010

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.))

D3.9 – INCONGRUENCES DETECTED BETWEEN TRACKERS WORKING WITH WEAKER AND STRONGER ASSUMPTIONS ABOUT THE WORLD

EIDGENOESSISCHE TECHNISCHE HOCHSCHULE ZUERICH (ETHZ)

Abstract:
We present an approach for unusual event detection based on a tree of trackers. At lower levels in the tree, the trackers are trained on broad classes of targets whereas at higher levels, they are more specific. For instance, at the root, a general blob tracker could operate and track any foreground object. At higher levels, more and more detailed information about human appearances or specific actions are included. As long as the target behaves according to expectations, a higher up tracker will always be better tuned to the data than its parent tracker at lower level. Typically, a better informed tracker behaves more robustly. But in cases where unusual events occur and the normal assumptions about the world no longer hold, they loose their reliability. In such situations, a less informed tracker, not relying on what has now become false information, has a good chance of performing better. Such performance inversion signals an unusual event. We show applications of the tracker tree concept in the scenario of autonomous living, where elderly people are monitored. We are able to spot rare events at different levels of semantic interpretation.

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.))

Table of Content

1. Introduction

In this deliverable, we describe the latest results for the incongruence concepts in tracking, as initially proposed by ETH and later generalized to become the overall vision of Dirac for `rare events'. In particular, the deliverable gives an overview of the work on tracker trees, which implement the initial idea of noting that more specialized trackers perform weaker than a less specialized one, which acts as their common parent. This reversal in performance is used as an indication that what is going on is not covered well by the collective events modeled by the more specialized trackers, hence is novel or `rare'.

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 [Dee08]. 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].

Within the variety of possible applications for abnormality detection, we address the issue of autonomous living, where mostly elderly people are monitored in their homes (*cf.* Fig. 1). In this context, fall detection is an important aspect and various solutions have been proposed for the automatic generation of alarms in suspect cases [Noury07]. The approaches include wearable, accelerometer based systems but also concepts relying on vision. Wireless wearable devices are very reliable, however they have the clear disadvantage that the concerned person may forget to wear or to recharge them. On the other hand, vision systems for fall detection (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.

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.))

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

Rather than trying to detect unusual events by modeling them directly, our approach follows the indirect route of detecting them as deviating from models of usual events. The latter are easier to come by. This said, in order to cover the wide range of unusual events that may be of interest, this calls for modeling a wide spectrum of usual events, often at different levels of granularity. For instance, the calamity of falling would be detected as deviating from all the normal categories like walking, standing, or sitting. Our proposal is to build an entire tree of trackers, as sketched in Fig. 2 and explained in detail in the following section. The aspiration is to detect an increasing gamut of unusual events, which will also gradually get more subtle and semantically rich.

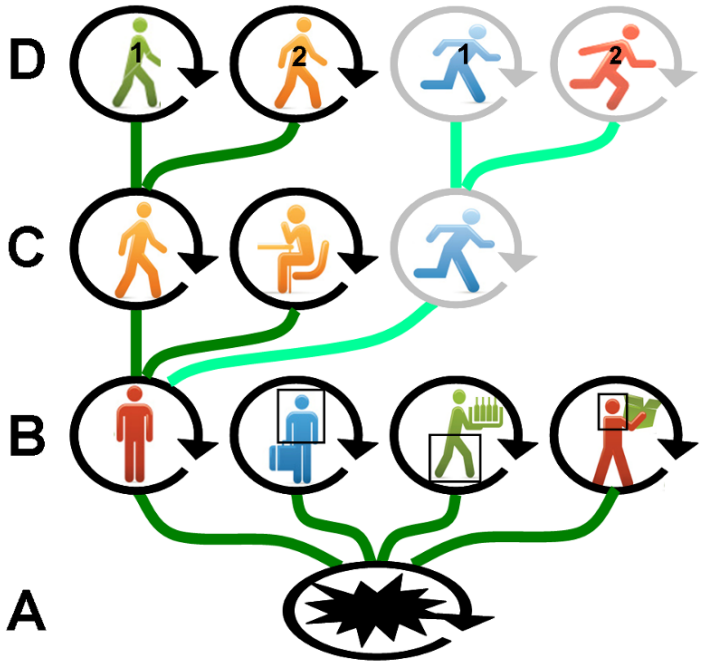


Figure 2. Overview of the proposed *tracker tree* with increasingly informed trackers for increasing levels A - D. Each black circle depicts one tracker, with the actually implemented ones in black and the pale ones for the purpose of illustration. On level A, any kind of object is tracked, level B encodes (partial) human appearances, level C goes after specific actions, whereas on level D action and person specific trackers are located.

2. Tracker Trees

2.1. Concept

The *tracker tree* described in this paper is geared toward the detection of unusual events in the home (*e.g.* for elderly care), the principle however is not restricted to this scenario.

The root node tracker is a simple, generic blob tracker, going after anything that is not background. One level up, a tracking-by-detection related framework is used for multiple body part trackers (full body, upper body, legs, head-shoulders). On this level, people are tracked, independent of their activity. One further level up, two trackers detect walking or sitting people, but more such trackers can be easily envisaged. This level could be considered an action-specific level. Then there is one

higher level, which specializes the walking tracker towards trackers that have been tuned particularly towards the gait of specific people. Hence, our hierarchy consists of multiple levels, within which families of trackers are trained to cover the normal conditions at that level. Notice that at one level there could be multiple such families. If one would *e.g.* also have a running tracker at the action level, it would make sense to also have a family of person specific running trackers one level up, just as is illustrated by pale trackers in Fig. 2. Going to higher levels, the trackers are endowed with stronger and stronger knowledge about the (normal) world. In our current implementation, all the different trackers operate autonomously, *i.e.* a tracker does not depend on the outcome of any other.

2.2. Unusual event detection

Unusual events are detected when and where a level can deal well with an event (can explain it with the available trackers), whereas none of the relevant trackers at the immediately higher level can. This is motivated by the fact that a tracker that uses more knowledge about the world should be more robust. If none of the more informed trackers can deal with the data, but the less informed one can, then this is a sign that something unusual is going on. Indeed, using more information is only advantageous as long as this information is correct. In the case of an unusual event, none of the usual, extra pieces of information apply. A performance reversal occurs in the sense that the weaker tracker better explains the data than any of the more informed trackers. An interesting aspect of the hierarchical approach is that unusual events at multiple, semantic levels can be handled and interpreted. For instance, if none of the people 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.

Before further characterizing our specific implementation of the *tracker tree* concept, we introduce in Sec. 3 a method of probabilistic shape tracking which will be used extensively in the realization of the system. Implementation details are presented in later on in Sec. 4. A number of experiments that illustrates the basic concept are shown in Sec. 5 and the report is concluded in Sec. 6.

3. Appearance based probabilistic tracking

In this section, we describe a shape based tracking approach which is based on manifold learning and nicely fits in the concept of more or less informed trackers. By means of this method, it is possible to create different trackers by combining different sets of training data and consequently making the trackers more or less informed. Most trackers in our tracker tree implementation have been generated this way.

Manifold learning is a popular technique in human activity modeling and recognition. The fact that consistent human actions have a small number of intrinsic degrees of freedom can be exploited for designing a low dimensional manifold which describes the principal aspects of the observed human activity while omitting details. Learning manifolds and mapping functions to appearance space and body pose space is for example used successfully for inferring 3D body pose from silhouettes [Elgammal04], [Jaeggli07] and also including dynamical information [Urtasun06]. When it comes to tracking, learned low dimensional manifolds can include dynamical models which are used for prior computation (*e.g.* [Lee07]).
Inspired by these ideas, we briefly describe our tracking method, which unlike other approaches does not infer 3D human poses, but is modeling and interpreting the person's appearances in a probabilistic form.

3.1. Representation

In order to encode the shape of the tracked persons, we use normalized and rescaled silhouettes obtained from background subtraction. As shown in Fig. 3 (a-c), these binary images are converted by a signed-distance transform, and each frame is reshaped in a vector.

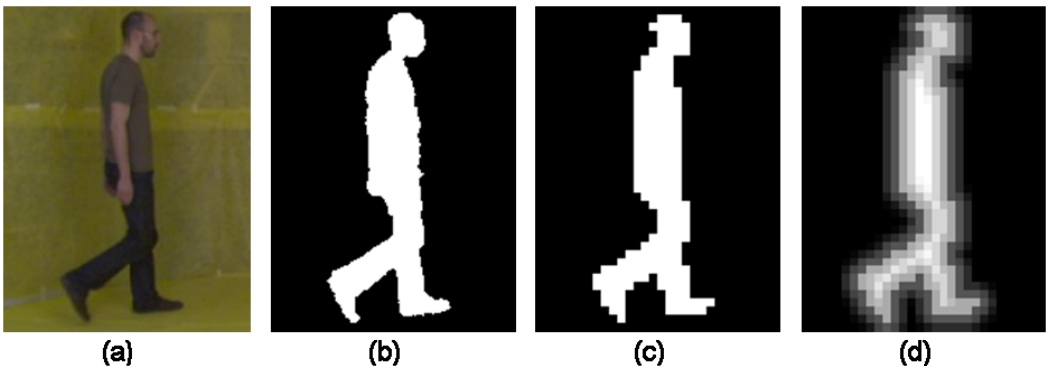


Figure 3. Image representation: (a) original, (b) segmented and rescaled, (c) distance transformed, (d) embedded and reconstructed.

3.2. Model learning

In the training phase, we generate a model representing the human appearances from a fixed number of image frames, the extracted feature vectors are combined to a feature matrix.

3.2.1. Dimensionality reduction

The high dimensionality of the shape representation space is reduced with respect to the included training data. For successful tracking we impose the requirement that the chronological order of the input frames has to be reproduced in embedding space. This means, that the Euclidean distances of consecutive frames, measured in latent space, should be small.

We use Isomap [Tenenbaum00] as a nonlinear dimensionality reduction technique, which has proven to meet our expectations and also produces interpretable manifolds. We fix the latent space to be three dimensional, encoding enough variance for successful reconstruction and still permitting efficient tracking. An example of such manifold is shown in Fig. 4. It was obtained from one person's continuous unconstrained walking. Points correspond to video frames and their temporal order is indicated by the connections. For some of the frames, the according silhouette is displayed. Grayscales mark the dimension which intuitively encodes the persons walking direction, reaching from right (light gray) to left (dark gray), with frontal/dorsal orientations in between. The manifold also represents the person's gait with open leg states being spatially separated from closed ones.

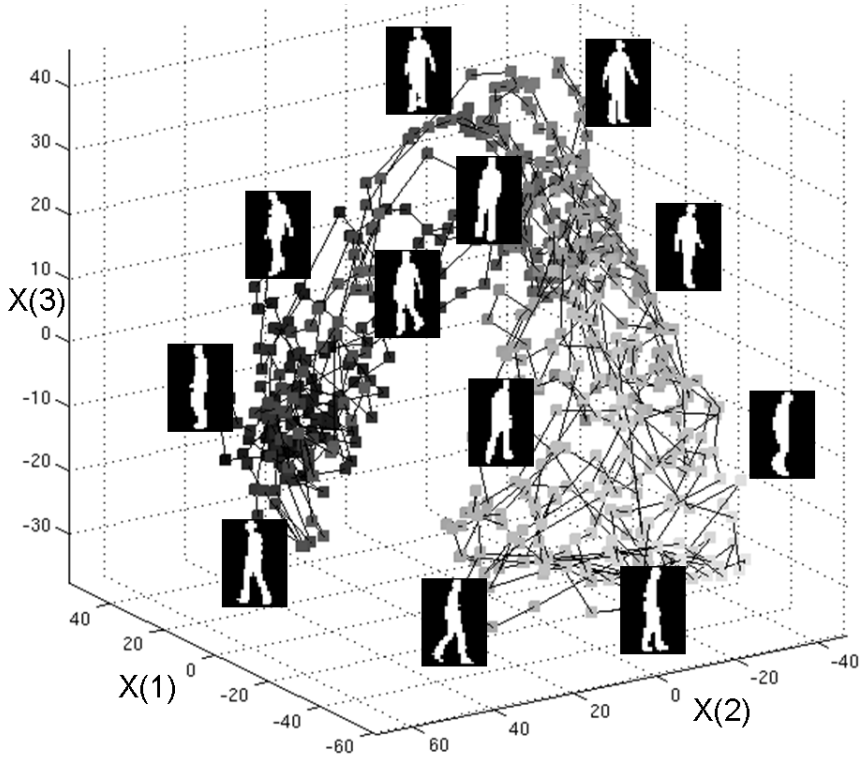


Figure 4. Visualization of the low-dimensional representation: Manifold of 600 encoded silhouettes taken from a sequence of one person's unconstrained walking (See text for details).

3.2.2. Gaussian Process regression

Isomap is a technique reducing the data based on local distances, therefore no explicit mapping is formulated in the framework (in contrast to linear dimensionality reduction techniques such as principal component analysis). For the generative association between latent space and image representation, we learn a regression function by using Gaussian Process [Rasmussen06}. To this end, we employ the Gaussian Process toolbox [Lawrence03] and compute the mapping function $\mathcal{M}: \mathbf{z} \mapsto \mathbf{y}$ which estimates the shape representation $\mathbf{y}$ and its variance for any latent point $\mathbf{z}$. For the ease of notation, we simply denote $\hat{\mathbf{y}}(\hat{\mathbf{z}})$ the predicted shape obtained by mapping a predicted latent point $\hat{\mathbf{z}}$. In Fig. 3 (d), the image space representation is shown after Isomap embedding and Gaussian Process reconstruction of Fig. 3 (c).

3.3. Tracking

After learning a low dimensional manifold representing a set of encoded silhouettes for a specific action class, the next step is to explain unseen test sequences within this model. This is done with a Bayesian tracking approach [Doucet00] by using a six dimensional particle filtering technique. For every hypothesized sample $\theta_i = \{u_i, v_i, s_i, \mathbf{z}_i\}$ the observation likelihood is evaluated, where $\{u, v\}$ is the bounding box location in the image, s its scale (with fixed aspect ratio), and $\mathbf{z}$ the tracked shape in the low-dimensional embedding space.

3.3.1. Likelihood formulation

For every particle θ_i, the likelihood of the shape observation given this sample is estimated, using the following formulation:

$$\mathbf{p}(\mathbf{y_{obs}}|\theta_i) \propto \mathcal{N}\{d(\mathbf{y_{obs}}, \hat{\mathbf{y}}(\theta_i)); 0, \sigma^2\} \qquad (1)$$

with $d(\mathbf{y_{obs}}, \hat{\mathbf{y}})$ a distance function between the observed $\mathbf{y_{obs}}$ and the predicted $\hat{\mathbf{y}}$ shapes, both represented as distance transformed silhouettes. The likelihood is in this case normally distributed with zero mean and σ^2 variance. More precisely, if we denote the shapes $\mathbf{y_{obs}}$ and $\hat{\mathbf{y}}$ as vectors of K elements y_k and $\hat{y}_k$, $k = 1 \ldots K$, the distance function becomes

$$d(\mathbf{y_{obs}}, \hat{\mathbf{y}}) = \sum_{k=1}^{K} \beta_k |y_k \hat{y}_k| \qquad (2)$$

with

$$\beta_k = \begin{cases} 1 & \text{if } \operatorname{sign}(y_k) \neq \operatorname{sign}(\hat{y}_k) \\ 0 & \text{if } \operatorname{sign}(y_k) = \operatorname{sign}(\hat{y}_k) \end{cases} \qquad (3)$$

such that equally signed pixels in the observed and the predicted shape do not increase the distance.

This likelihood formulation makes it possible to obtain a posterior probability density function over all samples θ_i given the shape observation $\mathbf{y_{obs}}$ for each frame in the test sequence.

3.3.2. Illustration

The proposed silhouette based tracking approach is illustrated in Fig. 5, where two frames of a publicly available video sequence (video downloaded from www.openvisor.org, 2009/05/27) are shown. On the upper left of each frame, the background subtracted silhouette is presented, and on the lower left, the image space representation of the particle filter sample with the highest weight is shown. The latter corresponds to the shape encoded in the low dimensional model which best matches the observed silhouette. Trained in a controlled lab setup, the learned model can nonetheless be applied on any sequence and accurate tracking is possible even with noisy background subtraction. The tracking approach works well as long as the observed shape can be well described by the model, *i.e.* the likelihood term has clearly pronounced peaks, whereas it results in small posterior probabilities for out-of-model observations.

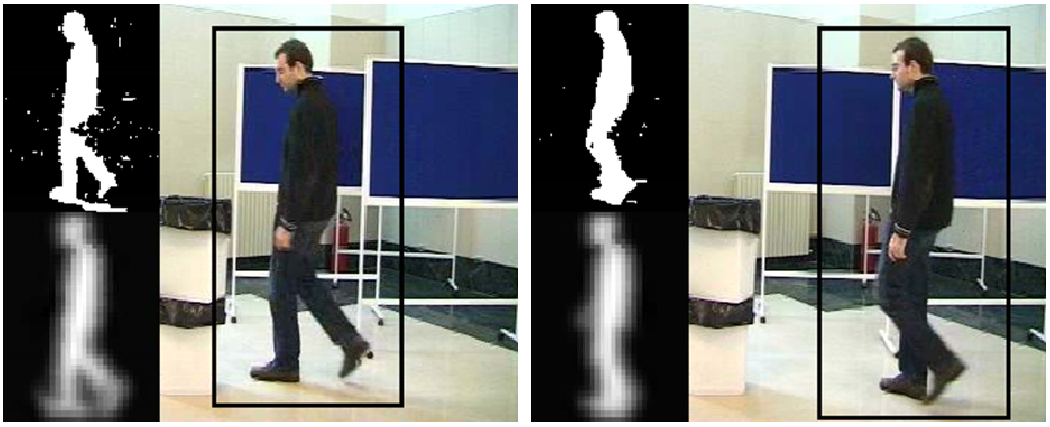


Figure 5. Application of the proposed silhouette based tracking approach on a publicly available video sequence, from which two frames are shown. The background subtracted image and the best corresponding shape in the model are depicted on the left.

A second sequence in Fig. 6 shows the applicability of the proposed manifold tracking technique to an outdoor video sequence. A human target is tracked accurately even with in noisy data (upper left: corrupted silhouette) and a corresponding silhouette can be reconstructed from the tracking in the underlying latent space (lower left).

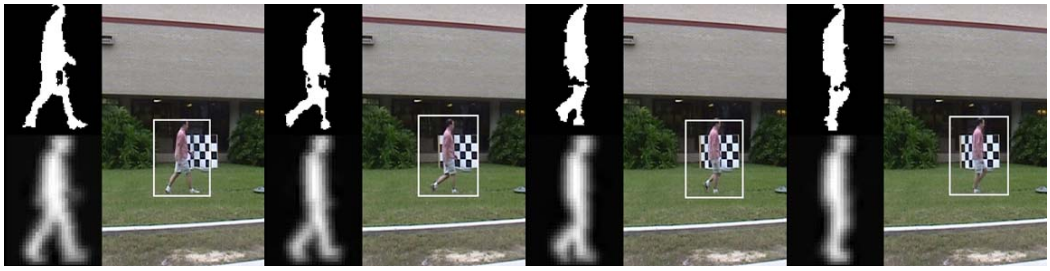


Figure 6. Application of the silhouette based tracking to an outdoor video sequence. The images show stable tracking performance as well as accurate silhouette reconstruction from noisy data.

3.3.3. Tracking Priors

In this probabilistic formulation it is possible to easily incorporate scene-specific knowledge by adding tracking priors. For example, a ground plane prior is used for the stabilization of the tracker and for limiting the search space of the particle filter.

4. System implementation

The outline of the proposed *tracker tree* was already presented in Fig. 2, where schematically all the implemented trackers were depicted in black circles. The tracker instances are placed on four hierarchical levels A - D. Next, each level will be detailed by explaining the corresponding trackers. In general, we use both, previously proposed state-of-the-art methods as well as the custom-built technique of Sec. 3 trained for different levels of generalization. The only restriction therein is the number of latent dimensions, which requires a limitation of the input data in terms of intrinsic variance such that the embedded representation remains meaningful[1].

4.1. Level A

The least informed and thus most general tracker in the tracker tree is meant to trace any foreground object, subject to any kind of deformation. This is a simple, generic blob tracker which has no information about the nature of the foreground object. For this purpose, a color histogram based implementation of the CAMShift tracking approach [Bradski98], [Comaniciu03] is used.

[1] Truly, this is a vague definition, we noticed however that the consistent human actions we considered can be represented in three dimensions with the method delivering well-interpretable results. More dimensions could of course be included

4.2. Level B

The trackers on this level make a first step towards the description of human body shapes. For tracking people independent of their activity, we use a set of body part trackers, namely for the lower body, the upper body and the head-shoulders, depicted on the right part of level B in Fig. 2. For these three trackers, we generate an embedding using the method of Sec. 3. The image data provided during the training procedure is chosen with respect to the specified target class of the tracker. For each of these three trackers, the obtained low dimensional manifolds are similar to the one shown in Fig. 4 and encode the principal motion such that tracking remains possible within this manifold. Following the method, particle filter based tracking is accomplished and the output is a probability that quantizes the match between observation and body part model.
In addition, the leftmost tracker instance of level B in Fig. 2 is an implementation of the human detector based on discriminatively trained part models [Felzenszwalb08]. It is used as provided by its author on the website and follows a tracking-by-detection approach.

4.3. Level C

On this level, we are interested in tracking different basic human actions. As shown in Fig. 2, we dispose of two different action trackers, describing walking and sitting humans respectively, both based on the method described in Sec. 3.
For generalization reasons, the person upright and person sitting trackers are both trained with image sequences from multiple persons. The resulting low dimensional manifold in the walking person case looks similar to the one in Fig. 4, but contains more data and hence includes more variance. For the sitting case, only frontal poses of persons sitting on different chairs are considered, with the manifold encoding mainly leg and arm positions as well as sitting height.

4.4. Level D

On the most specific level in the current tracker tree, the aim is to track one particular person performing a specified action. In that sense, this is a specialization of the action specific trackers one level down and we rely therefore on a modification of the nonpersonal walking tracker of level C. The goal is here to model the appearance of two persons separately by providing individual training data and learning two distinct manifolds. Besides tracking the considered person, the tracker outputs a probability quantizing how well the observed silhouette fits in the individual model. This output score can also be evaluated relative to the non-personal walking tracker on level C: The discrepancy in terms of posterior probability between person specific

and non-personal trackers provides information on the belonging of the observation to the individual model.

5. Experiments

5.1. Experimental setup

We illustrate our implementation of the *tracker tree* on three different video sequences. We use videos recorded with static cameras at a frame rate of at least 15 fps. Two of the videos were recorded in our lab setup, one is a public sequence. Concerning the trackers built with the method of Sec. 3, we apply background subtraction for the extraction of the silhouette in each frame. Shapes are then represented in a 30 x 40 pixel image for the entire body trackers, the body part trackers involve smaller representations. Training data was recorded in a controlled lab environment and manifolds were constructed from approximately 2500 frames. The parameters used for the Bayesian tracking (σ^2 and noise parameters in the particle filter) are estimated from the training data, the number of particles is empirically fixed to 1500 and initialization is done manually for the first frame of the tracking sequence.

5.2. Experiment 1: System operation

In a first experiment, we want to show the properties of the different trackers in the tracker tree applied in a scene without any abnormality. In Fig. 7 an extract of the publicly available sequence (video downloaded from www.openvisor.org, 2009/05/27) is given, showing a person entering the room, walking a couple of steps and sitting down. Four images from this video are displayed in Fig. 7 (a-d) with the tracker's output probabilities plotted on a logarithmic scale for the entire sequence in the lower part of the figure. The instants corresponding to the presented frames are indicated by vertical black lines in the plot. In the images, the white ellipse indicates the general object tracker [Bradski98}, the other bounding boxes correspond to the trackers as referenced in the plot legend. In the probability graph we introduce an empirically determined threshold which is used to decide on the reliability of the tracker. In other words, this threshold could indicate to the system whether the particular tracker is likely to explain the observation. The threshold is indicated by a black dotted horizontal line and accordingly, only trackers with above-threshold probabilities are visualized in the frames. Note that the yellow bounding box in the images corresponds to the part model detector [Felzenszwalb08], for which no

probability output is available and thus only a binary curve is plotted. In sequence 1 however, this detector always is active.

From the lower part of Fig.7 it can be seen that as long as the person is walking (a), the observation is well explained by the underlying model of the walking tracker (black bounding box and line) as the output probability is high. When the person starts to sit down (b) the walking tracker fails and also the lower body part tracker has a transitory instability. Thereafter, the person remains seated (c) and the sitting tracker is able to explain the situation. All the part trackers are also active. During transitions (d), when the person is turning on the chair, he apparently moves in a way which was not included in the training data. The sitting tracker is very sensitive to the person's rotation and therefore does not generalize to different viewing angles.

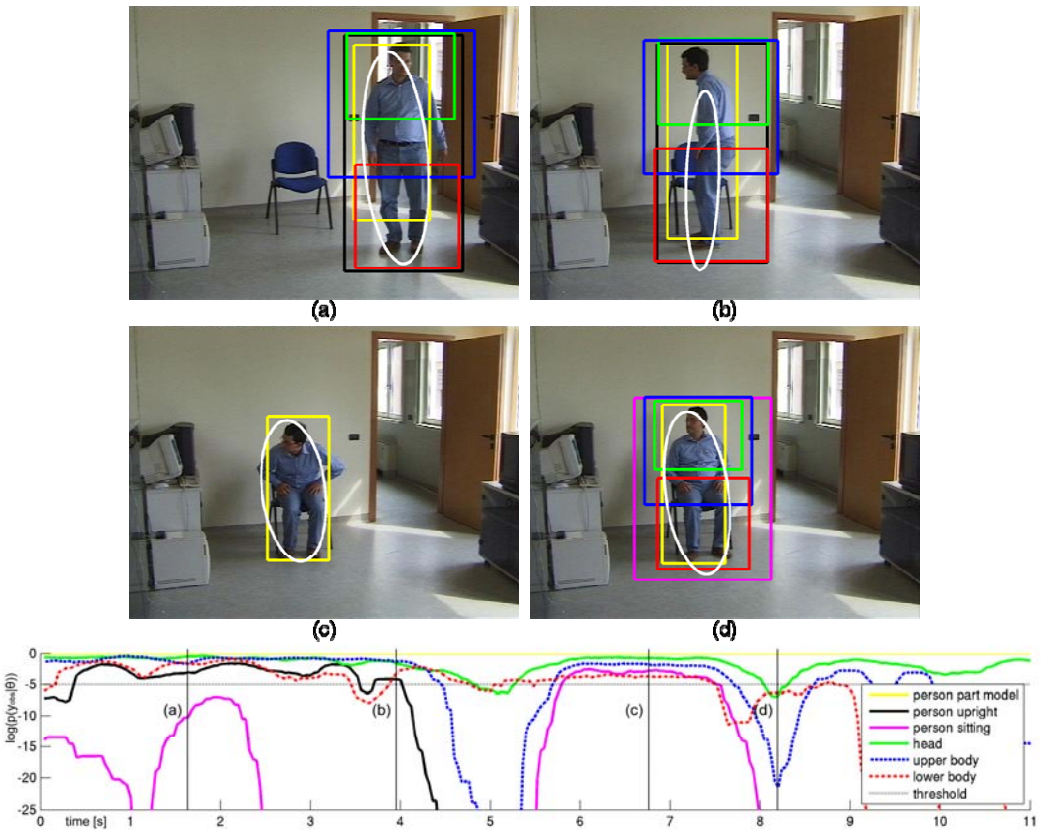


Figure 7. Results for the first test sequence: The behavior of the different trackers in the framework is presented. The images on top show four frames from the sequence, their corresponding instants are indicated in the plot on the lower part. The confident trackers are visualized in the frames with different colors (see text for interpretation, figure is best viewed in color, the full videos can be downloaded from www.vision.ee.ethz.ch/fnater/tracker-trees/.

5.3. Experiment 2: Fall detection

In a second sequence presented in Fig. 8, a person is in the room (a), walking behind the sofa towards the shelf on the left (b), taking a book and reading it (c), turning towards the other shelf (d, e), where the book is placed. Coming from behind the chair on the right (f), he wants to move to the front, when suddenly he stumbles across the edge of the carpet (g) and falls (h). The same color code is used as in the previous video, omitting the unused sitting tracker.
Again, the relative tracker outputs allow for interpretations of what is going on in the scene. For example, the system detects an occlusion (b, c, f) if the walking tracker fails, but some body parts, in this case head-shoulder and upper body are still visible. In the same category falls the event in (e), where the person is holding a book such that the upper body tracker is perturbed. This demonstrates the use of body part trackers, especially in living room scenarios where multiple occluders are usually present.

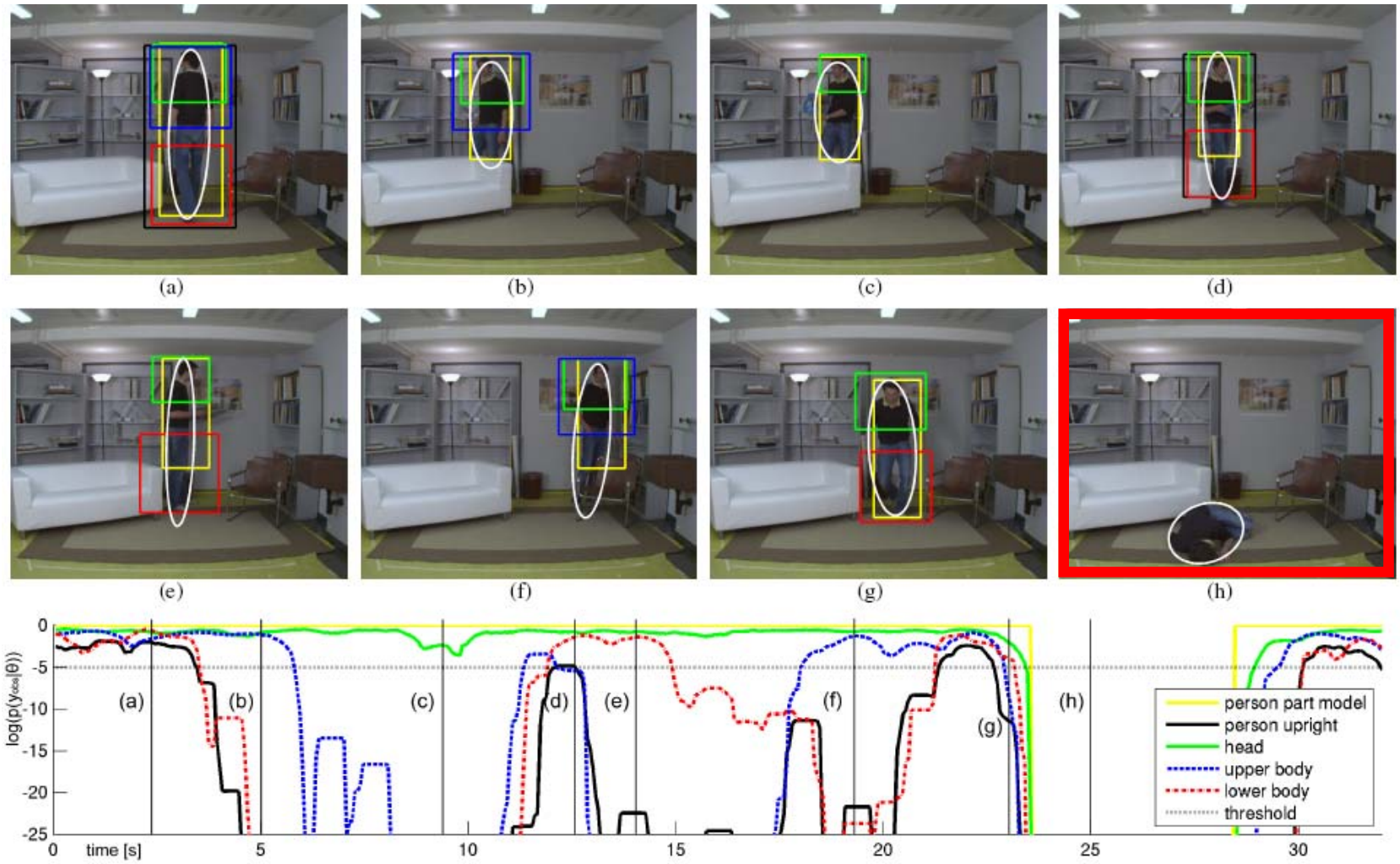
A fall is detected when a foreground object is tracked but cannot be explained by any of the more specific trackers, as seen in Fig. 8 (h). None of the tracking models trained for normal human behavior can cope with this special situation. Here we additionally make use of the sequential information that we had observed a person right before the fall happened.

Figure 8 (next page). Results for the second test sequence: The system's reaction in the case of abnormal events is illustrated, including occlusions, out-of-model appearances and a fall. The instants corresponding to the displayed frames are indicated in the plot by vertical bars. The confident trackers are displayed in the images.

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.))



5.4. Experiment 3: Limping

In a third experiment, we show the tracker tree's behavior to a person in the scene who is limping. This action was not included during the training phase and should therefore be detected as unusual.

In a short sequence shown in Fig. 9, a person is walking normally towards the right, turns around (a) and suddenly starts to limp (b). Finally, his legs are occluded by the desk in the foreground (c).

From the bottom part of Fig. 9, where the same signals are plotted as in the previous experiments, it can be seen that all the trackers behave normally until approximately second 3, when the person is walking normally. When he start to limp, all the body part trackers (on level B in the implemented tracker tree) are still well following the target, the walking tracker from level C however shows periodic drops for the probability. This occurs in the part of the walking cycle, where the limping is characterized and an abnormal event is noted from the fact that all lower level trackers agree, whereas no higher level tracker explains the situation. In the end of the sequence, where the legs are invisible, no evidence is given for abnormal walking, due to the fact that not all body part trackers remain active.

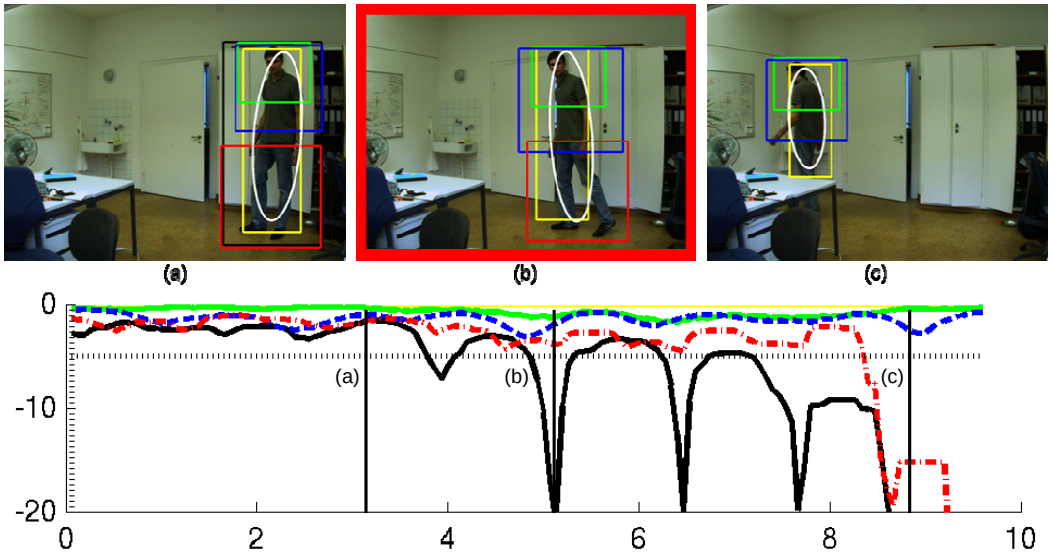


Figure 9 . Results of experiment 3 showing the tracker trees output to a limping action. The abnormal event is detected temporarily during the walking cycle, all part trackers agree whereas the (normal) walking tracker cannot cope with the situation.

5.5. Experiment 4: Intruder detection

For the intruder detection task, we include the two person specific walking trackers (Fig. 2, level D) as well as the multiple person walking tracker from level C. In Fig. 10, the principle is demonstrated with three short sequences, starring two familiar and one unknown person respectively. In Fig. 10 (a,d), we track the first familiar person. In this case, the probability outputs for the multiple person tracker (black) and the person 1 tracker (cyan) correlate while the person 2 tracker (orange) less well explains the situation. It is the opposite in Fig. 10 (b,e) where the second known person is tracked. If a third, unknown person is in the scene (Fig. 10 (c,f)), its appearance is well modeled by the multiple person tracker while the person specific ones tend to fail. The person must therefore be an intruder (in the sense of someone not known to the system yet).

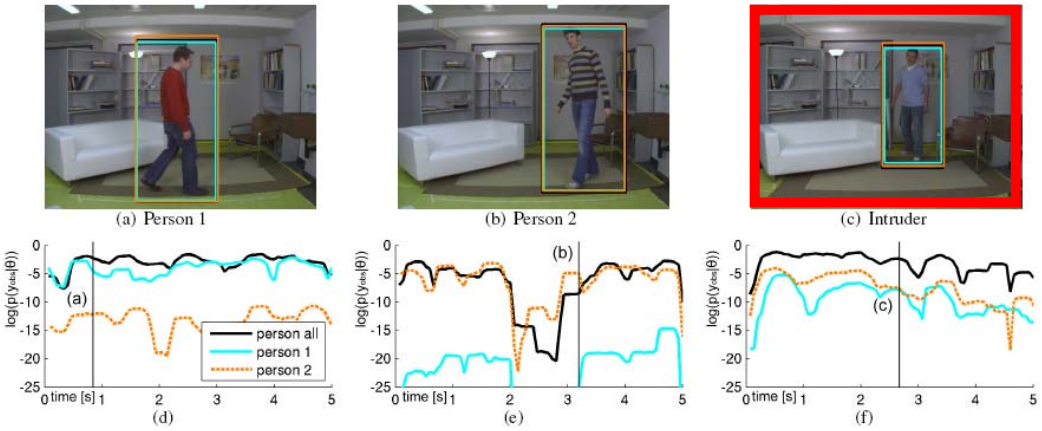


Figure 10. Results for the person identification task. Three different persons are shown walking through the room, the matching probabilities of the observation to the general (black) and the two person specific models (cyan and orange) are plotted beneath the corresponding frames. For the sake of visibility, all the other trackers in the system are omitted.

5.6. Discussion

We see from the experiments that it is useful to interpret the independent tracker outputs in a tree-like structure. Irregularities in the combination of the different trackers can be detected. Reasoning is possible on these irregularities since all the trackers have previously determined tracking capacities and target classes. Once

expanded and used together with additional temporal information, such a tracking framework is a powerful tool for unusual event detection.

Finally, the major limitation of the presented trackers built with the method of Sec. 3 should be pointed out. These trackers are all based on silhouette representations obtained from background subtraction and are therefore bound to perform badly under certain circumstances (moving camera, illumination changes, high noise level, *etc.*). The intruder detection as well task is only supported by the person's shape and works fine as long as this shape is characteristic for the individual. Other more sophisticated techniques (*e.g.* gait recognition) could be used instead for improving the performance. However, these issues concern the current implementation and do not affect the validity of the presented concept.

5.7. Tracker Tree extension: picking-up action

In a real-world application scenario, a fixed model is in most cases not suitable for the description of the entire range of human behavior that can occur in people's homes. It is much more desirable to start from a basic general model and adapt this model to the specific behavior of the monitored person. Within the tracker tree, the addition of such more specific trackers or trackers focusing at different aspects of human behavior is straightforward, since they can just be integrated at the correct location in the tree. The only requirement is the availability of labeled training data of one specific action, which we want to include in the new tracker.

We added the picking up action to the tracker tree, by training a low dimensional model (as proposed in Sec. 3) for with the according training data. To this end, we instructed a few persons to perform a picking up action multiple times, always being oriented in the same direction with respect to the camera. A three dimensional manifold was extracted from this training, in which run time observations can be tracked. The tracker was then added in the tracker tree on the action specific level, as seen in Fig. 11

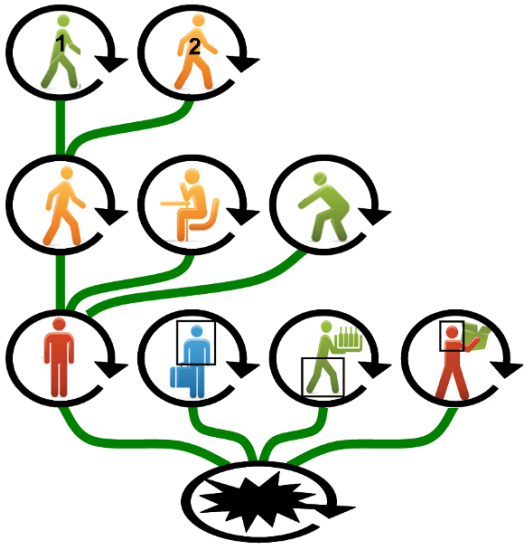


Figure 11. Augmented tracker tree, with the picking up tracker included on the appropriate level.

We subsequently performed tests with the augmented tracker tree in the same setting as for the previous experiments. As shown in Fig. 12, the picking up tracker (orange) tracks the person precisely, when he is performing this action. It is interesting to note that this tracker outperforms the (yellow) person detector and the (white) foreground object tracker in terms of accuracy. This confirms one of the basic assumptions of the tracker tree, that more informed trackers perform better as less informed ones, as long as the 'world' they see behaves according to their implicit assumptions. When it comes to a fall of the person in the scene, this event is still detected as abnormal in the tracker tree, as shown in Fig. 12 (c).

Figure 12. Living room sequence of a person who walks (a), wants to pick up a pen (b) and falls (c). The different trackers are indicated with the colored bounding boxes and the red frame marks the hierarchical incongruence in the tracker tree that is spotted in (c).

6. Conclusions and future work

We proposed *tracker trees* as a way to detect unusual events in cases where these are not modelled explicitly. The underlying idea is to build a hierarchy of trackers, where the position in the global structure depends on how strong the prior expectations about the world are that are being exploited. We have argued that the multi-layer hierarchy allows one to make rather specific interpretations about the kind of unusual event that has occurred. High up in the tree semantically more complicated events are detected than lower down.
Clearly, the tracker tree as proposed here is just an early example of how such structures may be put to use. More exhaustive and detailed experiments will be performed in the future and several improvements will be added.

Firstly, so far, the different trackers all run independently, and their performance levels are compared. As long as the assumptions used by a higher-level tracker are correct, such tracker has a better chance in successfully tracking the target. They can better distinguish target from background and can make better predictions about the future. Thus, it would be beneficial for the entire system that lower-level trackers, which operate without such strong expectations, would be helped by the successful higher-level ones to keep on track. On the other hand, the lower-level trackers are typically simpler and therefore less time consuming (*e.g.* a blob tracker vs. a full articulated body motion tracker). They could help the higher levels to focus their (expensive) attention on the most promising parts. Then again, with bi-directional information flows between lower and higher levels, a danger of instability will occur, especially in cases where the higher levels put wrong information into the system. This would be the case if the assumptions they make no longer correspond with reality. As a consequence, the detection of performance reversals as related to unusual events is also important to keep the interconnected tracker system stable.
Secondly, we think of expanding the system to include scene and temporal information, in that certain events are only normal at predefined image regions and during certain times (of the day). This information should of course be trained from run-time observations.
Thirdly, the current system is still pretty limited in what it can detect. The addition of complementary trackers and detectors is called for and one can think of different directions of extension. For instance, we are working on trackers that are dedicated to walking with different emotions (*e.g.* joyfully, depressed, with anger, with fear, neutrally). This family of trackers would then exist next to the person-specific

walking trackers at level D and represent a second, independent specialization of the generic walking tracker. The tracker tree could also be further extended on other levels. In particular if the application scenario is changed, the addition of different action trackers seems necessary. Moreover, the tree is not limited to four levels, if further granularity is required, additional levels could be added easily.
Finally, it is clear that unusual is not the same as important. Not all unusual events detected by such system will be relevant, and vice versa an extensive such system will contain explicit trackers and detectors for several relevant cases, *e.g.* for angry walking.

7. 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.

[Comaniciu03] D. Comaniciu, V. Ramesh, and P. Meer. "Kernel-Based Object Tracking." In *IEEE Trans. PAMI*, 25(5):564-575, 2003.

[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.

[Dee08] H. M. Dee and S. A. Velastin. "How close are we to solving the problem of automated visual surveillance?" *Machine Vision and Applications*, 19(5-6):329–343, 2008.

[Doucet00] A. Doucet, S. Godsill, and C. Andrieu. "On sequential monte carlo sampling methods for bayesian filtering." *Statistics and Computing*, 10:197–208, 2000.

[Elgammal04] A. Elgammal and C. Lee. "Inferring 3d body pose from silhouettes using activity manifold learning." In *Proc. CVPR*, 2004.

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

[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.

[Jaeggli07] T. Jaeggli, E. Koller-Meier, and L. van Gool. "Learning generative models for monocular body pose estimation." In *Asian Conference on Computer Vision*, 2007.

[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.

[Lee07] C.-S. Lee and A. Elgammal. "Modeling view and posture manifolds for tracking." In *Proc. ICCV*, 2007.

[Gong08] 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.

[Nater09] F. Nater, H. Grabner, T. Jaeggli and L. Van Gool. "Tracker Trees for Unusual Event Detection." *ICCV Workshop on Visual Surveillance*, 2009.

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

[Noury07] N. Noury, A. Fleury, P. Rumeau, A. Bourke, G. Laighin, V. Rialle, and J. Lundy. "Fall detection - principles and methods." In *IEEE Engineering in Medicine and Biology Society*, 2007.

[Rasmussen06] C. E. Rasmussen and C. K. I. Williams. "Gaussian Processes for Machine Learning." *The MIT Press*, 2006.

[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.

[Urtasun06] R. Urtasun, D. Fleet, and P. Fua. "3d people tracking with gaussian process dynamical models." In *Proc. CVPR*, 2006.

[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.