Chapter 1

Learning to learn new models of human activities in indoor settings[1]

1.1 Introduction

Biological cognitive systems have the great capability to recognize and interpret unknown situations. Equally, they can integrate new observations easily within their existing knowledge base. Autonomous artificial agents to a large extent still lack such capacities. In this paper, we work towards this direction, as we do not only detect abnormal situations, but are also able to learn new concepts during runtime.

We aim at the interpretation of human behavior in indoor environments. Possible applications go from the main IM2 scenario, i.e. analysis and understanding of meetings, to monitoring of elderly or handicapped people in their homes in order to ensure their well-being. The indoor setting triggers interesting issues, such as the adaptation of pre-trained knowledge to a particular room scene filmed with a different camera or to an unknown person with an individual behavior style, whereas *real* abnormalities must still be detected.

One main limitation of automated surveillance approaches is their need for an offline prior training with many labeled data. Furthermore, no training sequence contains a comprehensive set of all the situations to expect and any surprising new event can appear in only a few frames. In order to overcome these limitations, we propose to start from an initially trained set of basic activities and incorporate an on-line update mechanism. Minimal human annotation, *i.e.,* labeling one sample per new activity is required to include semantic meaning. Hence, we are able to incorporate new activity

[1] This chapter was written by Fabian Nater, Tatiana Tommasi, Luc Van Gool, and Barbara Caputo. (Equal contribution of both first authors.)

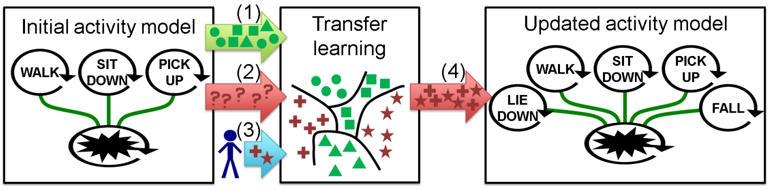


Figure 1.1: Schematic overview of our approach to combine activity tracking with transfer learning. In surveillance videos, an initial model recognizes familiar activities (1) or detects abnormalities (2). Together with minimal human interaction (3), the transfer learning algorithm returns labels (4) such that the activity model can be extended with new classes.

concepts during runtime and recognize them in the future.

In the rest of this chapter we review related work in Sec. 1.2 and present our approach in Sec. 1.3. The employed techniques for human activity tracking and transfer learning are briefly discussed respectively in Sec. 1.4 and Sec. 1.5. Sec. 1.6 reports on our experimental results, while Sec. 1.7 concludes the paper.

1.2 Related Work

The detection of abnormal events is a popular field of research, many techniques detect abnormalities as outliers to previously trained models of normality. Successes include surveillance scenarios, such as the works by Stauffer and Grimson (2000) and Adam et al. (2008) or human behavior analysis, as for example in Boiman and Irani (2005) or Nater et al. (2010). On the other hand, abnormalities can also be modeled explicitly. This is often done in our target scenario, the visual detection of a fall, *e.g.*, (Anderson et al., 2009; Nasution and Emmanuel, 2007; Cucchiara et al., 2005).

In order to interpret human motion, the person in the scene usually has to be tracked first, accounting for various appearance and scale changes. Such methods reach from generic blob-trackers (Comaniciu and Meer, 2002) to sophisticated articulated body motion trackers in tracking-by-detection frameworks (Andriluka et al., 2010). One step further, the recognition of human actions often refers to the classification of tracked motion patterns into multiple action categories, such as for instance (Efros et al., 2003; Liu et al., 2008)). In order to learn these action classes, a vast amount of labelled training data is required in most cases and it is thus hard to model very specific or unexpected activities that only occur rarely.

The use of transfer learning for activity recognition problems has been introduced in recent works for example for cross view action recognition (Liu et al., 2011), for domain adaptation (Xian-ming and Shao-zi, 2009; Yang et al., 2010; Hu et al., 2011) or to transfer across sensor networks (van

Kasteren et al., 2010). Furthermore, in a scenario similar to ours but not using video data, Rashidi and Cook (2011) use transfer learning to adapt models of daily activities between different residents in different smart homes. However none of these works consider the possibility to update the set of class knowledge models when the newly acquired information contains actions which were not seen before. In visual object classification, knowledge transfer is applied to solve a N' class problem when N classes are already known, with N' and N disjoint groups, as proposed in Lampert et al. (2009) and Jie et al. (2011). On the other hand, training to discriminate $(N + N')$ classes when the model for an N-class problem was previously learned, is known as *class-incremental learning* and only few attempts have been made to determine a principled technique for this process (Muhlbaier et al., 2009; Zhang et al., 2006).

1.3 Proposed Approach

We focus here on the problem of recognizing the activities of a person in an in-house scenario; our algorithms can be easily applied also to other settings, such as the IM2 meeting scenario. To this end, we propose to use a set of activity trackers. Each tracker is trained to one specific activity class. Known concepts can be recognized and labeled, while abnormal events are detected as unknown activities.

For an increased flexibility and to learn the unknown activities, we propose to augment this static model with an update procedure, based on transfer learning. To classify the unknown samples, we build a multiclass model which exploits prior knowledge of known classes and incrementally learns the new actions. The procedure is outlined in Fig. 1.1. The central block receives labeled (Arrow 1) and unknown (Arrow 2) samples from the activity trackers. Based on minimal human annotation (Arrow 3), it labels the previously unknown activities (Arrow 4). In a final step, the newly labeled activities are integrated in the previous model besides the initial trackers. In this sense, the transfer learning algorithm acts as an artificial expert.

The interaction of the two techniques is useful due to their complementary nature:

- Generative tracking with multiple activity trackers provides labels for familiar activities and detects abnormal situations. In both cases, the location of the person is determined as bounding box. (Sec. 1.4)

- Discriminative classification interprets the abnormal situations in order to label new activities. Knowledge transfer uses prior information from known classes for a more efficient and accurate labeling of new ones. Human annotation of at least one frame is necessary to provide the desired semantic label. (Sec. 1.5)

The approach has several application-specific advantages. Firstly, if only few labeled samples of some actions are available, we can exploit prior knowledge acquired under different conditions in terms of location, observed person and employed recording camera. Furthermore, human annotation of one sample per class enables the semantic interpretation of the activities. For example, it is now desirable to include a fall in the model, in order to automatically take appropriate action in case it is detected again, *i.e.,* call an ambulance. Besides that, the model continuously becomes richer in what it knows, such that diverse activity concepts can be recognized and the performance increases over time. Finally, a shift in an activity concept, *e.g.,* a person gradually starts to limp, can also be integrated.

In the following two sections we provide details for the activity tracking and the transfer learning and show how the two parts interact.

1.4 Activity Tracking for Unusual Event Detection

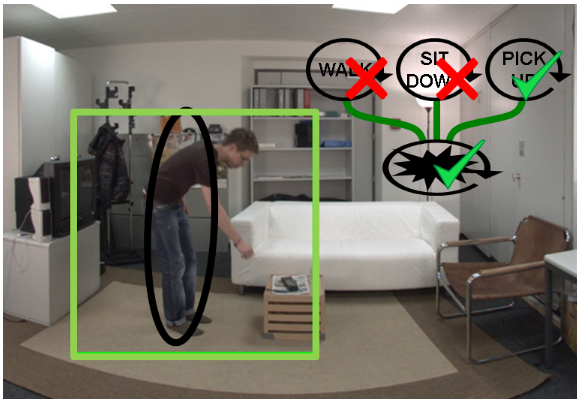


Figure 1.2: Simultaneous tracking and activity spotting: A person in the scene is always tracked by the foreground blob tracker in black. This tracker provides unlabeled samples (Arrow (2) in Fig. 1.1). The more specific activity trackers simultaneously track the person and determine his activity. If one is active (picking up in green), it overrules the blob tracker and provides labeled bounding boxes (Arrow (1) in Fig. 1.1).

In tracking, the aim is to follow the motion of the person throughout the video and account for various appearance and scale changes.

We follow the work of Nater et al. (2009) where simultaneously a person is tracked and the performed action is determined. To this end, multiple activity trackers are used and each of the trackers is trained to a specific aspect of human motion. As long as the person in the scene behaves according to the expectations, there will be one specific tracker which recognizes

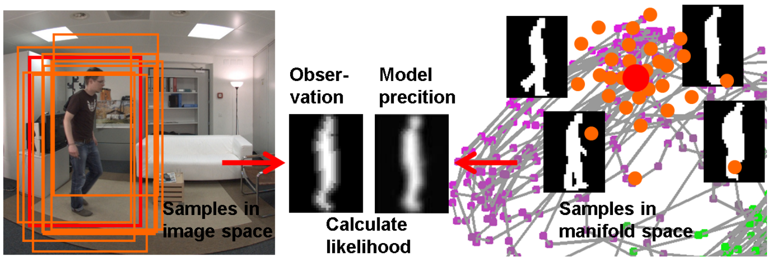


Figure 1.3: Schematic overview of the tracking technique: In a particle filter approach, samples live both, the image space and the manifold space. Comparison and likelihood estimation is performed on the silhouettes, and a posterior probability density is obtained.

the activity, as shown in Fig. 1.2. However, if none of the generative activity trackers can explain the situation, but a less informed foreground blob tracker still tracks the target, this performance reversal signals an abnormal event. In the following we briefly review the employed methods.

Activity modeling and tracking. In a first step, we train a low dimensional model in order to describe the observed training data. To this end, silhouettes of a human person are extracted from the training video sequences and are represented on a three-dimensional manifold. Isomap (Tenenbaum et al., 2000) is used as the dimensionality reduction technique, because it ensures that local distances remain similar as in the original data. To be able to infer the original silhouette space from the model, we learn a Gaussian Process regression (Rasmussen and Williams, 2006; Lawrence, 2003) on the training data. One model is learned for each activity. Initially, *walking, sitting down* and *picking up* are learned from training data in a lab setup. Typically, several hundred frames are required per activity to train non-overfitting models.

The models are subsequently applied to new sequences in living-room settings. After background subtraction, the binary observation image and the low dimensional manifold are sampled with a particle filter. From frame to frame, particles are propagated and re-weighted with respect to the likelihood between model prediction and observation. This is sketched in Fig. 1.3. At each time step, a posterior probability is available which gives an indication of how well the tracker explains the observation.

All available trackers are run in parallel. A user-defined threshold, applied on the activity trackers' posterior probabilities, determines active and inactive trackers. Of all the active trackers, the one with the maximal posterior probability specifies the activity label the current frame and the bounding box of the person. The cropped and labeled frames are delivered to the transfer learning stage (Arrow (1) in Fig. 1.1). If no available tracker is active for a certain observation, it is reported as *unknonw* and in our case corresponds to an abnormal event.

Blob tracking. A foreground blob tracker, CamShift in our implementation (Bradski, 1998; Comaniciu and Meer, 2002), initialized by a person detector (Felzenszwalb et al., 2008), tracks the human target as long as it is in the scene. In case of an abnormal event, this tracker determines the bounding box of the person, which is handed over (Arrow (2) in Fig. 1.1).

Update. Given the frames first labeled as abnormal and the new semantic activity labels obtained from the classifier stage, a new activity model is learned for each new class. The new activity trackers are added besides the existing ones and the initial and the new activities will be detected and recognized from now on. If a shift in one of the known concepts is observed, *i.e.,* activity detection with the initial set of trackers does not match the labeling of the transfer learning, existing activity models need to be replaced.

1.5 Knowledge Transfer for Unusual Event Learning

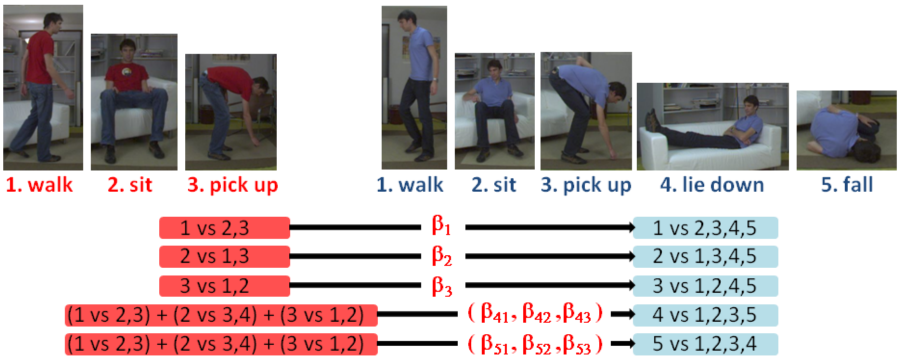


Figure 1.4: Description of the multiclass one-vs-all transfer learning strategy. The activity classes on the left (marked in red) correspond to prior knowledge. The classes on the right (marked in blue) correspond to the new target task. The new hyperplanes for classes 1,2 and 3 are obtained through transfer learning from the corresponding source knowledge while for classes 4 and 5 a weighted combination of all the known hyperplanes is used as prior.

Transfer learning can help to reduce the labeling effort which is in general necessary when recognizing a new set of activities. The idea is to transfer only the useful part of information from the already known activity classes when solving the new multiclass problem.

In the following we summarize the binary transfer learning method presented in Tommasi and Caputo (2009); Tommasi et al. (2010) and describe how to extend it to multiclass with the one-vs-all approach, mapping it to the task of learning to learn new models of human behaviors.

1.5.1 Adaptive knowledge transfer

Given a set of l samples $\{\mathbf{x}_i, y_i\}_{i=1}^{l}$, where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^d$ and $y_i \in \mathcal{Y} = \{-1, 1\}$, we want to learn a linear function $f(\mathbf{x}) = \mathbf{w} \cdot \phi(\mathbf{x}) + b$ which assigns the correct label to an unseen test sample $\mathbf{x}$. The function $\phi(\mathbf{x})$ maps the input samples to a high dimensional feature space where the inner product can be easily calculated through a kernel function $K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$ (Cristianini and Shawe-Taylor, 2000).

In Least-Square Support Vector Machine (LS-SVM) the model parameters $(\mathbf{w}, b)$ are found by solving the following optimization problem (Suykens et al., 2002):

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{2}\sum_{i=1}^{l} \zeta_i [y_i - \mathbf{w} \cdot \phi(\mathbf{x}_i) - b]^2 \ . \tag{1.1}$$

The weight ζ_i is introduced to take care of unbalanced distributions and it depends on the number of positive and negative available samples (Tommasi et al., 2010). It can be shown (Suykens et al., 2002) that the optimal $\mathbf{w}$ is expressed by $\mathbf{w} = \sum_{i=1}^{l} \alpha_i \phi(\mathbf{x}_i)$, and $(\boldsymbol{\alpha}, b)$ are obtained from:

$$\begin{bmatrix} \mathbf{K} + \frac{1}{C}\mathbf{W} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix}, \tag{1.2}$$

where $\mathbf{W} = diag\{\zeta_1^{-1}, \zeta_2^{-1}, \ldots, \zeta_l^{-1}\}$. Let us call $\mathbf{G}$ the first term in left-hand side of Eq. (1.2). Thus the optimization problem in Eq. (1.1) can be solved by simply inverting $\mathbf{G}$.

By slightly changing the classical LS-SVM regularization term, it is possible to define a learning method based on adaptation (Tommasi et al., 2010). The idea is to constrain a new model to be close to a set of k pre-trained models:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w} - \sum_{j=1}^{k} \beta_j \mathbf{w}'_\mathbf{j}\|^2 + \frac{C}{2}\sum_{i=1}^{l} \zeta_i [y_i - \mathbf{w} \cdot \phi(\mathbf{x}_i) - b]^2, \tag{1.3}$$

where $\mathbf{w}'_\mathbf{j}$ is the parameter describing each old model and β_j is a scaling factor necessary to control the degree to which the new model is close to the old one. The LS formulation gives the possibility to write the Leave-One-Out (LOO) prediction for each sample $\tilde{y}_i$ in closed form:

$$\tilde{y}_i = y_i - \frac{\alpha_i}{\mathbf{G}_{ii}^{-1}} - \sum_{j=1}^{k} \beta_j \frac{\alpha'_{i(j)}}{\mathbf{G}_{ii}^{-1}}, \tag{1.4}$$

where α_i and $\alpha'_{i(j)}$ are respectively elements of the vectors $\boldsymbol{\alpha} = (\mathbf{K}+\frac{1}{C}\mathbf{W})^{-1}\mathbf{y}$ and $\boldsymbol{\alpha_j}' = (\mathbf{K} + \frac{1}{C}\mathbf{W})^{-1}\mathbf{\hat{y}_j}$. $\mathbf{y}$ is the vector of the y_i and $\mathbf{\hat{y}_j}$ is the vector of the predictions of the j^{th} known model $\hat{y}_{i(j)} = (\mathbf{w}'_\mathbf{j} \cdot \phi(\mathbf{x}_i))$. Thus the LOO error can be easily evaluated as $r_i^{(-i)} = y_i - \tilde{y}_i$. It is an unbiased estimator of the classifier generalization error and can be used to find the best value of $\boldsymbol{\beta}$.

1.5.2 One-vs-All multiclass extension

We start from a prior knowledge problem with N different models of human behaviors and train a multiclass SVM classifier with the one-vs-all approach. Only the parameters that describe the hyperplanes $\{\mathbf{w}'_\mathbf{j}\}_{j=1}^{N}$ are memorized while the data are not stored. As target task we consider to solve a $(N+N')$ multiclass problem where N models of human activities are the same as in the original source task and N' models are new. However, now only very few samples for each model are available.

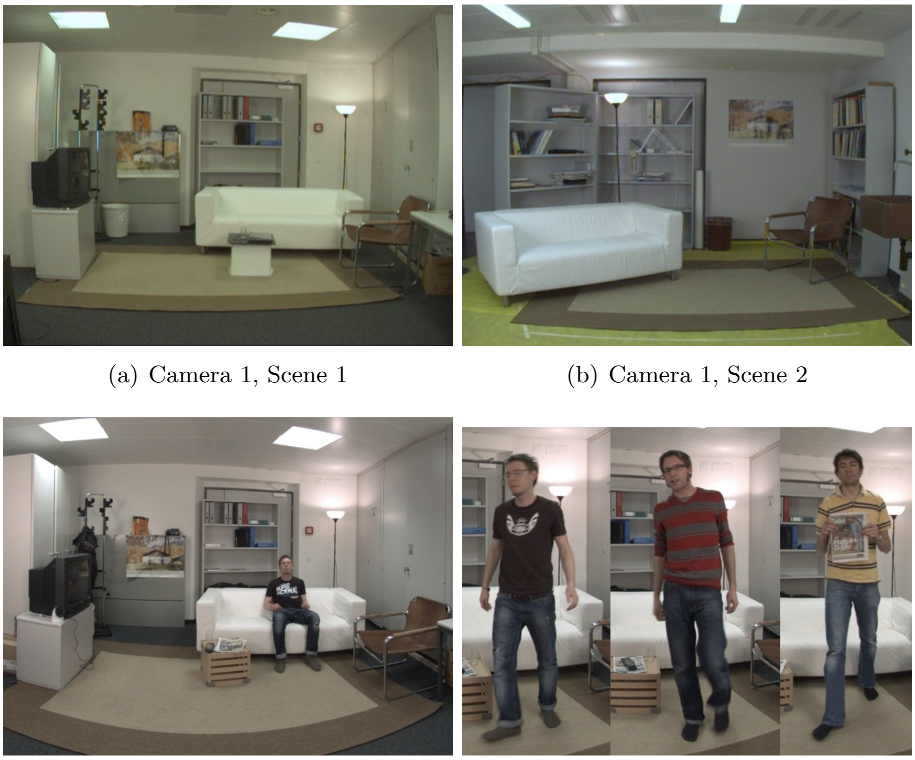
The binary transfer approach described previously can be used separately to learn each of the $(N + N')$ one-vs-all hyperplanes (see Fig. 1.4). The N hyperplanes associated to the same models of human behaviors considered in prior knowledge, are now trained to separate some new positive samples against a different negative set due to the presence of N' new models. In these cases the $\boldsymbol{\beta}$ vector reduces to one single value ranging in $[0, 1]$. The method also exploits a linear combination of prior knowledge hyperplanes to separate each of the N' new activity models from all the others. Here the idea is that a combination of visual characteristics which differentiate among *walk*, *sit* and *pick up* can still be useful to carachterize *lie down* and *fall* and can help when only few samples of the different actions are available.

1.6 Experiments

We demonstrate the activity classification via transfer learning, and show that the newly learned classes improve the performance of the activity model. We use the same data for both tasks.

1.6.1 Dataset and setting

In our experiments, we include 5 different activities to be recognized. These are *walk, sit down, pick up, lie down* and *fall.* We consider different cases that might also appear in real-life scenarios. As depicted in Fig. 1.5, we include two different indoor scenes, two camera types that were used for recording and three different persons.



(a) Camera 1, Scene 1

(b) Camera 1, Scene 2

(c) Camera 2, Scene 1

(d) Persons $1, 2, 3$

Figure 1.5: Different settings are used for the experiments. We recorded in two different indoor scenes, with two different cameras and three persons perform the activities.

Cameras. Camera 1 has VGA resolution and records at 15 frames per second. The used lens introduces minimal distortion. Camera 2 has a resolution of 1624×1234 pixels and records at 12 frames per second. A fish-eye lens with a large field of view introduces distortion, that needs to be corrected. To this end, we apply the technique of Havlena et al. (2009) and rectify the images cylindrically, *i.e.* straight, physically vertical lines are preserved. For visualization purposes, the relevant image region is cropped out in Fig. 1.5(c).

Sequences. We dispose of 12 video sequences, which were recorded as detailed in Tab. 1.1[2]. They contain between 1000 and 3000 frames and depict a single person who performs all the five activities. We manually provide a frame by frame ground truth annotation for each sequence. Transitions (*e.g.,* standing up after a fall) are termed with *no activity.*

Seq 1a, Seq 1b, Seq1c :	{Scene 1, Person 1, Camera 2}
Seq 2a, Seq 2b, Seq2c :	{Scene 1, Person 2, Camera 2}
Seq 3a, Seq 3b, Seq3c :	{Scene 1, Person 3, Camera 1}
Seq 4a, Seq 4b, Seq4c :	{Scene 2, Person 3, Camera 1}

Table 1.1: Three sequences were recorded for every parameter combination.

Initial processing. We run the three initial activity trackers (walk, sit down, pick up) and the blob tracker on all the sequences. The known activities are spotted and abnormal events are detected. Each frame is labeled and the bounding box of the person is obtained. This forms the basis for further analysis.

1.6.2 Transfer learning

As explained in Sec. 1.3 the transfer learning step is used as an expert exploiting prior knowledge and labeling new samples that are then used to update the tracking system. Having an accurate classification process is crucial for the efficiency of the final action recognition method. We validate the proposed transfer approach with four experiments. As prior knowledge we used Seq $*a$ with the $N = 3$ activities labeled in the initial processing. Seq $*b$ is used to extract randomly 10 frames for each of all the $N + N' = 5$ actions (initial processing and new activities). This defines the training set for the target task. Finally Seq $*c$ is used as test set.

The PHOG features (Bosch et al., 2007) (histogram bins=9, angle=180, levels=3) are calculated on the provided bounding box around the person and they are used together with the RBF kernel in all the experiments. The learning parameters are chosen by cross validation on prior knowledge. To implement the multiclass transfer learning method we started from Tommasi et al. (2010) using the code released by the authors[3].

We compare three methods that are applied to the test sequence:

- *Initial Model*: The prior knowledge model learned on the 3 initial activities.
- *No Transfer*: The model learned on few samples of the 5 activities.

[2] Data available from `www.vision.ee.ethz.ch/fnater`

[3] http://www.idiap.ch/∼ttommasi/source_code.html

- *Transfer*: The model learned on few samples of the 5 activities transferring from prior knowledge.

The plotted values correspond to the average recognition rate on 10 runs of the experiment (the random selection of training frames from Seq $*b$ is repeated 10 times). The significance of the comparison between *Transfer* and *No Transfer* is evaluated through the sign test (Gibbons, 1985): a square marker is reported on the graph if $p < 0.05$ (see Fig. 1.6). The four experiments differ by the existing relation between prior knowledge and target task.

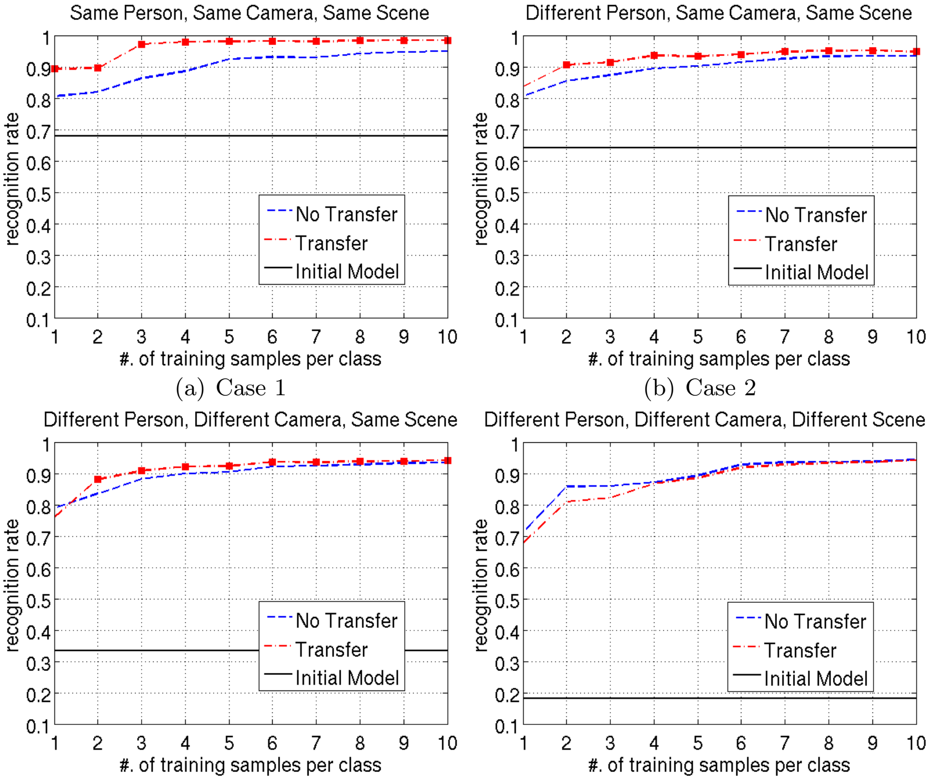
Case 1: same person, same camera, same scene. The acting person, the background scene and the recording camera are the same in prior and new sequence. Specifically we used Seq $1a$, Seq $1b$ and Seq $1c$. Classification results are reported in Fig. 1.6 (a): transferring from prior knowledge guarantees a significant advantage compared to learning from scratch. The same experiment was repeated using Seq $3a$, Seq $3b$ and Seq $3c$, with equal results.

Case 2: different person, same camera, same scene. The background scene and the recording camera are fixed, but the acting person in prior knowledge is different with respect to the one in the training and test videos. We used respectively Seq $2a$, Seq $1b$ and Seq $1c$. The results are reported in Fig. 1.6 (b). Even if the actions in prior knowledge are performed by a different person, transferring information still guarantees an advantage in learning. The same experiment was repeated inverting the role of the two acting persons and using Seq $1a$, Seq $2b$ and Seq $2c$ with analogous results.

Case 3: different person, different camera, same scene. Prior knowledge and new task involve different persons, they are also recorded with a different camera but the scene remains the same. Specifically we considered Seq $3a$, Seq $1b$ and Seq $1c$. Fig. 1.6 (c) shows the results: here *Transfer* is still significantly better than *No Transfer* but the gain in terms of recognition performance is small.

Case 4: different person, different camera, different scene. Finally we consider a prior knowledge setting where the person, the camera used and the background scene are different with respect to the one used in the training and test videos. We used Seq $4a$, Seq $1b$ and Seq $1c$ and the results are reported in Fig. 1.6 (d). Here the transfer learning system automatically realizes that the information coming from prior knowledge is not useful for the new task and *Transfer* performs as *No Transfer*.

Comparing all the four graphs in Fig. 1.6, the progressively lower relevance of prior knowledge with respect to the new target task can be read



(a) Case 1

(b) Case 2

(c) Case 3

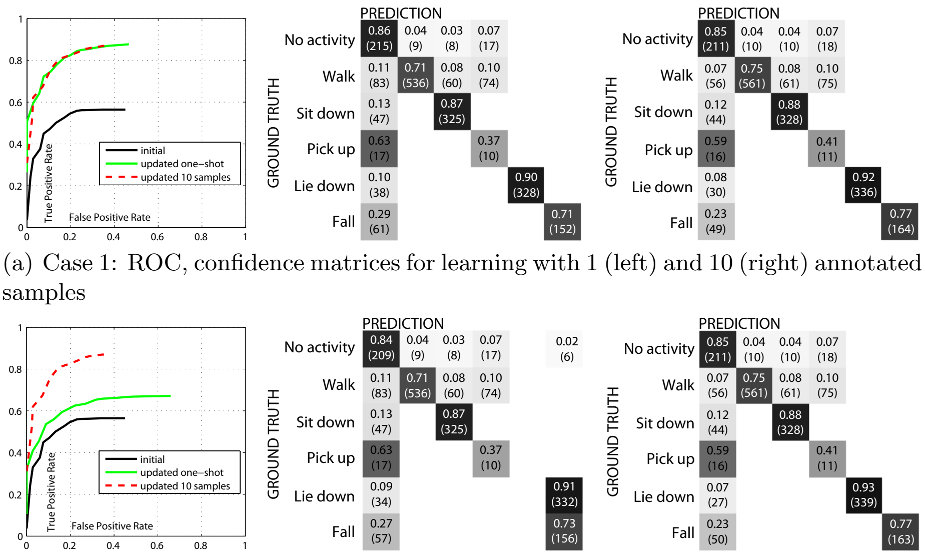
(d) Case 4

Figure 1.6: Average recognition rate results on ten runs evaluated varying the number of samples per class in the training set. The significance of the comparison between Transfer and No Transfer is evaluated through the sign test (Gibbons, 1985): a square marker is reported on the graph if $p < 0.05$. Passing from case 1 to case 4 the prior knowledge is less and less relevant, consequently the advantage of Transfer w.r.t. No Transfer decreases.

in the decreasing recognition rate result for the *Initial Model*. Globally, the classifiers obtained with *Transfer* learning perform better or at least equally to *No transfer*. Therefore we use the transfer learning to fix the activity class labels that are delivered to update the activity trackers.

1.6.3 Activity tracking

Given an updated set of activity trackers, we evaluate how the activity recognition performance increases with respect to the initial processing. The predicted activities are compared to the ground truth. We use Seq $*b$ since it was not used previously for testing the classification. Activities are predicted for three cases: (i) the initial tracker set, (ii) the tracker set after the update with one-shot learning and (iii) after the update with 10 manually labeled frames.



GROUND TRUTH	PREDICTION					
No activity	0.86 (215)	0.04 (9)	0.03 (8)	0.07 (17)		
Walk	0.11 (83)	0.71 (536)	0.08 (60)	0.10 (74)		
Sit down	0.13 (47)		0.87 (325)			
Pick up	0.63 (17)			0.37 (10)		
Lie down	0.10 (38)				0.90 (328)	
Fall	0.29 (61)					0.71 (152)

GROUND TRUTH	PREDICTION					
No activity	0.85 (211)	0.04 (10)	0.04 (10)	0.07 (18)		
Walk	0.07 (56)	0.75 (561)	0.08 (61)	0.10 (75)		
Sit down	0.12 (44)		0.88 (328)			
Pick up	0.59 (16)			0.41 (11)		
Lie down	0.08 (30)				0.92 (336)	
Fall	0.23 (49)					0.77 (164)

(a) Case 1: ROC, confidence matrices for learning with 1 (left) and 10 (right) annotated samples

GROUND TRUTH	PREDICTION					
No activity	0.84 (209)	0.04 (9)	0.03 (8)	0.07 (17)		0.02 (6)
Walk	0.11 (83)	0.71 (536)	0.08 (60)	0.10 (74)		
Sit down	0.13 (47)		0.87 (325)			
Pick up	0.63 (17)			0.37 (10)		
Lie down	0.09 (34)					0.91 (332)
Fall	0.27 (57)					0.73 (156)

GROUND TRUTH	PREDICTION					
No activity	0.85 (211)	0.04 (10)	0.04 (10)	0.07 (18)		
Walk	0.07 (56)	0.75 (561)	0.08 (61)	0.10 (75)		
Sit down	0.12 (44)		0.88 (328)			
Pick up	0.59 (16)			0.41 (11)		
Lie down	0.07 (27)				0.93 (339)	
Fall	0.23 (50)					0.77 (163)

(b) Case 4: ROC, confidence matrices for learning with 1 (left) and 10 (right) annotated samples

Figure 1.7: Activity tracking results. ROC curves and confusion matrices for case 1 (top row, corresponding to Fig. 1.6(a)) and case 4 (bottom row, corresponding to Fig. 1.6(d)). In the first row, the performances for one-shot learning and learning with 10 samples match, whereas in the more difficult case in the bottom row, more annotations improve the performance.

Tracker test sequence		1b				3b	2b
Tracker update sequence		1c				3c	2c
Transfer prior sequence		1a	2a	3a	4a	3a	1a
Corresponds to case		1	2	3	4	1	2
Initial processing	TPR	0.50				0.45	0.44
	FPR	0.13				0.17	0.06
Updated (1-shot)	TPR	0.78	0.78	0.78	0.59	0.72	0.62
	FPR	0.14	0.14	0.14	0.16	0.17	0.05
Updated (10 samples)	TPR	0.81	0.82	0.81	0.81	0.73	0.66
	FPR	0.15	0.17	0.15	0.15	0.22	0.13

Table 1.2: Results for different sequences, the predicted activity is compared to the ground truth. Different cases are reported in terms of true positive rate (TPR) and false positive rate (FPR). The updated activity set outperforms the initial one. In most situations, the results obtained with 10 labeled samples are only marginally better than using one-shot learning.

In Fig. 1.7 we provide detailed insights for the activity update. The cases 1 (same scene, same person, same camera) and 4 (different scene, different person, different camera) are depicted. In Fig. 1.7, ROC curves are shown for the initial and updated (one-shot and 10 samples) tracker sets. To this end, the threshold that determines the active trackers, is gradually increased. This results in different numbers of true-positives and false-positives. For the confusion matrices in Fig. 1.7 and all further experiments, the threshold is kept fix.

One-shot labeling already improves the activity tracking performance considerably with respect to the initial tracker set. If the labels provided by the one-shot learning are correct as in case 1, the benefit of labeling 10 frames is marginal. If it turns out that one manually labeled sample is not sufficient for a good classification accuracy, as in the most difficult case 4, manual annotation of 10 frames improves the final performance. In the confusion matrices, the predicted activities are reported *vs.* the ground truth in terms of number of frames and underlie this finding. Cases 2 and 3 are very similar to case 1, *i.e.,* the transfer learning with one manually labeled sample is sufficient.

In Tab. 1.2, we report the evaluation of the activity recognition in terms of overall true-positive-rate and false-positive-rate for different cases of prior knowledge and target tasks. The first four columns report results obtained on the same sequences used for the experiments in Fig. 1.6, the last two columns contain the results for other test sequences. In all cases, the augmentation of the tracker set with new trackers learned from the transferred labels helps. In five of the six evaluated cases however, the annotation of ten frames *vs.* one frame only improves the performance marginally. We underline that the number of labelled training samples needed is in any case two or at least one order of magnitude smaller than what originally requested to update the activity tracker in Nater et al. (2009).

1.7 Conclusions

Starting from the output of a method that detects known activities and unusual events in surveillance videos, we presented here a strategy to learn these new events. We only need a very small number of training samples since we exploit prior knowledge of activities that were known already. We extended an efficient transfer learning method from binary to multiclass and we tested it on the realistic and challenging scenario of learning new human activities. Finally, we show that the combination of activity tracking techniques with transfer learning can aid in determining the behavior of a person in an indoor scene. Future work will focus on merging the tracking and learning steps into an unified framework Tommasi et al. (2012).

Bibliography

Adam, A., Rivlin, E., Shimshoni, I., and Reinitz, D. (2008). Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):555–560.

Anderson, D., Luke, R., Keller, J., Skubic, M., Rantz, M., and Aud, M. (2009). Linguistic summarization of video for fall detection using voxel person and fuzzy logic. *International Journal on Computer Vision and Image Understanding (CVIU)*, 113(1):80–89.

Andriluka, M., Roth, S., and Schiele, B. (2010). Monocular 3d pose estimation and tracking by detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Boiman, O. and Irani, M. (2005). Detecting irregularities in images and in video. In *International Conference on Computer Vision (ICCV)*.

Bosch, A., Zisserman, A., and Munoz, X. (2007). Representing shape with a spatial pyramid kernel. In *ACM CIVR*.

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

Comaniciu, D. and Meer, P. (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603 –619.

Cristianini, N. and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines (and Other Kernel-Based Learning Methods)*. Cambridge University Press.

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

Efros, A., Berg, A. C., Mori, G., and Malik, J. (2003). Recognizing action at a distance. In *International Conference on Computer Vision (ICCV)*.

Felzenszwalb, P., Mcallester, D., and Ramanan, D. (2008). A discriminatively trained, multiscale, deformable part model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Gibbons, J. (1985). *Nonparametric Statistical Inference.* New York: Marcel Dekker.

Havlena, M., Ess, A., Moreau, W., Torii, A., Jancosek, M., Pajdla, T., and Van Gool, L. (2009). Awear 2.0 system: Omni-directional audio-visual data acquisition and processing. In *CVPR Workshop on Egocentric Vision.*

Hu, D. H., Zheng, V. W., and Yang, Q. (2011). Cross-domain activity recognition via transfer learning. *Pervasive Mob. Comput.*, 7:344–358.

Jie, L., Tommasi, T., and Caputo, B. (2011). Multiclass transfer learning from unconstrained priors. In *International Conference on Computer Vision (ICCV).*

Lampert, C. H., Nickisch, H., and Harmeling, S. (2009). Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Lawrence, N. (2003). Gaussian process latent variable models for visualisation of high dimensional data. In *Neural Information Processing Systems.*

Liu, J., Ali, S., and Shah, M. (2008). Recogniziong human actions using multiple features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Liu, J., Shah, M., Kuipers, B., and Savarese, S. (2011). Cross-view action recognition via view knowledge transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Muhlbaier, M. D., Topalis, A., and Polikar, R. (2009). Learn++.nc: combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes. *Transactions on Neural Networks*, 20:152–168.

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

Nater, F., Grabner, H., and Gool, L. V. (2010). Exploiting simple hierarchies for unsupervised human behavior analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Nater, F., Grabner, H., Jaeggli, T., and Van Gool, L. (2009). Tracker trees for unusual event detection. In *ICCV Workshop on Visual Surveillance.*

Rashidi, P. and Cook, D. J. (2011). Activity knowledge transfer in smart environments. *Pervasive and Mobile Computing*, 7(3):331 – 343.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

Stauffer, C. and Grimson, W. E. L. (2000). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757.

Suykens, J., Gestel, T. V., Brabanter, J. D., Moor, B. D., and Vanderwalle, J. (2002). *Least Squares Support Vector Machines*. World Scientific.

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

Tommasi, T. and Caputo, B. (2009). The more you know, the less you learn: from knowledge transfer to one-shot learning of object categories. In *British Machine Vision Conference (BMVC)*.

Tommasi, T., Orabona, F., and Caputo, B. (2010). Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tommasi, T., Orabona, F., Kaboli, M., and Caputo, B. (2012). Leveraging over prior knowledge for obline learning of visual categories. In *British Machine Vision Conference (BMVC)*.

van Kasteren, T., Englebienne, G., and Kröse, B. J. A. (2010). Transferring knowledge of activity recognition across sensor networks. In *Pervasive*, pages 283–300.

Xian-ming, L. and Shao-zi, L. (2009). Transfer adaboost learning for action recognition. In *IEEE International Symposium on IT in Medicine & Education*.

Yang, W., Wang, Y., and Mori, G. (2010). *Learning Transferable Distance Functions for Human Action Recognition*, pages 349–370. Advances in Pattern Recognition. Springer.

Zhang, B.-F., Su, J.-S., and Xu, X. (2006). A class-incremental learning method for multi-class support vector machines in text classification. In *ICMLC*.