Human Action Localization with Sparse Spatial Supervision

Philippe Weinzaepfel, Xavier Martin, Cordelia Schmid

Human Action Localization with Sparse Spatial Supervision

Philippe Weinzaepfel, Xavier Martin, Cordelia Schmid

Human Action Localization with Sparse Spatial Supervision

Philippe Weinzaepfel, Xavier Martin, and Cordelia Schmid, *Fellow, IEEE*

Abstract—We introduce an approach for spatio-temporal human action localization using sparse spatial supervision. Our method leverages the large amount of annotated humans available today and extracts human tubes by combining a state-of-the-art human detector with a tracking-by-detection approach. Given these high-quality human tubes and temporal supervision, we select positive and negative tubes with very sparse spatial supervision, i.e., only one spatially annotated frame per instance. The selected tubes allow us to effectively learn a spatio-temporal action detector based on dense trajectories or CNNs. We conduct experiments on existing action localization benchmarks: UCF-Sports, J-HMDB and UCF-101. Our results show that our approach, despite using sparse spatial supervision, performs on par with methods using full supervision, i.e., one bounding box annotation per frame. To further validate our method, we introduce **DALY** (Daily Action Localization in YouTube), a dataset for realistic action localization in space and time. It contains high quality temporal and spatial annotations for 3.6k instances of 10 actions in 31 hours of videos (3.3M frames). It is an order of magnitude larger than existing datasets, with more diversity in appearance and long untrimmed videos.

Index Terms—Spatio-temporal action localization, weak supervision, human tubes, CNNs, dense trajectories.

1 Introduction

Action classification has been widely studied over the past decade and state-of-the-art methods [1], [2], [3], [4], [5] now achieve excellent performance. However, to analyze video content in more detail, we need to localize actions in space and time. Detecting actions in videos is a challenging task which has received increasing attention over the past few years. Recently, significant progress has been achieved in supervised action localization, see for example [6], [7], [8], [9], [10]. However these methods require a large amount of annotation, i.e., bounding box annotations in every frame. Such annotations are, for example, used to train Convolutional Neural Networks (CNNs) [6], [7], [9], [10] at the bounding box level. Several works have suggested to generate action proposals before classifying them [11], [12], however they generate hundreds of proposals for a video, thus supervision is still required to label them in order to train a classifier. Consequently, all these approaches require full supervision, where action localization needs to be annotated in every frame. This makes scaling up to a large dataset difficult. The goal of this paper is to move away from full supervision, similar in spirit to recent work on weakly-supervised object localization [13], [14].

Recently, Mettes et al. [15] have addressed action localization with another annotation scheme, e.g. with pointly-supervised proposals. A large number of candidate proposals are obtained using APT [12], a method based on grouping dense trajectories. They show that Multiple Instance Learning (MIL) applied directly on these proposals performs poorly. They thus introduce point supervision and incorporate an overlap measure between annotated points and proposals into the mining process. This requires annotating a point in every frame. In this paper we go a step further and significantly reduce the number of frames to annotate. To this end, we leverage the fact that actors are humans and extract human tubes. Given these human tubes, our approach uses only one spatial annotation per action instance, see Figure 1. We show that such a sparse annotation scheme is sufficient to train state-of-the-art action detectors.

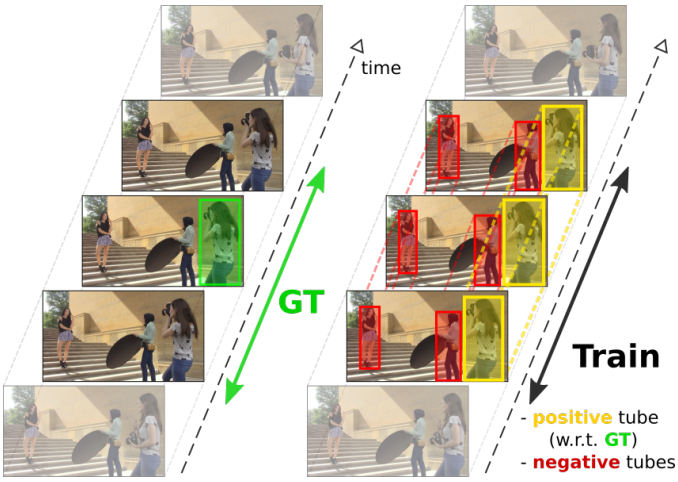

Fig. 1. We consider sparse spatial supervision: the temporal extent of the action as well as one box per instance are annotated in the training videos (left). To train an action detector, we extract human tubes and select positive and negative ones (right) according to the sparse annotations.

Our approach first extracts human tubes from videos. Using human tubes for action recognition is not a novel idea [16], [17], [18]. However, we show that extracting high quality human tubes is possible by leveraging a recent state-of-the-art object detection approach (Faster R-CNN [19]), a large annotated dataset of humans in a variety of poses

- *P. Weinzaepfel is with Xerox Research Centre Europe, Meylan, France. E-mail: philippe.weinzaepfel@xrce.xerox.com*
- *X. Martin and C. Schmid are with Inria, LJK, Grenoble, France. E-mail: firstname.lastname@inria.fr*

Fig. 2. Illustration of the 10 classes of our DALY dataset. One annotated frame is shown per class.

(MPII Human Pose [20]) and a state-of-the-art tracking-by-detection approach [21], [22]. Our experiments demonstrate that a small number of human tubes per video is sufficient to obtain a recall of 95% on the UCF-Sports and J-HMDB datasets, and 65% on the more challenging UCF-101 benchmark.

Assuming that the temporal extent of the action is given at train time, we study how much spatial supervision is required for training an action detector, when sparse spatial annotations are used to label human tubes as positive or negative, see Figure 1. We show that spatial supervision can be reduced to only one annotated frame without impacting performance significantly. This is observed both for learning a SVM on dense trajectories (IDT) [1] and for training CNNs [2]. Our action detector combining IDT and CNN obtains a mAP of 96%, 64% and 57% on UCF-Sports, J-HMDB and UCF-101 respectively, with only one annotated frame. The performance is comparable to full supervision, i.e., the drop is at most 2% while the annotation cost is drastically reduced. We also significantly outperform other methods that aim at reducing supervision. For example, we obtain 57% mAP on the UCF-101 dataset with one annotated bounding box, i.e., 2 points annotated per action instance, whereas Mettes et al. [15] obtain 32% mAP but use one point annotation in every frame. This represents 4030 bounding box annotations (i.e., 8060 points) for our sparse annotation scheme compared to almost 1.9 million points for [15].

To further validate our method, we introduce the Daily Action Localization in YouTube (**DALY**) dataset. It is designed to correct the drawbacks of existing datasets, which are trimmed (UCF-Sports, J-HMDB) or almost-trimmed videos (UCF-101) with specific action types, e.g. sports only, showing in most cases only one human per video. DALY is a large dataset with diverse actions in untrimmed videos, sampled from real-world data. It consists of more than 31 hours of videos (3.3M frames) from YouTube with 3.6k spatio-temporal action instances for 10 realistic daily actions, see Figure 2. Annotations indicate the start and end time of each action instance, with spatial annotations for a sparse subset of frames. The task is to localize relatively short actions (8 seconds in average) in long untrimmed videos (3min 45s in average). Furthermore, it includes videos with multiple humans performing actions simultaneously. On the DALY dataset our human tubes obtain a spatial recall of 95%, but the detection task is extremely challenging, we obtain a mean Average Precision of 14%.

This paper is organized as follows. We first review related work in Section 2. We then describe our approach for human action localization with sparse spatial supervision in Section 3. Section 4 presents the datasets used in experimental evaluation and introduces our DALY dataset. Next, we describe and evaluate our approach for extracting human tubes from videos in Section 5. Finally, Section 6 presents experimental results for action localization with sparse spatial supervision. The DALY dataset is available online at http://thoth.inrialpes.fr/daly/.

2 Related work

In this section we review related work on action localization with full supervision (Section 2.1) and partial supervision (Section 2.2).

2.1 Fully-supervised action localization

Initial attempts for temporal and spatio-temporal action localization are based on a sliding-window scheme and focus on improving the search complexity [23], [24], [25], [26]. Other approaches rely on figure-centric models. For instance, Lan et al. [27] consider the human position as a latent variable and infer it jointly with the action label. Kläser et al. [16] use a human detector and build human tracks using KLT features tracks. The human tracks are then classified with HOG-3D descriptors [28]. Our approach is also based on human tracks but is significantly more robust to huge variations in pose and appearance.

Several recent methods for action localization are based on action proposals to reduce the search complexity. Jain et al. [29] build action localization candidates by hierarchically merging supervoxels and use dense trajectory features for tube classification. Similarly, van Gemert et al. [12] cluster trajectories and use the resulting tubes for action detection. In [18], proposals are based on an actionness measure [30] which requires localized training samples. In parallel, several works [11], [31] have attempted to further improve the quality of tubes. Most of these methods generate thousands of proposals for a short video and require ground-truth annotations to label the proposals in order to learn a proposal classifier. In contrast, our approach relies on only a few human tube proposals per video and can, thus, reduce

spatial supervision to one annotated frame without drop of performance.

Recently, CNNs for human action localization have emerged [6], [7], [9], [10]. These approaches rely on R-CNNs for both appearance and motion, classifying region proposals in individual frames. Detection tubes are obtained by combining class-specific detections with either temporal linking based on proximity [6], or with a class-specific tracking-by-detection approach [7]. Both strategies need to be run independently for each action. State-of-the-art approaches [9], [10] rely on Faster R-CNN trained on appearance and flow. Note that all these methods make extensive use of bounding box annotations in every frame for training.

2.2 Action localization with partial supervision

Annotating all videos with bounding boxes in every frame is unrealistic for large-scale datasets, yet reducing spatial supervision has received little attention so far. Weakly-supervised temporal localization was studied in [32], [33], [34]. Bojanowski et al. [32] assume an ordered list of actions in each video as input. Duchenne et al. [33] use a discriminative clustering on short video segments to identify the temporal localization in the training set and learn a classifier. The detection is then performed using a sliding window. Hoai et al. [34] extend a Multiple Instance SVM to time series, allowing for discontinuities in the positive samples. In the context of object detection, Prest et al. [35] propose to extract spatio-temporal tubes and then perform tube selection before training a classifier. Siva and Xiang [36] apply multiple instance learning (MIL) on cuboids of various time lengths around detected humans, described by STIPs [37]. Their method is thus limited to static human actions, with a bounding box that does not move or change over time. Recently, Mettes et al. [15] propose to extract hundreds of action proposals and then apply MIL. Given the huge number of proposals, good performance requires pruning of the proposals. To this end they consider spatial supervision in the form of 2D points annotated for each frame, which they refer to as pointly-supervised.

Some other works also detect actions without spatial supervision. Mosabbeb et al. [38] use a subspace segmentation clustering approach applied on groups of trajectories, in order to segment videos into parts. A low-rank matrix completion method estimates the contribution of each cluster to the different labels, hence the approach detects several disjoint action parts and not one consistent spatio-temporal localization. Ma et al. [39] first extract a per-frame hierarchical segmentation, which is tracked over the video. Using foreground scoring, they obtain a hierarchy of spatio-temporal segments where the upper level corresponds to human body location candidates. However, they rely on parts segmentation which is challenging in low-quality real-world videos, e.g. with strong occlusion and compression artifacts. More recently, Chen and Corso [40] propose to generate unsupervised proposals by clustering intentional motion based on dense trajectories. Their method can handle only one action per training video and is not robust to nearby motions.

3 Our approach

In this section we describe our approach for learning an action detector with sparse spatial supervision. Sparse means that ground-truth bounding boxes around the actors are annotated in a few frames only. Sparse spatial annotation significantly reduces annotation cost, an important factor to create real-world large-scale datasets.

Our main hypothesis is that since actions are performed by humans, we can take advantage of existing human localization datasets. Thanks to the large amount of human annotations at our disposal [20], we can extract high-quality action-agnostic human tubes, which we then label as negative or positive samples using the sparse spatial annotations, see Figure 1. The obtained labeling is then used to train action detectors. In this paper we rely on two detectors, one based on dense trajectories [1] and the other on CNNs [19]. At test time, we also extract human tubes and score them with the trained detectors.

In this section, we first present how human tubes are extracted in Section 3.1. Next, we describe how these human tubes are labeled from the few annotated frames required to train a classifier (Section 3.2) and the different approaches we consider for human tube scoring (Section 3.3). We finally present how we perform temporal detection for untrimmed dataset in Section 3.4.

3.1 Extracting human tubes

The first step of our approach is to extract human tubes, where human tubes are sequences of bounding boxes following a particular person. We propose to extract human tubes by relying on additional training data, i.e., the pose annotations from the MPII Human Pose dataset [20]. This human pose dataset contains people in a wide range of poses, which allows us to detect humans even when they perform actions that involve unusual poses.

Our approach starts by detecting humans at the frame level. The human detector relies on the state-of-the-art Faster R-CNN [19] detector trained with the MPII Human Pose dataset. Once humans are detected in every frame, we track them throughout the video. Relying on a tracking strategy similar to Weinzaepfel et al. [7], we combine the human detection score from Faster R-CNN with an instance-level detector. The instance-level detector is a linear SVM learned on the features from the last fully-connected layer of Faster R-CNN. The tracker performs a sliding window search in the next frame around the location of the tracked box, and selects the highest scored box according to a combined score of the per-frame human and instance-level detectors. This box is refined according to the regressor branch of Faster R-CNN learned for human detection. At every frame, the instance-level detector is updated with the selected region.

We obtain an initial human tube by tracking the highest scoring human detection in the video sequence, both forward and backward in time. Having tracked this detection, we remove the human detections that have an Intersection Over Union (IoU) above 0.3 with any box of this track. We repeat this process by selecting and tracking the highest-scoring human detection among remaining ones. We stop when no detections are left to examine.

	DALY	UCF-Sports [41]	J-HMDB [42]	UCF-101 [43]
#classes	10	10	21	**24**
action types	**everyday**	sports	**everyday**	sports
#clips	**8133**	150	928	3207
avg resolution	**1290x790**	690x450	320x240	320x240
total #frames	**3.3M**	10k	32k	558k
avg video dur.	**3min 45s**	5.8s	1.4s	5.8s
avg action dur.	**7.9s**	5.8s	1.4s	4.5s
action dur. / video dur.	**4%**	100%	100%	78%
#instances	3637	154	928	**4030**
avg #instances/class	**364**	15	44	168
spatial annotation	subset	all	all	all

TABLE 1
Comparison of our DALY dataset with existing action localization datasets.

3.2 Learning with sparse spatial supervision

In this work, we consider that for each training video, only a subset of N frames are annotated in each action instance. For our experiments we regularly sample them from the ground-truth tubes, except in the case of DALY where only up to 5 annotations are provided per action instance. The human tubes are labeled as positive if they have an Intersection-over-Union (IoU) over 0.5 with any ground-truth, the IoU being computed only over the N annotated frames, see Figure 1. In our experiments, we vary the value of N from 1 to 5 frames.

3.3 Scoring human tubes

We consider different approaches to score the human tubes: improved dense trajectories (IDT) and two-stream R-CNNs (CNN), as well their combination with late fusion (CNN+IDT).

Improved Dense Trajectories

For each human tube, we extract Improved Dense Trajectories (IDT) and aggregate them with a Fisher Vector representation [1]. In more details, we start by extracting IDT for the entire video[1]. For each descriptor type (HOG, HOF, MBHx, MBHy), we reduce its initial dimension by a factor of 2 using PCA and learn a codebook of 256 Gaussians. For each tube, we build a Fisher Vector per descriptor type, using only the trajectories that start inside the tube. Each of the 4 Fisher Vectors is independently power-normalized and L2-normalized [44]. A tube is finally described by the concatenation of the 4 normalized Fisher Vectors, resulting in 102400 dimensions. To classify human tubes described with IDT, we learn a linear Support Vector Machine (SVM), which have demonstrated an excellent performance on the Fisher Vector representation [1]. We convert SVM scores into probabilities [45] in order to make them comparable to CNN softmax scores. We use as negatives all human tubes from negative videos, as well as human tubes for which the Intersection-over-Union over human tubes labeled as positives is below 0.5.

Two-stream R-CNNs

We also learn a two-stream Fast R-CNN [46] on the tubes selected with sparse spatial supervision. We use as input proposals to Fast R-CNN the ones from the human tubes, i.e., the Region Proposals Network of the human detector. We label them as positive or negative according to the overlap with the human tubes: proposals with IoU over 0.5 with positive human tubes are labeled as positive proposals, the others as negative. Thanks to the human tubes, we can thus label the human proposals from all frames for training, which increases significantly the training data compared to training on the sparse annotated frames only.

We learn the two streams independently, one on RGB images and the other on flow images. Flow is computed with [47] and converted to a jpg file following [6], [7]. As [5], we initialize the weights with ImageNet pretraining [48] for both streams. We perform late fusion of the scores from the RGB and the flow streams. During testing, we score the human tubes using the softmax probability scores averaged over all boxes of a tube. Note that we do not need to detect, but just to apply the classifier on the boxes from the human tubes. Thus, we do not need to learn region proposals. Furthermore, we do regress boxes, as (a) the human tubes are already regressed, and (b) the sparse supervision limits the number of ground-truth boxes, i.e., of regression targets at training.

Fusion of the features

We finally consider the fusion of the IDT and the CNN features, using late fusion. More precisely, we score a human tube using the average of the probabilities output by (a) the SVM learned on IDT features, (b) the RGB stream of Fast R-CNN and, (c) its flow stream.

3.4 Temporal detection

We now present how we perform temporal detection at test time for untrimmed dataset. During training, we assume that the temporal extent of the actions is given, and extract the human tubes for this temporal extent. At test time, for datasets with long videos that can contain multiple clips such as DALY, we first split videos into clips using an automatic shot detector[2]. For datasets with only one shot per video, e.g. the UCF-Sports, J-HMDB and UCF-101 datasets, we define clips as the full videos. We then extract the human tubes in each clip. The temporal detection is performed using a multi-scale temporal sliding window inside each tube. We use the same temporal lengths as [7] ($\{20, 30, 40, ..., 90, 100, 150, 300, 450, 600\}$

1. https://lear.inrialpes.fr/people/wang/improved_trajectories

2. https://github.com/johmathe/Shotdetect

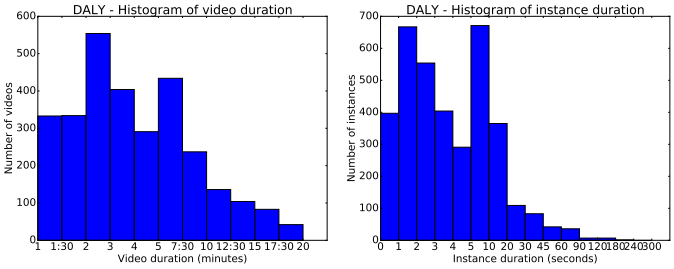

Fig. 3. Histogram of duration of the videos (left) and instances (right).

frames) and the same stride (10). For the DALY dataset, which contains longer actions, we add temporal lengths of $\{900, 1200, 1500, 1800\}$ and $\{2400, 3000, 3600, ..., 120000\}$. In order to penalize short action detections, we score a snippet of a tube using its CNN+IDT score minus α/L where α is a parameter experimentally set to 20 and L is the length of the detection.

4 Datasets

This section describes existing action localization datasets used in our experiments (Section 4.1) and their limitations. We then introduce our **DALY** (Daily Action Localization in YouTube) dataset (Section 4.2). We finally present the evaluation protocol (Section 4.3).

4.1 Action localization datasets

• The **UCF-Sports** dataset [41] is limited to 150 short sports videos with 10 actions, such as *diving* or *running*. It contains an average of 15 instances per class. Videos are trimmed to the action and every frame is annotated with a bounding box. For each class, sequences present similarities in background, camera viewpoint and actors, reducing diversity. We use the train/test split defined in [27].
• The **J-HMDB** dataset [42] is a subset of the HMDB benchmark [49]. It contains 928 videos with 21 actions, including *stand up*, *run* and *pour*. The videos are trimmed to the action, are very short (1.4 sec on average), and most of them contain a single human. On average 44 instances are annotated per action class. The annotations are human silhouettes in every frame. We use the bounding boxes around these silhouettes as ground-truth. The dataset has 3 train/test splits.
• The **UCF-101** dataset [43] contains spatio-temporal annotations for 24 actions in 3207 sports videos. Many videos are similar in terms of actors or background. In contrast to UCF-Sports and J-HMDB, the detection is also temporal but the videos remain short; for half of the classes, the action lasts for more than 80% of the video duration. There are 3 train/test splits. Results are reported for split 1 only, as in [7], [9], [10].

In summary, existing action localization datasets are limited in either diversity of actions (mainly sports), diversity of videos (similar backgrounds and/or actors across videos), video duration (short, trimmed to the action) or number of samples per class, see Table 1 for details. There is a clear need for a dataset that overcomes all of these limitations if we want to improve spatio-temporal action localization in real-world scenarios.

4.2 DALY dataset

We introduce DALY, a dataset for **D**aily **A**ction **L**ocalization in **Y**ouTube. The DALY dataset consists of 31 hours of YouTube videos, with spatial and temporal annotations for 10 everyday human actions with a total of 3.6k instances.

We describe in this section how the DALY dataset was collected. More precisely, we first explain the action class selection and the filtering of the videos. We then present the spatio-temporal annotation of action instances.

Picking action classes

In order to obtain a dataset that fairly evaluates action localization methods, action classes must have clearly defined temporal boundaries. Ambiguities introduce noise in label and temporal annotation, which makes the evaluation unreliable.

We thus select classes for which temporal boundary guidelines can be stated precisely and concisely. For instance, the *brushing teeth* action is defined as 'toothbrush inside the mouth'. Another example is *cleaning windows* for which the moment where 'the tool is in contact with the window' is annotated. The selected action categories are *applying make up on lips, brushing teeth, cleaning floor, cleaning windows, drinking, folding textile, ironing, phoning, playing harmonica and taking photos/videos*, see Figure 2.

Some of those action classes were picked to have similar body movement, in order to make them hard to distinguish. For instance, several action classes imply motion of the hands near the head (*taking photos, phoning*) or the mouth (*playing harmonica, drinking, brushing teeth, applying make up on lips*).

Video collection

The videos are retrieved from YouTube using manually defined queries related to the action labels. For example, the class *cleaning floor* relies on queries such as 'sweeping floor', 'mopping floor', 'cleaning floor', etc. We only collect videos that last between 1 and 20 minutes. A minimum duration of 1 minute ensures that temporal localization will be meaningful (in most cases, shorter videos contain only one action from the beginning to the end). The maximum duration of 20 minutes is to promote ease of use and, for some future methods, avoid disproportionate computational time. The average video duration is 3min 45s. Figure 3 (left) shows an histogram of video duration, and Table 2 displays per-class average video duration. Videos are longest for *ApplyingMakeUpOnLips*, mainly because this action tends to be present in long and detailed make-up tutorials.

Videos are filtered to remove cartoons, slideshows, actions performed by animals and first-person viewpoints. We also remove videos in which the human is not visible when the action occurs, for instance when the camera focuses on the mop while performing the *cleaning floor* action.

We keep 51 videos for each action class, where each video contains at least one instance of the action class. In total, this corresponds to 31 hours of video or 3.3 million frames. This represents 300 times more frames than UCF-Sports, 10 times more than J-HMDB and 6 times more than UCF-101, see Table 1. Our dataset contains 510 videos in total, downloaded from YouTube and used as-is. Thus,

class	avg video dur.	#inst.	avg inst. dur.
ApplyingMakeUpOnLips	376.8s ± 265.1	409	3.8s ± 3.4
BrushingTeeth	176.0s ± 120.3	257	9.2s ± 16.2
CleaningFloor	194.2s ± 128.7	187	14.9s ± 15.4
CleaningWindows	196.2s ± 131.9	468	7.1s ± 10.0
Drinking	202.4s ± 130.9	291	2.6s ± 3.0
FoldingTextile	184.1s ± 150.1	257	14.6s ± 22.4
Ironing	233.2s ± 183.8	424	7.2s ± 8.2
Phoning	217.9s ± 140.7	509	10.0s ± 30.3
PlayingHarmonica	190.2s ± 139.8	289	14.6s ± 21.8
TakingPhotosOrVideos	283.0s ± 207.3	546	3.1s ± 3.4
all	225.4s ± 175.7	3637	7.9s ± 16.6

TABLE 2
Statistics for each class showing the video duration (average and standard deviation), the number of instances, and the instance duration (average and standard deviation).

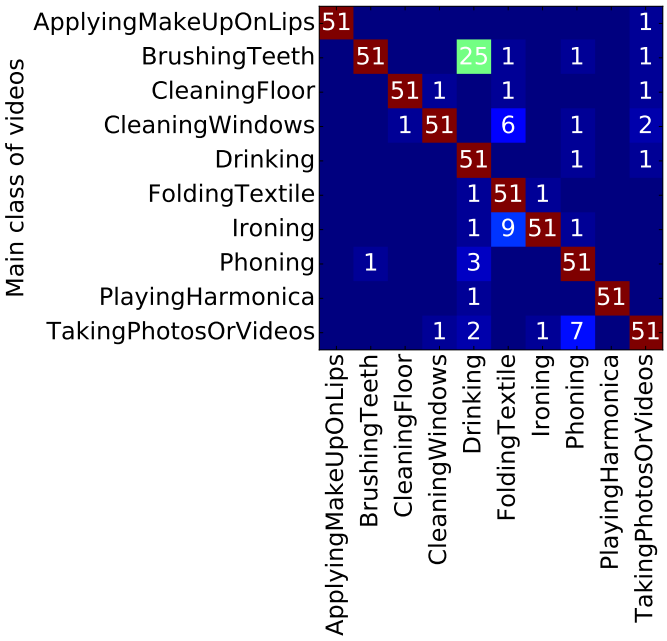

Fig. 4. Statistics of multiple classes per video. Each row comprises the 51 videos downloaded for a given class, each column counts the videos containing at least one instance of the column class.

those videos may contain many shots in contrast to existing datasets that have trimmed or almost-trimmed videos. Using an off-the-shelf automatic shot detector[3], we obtain a total of 8133 clips. We generate a split with 31 training videos and 20 test videos for each class, ensuring that videos uploaded by the same user are in the same set.

The selected action classes are sufficiently common such that multiple action classes can be found in a single video, see Figure 4. Each row of the matrix displays the presence of other action classes in the 51 videos of a given class. For example, out of the 51 videos selected for the class *brushing teeth*, 25 videos also contain *drinking* instances. There is overlap between *ironing* and *folding textile*; *Taking photos or videos*, *phoning* and *drinking* also occur together, as *taking photos* is mostly performed outdoors, where other people are *phoning* or *drinking*. In some cases, several instances are happening simultaneously, see Figure 5. We annotate all occurrences of the 10 classes exhaustively.

3. https://github.com/johmathe/Shotdetect

Temporal annotation

Videos are carefully annotated by members of our research team with the *begin* and *end* time for all action instances. Precise guidelines are established before annotation. For example, the *phoning* action lasts as long as the phone remains close to the ear. In case of a shot change during an action, we annotate it as two separate instances and set a 'shotcut' flag on the second instance. DALY contains 3637 action instances in total. Compared to the average video duration of about 4 minutes, actions are short with an average duration of 8 seconds. Figure 3 (right) shows an histogram of instance duration and per-class statistics are shown in Table 2. Most instances are shorter than 10 seconds, however DALY also contains instances of several minutes. Some classes have very brief instances (e.g. *drinking*), others are longer on average (e.g. *brushing teeth*). For actions that usually last many seconds (such as *cleaning window* or *brushing teeth*), a short instance duration can be explained by video editing: the uploader may have edited the videos, creating small shots that contain the action. The videos are untrimmed, 75% of the frames do not contain any of our 10 actions.

For each action instance, we add a set of 'flags' that state if an action is: (a) small compared to the image, (b) very big compared to the image (zoomed in), (c) largely occluded at some point, (d) outside the camera's field of view at some point. These flags will allow future work to focus on these challenging cases.

Not included in the above count are around 200 instances annotated as ambiguous or mirror reflections. The 'ambiguous' flag is applied when it is unclear whether the action is genuinely performed or not, e.g. the *toothbrush* is put inside the mouth without actually *brushing the teeth*, or the *squeegee* is in contact with the window but the actor is mainly talking instead of *cleaning windows*. In addition to the main actor performing the action, his reflection can sometimes be seen in a mirror or a window. In this case, we have added the flag 'mirror reflection'. This is in particular the case for *brushing teeth* that often occurs in a bathroom in front of a mirror. These annotations of ambiguous cases and mirror reflections are ignored during evaluation, following the Pascal VOC protocol [50]. In other words, we do not register these cases as missing positives or false positives.

Spatial annotation

An action is present in 700k frames out of 3.3M frames total. Annotating all of them would be very time consuming and clearly does not scale up to even larger collections.

Thus, we subsample the frames for spatial annotation. For each temporal instance, we pick 5 frames uniformly sampled over time, with a maximum of 1 frame per second. For each frame, annotators are asked to draw a bounding box around the actor, a bounding box around the object(s) involved in the action (e.g. the glass/cup for *drinking*), and the pose of the upper body of the actor (bounding box around the head and keypoints for shoulders, elbows and wrists). Some of the spatial annotations are completed by external workers, but all of them are reviewed in-house and adjusted when necessary. Figure 6 shows a few examples of spatial annotation.

Fig. 5. Example frames from the DALY dataset with simultaneous actions.

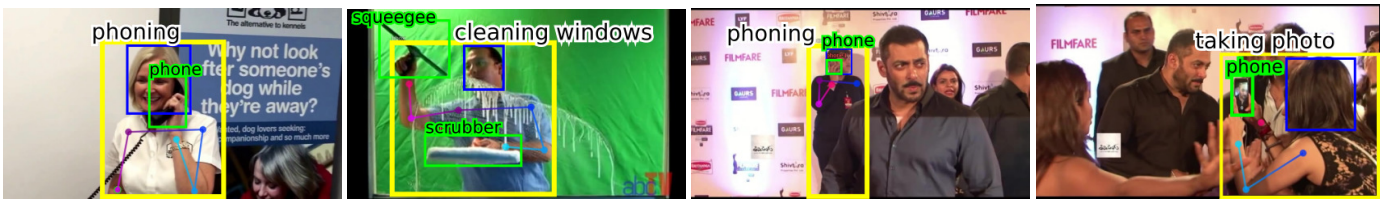

Fig. 6. Example of spatial annotation from the DALY dataset. In addition to the bounding box around the actor (yellow), we also annotate the objects (green) and the pose of the upper body (bounding box around the head in blue and joint annotation for shoulders, elbows and wrists).

4.3 Evaluation protocol

We measure detection performance using the standard mean Average Precision (mAP) metric. Following the Pascal VOC protocol [50], a detection is considered 'correct' if (a) the intersection over union (IoU) with the ground-truth is above a threshold δ, and (b) the detection is correctly classified. Duplicate detections are considered as false positives. The IoU between two spatio-temporal tubes is defined as the IoU in the temporal domain multiplied by the average of the spatial IoU between boxes averaged over all frames in the temporal intersection. For the DALY dataset, the averaged spatial overlap is computed only over the annotated frames in the temporal interval. Average Precision (AP) is computed for each class, and the mean over all classes (mAP) is reported. The IoU threshold δ is set to 0.5 when measuring spatial localization in trimmed clips and 0.2 for detection in space and time unless stated otherwise. We denote by mAP@δ the mAP computed at an IoU threshold δ.

5 Evaluating Human Tubes

In this section we experimentally evaluate our human tubes. We first evaluate the human detector in Section 5.1, and then the human tubes in Section 5.2. We also give implementation details.

5.1 Human detector

Implementation details

We use the state-of-the-art detector Faster R-CNN [19] to train our human detector. We use the approximate joint training[4] with the VGG-16 network architecture [51]. In Faster R-CNN, proposals are generated from anchors of size 8, 16 and 32 pixels on features maps with a stride of 16 pixels. We add a smaller scale of 4 pixels to the anchors, as it improves robustness to small humans. We found that this consistently boosts performance, i.e., the recall at an IoU threshold 0.5 for detections with a score over 0.5 is increased by 0.2% on UCF-Sports, 1.3% on J-HMDB and 3.7% on UCF-101. Other parameters are kept similar to the original implementation.

4. https://github.com/rbgirshick/py-faster-rcnn

score th.	UCF-Sports	J-HMDB	UCF-101	DALY
0.5	90.5% (33)	91.4% (27)	58.9% (15)	81.8% (19)
0.1	94.0% (51)	95.1% (43)	66.7% (24)	90.1% (56)

TABLE 3
Recall@0.5 of the human detections when thresholding them at a score over 0.5 or 0.1, on all annotated frames from the UCF-Sports, J-HMDB, UCF-101 and DALY datasets. The number in parenthesis indicates the average number of detections per frame before non-maximum suppression.

External training data

We use the MPII Human Pose dataset [20] to obtain training data with sufficient variability. The publicly available training set used here contains 28k annotated poses, including a bounding box around the head and joint positions. The images come from around 4000 videos, selected to contain around 500 different activities. We obtain a bounding box for each person by taking the box containing the head and all visible joints, with a fixed additional margin of 20 pixels. The bounding boxes are thus not perfect, see Figure 7. For instance, they can be slightly too large (top of bounding boxes from left image) or may not cover the extremity of the limbs (second image). The bounding boxes may also be cropped if some joints are not visible (last two images). Nevertheless, this dataset remains large enough and offers a huge variability in term of poses. It is thus well suited for training an accurate human detector.

Evaluation

Figure 8 shows some results of our human detector. It is robust to unusual poses (first two examples), to humans that are not fully visible (third image), and can detect multiple people (right example). To numerically evaluate the detection performance, we measure Recall@0.5, i.e., the ratio of ground-truth boxes for which at least one human detection has an Intersection Over Union (IoU) over 0.5. We consider as human detections the boxes for which the human probability is over 0.5. Table 3 reports the Recall@0.5

Fig. 7. Example of training examples from the MPII Human Pose dataset [20]. We display the joint annotation in yellow and the bounding box used for training in green.

Fig. 8. Example results of our human detector which consists of Faster R-CNN [19] trained on the Human MPII Pose dataset [20]. The first, second and fourth examples come from the UCF-Sports dataset, the third one from J-HMDB.

	UCF-Sports	J-HMDB	UCF-101
ImageNet (I)	29.9% (5.1)	63.1% (2.5)	10.4% (4.3)
Pascal VOC (I)	69.5% (3.4)	92.2% (1.5)	30.7% (2.8)
MPII (I)	**97.4%** (2.3)	95.2% (1.2)	**65.5%** (2.6)
MPII (H)	96.8% (3.2)	95.3% (1.4)	59.1% (3.2)
MPII (I+H)	**97.4%** (2.7)	**95.5%** (1.3)	65.0% (3.0)
Linking	96.1% (267.6)	97.7% (276.2)	53.9% (223.3)

TABLE 4
Recall@0.5 of our human tubes with variants of our tracker on all videos from the UCF-Sports, J-HMDB and UCF-101 datasets. The number in parenthesis indicates the average number of tubes per video. We study the impact of the training data, as well as the impact of the human detection score (H) and the instance-specific detection score (I) in the tracking strategy. We also compare our tracker to a linking approach.

on several datasets. We can see that our human detector performs well on all of them, with over 90% on the UCF-Sports and J-HMDB datasets. On UCF-101 we obtain a lower recall, which can be explained by the low quality (and high compression) of the videos. On the DALY dataset, we obtain a recall of around 80% and find that most missed humans are small or occluded. We also report the Recall@0.5 at a lower score threshold of 0.1 in Table 3 and we can see that the recall increases by 4 to 9%.

5.2 Evaluation of human tubes

Implementation details

The instance-level detector is a linear SVM on the last fully-connected layer of Faster R-CNN which has 4096 dimensions. The scores are converted to probabilities [45]. During initialization and update, we use the tracked box as positive samples. As negatives, we use the human proposals that have (a) an IoU below 0.1 and (b) a probability score over 0.1, i.e., hard negatives. For the sliding window performed in each frame, we use 5 different widths and heights equal to $\{80\%, 90\%, 100\%, 110\%, 120\%\}$ of the original size. In space, we use a grid of size 9×9 with stride 16 $pixels$ at each scale. Launching the tracker multiple times on a clip can be performed efficiently thanks to shared convolutional layers between all regions of a frame in Faster R-CNN. When detecting the humans, we can keep in cache the last convolutional layer (the one just before the Region-of-Interest pooling layer) as well as the features of the last fully connected layer. The first one will be used each time we perform a sliding window in the same frame, as the computation until the RoI pooling will be exactly the same. The second one will be used as the negative features for training the SVM.

Evaluation

To evaluate our human tubes we use Recall@0.5 at the tube level, i.e., the ratio of ground-truth instances for which at least one tube has an IoU over 0.5. We first measure the impact of the human-level detector and the instance-level detector, as well as the importance of training data. For the untrimmed UCF-101 dataset, we truncate all tracks to the ground-truth duration of the actions. Table 4 shows a comparison of multiple variants of the tracker with different pre-training sets for the network: (a) ImageNet (image classification task) [48], (b) Pascal VOC (object detection task) [50] and (c) MPII Human Pose. The tags (I) and (H) indicate that the instance-level and/or the human-level detectors are used in the tracking process. First, we can see that using features trained for human detection (I.e., on the MPII Human Pose dataset) is crucial: the performance significantly drops when using features from ImageNet or Pascal VOC. The fact that the network is pre-trained for human detection allows to effectively learn an instance-level detector for human tracking. Removing the instance-level detector decreases the performance by a few percent. Indeed, when multiple humans are present, relying only on the human-level detector may lead to drifting. The instance-level detector alone and its combination with the human-level detector give a similar performance. We are able to reach a recall of more than 95% with only 2 tubes on average per video, for UCF-Sports and J-HMDB. On the UCF-101 dataset, we obtain a Recall@0.5 of 65% with 3 tubes on average. We explain this lower recall by the lower performance

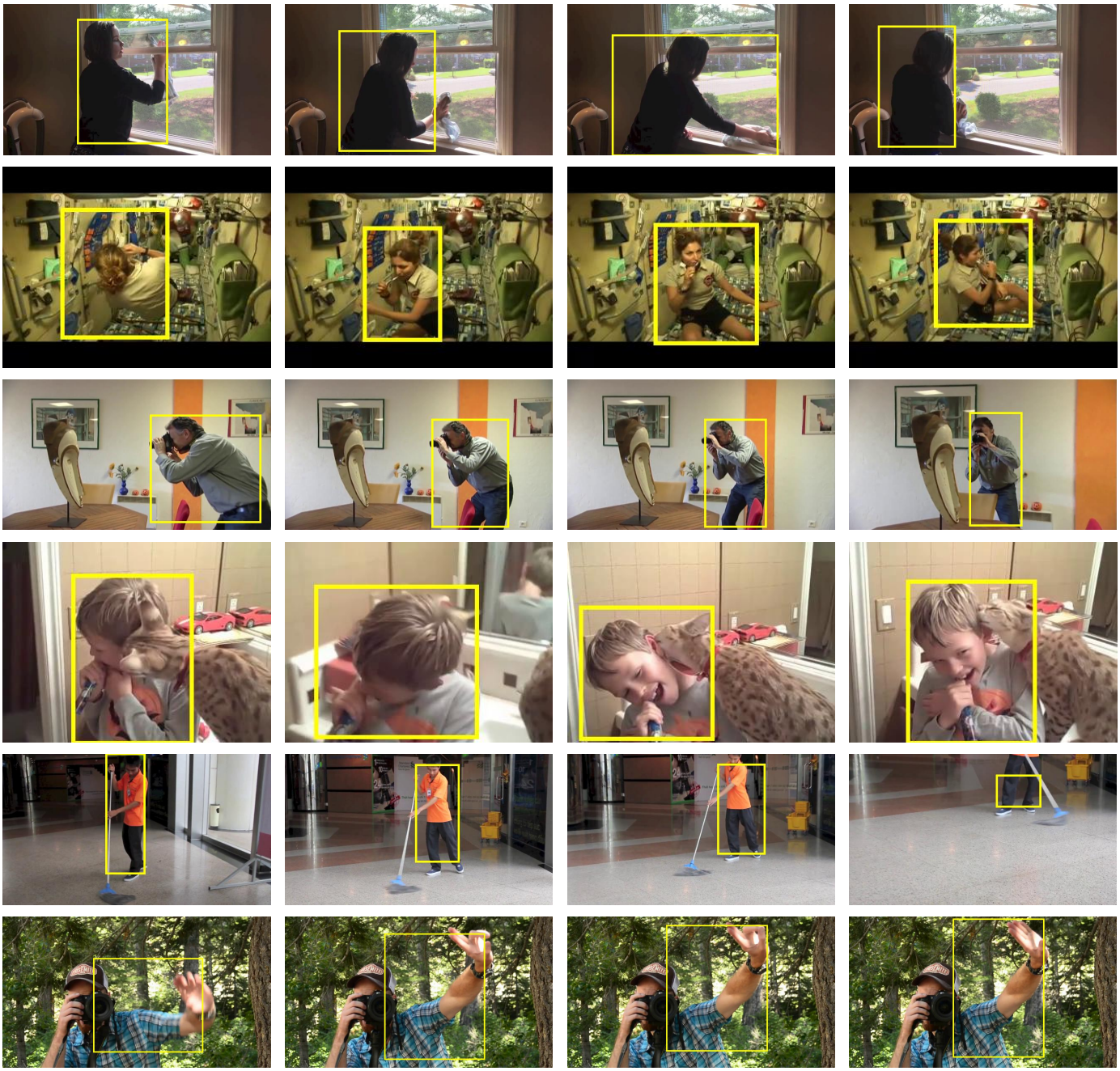

Fig. 9. Example of human tubes with successful human tube extraction in the first four rows, and some failure cases in the last two rows. Failures are caused by partial visibility of the human (end of fifth row) and missed human detection caused by an occluding camera (last row).

of the human detector, itself caused by the low quality of the videos that contain huge compression artifacts, as well as the fact that the humans are smaller. We also compare to a linking strategy similar to [6], [9], [10]. We can observe that the tracker reaches a higher recall, in particular for UCF-101. There is almost no difference on J-HMDB, which can be explained by the fact that the videos are extremely short and most of them contain only one human.

On the DALY dataset, the combination of the instance-level and human-level detectors in the tracker (MPII, I+H) obtains a Recall@0.5 of 91% with an average of 5 tubes per clip on the test set when measuring the spatial IoU on the annotated frames. We observe that most of the failure cases correspond to small humans or occlusions. Figure 9 shows the highest scoring human tube for several sequences of the DALY dataset. In the first four examples, we can see that the human tube performs well despite motion of one arm (first row), turning of the person (second and third row), camera motion (third row) or presence of an animal close to the human (fourth row). Nevertheless, there are some failure cases due to the fact that the full body disappears (fifth row). In this case, only the feet remain visible causing the failure of the human tracker, as the human detector performs poorly and the instance-level detector is trained on previous frames

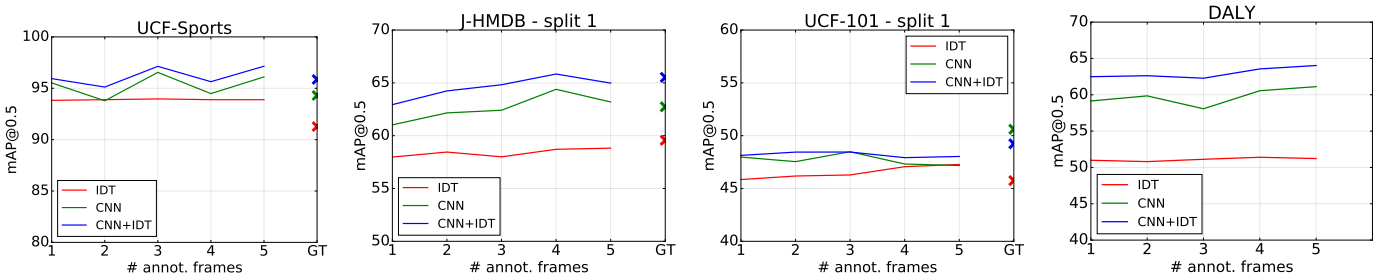

Fig. 10. Video-mAP@0.5 on trimmed clips when selecting positive tubes according to a varying number of frames. The cross indicates the performance when training on ground-truth tubes instead. Note that for the DALY dataset there is no ground-truth tube, and 5 frames means 'up to 5', as short instances may have less annotated frames than 5. We report results for CNN and IDT separately, and their late fusion of scores (CNN+IDT).

where the full body is visible. Another failure case is due to a partial occlusion by the camera (sixth row), which causes the human detector to fail.

6 Experimental results on action localization with sparse spatial supervision

In this section, we present experimental results measuring the performance of action localization when using sparse spatial supervision. In Section 6.1 we show that using few annotated frames and our human tube-based approach, we achieve results similar to fully-supervised methods. We compare this result to the impact of the number of annotated frames on Faster R-CNN used by state-of-the-art action localization methods in Section 6.2. Next, we compare our work with the state of the art on existing datasets in Section 6.3. We present experimental results on the DALY dataset in Section 6.4.

6.1 Impact of sparse spatial supervision

To measure the impact of sparse spatial supervision, we evaluate the video-mAP@0.5 on the trimmed clips using IDT, CNN and CNN+IDT features. Results are reported in Figure 10 for the UCF-Sports, J-HMDB, UCF-101 and DALY datasets. We compare to a baseline in which we train on the ground-truth tubes. More precisely, we use as positives the ground-truth tubes, and as negatives the human tubes that do not match any ground-truth (IoU below 0.5). The performance when training on the ground-truth is shown with the crosses.

We measure the impact of varying the number of annotated frames from 1 to 5 and compare to the performance with ground-truth annotations. Overall, the performance is not significantly impacted by the number of annotated frames, with variations limited to a maximum of 3% for all datasets. More interestingly, the difference between training on the ground-truth tubes and training with sparse supervision is also limited to this range. This is despite a significant decrease of annotation cost from 1 box per frame (i.e., 100 boxes for an instance of 100 frames) to 1 per action instance. In other words, for a fraction of the annotation cost we obtain similar results.

In more details, we can observe that the performance is almost constant in all cases for the UCF-Sports dataset, with a variation of less than 2% for all features. The slight variations when using the CNN features can be explained by the randomness when learning a CNN, e.g. weight initialization for the last layers or the random shuffling of the frames. On the other datasets, we can see a small increase in performance when moving from 1 annotated frame to 5, or when training with the ground-truth. However, this small increase is limited to around 2% in all cases.

In summary, the drop in performance when using only 1 annotated frame per instance is very small. This holds true for both CNN and IDT features as well as their combination. This demonstrates the effectiveness of leveraging human tubes in order to drastically reduce annotation effort when training an action detector.

So far, we consider that the sparse subset of the N annotated frames are uniformly sampled in time. The following experiments examine the impact of this choice, using IDT features and randomly choosing N annotated frames. We run this experiment 50 times and report the mean and the standard deviation (std) in Table 5 for different numbers of annotated frames. We obtain a std of around 0.5% on UCF-101 and J-HMDB. It is a bit higher with around 1.5% on UCF-Sports, which can be explained by the small number of videos resulting in higher variance. We can observe that the standard deviation decreases slightly as the number of annotated frames increases. The low standard deviation shows that the choice of the annotated frames has negligible impact, which can be explained by the high quality of our human tubes.

6.2 Impact of training data on Faster R-CNN

In this section, we measure the gain due to the additional annotations obtained by the human tubes, i.e., we train CNNs only on the sparse annotations of the actions without any additional human data. We resort to Faster R-CNN, which has shown state-of-the-art results for action detection at the frame level [9], [10]. We run a baseline two-stream Faster R-CNN on the same annotated frames as in our sparse supervision scheme, ranging from 1 to 5 annotated frames per action instance, uniformly sampled. To combine the two streams, we use the union of the Region-of-Interests from both streams [10] and fuse the scores of both streams using late fusion. We report the frame-mAP@0.5 in Figure 6 when varying the number of annotated frames. We also report the performance when training on all frames (GT).

#nframes	UCF-Sports	J-HMDB	UCF-101
1	92.2 ± 1.7	58.1 ± 0.6	45.5 ± 0.6
2	92.6 ± 1.6	58.5 ± 0.5	46.3 ± 0.4
3	93.0 ± 1.3	58.5 ± 0.4	46.7 ± 0.4
4	93.1 ± 1.4	58.5 ± 0.4	46.8 ± 0.5
5	93.0 ± 1.4	58.6 ± 0.3	46.9 ± 0.4

TABLE 5
Impact of random selection of annotated frames, the number of annotations per video ranges from 1 to 5. We report mean and standard deviation for video-mAP@0.5 on trimmed tracks using IDT features. Results on J-HMDB and UCF-101 are reported for the first split only.

#nframes	UCF-Sports	J-HMDB	UCF-101
1	67.9	50.9	53.6
2	77.4	50.7	59.1
3	79.6	55.8	61.2
4	81.0	56.2	62.5
5	81.2	56.5	63.8
GT	87.6	59.1	63.1

TABLE 6
Frame-mAP@0.5 of two-stream Faster R-CNN when training on the annotated frames only. GT refers to training on all frames. Results on J-HMDB and UCF-101 are reported for the first split only.

We can observe a clear drop between training on all frames (GT) and on one annotated frame per instance. The frame-mAP@0.5 decreases by 20% on the UCF-Sports, 8% on J-HMDB and 10% on UCF-101. The drop is lowest for J-HMDB, which can be explained by extremely short videos and thus reduced diversity in appearance (fewer unrelated images). The significant decrease of 20% on UCF-Sports is due to (a) a low number of videos in the training set and (b) longer instances with significant variation for some classes such as *Diving* or *Swinging*. When training on 5 annotated frames per video, there still exists a gap compared to training, in particular for small datasets (6% on UCF-Sports and 3% on J-HMDB). For larger datasets such as UCF-101, the frame-mAP@0.5 is on par with training on full ground-truth.

6.3 Comparison to the state of the art

We now compare our CNN+IDT approach to the state of the art on UCF-Sports, J-HMDB and UCF-101, see Table 7. We report our results with two settings: when training on sparse spatial annotations (first two rows) with 1 or 5 annotated frames per instance, and when training on the ground-truth tubes.

We first compare our results to Mettes et al. [15], a recent method which uses sparse supervision based on one point per frame instead of ground-truth tubes. We can observe that our approach outperforms theirs substantially by 25% despite the fact that their approach uses significantly more annotations: 1 point per frame compared to 1 box (i.e., 2 points) per instance. On the training set of the first split of UCF-101 this represents 1.9M points with their point annotation compared to 8k points in our approach. To examine the cause of the gain we report results with the same features and the same annotations. We use their point annotations, which are the center point of the bounding boxes, to select positive human tubes. We use as positives the human tubes for which at least 80% of the points are inside the tracks, and as negative the human tubes that have IoU below 0.5 with the positives. For a fair comparison, the results of our approach are reported using IDT features only. We report our results in Table 7. We obtain a similar performance with the point annotations scheme, 57.5% video-mAP@0.2 on UCF-101, than with our sparse supervision scheme: 57.1% when using IDT alone. These performances are still significantly higher than [15] with an improvement of 24% in video-mAP@0.2 on UCF-101. The remaining source of difference lies in the human tubes. To examine this, we compare the APT proposals [12] used by Mettes et al. [15] to our human tubes in Table 8 and report Recall@0.5. We can observe that the quality of our human tube is significantly higher, i.e., our human tubes obtain a significantly higher recall with only a few tubes. In contrast, APT outputs thousands of proposals and reaches a lower recall. This clearly explains the gap in performance.

We also compare our approach to state-of-the-art fully supervised approaches in Table 7. On the UCF-Sports dataset, we obtain state-of-the-art video-mAP with 95.9% when training on the ground-truth, and the same performance when using only one annotated frame per action instance. This performance is explained by the high quality of the human tubes (Section 5.2) and the fact that we combine IDT and CNN features.

For the J-HMDB dataset we obtain a video-mAP@0.5 of 65.8% with full supervision, which is slightly below the state of the art. The drop in performance compared to for example [10] can be explained by their additional multi-region and multi-flow description. Furthermore, using human tubes required in a weakly supervised case are likely to be the cause for an additional drop, as the regions are not selected and adapted to the action class. When considering that only one frame (resp. 5 frames) is spatially annotated for each action instance, we obtain a video-mAP of 63.9% (resp. 64.0%), which is less than 2% below the fully-supervised variant.

For the UCF-101 dataset, we obtain a video-mAP@0.2 of 58.9% with full supervision and of 57.4% with one box annotated per instance, i.e., there is almost no drop in performance due to using sparse supervision. This performance is on-par with most existing approaches but below [9], which can be explained by the relatively low recall of our tubes. At a threshold of 0.05 we perform better than most state-of-the-art methods. In particular, we significantly outperform [18] which also leverages a human detector.

6.4 Evaluation on DALY

We now present experimental results on DALY. We separately evaluate spatial detection in trimmed clips and spatio-temporal detection in full videos.

Action localization in trimmed clips

We first evaluate action localization in trimmed clips. To this end, we only test on the human tubes trimmed to the ground-truth temporal extent of the actions, i.e., we measure spatial detection performance, as in the case of the UCF Sports and J-HMDB datasets. Figure 10 reports the results when varying the maximum number of annotated frames from 1 to 5 using IDT, CNN and CNN+IDT features. We

Method	Annot.	features	UCF-Sports	J-HMDB	UCF-101 (split 1)		DALY
			mAP@0.5	mAP@0.5	mAP@0.05	mAP@0.2	mAP@0.2
ours (sparse)	1 frame	CNN (Fast R-CNN) + IDT	**95.9%**	**63.9%**	**70.0%**	**57.4%**	**14.5%**
	5 frames		**97.1%**	**64.0%**	**67.1%**	**57.3%**	**13.9%**
Mettes et al. [15]	points	IDT	-	-	-	32.4%	-
ours			**94.3%**	59.7%	-	**57.5%**	-
ours (sparse)	1 frame		93.9	**59.8**	-	57.1%	14.2%
ours	GT	CNN (Fast R-CNN) + IDT	**95.9%**	65.8%	71.1%	58.9%	-
Gkioxari and Malik [6]		CNN (R-CNN)	75.8%	53.3%	-	-	-
Weinzaepfel et al. [7]		CNN (R-CNN) + Handcrafted	90.5%	60.7%	54.3%	46.8%	-
van Gemert et al. [12]		IDT	-	-	58.0%	37.8%	-
Yu and Yuan [18]		IDT	-	-	42.8%	-	-
Saha et al. [9]		CNN (Faster R-CNN)	-	71.5%	**79.1%**	66.7%	-
Peng and Schmid [10]		CNN (Faster R-CNN)	94.8%	**73.1%**	78.8%	**72.9%**	-

TABLE 7
Comparison to the state of the art with video-mAP@0.5 on spatial localization datasets (UCF-Sports and J-HMDB) and video-mAP@0.2 for spatio-temporal action localization benchmarks (UCF-101 and DALY). For UCF-101 we also report video-mAP@0.05 to compare to [18] which also leverages a human detector. We first present the results of our method with sparse spatial annotation (first two rows). For comparison to Mettes et al. [15], we also report our performance using IDT only and their point annotation scheme that assumes that the center point of each box is given. We finally report our results when training on ground-truth as well as state-of-the-art fully-supervised approaches (GT).

	UCF-Sports	J-HMDB	UCF-101
Human Tubes	**97.4%** (2.7)	**95.3%** (1.3)	**65.0%** (3.0)
APT [[12]	89.4% (1449)	-	36.8 (2299)

TABLE 8
Comparison of our human tubes and APT [12] (used by Mettes et al. [15]) with Recall@0.5 over all videos for the UCF-Sports, J-HMDB and UCF-101 datasets. The average number of proposals is in parenthesis.

obtain a video-mAP@0.5 of 64.0% with CNN+IDT when using all annotated frames, i.e., up to 5 per instance, compared to 62.5% when using only one frame annotation. Once again, this gap is extremely small and validates the effectiveness of our training from sparse supervision. On the DALY dataset, we observe that IDT performs significantly worse than CNN. This can be explained by the fact that many instances are short in time, and consequently contain a relatively small number of trajectories when building the Fisher Vector representation with IDT.

Spatio-temporal action localization

We finally evaluate our method on action localization in space and time. We report a video-mAP@0.2 of 13.9% when training on all annotated frames, i.e., 5 per instance, see Table 7. We obtain a similar performance of 14.5% when considering only 1 annotated frame per action instance. The drop compared to spatial localization can be explained by the difficulty of temporal detection as the dataset contains both short and long actions in long untrimmed videos.

Figure 11 shows a few detection examples. In the first row, we are able to distinguish multiple instances of drinking performed by two different actors, with only a small false positive detection in the third column. In the second row we can observe that our detection is a bit too short, but well localized. Similar findings are also valid for the third row with two instances of the phoning actions. In the fourth row we detect both actions, folding textile and ironing, with an accurate time overlap. However, there is one short-lived false detection at the end of each detected instance. In the fifth row, our human tube is able to track and detect the cleaning floor action despite significant motion in the scene, but the temporal detection is cut into parts towards the end of the action. The last row shows a failure case in which a long action, playing harmonica, is detected as many small chunks. We can observe that this dataset requires a more sophisticated approach for localization in time.

7 Conclusion

We have presented an effective approach for extracting human tubes using a generic human detector. We have shown that this allows to significantly reduce the level of spatial supervision for training an action detector. In particular, when considering only one annotated frame per instance, the drop in performance is almost insignificant while the annotation cost is drastically reduced. We also introduced DALY, the first dataset for action localization in space and time in real-world untrimmed videos. It overcomes the limitations of existing datasets and will allow to measure progress in the field over the next years.

Acknowledgments

This work was supported in part by the ERC advanced grant ALLEGRO, the MSR-Inria joint project, a Google research award and a Facebook gift. We gratefully acknowledge the support of NVIDIA with the donation of GPUs used for this research.

References

[1] H. Wang, D. Oneata, J. Verbeek, and C. Schmid, "A robust and efficient video representation for action recognition," *IJCV*, 2015.
[2] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*, 2014.
[3] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *ICCV*, 2015.
[4] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *CVPR*, 2015.
[5] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Val Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *ECCV*, 2016.
[6] G. Gkioxari and J. Malik, "Finding action tubes," in *CVPR*, 2015.
[7] P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Learning to track for spatio-temporal action localization," in *ICCV*, 2015.

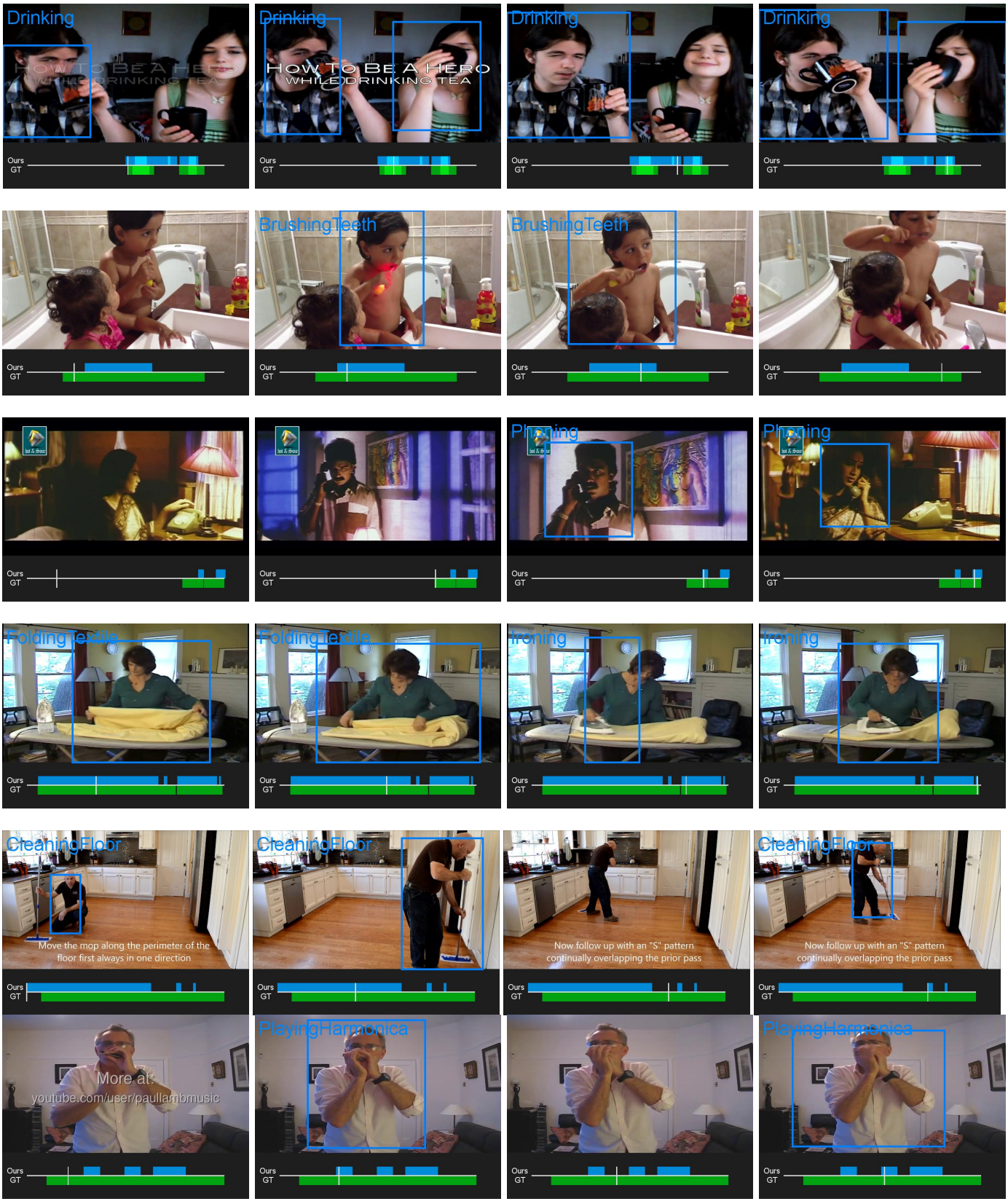

Fig. 11. Spatio-temporal detection examples on the DALY dataset. The timeline shows the ground-truth and our detection for a clip. The video cursor indicates the displayed frame, for which we show the action detection.

[8] L. Wang, Y. Qiao, and X. Tang, "Video action detection with relational dynamic-poselets," in *ECCV*, 2014.

[9] S. Saha, G. Singh, M. Sapienza, P. H. S. Torr, and F. Cuzzolin, "Deep learning for detecting multiple space-time action tubes in videos," in *BMVC*, 2016.

[10] X. Peng and C. Schmid, "Multi-region two-stream R-CNN for action detection," in *ECCV*, 2016. [Online]. Available: https://hal.inria.fr/hal-01349107v3

[11] M. Marian Puscas, E. Sangineto, D. Culibrk, and N. Sebe, "Unsupervised tube extraction using transductive learning and dense trajectories," in *ICCV*, 2015.

[12] J. C. van Gemert, M. Jain, E. Gati, and C. G. Snoek, "APT: Action localization proposals from dense trajectories," in *BMVC*, 2015.

[13] R. G. Cinbis, J. Verbeek, and C. Schmid, "Weakly Supervised Object Localization with Multi-fold Multiple Instance Learning," *IEEE Trans. PAMI*, 2016.

[14] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *CVPR*, 2016.

[15] P. Mettes, J. C. van Gemert, and C. G. Snoek, "Spot on: Action localization from pointly-supervised proposals," in *ECCV*, 2016.

[16] A. Kläser, M. Marszalek, C. Schmid, and A. Zisserman, "Human focused action localization in video," in *International Workshop on Sign, Gesture, and Activity (SGA)*, 2010.

[17] A. Prest, V. Ferrari, and C. Schmid, "Explicit modeling of human-object interactions in realistic videos," *IEEE Trans. PAMI*, 2013.

[18] G. Yu and J. Yuan, "Fast action proposals for human action detection and search," in *CVPR*, 2015.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.

[20] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *CVPR*, 2014.

[21] S. Hare, A. Saffari, and P. Torr, "Struck: Structured output tracking with kernels," in *ICCV*, 2011.

[22] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. PAMI*, 2012.

[23] I. Laptev and P. Pérez, "Retrieving actions in movies," in *ICCV*, 2007.

[24] L. Cao, Z. Liu, and T. S. Huang, "Cross-dataset action detection," in *CVPR*, 2010.

[25] J. Yuan, Z. Liu, and Y. Wu, "Discriminative subvolume search for efficient action detection," in *CVPR*, 2009.

[26] A. Gaidon, Z. Harchaoui, and C. Schmid, "Temporal localization of actions with actoms," *IEEE Trans. PAMI*, 2013.

[27] T. Lan, Y. Wang, and G. Mori, "Discriminative figure-centric models for joint action localization and recognition," in *ICCV*, 2011.

[28] A. Kläser, M. Marszaek, and C. Schmid, "A spatio-temporal descriptor based on 3D-gradients," in *BMVC*, 2008.

[29] M. Jain, J. van Gemert, H. Jégou, P. Bouthemy, and C. Snoek, "Action localization by tubelets from motion," in *CVPR*, 2014.

[30] W. Chen, C. Xiong, R. Xu, and J. Corso, "Actionness ranking with lattice conditional ordinal random fields," in *CVPR*, 2014.

[31] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid, "Spatio-temporal object detection proposals," in *ECCV*, 2014.

[32] P. Bojanowski, R. Lajugie, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic, "Weakly supervised action labeling in videos under ordering constraints," in *ECCV*, 2014.

[33] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce, "Automatic annotation of human actions in video," in *ICCV*, 2009.

[34] M. Hoai, L. Torresani, F. De la Torre, and C. Rother, "Learning discriminative localization from weakly labeled data," *Pattern Recognition*, 2014.

[35] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning Object Class Detectors from Weakly Annotated Video," in *CVPR*, 2012.

[36] P. Siva and T. Xiang, "Weakly supervised action detection," in *BMVC*, 2011.

[37] I. Laptev, "On space-time interest points," *IJCV*, 2005.

[38] E. A. Mosabbeb, R. Cabral, F. De la Torre, and M. Fathy, "Multi-label discriminative weakly-supervised human activity recognition and localization," in *ACCV*, 2014.

[39] S. Ma, J. Zhang, N. Ikizler-Cinbis, and S. Sclaroff, "Action recognition and localization by hierarchical space-time segments," in *ICCV*, 2013.

[40] W. Chen and J. J. Corso, "Action detection by implicit intentional motion clustering," in *ICCV*, 2015.

[41] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action MACH: A spatio-temporal maximum average correlation height filter for action recognition," in *CVPR*, 2008.

[42] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black, "Towards understanding action recognition," in *ICCV*, 2013.

[43] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild," in *CRCV-TR-12-01*, 2012.

[44] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *IJCV*, 2013.

[45] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *Journal of Machine Learning Research*, 2004.

[46] R. Girshick, "Fast R-CNN," in *ICCV*, 2015.

[47] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *ECCV*, 2004.

[48] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *IJCV*, 2015.

[49] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: a large video database for human motion recognition," in *ICCV*, 2011.

[50] M. Everingham, L. Van Gool, C. Williams, W. J., and A. Zisserman, "The PASCAL Visual Object Classes (VOC) Challenge," *IJCV*, 2010.

[51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

Philippe Weinzaepfel received a Master degree from Université Grenoble Alpes, France, and Ecole Normale Supérieure de Cachan, France, in 2012. He was a doctoral student in the THOTH team, at INRIA Grenoble and Laboratoire Jean Kuntzmann, from 2012 until 2016, and received a PhD degree in computer science from Université de Grenoble, France, in 2016. He is currently a research scientist at Xerox Research Centre Europe, France, in the computer vision group. His research interests include computer vision and machine learning, with special interest in video understanding and action recognition.

Xavier Martin is currently a research engineer in the THOTH team (INRIA Grenoble) since 2014. He received a Master's degree of Engineering from École Nationale d'Informatique et de Mathématiques Appliquées de Grenoble (ENSIMAG) in 2014. He participated in the European project AXES, the ERC grant Allegro and was an apprentice engineer in research team Inria MOAIS while preparing his degree (2011-2014).

Cordelia Schmid Cordelia Schmid holds a M.S. degree in Computer Science from the University of Karlsruhe and a Doctorate, also in Computer Science, from the Institut National Polytechnique de Grenoble (INPG). Her doctoral thesis received the best thesis award from INPG in 1996. Dr. Schmid was a post-doctoral research assistant in the Robotics Research Group of Oxford University in 1996–1997. Since 1997 she has held a permanent research position at INRIA Grenoble Rhone-Alpes, where she is a research director and directs an INRIA team. Dr. Schmid has been an Associate Editor for IEEE PAMI (2001–2005) and for IJCV (2004–2012), editor-in-chief for IJCV (2013—), a program chair of IEEE CVPR 2005 and ECCV 2012 as well as a general chair of IEEE CVPR 2015 and ECCV 2020. In 2006, 2014 and 2016, she was awarded the Longuet-Higgins prize for fundamental contributions in computer vision that have withstood the test of time. She is a fellow of IEEE. She was awarded an ERC advanced grant in 2013, the Humbolt research award in 2015 and the Inria & French Academy of Science Grand Prix in 2016. She was elected to the German National Academy of Sciences, Leopoldina, in 2017.