HACS: Human Action Clips and Segments Dataset for Recognition and Temporal Localization

Hang Zhao[†], Antonio Torralba[†], Lorenzo Torresani[‡], Zhicheng Yan[♭]
[†]Massachusetts Institute of Technology, [‡]Dartmouth College, [♭]University of Illinois at Urbana-Champaign

Abstract

*This paper presents a new large-scale dataset for recognition and temporal localization of human actions collected from Web videos. We refer to it as **HACS** (**H**uman **A**ction **C**lips and **S**egments). We leverage both consensus and disagreement among visual classifiers to automatically mine candidate short clips from unlabeled videos, which are subsequently validated by human annotators. The resulting dataset is dubbed **HACS Clips**. Through a separate process we also collect annotations defining action segment boundaries. This resulting dataset is called **HACS Segments**. Overall, HACS Clips consists of 1.5M annotated clips sampled from 504K untrimmed videos, and HACS Segments contains 139K action segments densely annotated in 50K untrimmed videos spanning 200 action categories. HACS Clips contains more labeled examples than any existing video benchmark. This renders our dataset both a large-scale action recognition benchmark and an excellent source for spatiotemporal feature learning. In our transfer learning experiments on three target datasets, HACS Clips outperforms Kinetics-600, Moments-In-Time and Sports1M as a pretraining source. On HACS Segments, we evaluate state-of-the-art methods of action proposal generation and action localization, and highlight the new challenges posed by our dense temporal annotations.*

1. Introduction

Recent advances in computer vision [22, 23] have been fueled by the steady growth in the scale of datasets. For image categorization, in the span of just a few years we transitioned from Caltech101 [15], which contained only 9.1K examples, to the ImageNet dataset [12], which includes over 1.2M examples. In object detection, we have seen a similar trend in scaling-up dataset sizes. Pascal VOC [13] was first released with 1.6K examples, while the COCO dataset [36] today consists of 200K images and 500K object-instance annotations. Open Images V4 [28] further scales up the size of image datasets to the next level.

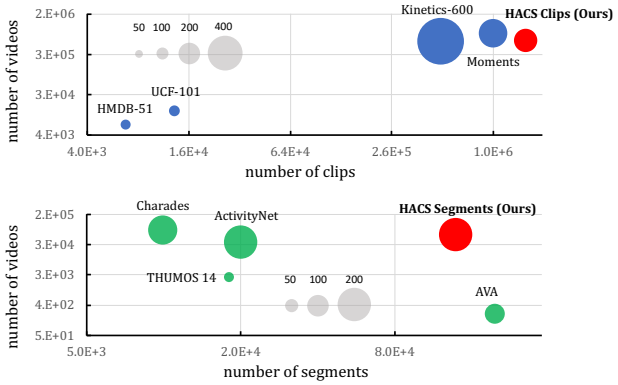

Figure 1: Comparisons of manually labeled action recognition datasets (**Top**) and action localization datasets (**Bottom**), where ours are marked as red. The marker size encodes the number of action classes in logarithmic scale.

It currently contains 9M images with image-level label and 1.7M images with 14.6M bounding boxes, and has greatly pushed the advances of research work in those fields [1, 19].

In the video domain, we have witnessed an analogous growth in the scale of action recognition datasets. While video benchmarks created a few years ago consists of only a few thousands examples (7K videos in HMDB51 [29], 13K in UCF101 [52], 3.7K in Hollywood2 [38]), more recent action recognition datasets, such as Sports1M [25], Kinetics [27] and Moments-in-Time [39], include two orders of magnitude more videos. However, for action localization, we have not seen a comparable growth in dataset sizes. THUMOS [24] was created in 2014 and contains 2.7K untrimmed videos with localization annotations over 20 classes. ActivityNet [6] only includes 20K videos and 30K annotations. AVA [42] includes 58K clips, and Charade [51] contains 67K temporally localized intervals. We argue that the lack of large-scale action localization datasets is impeding the exploration of more sophisticated models.

Motivated by the needs of large-scale action datasets, we introduce a new video benchmark, named *Human Action Clips and Segments* (*HACS*)*. It includes two types of man-

*Homepage: http://hacs.csail.mit.edu

ual annotations. The first type is action labels on *1.5M* clips of 2-second duration sparsely sampled from a half million of videos. We refer to this dataset as *HACS Clips*. It is designed to serve as a benchmark and as a pretraining source for action recognition. In our empirical study we compare different clip sampling methods and we observe that both consensus and disagreement over different visual classifiers can be used as criteria to identify clips especially worthy of annotation. Clips sampled from a large pool of videos according to such criteria capture large variations in action dynamics, context, viewpoint, lighting and other conditions. We demonstrate that spatiotemporal features learned on *HACS Clips* generalize well to other datasets.

The second type of annotation involves temporal localization labels on *50K* untrimmed videos, where both the temporal boundaries and the action labels of action segments are annotated. We call it *HACS Segments*. Thanks to our stringent guidelines on how to distinguish action and non-action segments, the resulting dataset has 1.8× more action segments per video and segments of shorter duration compared to ActivityNet. We demonstrate that this poses bigger challenges in action localization, as localizing short segments requires finer temporal resolution and more discriminative feature representations. Both types of annotation share the same taxonomy of 200 action classes, which we take from ActivityNet. We compare HACS with other video datasets in Figure 1. Despite being in its very first version, HACS compares favorably in scale to most prior benchmarks in this area. In summary, we make the following contributions in this paper.

1. We present a thorough empirical study on clip sampling methods, and use the nontrivial findings to sample a large number of clips for further manual verification. The resulting *HACS Clips* dataset has 2.5× more clip annotations compared to Kinetics-600.
2. We benchmark state-of-the-art action recognition models on *HACS Clips*. We show that *HACS Clips* outperforms Kinetics-600, Moments-In-Time and Sports1M as a pretraining dataset for action recognition on other benchmarks.
3. We collect action segment boundaries on 50K videos, based on annotation guidelines that reduce the ambiguity in the action definition and localization. The resulting *HACS Segments* has 2.5× more videos and 4.7× more action segments compared to ActivityNet.
4. On *HACS Segments*, we evaluate state-of-the-art methods of both action proposal generation and action localization, and highlight the new challenges.

2. Related Work

Action Recognition. In action recognition, the HMDB51 [29] and the UCF101 [52] datasets were created to provide benchmarks with higher variety of actions compared to precedent datasets, such as KTH [47]. These benchmarks have have enabled hand-design of motion features such as Spatial-Time Interest Point [30], Spatiotemporal Histogram of Oriented Gradient and Optical Flow [56, 31] and Fisher Vector feature encoding [41]. However, these datasets are not large enough to support modern end-to-end training of deep models. The large-scale Sports1M [26] and Kinetics datasets [27], which are over 20× larger than UCF101, were recently introduced to fill this gap. They enable the training of deep models from scratch [8, 43, 53]. However, these benchmarks cannot be used to train action localization models as they do not contain temporal boundary annotation. Collecting annotations on large-scale video datasets is time-consuming [50]. Previous work [37, 32] have shown that Web action images, which are widely available, can be exploited to train action classifiers, but such images cannot be used to learn motion features. Researchers have also explored synthetic generation of videos (e.g. VGAN [55], PHAV [11] for training action recognition models. Although this eliminates the need for human annotation, models trained on synthetic videos are still inferior to those trained on natural videos with human annotation.

Action Localization. Action localization in untrimmed videos is crucial to understanding Internet videos. Recently, several datasets for have been presented. THUMOS Challenge 2014 [24] includes 2.7K trimmed videos on 20 actions. It was subsequently extended into MultiTHUMOS [60] to have 65 action classes. Other datasets with fine granularity of classes but focused on narrow domains include MPII Cooking [45, 46] and EPIC-Kitchens [10]. Unfortunately models trained on such domain-focused datasets may not generalize well to every-day activities. Conversely, the Charades dataset [51] was purposefully designed to include more general, daily activities. ActivityNet-v1.3 [6] includes 20K untrimmed videos and 30K temporal action annotations. More recently, the AVA dataset [20] was introduced to provide person-centric spatiotemporal annotations on atomic actions. These datasets have substantially advanced the progress of research on action proposal generation [17, 18, 5, 35] and action localization [59, 48, 61, 34, 9, 2, 4].

3. Dataset Collection

3.1. HACS Dataset at a Glance

HACS uses a taxonomy of 200 action classes, which is identical to that of the ActivityNet-v1.3 dataset. It has 504K videos retrieved from YouTube. Each one is strictly shorter than 4 minutes, and the average length is 2.6 minutes. A total of 1.5M clips of 2-second duration are sparsely sampled by methods based on both uniform randomness and consen-

sus/disagreement of image classifiers. 0.6M and 0.9M clips are annotated as positive and negative samples, respectively. We split the collection into training, validation and testing sets of size 1.4M, 50K and 50K clips, which are sampled from 492K, 6K and 6K videos, respectively. We refer to this benchmark as *HACS Clips*. Furthermore, on a subset of 50K videos (38K for training, 6k for validation and 6K for testing) we collect manual boundaries defining the start, the end and the action label of every action segment in the video. All videos contain at least one action segment. We refer to this collection as *HACS Segments*.

3.2. Video Retrieval and De-duplication

We use 200 action labels to query the YouTube video search engine, and retrieve 890K potentially-relevant videos. The number of videos per class ranges from 1100 to 6600. We then perform two types of de-duplications. First, duplicate videos within *HACS* are removed. Second, to support fair assessment on other benchmarks, we remove videos that overlap with samples in the validation or test sets of other datasets, including Kinetics, ActivityNet, UCF-101 and HMDB-51. More details of video de-duplication are included in the supplement.

3.3. Sparse Clip Sampling

Manually annotating the start and the end of action segments in untrimmed videos is time-consuming. If the objective is to create a dataset for action recognition, it is more efficient to sparsely sample clips of short duration from a large number of videos, and ask annotators to quickly verify whether the presumed action is truly happening in the clip. This procedure can be used to gather a large-scale action clip dataset that can not only serve as an action recognition benchmark alone, but can also be leveraged for transfer learning, *e.g.*, by enabling the training of general deep models that can then be transferred for finetuning on smaller-scale datasets or employed in other downstream tasks.

One challenge in sampling clips is that the frequency of positive examples is arguably much smaller than that of negative examples. Thus, uniform random clip sampling would inevitably yield a large number of negative examples which are far less useful than positive examples for video modeling. On the other hand, using machine learning classifiers to guide the clip sampling can introduce dataset biases. For example, the collection of Kinetics [27] clips leveraged an image classifier trained on images automatically labeled by user feedback from Google Image Search. This classifier was used to sample clips with top action scores. The bias induced by such image classifier is certainly present in the data, yet it is difficult to assess.

In this section we are interested in the following two questions. First, *how can we assess the quality of clips sampled by different methods*? Second, *which clip-sampling method gives rise to the best training dataset*? To answer these questions, we present a thorough empirical study of clip-sampling strategies. An overview of the clip sampling pipeline used in our study is shown in Figure 2.

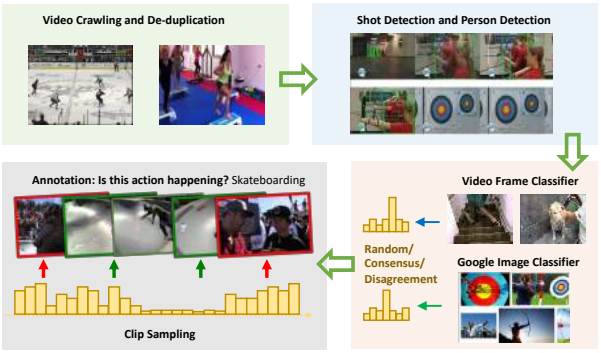

Figure 2: Our pipeline of sparse clip sampling and labeling.

3.3.1 Preprocessing: Removing Non-Person Clips

As a preprocessing step, we exclude clips that do not contain people since our aim is to create a dataset of human actions. To accomplish this, we first run a shot detection based on color histogram distance between video frames to segment the video into shots. After that we run a Faster R-CNN [44] person detector on two frames uniformly spaced in each shot, and remove shots with low average person detection scores.

3.3.2 A Study on Clip Sampling Methods

In this study, we compare three sampling methods: random sampling and two image classifier-based sampling methods. Prior work [37, 32, 21] on exploiting still images for action recognition has shown that still-image classifiers can predict actions in video reasonably well, despite their inability to model motion. Action context, such as objects typically involved in the action, prototypical scenes where actions occur, and other visual patterns that frequently co-occur with the action, can be captured by the image classifier for recognizing actions. To support our study, we first train two distinct image classifiers using training data from two different domains:

- **YouTube Frame Model**. The first model is trained on frames extracted from the top-500 videos retrieved by YouTube for each action class. Only video frames with person detected are used as positive samples for training. This gives a total of over 600K frames. As background (negative) samples we randomly choose frames with low person-score.

Clip Type	ME	Random	MC
Positive clips	71.3K	82.2K	**100.3K**
Negative clips	**168.7K**	157.8K	139.7K

Table 1: Comparing the frequency of positive and negative clips in three *Train-mini* sets sampled by different methods.

- **Google Image Model**. The second model is trained on images retrieved from the Google Image Search engine using the class labels as queries. We collect a total of 304K images after thresholding on person detection score. We use random samples from ImageNet as the examples of the background class. The image distribution is different from that of video frames, in terms of scene composition, background, and viewpoint.

For both classifiers we use a ResNet-50 trained with cross-entropy loss over 201 classes (200 action classes and 1 background class). The classifiers are applied to the central frame in each shot to get a probabilistic action prediction.

Next, we consider three different clip sampling methods:

1. **Random**. We randomly sample frames from each video.
2. **Maximum Entropy (ME)**. Within each video, we define the unnormalized sampling probability for the central frame of each shot as the average entropy of probabilistic predictions from the two image classifiers. We then apply L1-normalization to obtain a proper sampling distribution over the video. This method prefers to sample frames where the two classifiers disagree the most.
3. **Maximum Consensus (MC)**. Different from ME, the MC method defines the unnormalized sampling probability as the average action score from the two image classifiers for the action label that is used to retrieve the video. L1-normalization is also used. This method biases the sampling towards clips that receive a high score from both classifiers for the action label of interest.

Using these 3 sampling strategies, we collect 3 different sets of clips from a subset of training videos, which are denoted as *Train-mini-Random*, *Train-mini-ME* and *Train-mini-MC*, respectively. For each strategy, we randomly select 400 training videos per class, and sample 3 frames per video. Clips of 2-second duration centered around these sampled frames are sent to human annotators for manual verification, and each clip is marked as either positive or negative w.r.t the label of interest. Most action classes in our taxonomy are sufficiently distinct when observed in 2-second clips and annotating 2-second clips is also efficient.

Statistics of sampled clips. As shown in Table 1, MC method samples the highest number of true positive clips since clips with high scores based on the consensus of image classifiers are more likely to be true positive. However, these are also likely to be easy positive examples as they can be recognized by image classifiers. On the other side, ME yields the smallest number of true positives since it samples clips with conflicting predictions from image classifiers. This implies more uncertainty about the action class.

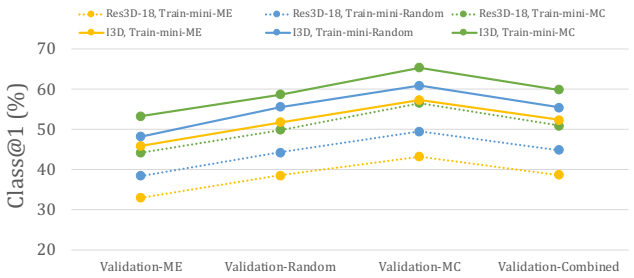

Figure 3: Evaluating Res3D-18 and I3D models trained on 3 different *Train-mini* sets on 4 different validation sets.

Evaluating clip sampling methods. We perform an empirical evaluation to address the two questions we asked in Section 3.3. Two models are used. A Res3D model [53] with 18 residual units (*i.e. Res3D-18*) and a I3D model [8]. Both take sequences of 16 frames as input. At training time, a random sequence of 16 frames within the clip is used. At evaluation time, 4 evenly spaced sequences of 16 frames are used and their predictions are averaged to obtain the final prediction. We train 3 separate instances of each model on the 3 different *Train-mini* sets. Since positive and negative clips are imbalanced, we adopt weighted sampling during training where the weight of each example is inversely proportional to the square root of the size of its class.

We also apply each sampling method to validation videos, and obtain 3 different sets of clips, namely *Validation-Random*, *Validation-ME* and *Validation-MC*, respectively, They are also manually verified by humans. We also combine all of 3 validation sets into a single one, namely *Validation-Combined*. We evaluate each trained model instance on all of 4 validation sets. Since the validation sets are also class-imbalanced, we report mean class accuracy (Class@1), which is obtained by averaging per-class accuracy over the 201 classes.

The results are shown in Figure 3. Models trained on the *Train-mini-MC* set consistently outperforms models trained on *Train-mini-Random* and *Train-mini-ME* sets on all validation sets. This suggests that for constructing a large-scale training set of clips under a constant human annotation budget, MC is the best method among those considered here because models trained on *Train-mini-MC* generalize best to all types of validation sets. On the other side, the *Validation-MC* set is easier than the others (models achieve higher accuracy), while *Validation-ME* is the most difficult for all models. This indicates that to construct a less biased validation/testing set, we should not rely on a single sampling method. Therefore, we propose to combine clips sampled by all of 3 methods in the final validation/testing set.

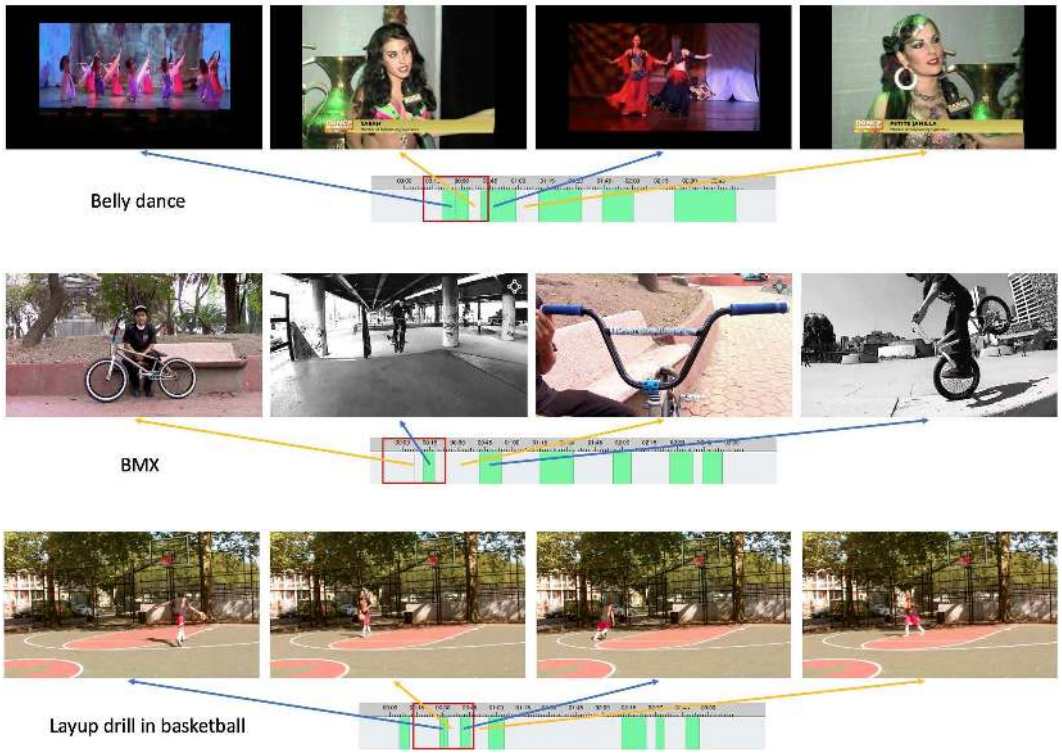

Figure 4: Examples of dense segment annotations. Action definition is clarified in the guideline to reduce the ambiguity of action boundaries.

3.4. Sparse Clip Annotation

We set up annotation tasks to label the sampled clips. **Annotation Guideline**. Different people may have different understandings of what constitutes a given action. To reduce the ambiguity, we prepare a detailed annotation guideline, which includes both clear action definitions as well as positive/negative examples with clarifications separately for each action. See more detailed guideline in the supplement. **Annotation Tool**. Our annotation tool supports display of up to 200 clips in a single page. We present clips sampled from the same video together. This not only reduces annotation inconsistency but also makes the annotation faster. **Quality Control**. We make two efforts to improve the annotation quality. First of all, each clip is labeled by three annotators, and only those clips with consensus from at least two annotators are included in the final dataset. Secondly, we ensure clips from the same class are labeled by the same group of annotators. This removes the inter-annotator noise.

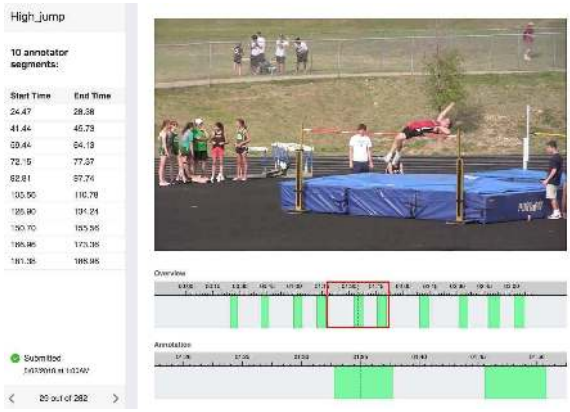

Figure 5: Action segment annotation tool. A timeline overview is shown below the video player, and a zoom-in view of current time window is shown in the bottom for accurate temporal annotation.

3.5. Dense Segment Annotation

HACS Clips alone are not sufficient for training and evaluating action localization methods as they lack temporal boundaries. Therefore, we ask annotators to densely label the start, the end and the action class of all action segments in a subset of 50K videos. A screenshot of our dense segment annotation tool is shown in Figure 5. We prepare clear annotation guidelines on distinguishing foreground action segments, where the action is being performed, and background segments, where both the person and the context (*e.g.* objects, scene) may appear but the action is not present. More importantly, we identify common patterns of the start and end of each action class. This helps annotators to better annotate the action segment boundaries. Examples of dense segment annotations are shown in Figure 4. For instance, for action *Belly Dance*, we consider the part of the video where dancers are being interviewed as background. For action *BMX*, we suggest to mark as background the part of the video where the person is explaining how to ride BMX bikes even though the rider and BMX bikes are visible. For action *Layup drill in backetball*, we clarify that the part of

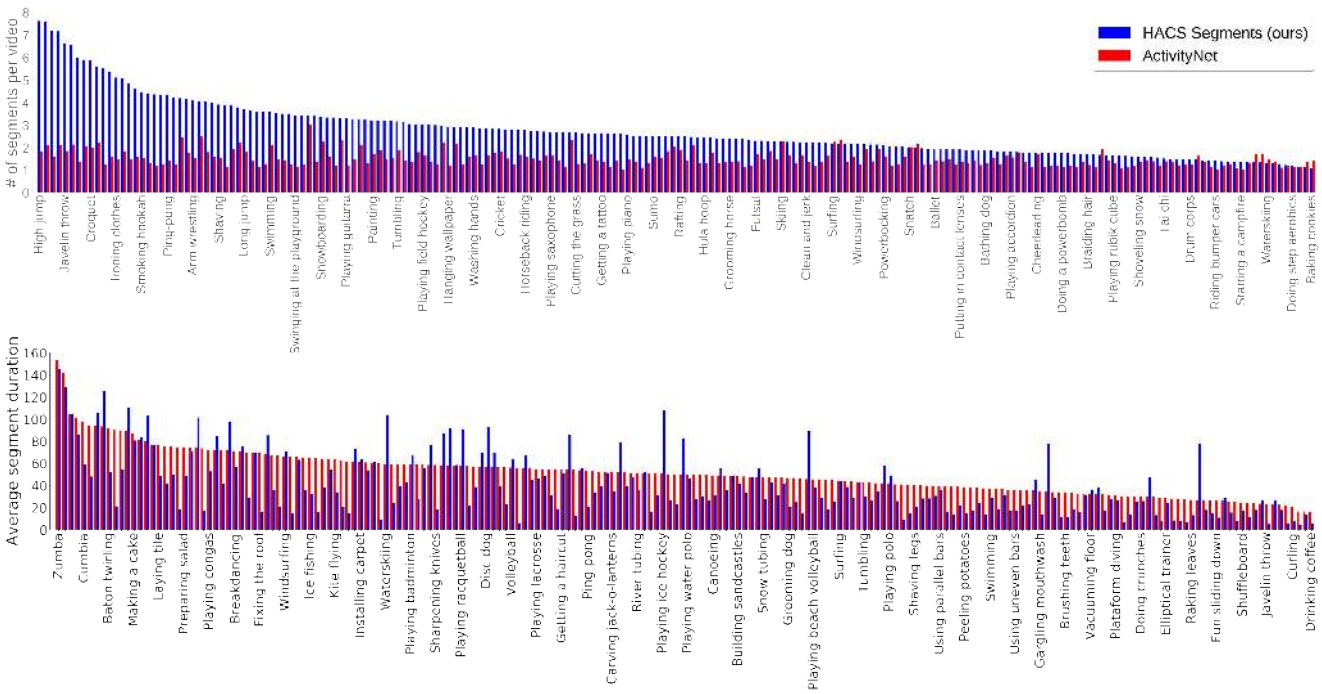

Figure 6: Comparing HACS Segments and ActivityNet. **Top**: comparing average number of action segments per video. On average, HACS Segments has 1.8× action segments per video (2.8 Vs. 1.5 segments). **Bottom**: comparing average segment duration. HACS segments are significantly shorter than those in ActivityNet (40.6 Vs. 51.4 seconds).

Input	RGB	Flow	RGB+Flow
Class@1	80.3	72.2	83.5

Table 2: Evaluating I3D models [8] on the validation set of HACS Clips.

the video where the player stands still or has finished the shooting should be marked as background.

The efficacy of our guidelines can be measured in numbers: compared to ActivityNet, HACS has on average 1.8× more action segments per video, and the average segment duration is about 20% shorter, as shown in Figure 6. This poses new challenges to action localization methods, which have to localize more segments of shorter duration.

3.6. Distinguishing Properties of HACS

Unlike other recognition datasets where only a single positive example is collected per video, HACS Clips includes also negative examples (each video contains 3 clips, with the negative to positive ratio being roughly 1 to 2). This could be used to model the discrepancy between action and non-action content. Moreover, videos in HACS Segments have both sparse clip annotation, which is a weak form of supervision for localization [57, 49, 40], and dense segment annotation, which can serve as the ground-truth of localization. Such hybrid annotation can be used for the task of weakly supervised action localization [57, 49], reminiscent of point supervision [3] and scribble supervision [33] in image semantic segmentation.

4. Action Recognition on HACS Clips

4.1. Action Clip Classification

In this section, we train I3D [8] on the full HACS Clips training set, and evaluate it on the validation set. We experiment with both RGB frames and optical flow as input. For efficiency, Farneback's algorithm [14] is adopted to compute optical flow. We also report results of combining RGB and optical flow by late fusion, where the final prediction score is a weighted sum of the prediction scores obtained from RGB and optical flow. We empirically set fusion weights for RGB and optical flow to 0.6 and 0.4, respectively. The results are shown in Table 2. We also show the class-specific accuracy, as well as the distribution of positive and negative clips per class in the supplement.

4.2. Results of Transfer Learning

Models trained on HACS Clips can be finetuned on other recognition datasets. By comparing finetuned models with models trained from scratch, we can assess the generalization performance of spatial-temporal features learned on HACS Clips. We evaluate the transfer learning on 3 action recognition benchmarks. On all benchmarks we observe substantial gains by pre-training on HACS Clips.

Datasets. We use a total of 6 additional datasets for our assessment. UCF-101, HMDB-51 and Kinetics-400 are used as target benchmarks. Sports1M, Moments-in-Time and Kinetics-600 [7], which is an extended version of the original Kinetics-400 dataset, are used as comparative pre-training datasets. For Kinetics-400, we report the accu-

Input	Pretraining	UCF101	HMDB51	Kinetics400
RGB	None	75.0	39.4	69.9
	Sports1M	92.8	68.3	71.0
	Moments	92.4	69.6	71.6
	Kinetics-600	94.9	73.4	72.9
	HACS Clips	**95.1**	**73.6**	**73.4**
Flow	None	85.2	56.1	62.9
	Sports1M	92.7	71.1	63.4
	Moments	94.6	75.3	63.9
	Kinetics-600	**96.0**	76.2	66.7
	HACS Clips	95.7	**76.5**	**67.2**

Table 3: Comparisons of HACS Clips with other datasets for pre-training I3D models. Results of UCF-101 and HMDB-51 are computed on split 1. *Moments* denotes Moments-In-Time dataset.

Pretrain Data	Method	UCF101	HMDB51	Kinetics-400
ImageNet	LTC-CNN [54]	92.7	67.2	N/A
	ST-Multiplier Net [16]	94.2	68.9	N/A
	TSN [58]	94.2	69.4	N/A
Sports1M	T-S R(2+1)D-34 [53]	97.3	78.7	75.4
Kinetics-400	T-S I3D [8]	98.0	80.7	75.7
HACS Clips	T-S I3D	**98.2**	**81.3**	76.4
	T-S R(2+1)D-34	98.0	79.8	76.1
	T-S R(2+1)D-101	N/A	N/A	**77.0**

Table 4: Comparing I3D and R(2+1)D models pretrained on *HACS Clips* with prior work. For UCF-101 and HMDB-51, average results over 3 splits are reported. Because R(2+1)D-101 model has 2× more residual units and 1.3× more parameters compared to R(2+1)D-34, it heavily overfits to the small datasets of UCF-101 and HDMB-51. Thus, we omit these results. We use *T-S* to denote Two-Stream.

racy on the validation set. For evaluation metric, we use Video@1 which is obtained by evenly sampling 10 clips in the video, and averaging the predictions.

Results. We train I3D models [8] without any use of 2D-to-3D inflation. When I3D models are pre-trained, we further fine-tune them on target benchmarks. As shown in Table 3, by pretraining on HACS Clips, the metrics are substantially improved on all 3 benchmarks. On all target datasets, HACS Clips shows better generalization performance compared to Sports1M, Moments-in-Time and Kinetics-600, where Kinetics-600 is the strongest competitor in this set. Sports1M annotations are noisy as they are generated by a tag prediction algorithm. Also, the average length of Sports1M videos is over 5 minutes and the tagged action may only be present for a short period of time. This introduces substantial temporal noise in learning spatial-temporal feature representation. Compared to Moments-in-Time, HACS Clips has a more fine-grained taxonomy for human actions, which helps generalization to other datasets. Compared to Kinetics-600, HACS Clips contains over 3× more annotations in the training set, which also contributes to the superior transfer learning performance.

Comparisons with other methods. In Table 4 we compare with the state-of-the-art. Both I3D [8] and R(2+1)D [53] model architectures are used here. For R(2+1)D, we report results of models with both 34 and 101 residual units after late fusion of RGB and flow scores. We compute video classifications by averaging predictions over 20 evenly-sampled clips in each video. By using the off-the-shelf I3D and R(2+1)D models, and leveraging a large-scale clip dataset, our approach outperforms other methods [54, 16, 58, 8, 53] on all 3 benchmarks.

Transfer learning on action localization. HACS Clips can also be used to pretrain action localization models. Compared with training from scratch, pretraining CDC models [48] on *HACS Clips* improves the average mAP by 8.6% on THUMOS 14 and by 2.5% on ActivityNet, respectively. See more detailed results in the supplement.

5. Action Localization on HACS Segments

We evaluate two action proposal generation methods and one action localization approach on HACS Segments.

5.1. Results of Action Proposal Generation

Two action proposal generation methods are evaluated: Boundary Sensitive Network (BSN) [35] and Temporal Actionness Grouping (TAG) [61]. We choose them because they achieve SoTA results on THUMOS 14 and ActivityNet benchmarks, and open-source implementations of these methods are available. We mostly follow the original training settings, and only highlight the differences below.

BSN Experiments. In the original work, snippet-level features are 400D, arising from a concatenation of two 200D probability vectors extracted from two TSN [58] models trained on 200 action classes of ActivityNet using RGB input and optical flow input, respectively. Analogously, here we train two TSN models (respectively taking RGB and flow as inputs) on HACS Clips with 200 action classes and 1 background class. The two 201D probability vectors from the trained models are concatenated to form 402D snippet-level features.

TAG Experiments. In the original work, two binary classifiers (based on TSN [58]) are trained on ActivityNet using RGB input and optical flow input, respectively. We use annotation in HACS Segments to train such binary classifiers.

We follow the original evaluation protocols, and report two metrics: 1) Average Recall (AR) vs Average Number (AN) of proposals per video and 2) area under AR-AN curve (AUC). Both are averaged over temporal Intersection over Union (tIoU) thresholds from 0.5 to 0.95 at increments of 0.05. Results are shown in Table 5 (Row 4 & 5) and

Method	Train/Test Dataset	AR@100	AUC
BSN	ActivityNet	74.16	66.17
	HACS Segments Mini	61.85	51.59
	HACS Segments	63.62	53.41
TAG	HACS Segments	55.88	49.15

Table 5: Action proposal generation results on ActivityNet and HACS Segments. BSN results on ActivityNet are from the original work [35]. Other results are obtained by running open-source implementations on HACS Segments.

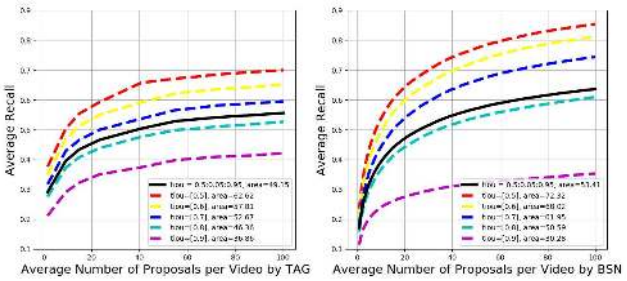

Figure 7: Action proposal generation results of TAG (**Left**) and BSN (**Right**) methods on HACS Segments.

Figure 7. Compared to TAG, BSN achieves both better AR@100 and better AUC score. However, TAG achieves higher AUC at high tIoU threshold 0.9 in Figure 7, indicating it is able to better localize action segment boundaries.

Comparing HACS Segments with ActivityNet. We use BSN to compare the difficulty of action localization on *HACS Segments* vs ActivityNet. While *HACS Segments* and ActivityNet have validation sets of similar size (6K *vs* 5K videos), the training set of *HACS Segments* is 3.8× larger than that of ActivityNet (38K *vs* 10K videos). To have a more fair comparison, we create *HACS Segments Mini*, which contains 10K training videos (50 videos per class) and the original *HACS Segments* validation set. We train and test each model on the training and validation splits of the same dataset (e.g., a model trained on *HACS Segments Mini* is tested only on the validation set of *HACS Segments*, not that of ActivityNet).

As shown in Table 5 (Row 2 & 3), compared to ActivityNet, BSN achieves much lower AR@100 and AUC score on HACS Segments Mini. This suggests HACS Segments Mini is a more challenging localization benchmark as it has more segments to localize in each video, and those segments have shorter duration. Note we do not experiment with models trained on one dataset and tested on a different one (say, trained on HACS Segments Mini and tested on ActivityNet) as the definitions where actions start, last and end may vary across datasets. Another finding is by training BSN models on the HACS Segments full dataset, AR@100 and AUC are improved by 1.77% and 1.82% in Table 5 (Row 4), which suggests that larger training sets lead to better accuracy.

Dataset	0.50	0.75	0.95	Average
ActivityNet [61]	43.26	28.70	5.63	28.28
HACS Segments Mini	24.89	16.04	4.50	15.93
HACS Segments	28.82	18.80	5.32	18.97

Table 6: Action localization results of SSN method for tIoU thresholds ranging from 0.5 to 0.95. Metric is mAP (%) . Results on ActivityNet are from the original work. Results on HACS Segments are obtained by late fusion of scores from RGB and Flow models.

Exploiting Negative Examples in HACS Clips. In HACS Clips, we annotated 1M negative clips. Due to the proposed clip sampling method, they include many hard negative examples, such as clips where both person and context are present, but action is not happening. We have conducted an ablation study on how they can help learn more useful features for action proposal generation. Due to space constraints, the results are presented in supplement.

5.2. Results of Action Localization

We train and test the Structured Segment Network (SSN) [61] on HACS Segments using its open-source implementation.[†]. Results are reported in Table 6. Compared to ActivityNet, localization average mAP on HACS Segments Mini is 12.35% lower. Given that ActivityNet and HACS Segments Mini have similar numbers and durations of untrimmed videos, the challenging nature of HACS comes from precise segment annotations. The average mAP gap between HACS Segments Mini and HACS Segments is 3.04%. This suggests that the reduction of training data hinders the action localization performance, and that our full-scale training set boosts the accuracy by a large margin.

6. Conclusion

We introduced a new video dataset with both sparse and dense annotations. We have demonstrated the excellent generalization performance of spatial-temporal feature learned on *HACS Clips* due to its large scale. Compared to other localization datasets, *HACS Segments* is not only larger, but it also poses new challenges in action localization through finer-scale temporal annotations. We hope the new challenges in action recognition and localization posed by *HACS* will inspire a new generation of methods and architectures for modeling the high complexity of human actions.

[†]BSN [35] is not benchmarked because its open-source code does not implement proposal classification.

References

[1] Takuya Akiba, Tommi Kerola, Yusuke Niitani, Toru Ogawa, Shotaro Sano, and Shuji Suzuki. Pfdet: 2nd place solution to open images challenge 2018 object detection track. *arXiv preprint arXiv:1809.00778*, 2018. 1

[2] Yancheng Bai, Huijuan Xu, Kate Saenko, and Bernard Ghanem. Contextual multi-scale region convolutional 3d network for activity detection. *arXiv preprint arXiv:1801.09184*, 2018. 2

[3] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. Whats the point: Semantic segmentation with point supervision. In *European Conference on Computer Vision*, pages 549–565. Springer, 2016. 6

[4] Shyamal Buch, Victor Escorcia, Bernard Ghanem, Li Fei-Fei, and Juan Carlos Niebles. End-to-end, single-stream temporal action detection in untrimmed videos. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017. 2

[5] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. Sst: Single-stream temporal action proposals. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6373–6382. IEEE, 2017. 2

[6] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015. 1, 2

[7] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 6

[8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 4, 6, 7

[9] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1130–1139, 2018. 2

[10] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[11] César Roberto de Souza, Adrien Gaidon, Yohann Cabon, and Antonio Manuel López. Procedural generation of videos to train deep action recognition networks. 2017. 2

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009. 1

[13] Mark Everingham, Luc. Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 1

[14] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. *Image analysis*, pages 363–370, 2003. 6

[15] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006. 1

[16] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4768–4777, 2017. 7

[17] Jiyang Gao, Kan Chen, and Ram Nevatia. Ctap: Complementary temporal action proposal generation. *arXiv preprint arXiv:1807.04821*, 2018. 2

[18] Jiyang Gao, Zhenheng Yang, Chen Sun, Kan Chen, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals, 2017. 2

[19] Yuan Gao, Xingyuan Bu, Yang Hu, Hui Shen, Ti Bai, Xubin Li, and Shilei Wen. Approach for large-scale hierarchical object detection. 1

[20] Chunhui Gu, Chen Sun, Sudheendra Vijayanarasimhan, Caroline Pantofaru, David A. Ross, George Toderici, Yeqing Li, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. AVA: A video dataset of spatio-temporally localized atomic visual actions. *CoRR*, abs/1705.08421, 2017. 2

[21] Guodong Guo and Alice Lai. A survey on still image based human action recognition. *Pattern Recognition*, 47(10):3343–3361, 2014. 3

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. *arXiv preprint arXiv:1703.06870*, 2017. 1

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1

[24] Yu-Gang Jiang, Jingen Liu, Amir Roshan Zamir, George Toderici, Ivan Laptev, Mubarak Shah, and Rahul Sukthankar. Thumos challenge: Action recognition with a large number of classes, 2014. 1, 2

[25] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 1

[26] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 2

[27] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 1, 2, 3

[28] Ivan Krasin, Tom Duerig, Neil Alldrin, Andreas Veit, Sami Abu-El-Haija, Serge Belongie, David Cai, Zheyun Feng, Vit-

torio Ferrari, Victor Gomes, et al. Openimages: A public dataset for large-scale multi-label and multi-class image classification. 2(6):7, 2016. 1

[29] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2556–2563. IEEE, 2011. 1, 2

[30] Ivan Laptev. On space-time interest points. *International journal of computer vision*, 64(2-3):107–123, 2005. 2

[31] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 2

[32] Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Attention transfer from web images for video recognition. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1–9. ACM, 2017. 2, 3

[33] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3159–3167, 2016. 6

[34] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 988–996. ACM, 2017. 2

[35] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *European conference on computer vision*, 2018. 2, 7, 8

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1

[37] Shugao Ma, Sarah Adel Bargal, Jianming Zhang, Leonid Sigal, and Stan Sclaroff. Do less and achieve more: Training cnns for action recognition utilizing action images from the web. *Pattern Recognition*, 68:334–345, 2017. 2, 3

[38] Marcin Marszalek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2929–2936. IEEE, 2009. 1

[39] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Yan Yan, Lisa Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 1

[40] Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. Weakly supervised action localization by sparse temporal pooling network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6752–6761, 2018. 6

[41] Dan Oneata, Jakob Verbeek, and Cordelia Schmid. Action and event recognition with fisher vectors on a compact feature set. In *Proceedings of the IEEE international conference on computer vision*, pages 1817–1824, 2013. 2

[42] Caroline Pantofaru, Chen Sun, Chunhui Gu, Cordelia Schmid, David Ross, George Toderici, Jitendra Malik, Rahul Sukthankar, Sudheendra Vijayanarasimhan, Susanna Ricco, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. 2017. 1

[43] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5534–5542. IEEE, 2017. 2

[44] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 3

[45] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele. A database for fine grained activity detection of cooking activities. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 1194–1201, 2012. 2

[46] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, pages 1–28, 2015. 2

[47] Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 32–36. IEEE, 2004. 2

[48] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-deconvolutional networks for precise temporal action localization in untrimmed videos. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1417–1426. IEEE, 2017. 2, 7

[49] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weaklysupervised temporal action localization in untrimmed videos. In *ECCV*, pages 162–179, 2018. 6

[50] Gunnar A Sigurdsson, Olga Russakovsky, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Much ado about time: Exhaustive annotation of temporal data. *arXiv preprint arXiv:1607.07429*, 2016. 2

[51] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, 2016. 1, 2

[52] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 1, 2

[53] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. *arXiv preprint arXiv:1711.11248*, 2017. 2, 4, 7

[54] Gul Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *IEEE Trans-*

actions on Pattern Analysis and Machine Intelligence, 2017. 7

[55] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Advances In Neural Information Processing Systems*, pages 613–621, 2016. 2

[56] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3169–3176. IEEE, 2011. 2

[57] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. Untrimmednets for weakly supervised action recognition and detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 6

[58] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Val Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 7

[59] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: region convolutional 3d network for temporal activity detection. In *IEEE Int. Conf. on Computer Vision (ICCV)*, pages 5794–5803, 2017. 2

[60] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *International Journal of Computer Vision*, pages 1–15, 2015. 2

[61] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017. 2, 7, 8