Scenes-Objects-Actions: A Multi-task, Multi-label Video Dataset

Jamie Ray[1], Heng Wang[1(✉)], Du Tran[1], Yufei Wang[1], Matt Feiszli[1], Lorenzo Torresani[1,2], and Manohar Paluri[1]

[1] Facebook AI, Menlo Park, USA
[2] Dartmouth College, Hanover, USA
{jamieray,hengwang,trandu,yufei22,mdf,torresani,mano}@fb.com

Abstract. This paper introduces a large-scale, multi-label and multi-task video dataset named **S**cenes-**O**bjects-**A**ctions (**SOA**). Most prior video datasets are based on a predefined taxonomy, which is used to define the keyword queries issued to search engines. The videos retrieved by the search engines are then verified for correctness by human annotators. Datasets collected in this manner tend to generate high classification accuracy as search engines typically rank "easy" videos first. The SOA dataset adopts a different approach. We rely on uniform sampling to get a better representation of videos on the Web. Trained annotators are asked to provide free-form text labels describing each video in three different aspects: scene, object and action. These raw labels are then merged, split and renamed to generate a taxonomy for SOA. All the annotations are verified again based on the taxonomy. The final dataset includes 562K videos with 3.64M annotations spanning 49 categories for scenes, 356 for objects, 148 for actions, and naturally captures the long tail distribution of visual concepts in the real world. We show that datasets collected in this way are quite challenging by evaluating existing popular video models on SOA. We provide in-depth analysis about the performance of different models on SOA, and highlight potential new directions in video classification. We compare SOA with existing datasets and discuss various factors that impact the performance of transfer learning. A key-feature of SOA is that it enables the empirical study of correlation among scene, object and action recognition in video. We present results of this study and further analyze the potential of using the information learned from one task to improve the others. We also demonstrate different ways of scaling up SOA to learn better features. We believe that the challenges presented by SOA offer the opportunity for further advancement in video analysis as we progress from single-label classification towards a more comprehensive understanding of video data.

Keywords: Video dataset · Multi-task · Scene · Object · Action

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11218, pp. 660–676, 2018.
https://doi.org/10.1007/978-3-030-01264-9_39

1 Introduction

In this work we introduce a new video dataset aimed at advancing research on video understanding. We name the dataset **S**cenes-**O**bjects-**A**ctions (SOA), as each video is annotated with respect to three different aspects: scenes, objects, and actions. Our objective is to introduce a benchmark that will spur research in video understanding as a comprehensive, multi-faceted problem. We argue that in order to achieve this goal a video dataset should fulfill several fundamental requirements, as discussed below.

Table 1. Statistics of the SOA dataset for different tasks.

Task	Scene	Object	Action	SOA
# videos	173K	560K	308K	562K
# classes	49	356	148	553
# annotations	223K	2.93M	484K	3.64M

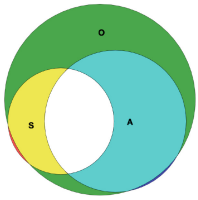

Fig. 1. Coverage of Scene, Object and Action labels on SOA videos. 105K videos (18.7%) have all three types of labels.

1. **Large-scale.** While KTH [29], HMDB51 [22] and UCF101 [34] have played a fundamental role in the past by inspiring the design of effective hand-engineered features for action recognition [23,40], larger video datasets are necessary to support modern end-to-end training of deep models. Datasets such as Sports1M [18], Kinetics [19] and AVA [27] were recently introduced to fill this gap and they have already led to the development of a new generation of more effective models based on deep learning [2,4,6,18,35,38,43]. SOA belongs to this new genre of large-scale video datasets. Despite being only in its first version, SOA already includes as many videos as Kinetics while containing ten times more annotations. Compared to crowdsourced datasets such as Charades [31] and Something-Something [9], SOA is both larger and more densely labeled. Table 1 summarizes the statistics of SOA.
2. **Unbiased Videos.** It is useful to fairly represent in the dataset the distribution of videos on the Internet. By doing so, models trained on the dataset can be directly applied to understand and recognize popular concepts in everyday Internet videos. For this purpose we build SOA by uniformly sampling videos from Web platforms. This procedure avoids biases on video length, content, metadata, and style. It provides a diverse collection of samples matching the actual distribution of Internet videos. On the contrary, prior datasets [1,18,19,34] have used keyword-based searches to find Web videos matching predefined concepts. The tags used for the searches skew the distribution of the dataset. Furthermore, search engines typically returns in the

top positions videos that match unambiguously the query. This yields prototypical examples that tend to be easy to classify. As evidence, the top-5 accuracy on Kinetics is already over 93% [24] less than one year from its public release. In our experiments we demonstrate that SOA is a much more challenging benchmark than prior datasets, with even the best video classification models hovering only around 45% top-5 accuracy[1].

3. **Unbiased Labels.** Rather than constraining annotators to adopt a predefined ontology to label the videos, as done in most prior video datasets, we allow annotators to enter free-from textual tags describing the video. We argue that this yields a more fitting set of annotations than those obtained by forcing labeling through a fixed ontology. The collection of free-form tags is then manually post-processed via concept renaming, deleting, merging and splitting to give rise to a final taxonomy, which directly reflects the distribution of labels given by annotators labeling the data in an unconstrained fashion. Moreover, SOA naturally captures the long tail distribution of visual labels in the real world, whereas existing datasets are often hand designed to be well balanced. This opens the door of studying few shot learning and knowledge transfer to model the long tail [41] on a large scale video dataset.
4. **Multi-task.** A video is much more than the depiction of a human action. It often portrays a scene or an environment (an office, a basketball court, a beach), and includes background objects (a picture, a door, a bus) as well as objects manipulated or utilized by a person (e.g., lipstick, a tennis racquet, a wrench). An action label provides a human-centric description of the video but ignores this relevant contextual information. Yet, today most existing video classification datasets contain only human action tags. While a few object-centric video datasets have been proposed [14,28], there is no established video benchmark integrating joint recognition of scenes, objects and actions. To the best of our knowledge the only exceptions are perhaps YouTube-8M [1] and Charades [31], where some of the classes are pure actions (e.g., wrestling), some represent objects (e.g., bicycle), and some denote "objects in action" (e.g., drinking from a cup). Unlike in these prior datasets, where contextual information (scenes and objects) is coupled with action categorization in the form of flat classification, we propose a dataset that integrates scene, object, and action categorization in the form of multi-task classification, where labels are available for each of these three aspects in a video. This makes it possible to quantitatively assess synergy among the three tasks and leverage it during modeling. For example, using SOA annotations it is possible to determine how object recognition contributes to disambiguating the action performed in the video. Furthermore, this multi-task formulation recasts video understanding as a comprehensive problem that encompasses the recognition of multiple semantic aspects in the dynamic scene. Figure 1 shows the coverage of annotations from different tasks on SOA videos.

[1] Top-5 accuracy on SOA is computed by considering each label from a given video independently, i.e., matching each label against top-5 predictions from the model.

5. **Multi-label.** Finally, we argue that a single class label per task is often not sufficient to describe the content of a video. Even a single frame may contain multiple prominent objects; the addition of a temporal dimension makes multi-label even more important for video than for images. As discussed above, datasets that use search queries to perform biased sampling can sidestep this issue, as they mostly contain prototypical examples for which a single-label assumption is reasonable. With closer fidelity to the true distribution and all of the hard positives it contains, the content of a given video is no longer dominated by a given label. In SOA we ask the annotator to provide as many labels as needed to describe each of the three individual aspects of recognition (Scenes, Objects and Actions) and we adopt mAP (mean Average Precision) as the metric accordingly.

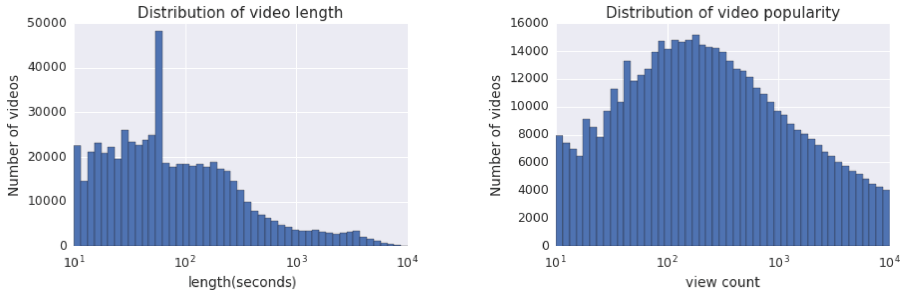

Fig. 2. Histograms of length and view count for sampled videos. These distributions contain heavy tails that will be lost by biased sampling.

2 Scenes-Objects-Actions

This section describes the creation of SOA in four steps: sampling videos, open-world annotation, generating the taxonomy and closed-world verification.

2.1 Sampling Videos

We sample publicly available videos shared on Facebook. The sampling is not biased by length or view count. The resulting videos are diverse and approximate the true distribution of Internet videos, as demonstrated by Fig. 2. From each video, we sample only one clip of about 10 s with the start time selected uniformly across the whole video. It is important to note that unbiased sampling yields an unbalanced long-tail class distribution, with many more videos containing mundane labels like "speaking to camera" compared to the kinds of actions popular in existing action recognition datasets, e.g., "ice skating".

After collecting the videos, we follow the protocol used for Kinetics [19] to de-duplicate videos within the SOA dataset. Our only modification is to use a ResNet-50 [11] image model as the feature extractor. We use the same protocol to remove SOA videos that match the testing and validation sets of the following action recognition datasets: Kinetics [19], UCF101 [33], and HMDB51 [21].

2.2 Open-World Annotation

The first stage of annotation provides an interface with a video player and three text-entry fields, one for each of the three SOA aspects (Scenes, Objects, and Actions). The annotator watches a clip (often multiple times) and types in any applicable textual tags corresponding to these three aspects. Note that the set of tags are not predefined. Each field includes an auto-complete mechanism so that the annotator does not need to type the whole tag. Each annotator is required to enter at least one label per aspect per clip. To improve recall, we send each clip to at least two annotators. The process takes on average 80 seconds per clip for a trained annotator.

2.3 Generating the Taxonomy

As described above, the initial round of labeling was unconstrained. The resulting free-form annotations were then cleaned in several ways. They were first sanitized to correct typos, unify synonyms and plurals, and merge similar terms. After this pass, only labels with more than 1500 samples were kept. The kept labels were then manually inspected and refined into a final taxonomy. The goals of the final taxonomy included:

1. Reduce label noise. Labels like "headphone" vs "headset", or "snowboard" vs "skateboard" were often confused, and we established guidelines for their use. In some cases this resulted in labels with less than 1500 samples being reintroduced.
2. Visual coherence. Certain free-form labels like "jumping" or "weight lifting" lacked visual coherence, and were replaced with more fine-grained labels. If there were not enough samples to split a label into multiple labels, we eliminated the incoherent label.
3. Sharing terminology. In structuring the final taxonomy we appealed to existing datasets and ontologies (e.g., MIT Places dataset [45], WordNet [26]) for guidance when possible, but there is no strict mapping to any existing taxonomy.

In particular, this process was aimed to preserve the true distribution of labels. The taxonomy was refined in certain areas and coarsened in others, so the granularity was changed, but additional videos were not retrieved to support new labels. Instead, all the videos were re-annotated with the new list of labels, as described below.

2.4 Closed-World Verification

When placing these labels into a visual taxonomy, we produced a set of mappings from free-form labels to curated labels. Many free-form labels were unchanged and mapped to a single curated label. Others were split or merged with other labels. These created mappings from free-form labels to groups of multiple curated labels.

Fig. 3. Different labels tend to co-occur in SOA. Here we visualize their relationship with t-SNE [25]. This embedding is purely based on label co-occurrence, without using video content. The superscript indicates the number of samples for each class. Scenes, Objects and Actions are in red, green and blue respectively. (Color figure online)

These mappings define a set of verification tasks for the second stage of annotation. Each label from the first stage may correspond to n labels in the new taxonomy (where n is zero if the label was discarded) for each aspect (Scenes, Objects, and Actions). These are provided to a second annotation tool which plays the video and displays these n choices as options (selected via hotkeys), with a default "NONE OF THE ABOVE" option included. Trained annotators watch a video and then select all labels that apply. This verification step takes about 30 seconds per clip on average. In practice, n is often equal to 1, making the task binary. This process can filter out erroneous labels, improving precision, but may yield low recall if the original labels or the mapping were too sparse. We noticed low recall for a small subset of labels and densified the mapping to correct for it. We measured the rate of "NONE OF THE ABOVE" to be about 30%. This indicates that our defined mapping provided a true label for 70% of the verification tasks.

Finally, we remove all the labels with less than 200 samples, and summarize the statistics of SOA in Table 1. Semantically related labels tend to co-occur on SOA, which we visualize using t-SNE in Fig. 3.

3 Comparing Video Models on SOA

This section compares different video models on SOA. We outline the experimental setup and three models used, then present and discuss the results.

3.1 Experimental Setup

SOA includes a total of 562K videos, which are randomly split into training, validation and testing with a percentage of 70, 10 and 20, respectively. For all the

experiments, we only use the training set for training and report metrics on the validation set. The performance on SOA is measured by computing the average precision (AP) for each class since it is a multi-label dataset. For each individual task (e.g., Scenes), we report the mean AP over all its classes (mAP). To measure the overall multi-task performance on SOA, we use a weighted average over the three tasks, by weighting each task differently to reflect the perceived importance of the three tasks to video understanding: $mAP_{SOA} = 1/6 * mAP_{Scene} + 1/3 * mAP_{Object} + 1/2 * mAP_{Action}$.

3.2 Video Models

We briefly describe the three popular video models used for evaluation on SOA.

Res2D. ResNet [11] is among the most successful CNN models for image classification. Res2D [39] applies a ResNet to a group of video frames instead of a single image. The input to Res2D is $3L \times H \times W$ instead of $3 \times H \times W$, where L is the number of frames and $H \times W$ is the spatial resolution. As the channel and temporal dimension are combined into a single dimension, convolution in Res2D is only on the two spatial dimensions. Note that 2D CNNs for video [32] ignore the temporal ordering in the video and are in general considered to be inferior for learning motion information from video.

Res3D. 3D CNNs [16,38] are designed to model the temporal dynamics of video data by performing convolution in 3D instead of 2D. Res3D [39] applies 3D convolutions to ResNet. Unlike Res2D, the channel and temporal dimensions are treated separately. As a result, each filter is 4-dimensional (channel, temporal and two spatial dimensions), and is convolved in 3D, i.e., over both temporal and spatial dimensions. Both Res2D and Res3D used in this paper have 18 layers.

I3D. The inflated 3D ConvNet (I3D) [4] is another example of 3D CNN for video data. It is based on the Inception-v1 [36] model with Batch Normalization [15]. I3D was originally proposed as a way to leverage the ImageNet dataset [5] for pre-training in video classification via the method of 2D-to-3D inflation. Here we only adopt this model architecture without pre-training on ImageNet as we are interested in comparing different model architectures on SOA trained under the same setup (no pre-training).

For a fair comparison, we use the same input to all three models, which is a clip of 32 consecutive frames containing RGB or optical flow. We choose the Farneback [7] algorithm to compute optical flow due to its efficiency. For data augmentation, we apply temporal jittering when sampling a clip from a given video. A clip of size 112×112 is randomly cropped from the video after resizing it to a resolution of 171×128. Training is done with synchronous distributed SGD on GPU clusters using Caffe2 [3]. Cross entropy loss is used for multi-label classification on SOA. For testing, we uniformly sample 10 clips from each video and do average pooling over the 10 clips to generate the video level predictions. We train all models from scratch with these settings unless stated otherwise.

Table 2. Three models trained with different inputs on SOA. For each task, we only use the videos and labels from that task for training and testing as listed in Table 1. Parameters and FLOPs are computed for RGB input. For optical flow, they are about the same as RGB.

Model	# params	FLOPs	Input	Scenes	Objects	Actions	SOA
Res2D	11.5M	2.6G	RGB	44.1	22.8	26.8	23.0
			Optical flow	29.7	14.6	21.5	16.7
			Late fusion	48.7	24.7	32.2	27.6
Res3D	33.2M	81.4G	RGB	48.0	25.9	33.6	27.3
			Optical flow	39.4	20.2	32.1	23.6
			Late fusion	**51.5**	**27.4**	**37.7**	**30.9**
I3D	12.3M	13.0G	RGB	45.4	22.6	30.3	24.5
			Optical flow	34.0	16.3	29.2	20.5
			Late fusion	49.4	24.4	35.4	28.5

3.3 Classification Results on SOA

Table 2 presents the mAP of each model, input, and task. For late fusion of RGB and optical flow streams, we uniformly sample 10 clips from a given video, and extract a 512-dimensional feature vector from each clip using the global average pooling layer of the trained model. Features are aggregated with average pooling over the 10 clips. We normalize and concatenate the features from RGB and optical flow. A linear SVM is trained to classify the extracted features.

Model vs. Task. Comparing the performance of different models in Table 2, we find that 3D models (i.e., Res3D and I3D) are consistently better than 2D models (i.e., Res2D) across different tasks. This indicates that 3D CNNs are generally advantageous for video classification problems. The gap between 2D and 3D models becomes wider when we move from Scene and Object tasks to Action task. This is presumably due to the fact that Scenes and Objects can often be recognized from a single frame, whereas Actions require more temporal information to be disambiguated and thus can benefit more from 3D CNNs.

Input vs. Model. We observe an interaction between the input modality and the model type. Optical flow yields much better accuracy when using 3D models, while in the case of RGB the performances of 2D and 3D CNNs are closer. For example, optical flow yields about the same mAP as RGB for Actions when using Res3D and I3D, but the accuracy with optical flow drops by about 5% when switching to Res2D. A similar observation applies to Scenes and Objects. This again suggests that 3D models are superior for leveraging motion information.

Task vs. Input. Choosing the right input for a target task is critical, as the input encapsulates all the information that a model can learn. RGB shows a great advantage over optical flow for Scenes and Objects. As expected, optical flow is more useful for Actions. Late fusion has been shown to be very effective for combining RGB and optical flow in the two-stream network [32]. The mAP of late fusion is about $2-4\%$ higher than each individual input in Table 2.

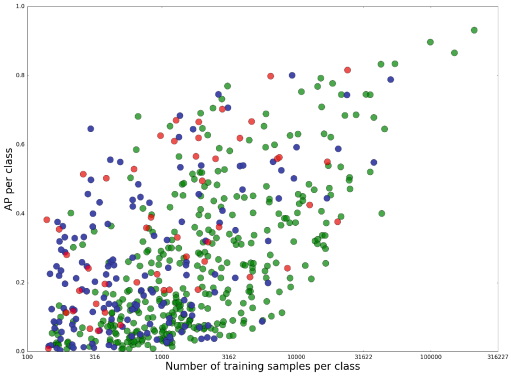

Fig. 4. The relationship between the Average Precision of each class and the number of training samples from that class. Scene, Object and Action classes are plotted in red, green and blue respectively. (Color figure online)

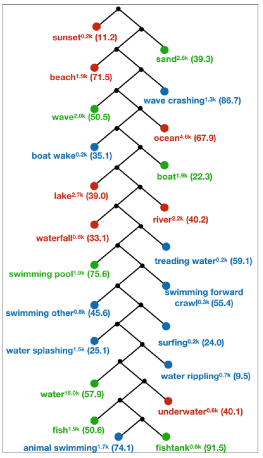

Fig. 5. Tree structure recovered from confusion matrix. We mark the number of training samples and testing AP for each class.

Overall, Res3D performs the best but is also the most computationally expensive, with the highest FLOPs and the most parameters, as shown in Table 2. Due to its strong performance, we use Res3D for the remaining experiments.

3.4 Discussion

In this section, we analyze the results from SOA in detail and highlight our findings. We choose the Res3D model with RGB as the input, which gives an mAP of 27.3 in Table 2. Figure 4 shows a strong correlation between AP and the number of positive samples in each class. The best two recognized classes for each task are man, overlaid text, grass field, gymnasium indoor, exercising other, speaking to camera, which are all very common categories in SOA.

To further understand the performance of the model, we construct a confusion matrix. As SOA is a multi-label dataset, we take the top-5 predictions of each sample, and consider all the pair combinations for each prediction and each ground truth annotation. All these combinations are accumulated to compute the final confusion matrix. To find meaningful structures from the confusion matrix, we recursively merge the two classes with the biggest confusion. This results in different tree structures where many classes are progressively merged together. Figure 5 shows such an example. We can clearly see that concepts appearing in the tree are semantically related with an increasing level of abstraction. There is also a gradual shift of concepts from fish to water, then water related scenery and activities, and drifting away to beach, sand and sunset.

Table 3. Comparison of SOA with Kinetics and Sports-1M for transfer learning. We consider four target datasets for fine-tuning including UCF101, HMDB51, Kinetics and Charades. Note that all these experiments are based on the Res3D model with RGB as the input. We report mAP on Charades and accuracy on the other three datasets.

Pre-training \ Fine-tuning	UCF101	HMDB51	Kinetics	Charades
From scratch	67.6	33.1	63.9	9.8
Kinetics	92.6	69.6	N/A	16.8
Sports-1M	90.2	63.7	64.6	13.7
SOA	84.7	57.2	63.9	15.3

We also found other trees that centered around concepts that are related to animals, cosmetics, vehicles, gym activities, etc. As in Fig. 5, these trees typically include multiple labels covering Scene, Object and Action. This is another evidence that Scene, Object and Action tasks should be solved jointly for video understanding and SOA provides an opportunity for driving computer vision research along this direction.

4 Transfer Learning

Strong transfer learning performance was not a design goal for SOA, however it is quite natural to ask what the strengths and weaknesses are with respect to this objective. The section discusses the results of using SOA for transfer learning, i.e., pre-training on SOA and fine-tuning on smaller datasets. We briefly describe the datasets used, and compare SOA with existing large-scale video datasets. We then discuss features of SOA that may influence its transfer learning ability and conclude by comparing with the state of the art.

4.1 Datasets

We compare SOA with Sports-1M [18] and Kinetics [19] for pre-training, and evaluate the performance of fine-tuning on four target datasets, i.e., UCF101 [34], HMDB51 [21], Kinetics and Charades [31].

Sports-1M is a large-scale benchmark for fine-grained classification of sport videos. It has 1.1M videos of 487 fine-grained sport categories. We only use the training set of Sports-1M for pre-training. **Kinetics** has about 300K videos covering 400 action categories. The annotations on the testing set are not public available. Here we use the training set for pre-training and report the accuracy on the validation set. **UCF101** and **HMDB51** are among the most popular datasets for action recognition. UCF101 has 13k videos and 101 classes, whereas HMDB51 is slightly smaller with 7k videos and 51 classes. Both datasets provide three splits for training and testing. We only use the first split in our experiments.

Unlike the other datasets, **Charades** is collected by crowdsourcing. It consists of 10k videos across 157 action classes of common household activities. We report mAP on the validation set of Charades.

Table 4. Compare the effectiveness of pre-training on SOA with the state of the art. For late fusion, we follow the same procedure described in Sect. 3.3 by combining the RGB results from Table 3 with the optical flow results listed in this table.

Methods	UCF101	HMDB51	Kinetics	Charades
ActionVLAD+iDT [8]	93.6	69.8	-	21.0
I3D (two-stream) [4]	98.0	80.7	75.7	-
MultiScale TRN [44]	-	-	-	25.2
S3D-G [42]	96.8	75.9	77.2	-
ResNeXt-101 (64f) [10]	94.5	70.2	65.1	-
SOA (optical flow)	86.5	65.6	59.1	16.1
SOA (late fusion)	90.7	67.0	67.9	16.9

4.2 Transfer Learning Results

We compare SOA with two popular large-scale datasets: Sports-1M and Kinetics. Fine-tuning performance is evaluated on UCF101, HMDB51, Kinetics, and Charades. The results are presented in Table 3. First, the improvement from pre-training is inversely related to the size of the fine-tuning dataset. For large datasets (e.g., Kinetics), the gain by pre-training is much smaller than datasets with less samples (e.g., UCF101, HMDB51, Charades). Pre-training is often used to mitigate scarcity of training data on the target domain. If the fine-tuning dataset is large enough, pre-training may not be needed.

Our second observation is that the improvements are also related to the source of the videos used for creating the datasets. UCF101, HMDB51, Kinetics and Sports-1M are all created with YouTube videos, whereas SOA uses publicly available videos shared on Facebook. Charades is built by crowdsourcing. Typically, improvements are largest when the pre-training and fine-tuning datasets use the same video source (e.g. YouTube) and sampling method (e.g., querying search engines). This is connected to the issue of dataset bias, which has already been observed on several datasets [37]. In Table 3, Kinetics performs remarkably well on UCF101 and HMDB51, but the gain becomes less pronounced on Charades. For SOA, its transfer learning ability is on par with Sports-1M and Kinetics on Charades, but is worse on UCF101 and HMDB51.

In Table 4 we compare against the state of the art in video classification by using SOA as a pre-training dataset for Res3D. State-of-the-art models tend to use more sophisticated architectures [42,44], more advanced pooling

mechanisms [8], deeper models [10], and heavyweight inputs [4,10] (long clips with higher resolution). Pre-training on SOA with a simple Res3D model gives competitive results in general. As shown in Sect. 5.3, the improvement from pre-training on SOA can be more significant as we scale up the dataset by either adding more videos or increasing the number of categories.

Table 5. Rows correspond to the target task, columns to the type of features extracted. Res3D with RGB input was used for all experiments.

Task \ Feature	Scene	Object	Action
Scene	49.7	53.9	45.6
Object	14.2	26.5	18.2
Action	18.3	29.9	34.8

(a) Correlation among the three tasks

Scene	S+O	S+A	SOA
	52.4	50.9	53.2
Object	O+S	O+A	SOA
	26.5	27.3	27.0
Action	A+S	A+O	SOA
	34.7	36.0	35.9

(b) How much one task can help another

5 Multi-task Investigations

SOA is uniquely designed for innovation in the large-scale multi-task arena. In this section we establish what we hope will be some compelling baselines about the interaction between features learned across tasks as an example of these kinds of questions. To our knowledge, SOA is the only dataset currently available on which such experimentation can be done. Previously, Jiang et al. [17] proposed to use context knowledge extracted from scene and object recognition to improve action retrieval in movie data. Ikizler-Cinbis et al. [13] extracted different types of features that can capture object and scene information, and combined them with multiple-instance learning for action recognition. More recently, Sigurdsson et al. [30] studied the effectiveness of perfect object oracles for action recognition.

5.1 Correlations Among the Three Tasks

For this experiment, we take the Res3D models (with RGB as the input) trained on the three individual tasks. We use each model in turn as a feature extractor for Scenes, Objects and Actions separately. The feature extraction process is the same as Sect. 3.3, i.e., average pooling the 512-dimensional Res3D feature vector over 10 clips for a given video. We then train a linear SVM on each of these three features for each of the three tasks (9 training runs in total).

The results are summarized in Table 5(a). It is interesting to compare the performance of the three task-specific Res3D models using RGB from Table 2 with the numbers on the diagonal axis of Table 5(a). The differences are explained by the usage of the SVM classifier on top of the Res3D features. In terms of overall performance considering all three tasks, Object features are the strongest

while Scene features are the weakest. Note that this ranking is also consistent with the number of annotations we have for each task (listed in Table 1).

Overall there are strong correlations among different tasks from our preliminary results in Table 5(a). For example, even when applying the weakest Scene feature on the hardest Object task, we achieve an mAP of 14.2, which is a decent result considering the difficulty of the Object task. This highlights the potentials of leveraging different information for each task and the usefulness of SOA as a test bed to inspire new research ideas.

At first glance, Table 5(a) appears to suggest that Object features are inherently richer than Scene features: Object features gives better accuracy (53.9 mAP) than Scene features (49.7 mAP) on Scene classification. However, SOA has over 13 times more annotations for Objects than Scenes. When we control the label count by reducing the number of feature-learning samples for Objects to be the same as Scenes, the mAP drops from 53.9 to 46.5, demonstrating that there is likely inherent value in the Scene features, despite the much smaller label space for Scene.

5.2 How Multiple Tasks Can Help Another

Here we study the effectiveness of leveraging several tasks to solve another. We follow the same procedure described in Sect. 5.1 with the difference that we combine multiple features by concatenating them together for each task. The results are presented in Table 5(b).

At a glance, simply concatenating different features does not seem to boost the performance of each individual task significantly. For the Scene task, combining all three features does improve the mAP from 49.7 to 53.2. However, the improvement becomes marginal for both the Object and the Action task. As Scene is the weakest descriptor, combining it with stronger features (such as Object) can make the Scene task easier, but not the other way around.

Moreover, fusing different features by concatenating them implies that each feature has the same weight in the final classifier. This is not ideal as the strength of each feature is different. It is, thus, appealing to design more sophisticated mechanisms to adaptively fuse different features together. There are many creative ways of exploiting the correlation among different tasks, such as transfer learning and graphical models [20] that we hope to see in future research.

5.3 Number of Videos vs. Number of Categories

The comparison of the Scene features with Object features in Sect. 5.1 suggests a more careful investigation of the tradeoffs between label diversity and number of labeled samples. Given a limited budget, and assuming the resource required for each annotation is the same, how should we spend our budget to improve the representational ability of SOA? As a proxy for richness of representation, we choose to use transfer learning ability. Huh et al. [12] investigated different factors that make ImageNet [5] good for transfer learning. Here we consider the effects of varying the number of samples and the number of categories for

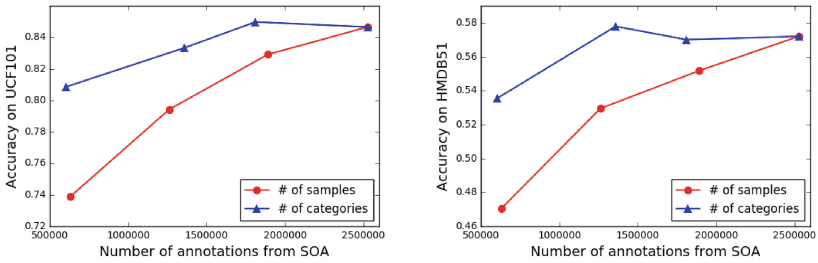

Fig. 6. How to scale up the transfer learning ability of SOA effectively: number of videos vs number of categories.

SOA. We then consider transfer performance as a function of the total number of annotations (as opposed to the total number of videos).

We randomly sample a subset (i.e., 25%, 50%, 75%, 100%) of either samples or categories to build a smaller version of SOA. In the first case, we randomly choose a given fraction of videos. In the second case, we randomly choose a given fraction of labels, remove all other labels from the dataset, and discard videos with no labels remaining. The second case generally yields more videos than the first case. A Res3D model is pre-trained with the smaller versions of SOA, and then fine-tuned on UCF101 and HMDB51.

The results in Fig. 6 are unequivocal: for a fixed number of annotations, a smaller label set applied to more videos produces better results. Fine-tuning accuracy on UCF101 and HMDB51 increases rapidly with respect to the number of videos used from SOA for pre-training, while performance seems to saturate as the number of categories is increased. This suggests that we can further boost the accuracy on UCF101 and HMDB51 by annotating more videos for SOA. This gives us a relevant guideline on how to extend SOA in the future.

6 Conclusions

In this work we introduced a new large-scale, multi-task, multi-label video dataset aimed at casting video understanding as a multi-faceted problem encompassing scene, object and action categorization. Unlike existing video datasets, videos from SOA are uniformly sampled to avoid the bias introduced by querying search engines, and labels originate from free-form annotations that sidestep the bias of fixed ontologies. This gives rise to a benchmark that appears more challenging than most existing datasets for video classification. We also present a comprehensive experimental study that provide insightful analyses on several factors of SOA, including performance achieved by popular 2D and 3D models, the role of RGB vs optical flow, transfer learning effectiveness, synergies and correlations among the three SOA tasks, as well as some observations that will guide future extensions and improvements to SOA.

As the design of SOA departs significantly from those adopted in previous datasets, we argue that the current and future value of our benchmark should be measured by its unique ability to support a new genre of experiments across different aspects of video recognition. We believe that this will inspire new research ideas for video understanding.

References

1. Abu-El-Haija, S., et al.: Youtube-8m: a large-scale video classification benchmark. arXiv preprint arXiv:1609.08675 (2016)
2. Ballas, N., Yao, L., Pal, C., Courville, A.: Delving deeper into convolutional networks for learning video representations. arXiv preprint arXiv:1511.06432 (2015)
3. Caffe2-Team: Caffe2: A New Lightweight, Modular, and Scalable Deep Learning Framework. https://caffe2.ai/
4. Carreira, J., Zisserman, A.: Quo vadis, action recognition? A new model and the kinetics dataset. In: CVPR (2017)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 248–255. IEEE (2009)
6. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2625–2634 (2015)
7. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Bigun, J., Gustavsson, T. (eds.) SCIA 2003. LNCS, vol. 2749, pp. 363–370. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-45103-X_50
8. Girdhar, R., Ramanan, D., Gupta, A., Sivic, J., Russell, B.C.: ActionVLAD: learning spatio-temporal aggregation for action classification. In: CVPR (2017)
9. Goyal, R., et al.: The? Something something? Video database for learning and evaluating visual common sense. In: Proceedings of ICCV (2017)
10. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3D CNNS retrace the history of 2D CNNS and ImageNet? arXiv preprint arXiv:1711.09577 (2017)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
12. Huh, M., Agrawal, P., Efros, A.A.: What makes ImageNet good for transfer learning? arXiv preprint arXiv:1608.08614 (2016)
13. Ikizler-Cinbis, N., Sclaroff, S.: Object, scene and actions: combining multiple features for human action recognition. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6311, pp. 494–507. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15549-9_36
14. ILSVRC-2015-VID: ImageNet Object Detection from Video Challenge. https://www.kaggle.com/c/imagenet-object-detection-from-video-challenge
15. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: ICML (2015)
16. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE TPAMI **35**(1), 221–231 (2013)
17. Jiang, Y.G., Li, Z., Chang, S.F.: Modeling scene and object contexts for human action retrieval with few examples. IEEE Trans. Circuits Syst. Video Technol. **21**(5), 674–681 (2011)

18. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: CVPR (2014)
19. Kay, W., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
20. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press, Cambridge (2009)
21. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB51: a large video database for human motion recognition. In: ICCV (2011)
22. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2556–2563. IEEE (2011)
23. Laptev, I.: On space-time interest points. Int. J. Comput. Vis. **64**(2–3), 107–123 (2005)
24. Long, X., et al.: Multimodal keyless attention fusion for video classification (2018)
25. Maaten, L.V.D., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(Nov), 2579–2605 (2008)
26. Miller, G.A.: Wordnet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)
27. Pantofaru, C., et al.: AVA: a video dataset of spatio-temporally localized atomic visual actions (2017)
28. Real, E., Shlens, J., Mazzocchi, S., Pan, X., Vanhoucke, V.: YouTube-BoundingBoxes: a large high-precision human-annotated data set for object detection in video. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7464–7473. IEEE (2017)
29. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, vol. 3, pp. 32–36. IEEE (2004)
30. Sigurdsson, G.A., Russakovsky, O., Gupta, A.: What actions are needed for understanding human actions in videos? In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, 22–29 October 2017, pp. 2156–2165 (2017)
31. Sigurdsson, G.A., Varol, G., Wang, X., Farhadi, A., Laptev, I., Gupta, A.: Hollywood in homes: crowdsourcing data collection for activity understanding. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 510–526. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_31
32. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: NIPS (2014)
33. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human action classes from videos in the wild. In: CRCV-TR-12-01 (2012)
34. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
35. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using LSTMS. In: International Conference on Machine Learning, pp. 843–852 (2015)
36. Szegedy, C., et al.: Going deeper with convolutions. In: CVPR (2015)
37. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1521–1528. IEEE (2011)
38. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: ICCV (2015)

39. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. arXiv preprint arXiv:1711.11248 (2017)
40. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3169–3176. IEEE (2011)
41. Wang, Y.X., Ramanan, D., Hebert, M.: Learning to model the tail. In: Advances in Neural Information Processing Systems, pp. 7029–7039 (2017)
42. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning for video understanding. arXiv preprint arXiv:1712.04851 (2017)
43. Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: deep networks for video classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4694–4702 (2015)
44. Zhou, B., Andonian, A., Torralba, A.: Temporal relational reasoning in videos. arXiv preprint arXiv:1711.08496 (2017)
45. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: NIPS (2014)