# The Action Similarity Labeling Challenge

Orit Kliper-Gross, Tal Hassner, Lior Wolf

**Abstract**—Recognizing actions in videos is rapidly becoming a topic of much research. To facilitate the development of methods for action recognition, several video collections, along with benchmark protocols, have previously been proposed. In this paper we present a novel video database, the "Action Similarity LAbeliNg" (ASLAN) database, along with benchmark protocols. The ASLAN set includes thousands of videos collected from the web, in over 400 complex action classes. Our benchmark protocols focus on action *similarity* (same/not-same), rather than action classification, and testing is performed on *never-before-seen* actions. We propose this data set and benchmark as a means for gaining a more principled understanding of what makes actions different or similar, rather than learning the properties of particular action classes. We present baseline results on our benchmark, and compare them to human performance. To promote further study of action similarity techniques, we make the ASLAN database, benchmarks, and descriptor encodings publicly available to the research community.

## 1 Introduction

Recognizing human actions in videos is an important problem in Computer Vision with a wide range of applications, including video retrieval, surveillance, man-machine interaction, and more. With the availability of high bandwidth communication, large storage space and affordable hardware, digital video is now everywhere. Consequently, the demand for video processing, particularly effective action recognition techniques, is rapidly growing. Unsurprisingly, action recognition has recently been the focus of much research.

Human actions are complex entities taking place over time and over different body parts. Actions are either connected to a context (e.g., swimming) or context free (e.g., walking). What constitutes an "action" is often undefined, and so the number of actions being performed is typically uncertain. Actions can vary greatly in duration; some actions being instantaneous whereas others prolonged. They can involve interactions with other people, or static objects. Finally, they may include the whole body or be limited to one limb. Figure 1 provides examples, from our database, of these variabilities.

To facilitate the development of action recognition methods, many video sets, along with benchmark protocols, have been assembled in the past. These attempt to capture the many challenges of action recognition. Some examples include the KTH [1] and Weizmann [2] databases, and the more recent, Hollywood, Hollywood2 [3], [4], and YouTube-actions databases [5].

This growing number of benchmarks and data sets is reminiscent of the data sets used for image classification and face recognition. However, there is one important difference: image sets for classification and recognition now typically contain hundreds, if not thousands, of object classes or subject identities (see for example: [6], [7], [8]), whereas existing video data sets typically provide only around 10 classes (see Section 2).

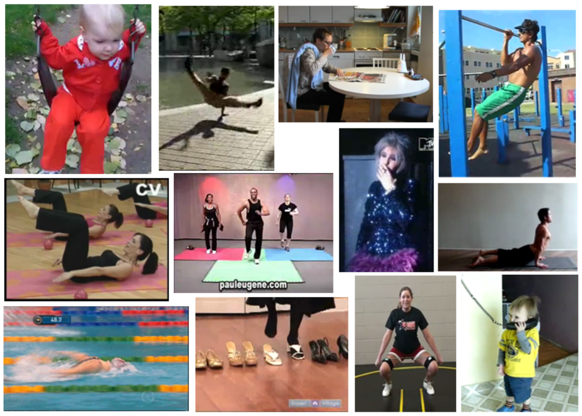

Fig. 1: Examples of the diversity of "real-world" actions

We believe one reason for this disparity between image and action classification is the following. Once many action classes are assembled, classification becomes ambiguous. Consider, for example, a high jump. Is it "running"? "jumping"? "falling"? Of course, it can be all three and possibly more. Consequently, labels assigned to such *complex* actions can be subjective and may vary from one person to the next. To avoid this problem, existing data sets for action classification offer only a small set of well-defined, *atomic* actions, which are either periodic (e.g., walking), or instantaneous (e.g. answering the phone).

In this paper we present a new action recognition data set, the "Action Similarity LAbeliNg" (ASLAN) collection. This set includes thousands of videos collected from the web, in over 400 complex action classes. [1].

To standardize testing with this data, we provide a "same/not-same" benchmark, which addresses the action recognition problem as a non class-specific similarity problem and which is different from more traditional multi-class recognition challenges. The rationale is that such a benchmark requires that methods learn to evaluate the similarity of actions rather than be able to

- O. Kliper-Gross is with the Department of Mathematic and Computer Science, The Weizmann Institute of Science, Israel.
E-mail: orit.kliper@weizamnn.ac.il
- T. Hassner is with The Computer Science Devision, The Open University, Israel,
- L. Wolf is with the Blavatnik School of Computer Science, Tel-Aviv University, Israel

[1] Our video collection, benchmarks, and related additional information is available at:
http://www.wisdom.weizmann.ac.il/~kliper/ASLAN7/ASLAN.html.

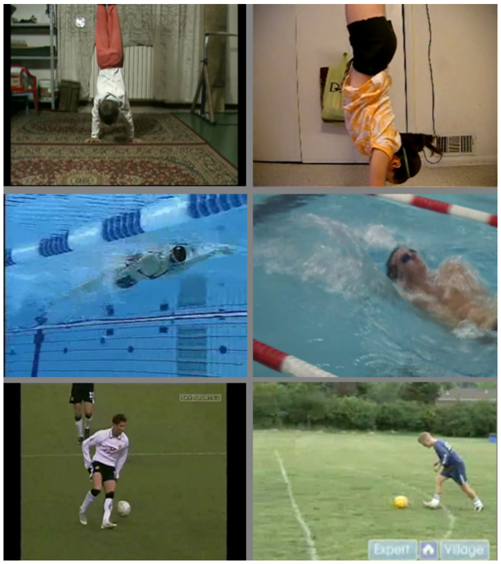

Fig. 2: Examples of "same" pairs from our database.

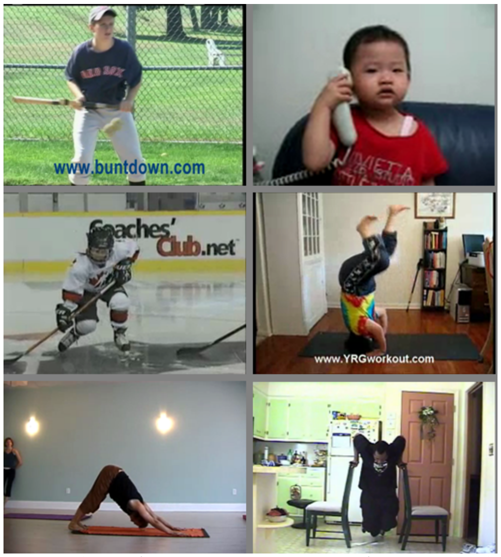

Fig. 3: Examples of "not-same" pairs from our database.

recognize particular actions.

Specifically, the goal is to answer the following binary question – "does a pair of videos present the same action, or not?". This problem is sometimes referred to as the "unseen pair matching problem" (see for example [8]). Figures 2 and 3 show some examples of "same" and "not-same"-labeled pairs from our database.

The power of the same/not same formulation is in diffusing a multi-class task into a manageable binary class problem. Specifically, this same/not-same approach has the following important advantages over multi-class action labeling: (a) It relaxes the problem of ambiguous action classes - it is certainly easier to label pairs as same/not-same rather than pick one class out of over a hundred, especially when working with videos. Class label ambiguities make this problem worst. (b) By removing from the test set all the actions provided for training, we focus on learning *action similarity*, rather than the distinguishing features of particular actions. Thus, the benchmark aims to gain a generalization ability which is not limited to a predefined set of actions. Finally, (c) besides providing insights towards better action classification, pair-matching has interesting applications in its own right. Specifically, given a video of an (unknown) action, one may wish to retrieve videos of a similar action, without learning a specific model of that action, and without relying on text attached to the video. Such applications are now standard features in image search engines (e.g., Google images).

To validate our data set and benchmarks we code the videos in our database using state-of-the-art action features, and present baseline results on our benchmark using these descriptors. We further present a human survey on our database. This demonstrates that our benchmark, although challenging to modern Computer Vision techniques, is well within human capabilities.

To summarize, we make the following contributions:

1) We make available a novel collection of videos and benchmark tests for developing action similarity techniques. This set is unique in the number of categories it provides (an order of magnitude more than existing collections), its associated pair-matching benchmark and the realistic, uncontrolled settings used to produce the videos.
2) We report performance scores obtained with a variety of leading action descriptors on our benchmark.
3) We have conducted an extensive human survey which demonstrates the gap between current state-of-the-art performance and human performance.

## 2 Existing data-sets and benchmarks

### 2.1 Data-sets

In the last decade, image and video databases have become standard tools for benchmarking the performance of methods developed for many Computer Vision tasks. Action recognition performance in particular has greatly improved due to the availability of such data sets. We present a list highlighting several popular data sets in Table 1. All these sets typically contain around ten action classes and vary in the number of videos available, the video source, and the video quality.

Early sets, such as KTH [1] and Weizmann [2], have been extensively used to report action recognition performance (e.g. [19], [20], [21], [22], [23], [24], [25], [26], to name a few). These sets contain few, "atomic" classes such as walking, jogging, running, and boxing. The videos in both these sets were acquired under controlled settings: static camera and un-cluttered, static background.

Over the last decade the recognition performance on these sets has saturated. Consequently, there is a growing need for new sets, reflecting general action recognition tasks with a wider range of actions. Attempts have been made to manipulate acquisition parameters in the laboratory. This was usually done for specific purposes, such as studying viewing variations [10], occlusions [27]

TABLE 1: **Popular Action Recognition Databases**

| Database | Classes | Videos | Setting | Data description |
|---|---|---|---|---|
| KTH [1] | 6 | 600 | Laboratory: 25 actors, 4 conditions, 4 repetitions = 2391 sub-sequences | Homogenous background, static camera, 25fps, 160x120px, 4s duration, AVI DVIX-compressed |
| WEIZMANN [2] | 9 | 81 | Laboratory: 9 different actors | Static background, resolution 180x144px, 25fps |
| UMD [9] | 10 | 100 | Laboratory: 1 actor, many repetitions | Resolution 300px |
| IXMAS [10] | 11 | 110 | Laboratory: 10 actors arbitrary orientation, 5 view points with multiple cameras, 30 sub-sequences | Resolution 100-200px, very short sequences |
| UIUC1 [11] | 14 | 532 | Laboratory: 8 actors, single view, extensive repetitions | Resolution 400px |
| UIUC2 [11] | 2-4-5 | 3 | YouTube videos: 3 different games badminton world cup 2006 | Resolution 80px |
| UCF-sports [12] | 9 | 200 | Real sports broadcasts | Unconstrained: wide range of scenes and viewpoints, simple background, resolution 720x480px |
| K/S [12] | 2 | 200 | Feature films | Large variability in genres, scenes and views, actors |
| Olympic-games [13] | 17 | 166 | Video footage from olympic games | 5065 manually annotated frames: high intra class variability, background clutter, large camera motion, motion blur, occlusions and appearance variations |
| Hollywood [3], [4] | 8/12 | 430/3669 | 32/69 movies | Large intra-class variability, label ambiguity, multiple persons, challenging camera motion, rapid scene changes, unconstrained and cluttered background, high quality, 240x450px, 24fps |
| UFC [14] | 2 | 20min | Broadcast videos | Varying views, appearance, camera motion, action frequencies, simultaneous actions by different actors |
| Youtube-actions [5] | 11 | 1168 | Youtube videos and personal videos | 25 sub-groups: different environments and photographers. Mix of steady and shaky cameras, cluttered background, variation in object scale, views points and illumination, low resolution (mpeg4-codec) |
| ADL [15] | 10 | 150 | Laboratory: 5 actors, 3 repetitions | Complex activities in living environment, static background, res. 1280x720px, 30fps, duration 10-60s |
| High-Five [16] | 4 | 300 | 23 different TV shows | 30-600 frames, realistic human interactions: varying number of actors, scale, and views |
| YouTube Olympic Sports [17] | 16 | 800 | YouTube vidoes | Complex activities, 50 sequences per class, labeled by Amazon Mechanical Turk |
| MSR II [18] | 3 | 54 | recorded in crowded environment | Multiple actions for the purpose of action detection, 203 action instances, 320x240px, 15fps. |

or recognizing daily actions in static scenes [15]. Although these databases have contributed much to specific aspects of action recognition, one may wish to develop algorithms for more realistic videos and more diverse actions. Real videos rarely exhibit consistent and controlled settings and it is further unclear how the full variety of human actions can be constructed under such conditions.

TV and motion pictures videos have been used as alternatives to controlled sets. The biggest such database to date was constructed by Laptev et al. [3]. Its authors, recognizing the lack of realistic annotated data sets for action recognition, proposed a method for automatic annotation of human actions in motion pictures based on script alignment and classification. They have thus constructed a large data set of $8$ action classes from $32$ movies. In a subsequent work [4], an extended set was presented, containing $3,669$ action samples of $12$ action and $10$ scene classes acquired from $69$ motion pictures. The videos included in it are of high quality and contain no unintended camera motion. In addition, the actions they include are non-periodic and well defined in time. These sets, although new, have already drawn a lot of attention (see for example [14], [28], [29], [30]).

Other data sets employing videos from such sources are the data set made available in [31], which includes actions extracted from 144 episodes of the TV series “Buffy the vampire Slayer”, the work of [12], which classifies actions in broadcast sports videos, and the recent work of [16], which explores human interactions in TV shows. All these sets offer only a limited number of well defined action categories.

While most action recognition research has focused on *atomic* actions, the recent work of [32] and [17] address complex activities, i.e. actions composed of few simpler or shorter actions. Ikizler and Forsyth [32] suggest learning complex activity models by joining atomic action models built separately across time and across the body. Their method has been tested on a controlled set of complex motions and on challenging data from the TV series Friends. Niebles and et al. [17] propose a general framework for modeling activities as temporal composition of motion segments. The authors have collected a new data set of $16$ complex Olympic Sports activities downloaded from YouTube.

Websites such as YouTube make huge amounts of video footage easily accessible. Videos available on these websites are produced under diverse, realistic conditions and have the advantage of having a huge variability of actions. This naturally brings to light new opportunities for constructing action recognition benchmarks. Such web data is increasingly being used for action recognition related problems. This includes [33], [34], performing automatic categorization of web videos, and [35], [36] which categorize events in web videos. These do not directly address action recognition but inspire further research in using web data for action recognition.

Most closely related to our ASLAN set, is the YouTube Action Data Set [5]. As far as we know, it is the first action recognition database containing videos "in the wild". This database, already used in a number of recent publications (for example, [30], [37], [38]), contains $1,168$ complex and challenging video sequences from YouTube and personal home-videos. Since the videos' source is mainly the web, there was no control over the filming and therefore the database contains large variations in camera motion, scale, view, background, illumination conditions, etc. In this sense, this database is similar to our own. However, unlike the ASLAN set, the YouTube set contains only $11$ action categories, which although exhibiting large intra-class variation, are still relatively well separated.

To summarize, existing databases contain far fewer action classes than the ASLAN set and the challenges they propose are different from the one described here.

### 2.2 Benchmarks

Most research on action recognition focuses either on multi-label action classification or on action detection. Existing methods for action similarity [22], [39], [40], [41] mainly focus on spatiotemporal action detection [39], [41], or classification [22], [40]. Action recognition has additionally been considered for never-before-seen views of a given action class (see, e.g., the work of [10], [22], [42]). None of these provide data or standard tests for the purpose of matching pairs of never-before-seen actions.

The benchmark proposed here attempts to address another shortcoming of existing benchmarks, namely, the lack of established, standard testing protocols. Different researchers use varying sizes of training and testing sets, different ways of averaging over experiments, etc. We hope that by providing a unified testing protocol we may provide an easy means of measuring and comparing performance of different methods.

Our work has been motivated by recent image sets, such as the Labeled Faces in the Wild (LFW) [8] for face recognition, and the extensive Scene Understanding (SUN) database [43] for scene recognition. In both cases, very large image collections were presented, answering a need for larger scope in complementary vision problems. The unseen pair-matching protocol presented in [8] motivated the one proposed here.

We note that same/not-same benchmarks such as the one described here have been successfully employed for different tasks in the past. Face recognition "in the wild" is one such example [8]. Others include historical document analysis [44], face recognition from YouTube videos [45], and object classification (e.g., [46]).

## 3 Goals of the proposed benchmark

### 3.1 The same/not-same challenge

In a same/not-same setting the goal is to decide if two videos present the same action or not, following training with same and not-same labeled video pairs. The actions in the test set are *not available during training*, but rather belong to separate classes. This means that there is no opportunity during training to learn models for actions presented for testing.

We favor a same/not-same benchmark over multi-label classification as its simple binary structure makes it far easier to design and evaluate tests. However, we note that typical action recognition applications label videos using one of several different labels rather than making similarity decisions. The relevance of a same/not-same benchmark to these tasks is therefore not obvious. Recent evidence obtained using the LFW benchmark suggests, however, that successful pair-matching methods may be applied for multi-label classification with equal success [47].

### 3.2 The testing paradigm

The setting of our testing protocol is similar to the one proposed by the LFW benchmark [8] for face recognition. The benchmarks for the ASLAN database are organized into two "Views". View-1 is for algorithm development and general experimentation, prior to formal evaluation. View-2 is for reporting performance and should only be used for the final evaluation of a method.

**View-1: Model selection and algorithm development**. This view of the data consists of two independent subsets of the database, one for training, and one for testing. The training set consists of $1,200$ video pairs: $600$ pairs with similar actions, and $600$ pairs of different actions. The test set consists of $600$ pairs: $300$ "same" and $300$ "not-same"-labeled pairs. The purpose of this view is for researchers to freely experiment with algorithms and parameter settings without worrying about over-fitting.

**View-2: Reporting performance**. The second view consists of $10$ subsets of the database, mutually exclusive in the actions they contain. Each of the subsets contains $600$ video pairs: $300$ same and $300$ not-same. Once the parameters for an algorithm have been selected the performance of that algorithm can be measured using View-2.

ASLAN performance should be reported by aggregating scores on $10$ separate experiments in a leave-one-out cross validation scheme. In each experiment, nine of the subsets are used for training, with the tenth used for testing. It is critical that the final parameters of the classifier under each experiment be set using only the training data for that experiment, resulting in $10$ separate classifiers (one for each test set).

TABLE 2: **ASLAN Database Statistics**

| **General Statistics:** | |
|---|---|
| ♯ classes | 432 |
| ♯ action samples | 3, 697 |
| ♯ unique samples | 3, 631 |
| ♯ unique urls | 1, 571 |
| ♯ unique titles | 1, 561 |
| **Class Statistics:** | |
| average ♯ of samples per class | 8.5 |
| ♯ classes with > 1 samples | 316 |
| Largest number of samples | "Handstand" 91 seq./64 urls |
| **Samples Statistics:** | |
| long samples (duration> 10sec) | 71 |
| short samples (duration< 1sec) | 187 |
| **Training/Testing:**[2] | |
| ♯ test pairs / ♯ training pairs | 600 / 5400 |
| **STIP Statistics** | |
| min ♯ STIP / max ♯ STIP | 3 / 26052 |
| Average ♯ STIP | 1549.9 |

For reporting final performance of the classifier, we use the same method as in [8] and ask that each experimenter report the **estimated mean accuracy** and the **standard error of the mean** (SE) for View-2 of the database. Namely, the **estimated mean accuracy** $\hat{\mu}$ is given by $\hat{\mu} = \frac{\sum_{i=1}^{10} P_i}{10}$ where $P_i$ is the percentage of correct classifications on View-2, using subset $i$ for testing. The **standard error of the mean** is given by,

$$S_E = \frac{\hat{\sigma}}{\sqrt{10}}, \quad \hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{10}(P_i - \hat{\mu})^2}{9}}$$

In our experiments (see Section 5) we also report the **Area Under the Curve** (AUC) of the ROC curve produced for classifiers used on the 10 test sets.

## 4 ASLAN Database

ASLAN was assembled in over 5 months of work, which included the downloading and the processing of around $10,000$ videos from YouTube. Construction was performed in two phases. In each phase we followed the following steps: (1) defining search terms, (2) collecting raw data, (3) extracting action samples, (4) labeling and, (5) manual validation. After the database was assembled we defined the two views by randomly selecting video pairs. We next describe the main construction details. For further details please refer to the project web page.

### 4.1 Main construction details

Our original search terms were based on the terms defined by the CMU Graphics Lab Motion Capture Database [3]. The CMU database is organized as a tree, where the final description of an action sequence is at the leaf. Our basic search terms were based on individual action terms from the CMU leaves. For some of the search terms we also added a context term (usually taken from a higher level in the CMU tree). For example: one search term could be *climb* and another could be *playground-climb*. This way, several query terms can retrieve the same action in different contexts.

In the first phase, we used a search list of 235 such terms, and automatically downloaded the top 20 YouTube video results for each term, resulting in $\sim 3,000$ videos. Action labels were defined by the search terms, and we validated these labels manually.

Following the validation, only $\sim 10\%$ of the downloaded videos contained at least one action, demonstrating the poor quality of keyword-based search as noted also in [33], [48]. We further dismissed cartoons, static images, and very low quality videos. The intra-class variability was extremely large and search terms only generally described the actions included in each category. We were consequently required to use more subtle action definitions, and a more careful labeling process.

In the second phase, $174$ new search terms were defined based on first phase videos. $50$ videos were downloaded for each new term, totaling $\sim 6,400$ videos. YouTube videos often present more than one action. Since ASLAN is designed for action similarity, not detection, we manually cropped the videos into action samples. **An action sample** is defined as a sub-sequence of a shot presenting a detected action, that is, a consecutive set of frames taken by the same camera presenting one action. The action samples were then manually labeled according to their content; a new category was defined for each new action encountered. We allowed each action sample to fall into several categories whenever the action could be described in more than one way.

### 4.2 Database statistics

The final database contains $3,631$ unique action samples from $1,571$ unique urls, and $1,561$ unique titles, in $432$ action classes. Table 2 provides some statistical information on our database. Additional information may be found on our website.

All the action samples are encoded using mp4 (codec h264) high resolution format (highest available for download), as well as AVIs (xvid codec). The database contains videos of different resolution, frame size, aspect ratio, and frame rate. Most videos are in color, but some are gray-scale.

Before detailing the views' construction we note the following. Action recognition is often used for video analysis and/or scene understanding. The term itself sometimes refers to action detection, which may involve selecting a bounding box around the actor, or marking the time an action is performed. Here we avoid detection by constructing our database from short video samples that could in principle be the output of an action detector. In particular, since every action sample in our database is manually extracted, there is no need

[2] Numbers relate to View-2, for each of the 10 experiments.

[3] http://mocap.cs.cmu.edu/.

to temporally localize the action. We thus separate action detection from action similarity and minimize the ambiguity that may arise by determining action durations.

### 4.3 Building the views

To produce the views for our database, we begin by defining a list of valid pairs. Valid pairs are any two distinct samples which were not originally cut from the same video; pairs of samples originating from the same video were ignored. The idea was to avoid biases for certain video context/background in same-labeled pairs, and to reduce confusion due to similar background for not-same pairs.

View-1 test pairs were chosen out of the valid pairs in $40$ randomly selected categories. The pairs in the training set of View-1 were chosen out of the valid pairs in the remaining categories.

To define View-2, we randomly split the categories into $10$ subsets, ensuring that each has at least $300$ valid same pairs. To balance each subset's categories, we allow only up to $30$ same pairs from each label. Once the categories of the subsets were defined, we randomly selected $300$ same and $300$ not-same pairs from each subset's valid pairs.

## 5 Baseline Performance

To demonstrate the challenge of the ASLAN data and benchmark, we report performance obtained with existing leading methods on View-2 of the database. To this end, we encoded ASLAN video samples using leading video descriptors [4]. We then used linear Support Vector Machine (SVM) [49] to classify pairs of same/not same actions, using combinations of (dis-)similarities and descriptors as input.

To validate these tests we further report the following results: (a) Human performance on our benchmark, demonstrating the feasibility of the proposed pair-matching task on our videos. (b) Results obtained using the same descriptors on KTH videos with a similar pair-matching protocol, illustrating the challenge posed when using videos collected in unrestricted conditions compared to laboratory produced videos.

### 5.1 State-of-the-art video descriptors

We have followed [3] and used the code supplied by the authors. The code detects Space-Time Interest Points (STIPs) and computes three types of local space-time descriptors: Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF), and a composition of these two referred to as HNF. As in [3] we used their version of the code without scale selection, using instead a set of multiple combinations of spatial and temporal scales.

The currently implemented variants of descriptors HOG, HOF, and HNF are computed on a 3D video patch in the neighborhood of each detected STIP. Each patch is partitioned into a grid with $3 \times 3 \times 2$ spatiotemporal blocks. $4$-bin HOG descriptors, $5$-bin HOF descriptors and $8$-bin HNF descriptors, are computed for each block. The blocks are then concatenated into a $72$-element, $90$-element descriptors and $144$-element descriptors, respectively.

We followed [3] in representing videos using a spatiotemporal bag-of-features (BoF). This requires assembling a visual vocabulary for each of our $10$ experiments. For each experiment, we used k-means ($k = 5,000$) to cluster a subset of 100k features randomly sampled from the training set. We then assigned each feature to the closest vocabulary word (using Euclidean distance) and computed the histogram of visual word occurrences over the space-time volume of the entire action sample.

We ran this procedure to create the $3$ types of global video descriptors of each video in our benchmark. We used the default parameters, i.e. $3$ levels in the spatial frame pyramid and initial level of $0$. However, when the code failed to find interest points, we found that changing the initial level improved the detection.

### 5.2 Experimental results

We performed $10$-fold cross validation tests as described in Section 3.2. In each, we have calculated $12$ distances/similarities between global descriptors of the benchmark pairs. For each of these (dis-)similarities taken separately, we found an optimal threshold on the same/not-same labeled training pairs using linear SVM classifier. Then we have used this threshold to label the test pairs. Table 3 reports the results on the test pairs (averaged over the $10$-folds).

In order to combine various features together, we have used the stacking technique [50]. In particular, we have concatenated the values of the $12$ (dis-)similarity into vectors, each such vector representing a pair of action samples from the training. These vectors, along with associated same/not-same labels, were used to train a linear SVM classifier. This is similar to what was done in [47]. Prediction accuracies based on these values are presented in last row of Table 3. In the last column, we further show the results produced by concatenating the (dis-)similarity values of all three descriptors, and use these vectors to train a linear SVM classifier.

The best results of $60.88 \pm .77$ accuracy and $65.30\%$ AUC were achieved using a combination of the $3$ descriptor types, and the $12$ (dis-)similarities i.e. vectors of length $36$ (see Figure 4).

### 5.3 Human survey on ASLAN

To validate our database we have conducted a human survey on a randomly selected subset of ASLAN [5].

[4] Descriptor encodings are made available for the research community.

[5] Our survey form can be accessed at the following URL: http://www.wisdom.weizmann.ac.il/~kliper/ActionsSim/Actions.htm

TABLE 3: ASLAN Performance: Accuracy±SE and (AUC), mean over the 10-folds. Best local results in blue, best overall results in red. In the last 4 rows original vectors were normalized before calculating (dis-)similarities.

| **(Dis-)Similarity** | **HOG** | **HOF** | **HNF** | **ALL Descriptors** |
|---|---|---|---|---|
| $\sum(x_1.*x_2)$ | 56.58 ± .74 **(61.61)** | 52.25 ± .35 (58.43) | 56.45 ± .61 (62.09) | 58.60 ± .82 (64.33) |
| $\sqrt{\sum(x_1.*x_2)}$ | **58.55 ± .80** (61.59) | **56.82 ± .57 (58.56)** | **58.87 ± .89 (62.16)** | **60.08 ± 1.08 (63.89)** |
| $\sqrt{\sum(\sqrt{x_1}.*\sqrt{x_2})}$ | 52.82 ± .81 (54.17) | 54.25 ± .86 (55.99) | 53.38 ± .73 (55.78) | 54.82 ± .78 (57.13) |
| $\sqrt{\sum\frac{x_1.*x_2}{x_1+x_2}}$ | 52.93 ± .80 (54.40) | 54.13 ± .79 (55.76) | 53.50 ± .71 (55.95) | 54.80 ± .91 (57.09) |
| $\sum(x_1.*x_2)/(\sqrt{\sum(x_1^2)}*\sqrt{\sum(x_2^2)})$ | 54.27 ± .66 (55.77) | 54.12 ± .74 (56.54) | 54.50 ± .63(57.63) | 55.72 ± .83(58.93) |
| $\sqrt{\sum\frac{(x_1-x_2)^2}{x_1+x_2}}$ | 53.28 ± .69 (54.42) | 53.42 ± .62 (55.79) | 53.87 ± .72 (55.97) | 54.97 ± .97 (57.13) |
| $\sum\lvert x_1-x_2\rvert$ | 53.23 ± .76 (54.41) | 53.53 ± .73 (55.59) | 53.75 ± .71 (55.90) | 54.80 ± .95 (57.01) |
| $\sqrt{\sum(\sqrt{x_1}-\sqrt{x_2})^2}$ | 53.22 ± .61 (54.19) | 53.77 ± .72 (56.00) | 53.77 ± .73 (55.80) | 54.83 ± .90 (57.18) |
| $\sqrt{\sum(x_1-x_2)^2}$ | 54.20 ± .68 (55.78) | 53.95 ± .66 (56.56) | 54.53 ± .71(57.63) | 55.43 ± .87(58.90) |
| $(\sum x_1\log\frac{x_1}{x_2}+\sum x_2\log\frac{x_2}{x_1})/2$ | 54.80 ± .74 (57.41) | 54.85 ± .74 (56.41) | 55.48 ± .52 (57.80) | 55.20 ± .56 (57.96) |
| $\sum\frac{\sqrt{\max(x_1,x_2)}}{\sqrt{(x_1+x_2)}}$ | 51.28 ± .61 (53.50) | 51.60 ± .72 (54.02) | 51.43 ± .65 (53.48) | 51.50 ± .73 (53.61) |
| $\sum\min(x_1,x_2)/(\sum x_1\sum x_2)$ | 55.18 ± .81 (57.58) | 54.63 ± .48 (57.45) | 56.08 ± .71 (59.28) | 57.23 ± .61 (60.43) |
| All (Dis-)Similarities | 59.78 ± .82 (63.20) | 56.68 ± .56 (58.97) | 59.47 ± .66 (63.30) | **60.88 ± .77 (65.30)** |

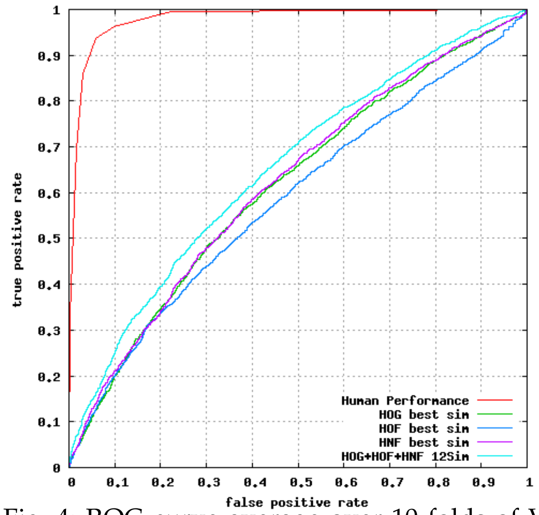

Fig. 4: ROC curve average over 10 folds of View-2.

The survey results were used for the following purposes: (a) Test the difficulty posed by our selections to human operators. (b) Verify whether the resolution of our action labels is reasonable. That is, if our definition of different actions is indeed perceived as such by people not part of the original collection process, and (c) The survey also provides a convenient means of comparing human performance to that of the existing state-of-the-art. Specifically, it allows us to determine which categories are inherently harder to distinguish than others.

The human survey was conducted on 600 pairs in 40 randomly selected categories. Each user viewed 10 randomly selected pairs, and asked to rate his or her confidence that each of these pairs represents the same action on a $1-7$ Likert scale. We have so far collected 1,890 answers from 189 users on the 600 pairs, an average of 3 users per pair of videos.

User votes for each pair are treated as independent experts and their median answer is the selected human score. The top curve in Figure 4 shows the performance of humans. The AUC computed for our survey is 97.86%. Note that the results are not perfect, suggesting either that the task is not totally trivial even for humans, or else that some videos may be mislabeled.

These results show that although challenging, the ASLAN benchmark is well within human capabilities. Figure 4 thus highlights the significant performance gap between humans and the baseline on this benchmark data set. Doing so, it strongly motivates further research into action similarity methods, with the goal of closing this performance gap.

### 5.4 The same/not-same setting on KTH

To verify the validity of our settings and the ability of the given descriptors to infer same/not-same decisions on never-before-seen data, we have defined a same/not-same protocol using the videos included in the KTH set [1]. We randomly chose 3 mutually-exclusive subsets on the 6 actions of the KTH set, and performed 3-fold cross-validation tests using the same (dis-)similarities for the classifier as in the ASLAN experiments. Best performing (dis-)similarities are presented in Table 4.

The performance on the KTH data reached 90% accuracy and 97% AUC, even using a single descriptor score. Clearly, methods applied to ASLAN perform far better when applied to videos from the KTH data set. The lower performance on ASLAN may indicate that there is a need for further research into action descriptors for such "in the wild" data.

## 6 Summary

We have introduced a new database and benchmarks for developing action similarity techniques: The Action Similarity LAbeliNg (ASLAN) collection. The main contributions of the proposed challenge are: First, it provides researchers with a large, challenging database with hundreds of complex action categories. The videos are

TABLE 4: Selected classification performance on the KTH data set: Accuracy±SE and (AUC), averaged over the 3-folds. Locally best results are marked in blue. Overall best results are marked in red.

| (Dis-)Similarity | HOG | HOF | HNF | ALL Descriptors |
|---|---|---|---|---|
| $\sqrt{\sum(\sqrt{x_1}-\sqrt{x_2})^2}$ | 82.61 ± 1.54 (**89.77**) | 85.33 ± .83 (90.25) | **90.00 ± .84** (96.35) | 88.39 ± .24 (97.19) |
| $\sqrt{\sum(x_1.*x_2)}$ | 80.39 ± 1.45 (89.55) | 81.44 ± .83 (89.94) | 88.83 ± 1.92 (96.34) | 82.50 ± 1.20 (96.59) |
| $\sqrt{\sum(\sqrt{x_1}.*\sqrt{x_2})}$ | 72.00 ± .4.40 (89.66) | 77.61 ± .58 (90.18) | 82.06 ± 3.36 (**96.75**) | 79.00 ± 2.83 (**97.76**) |
| $\sqrt{\sum\frac{(x_1-x_2)^2}{x_1+x_2}}$ | **82.78 ± 1.85** (89.26) | **85.44 ± .95** (90.13) | 89.00 ± 1.42 (96.30) | **88.67 ± .35** (97.06) |
| $(\sum x_1\log\frac{x_1}{x_2}+\sum x_2\log\frac{x_2}{x_1})/2$ | 78.44 ± 1.88 (87.27) | 83.28 ± 1.62 (**90.29**) | 87.94 ± 1.47 (94.13) | 83.94 ± 2.27 (96.52) |
| $\sum\|x_1-x_2\|$ | 82.28 ± 2.07 (89.11) | 84.89 ± .56 (90.06) | 89.67 ± 1.26 (96.32) | 88.56 ± .15 (97.11) |

produced under challenging, uncontrolled conditions, raising the bar for action recognition techniques.

Second, our benchmarks focus on action *similarity* (same/not-same), rather than action classification. This binary classification task makes it easier to design and evaluate performance tests, particularly when dealing with so many classes, where action labels are often ambiguous. The proposed benchmark tests the accuracy of similarity classification based on training with actions not included in the test. The purpose of this is to gain a more principled understanding of what makes actions different or similar, rather than learn the properties of particular actions. Finally, the benchmarks described in this paper provide a unified testing protocol and an easy means for reproducing and comparing different action similarity methods.

We tested the validity of our database by evaluating human performance, as well as reporting baseline performance achieved by using state-of-the-art descriptors. We show that while humans achieve very high results on our database, state-of-the-art methods are still far behind with only around 65% success. We believe this gap in performance strongly motivates further study of action similarity techniques. We propose this challenging database and benchmarks for the research community in order to stimulate further research of this theme.

## REFERENCES

[1] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing human actions: A local svm approach,” in *Proc. 17th Int. Conf. Pattern Recognition*, vol. 3, 2004, pp. 32–36.

[2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, “Actions as space-time shapes,” in *Proc. Int. Conf. Comput. Vision*, 2005, pp. 1395–1402.

[3] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic human actions from movies,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2008, pp. 1–8.

[4] M. Marszalek, I. Laptev, and C. Schmid, “Actions in context,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2009, pp. 2929–2936.

[5] J. Liu, J. Luo, and M. Shah, “Recognizing realistic actions from videos ’in the wild’,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2009, pp. 1996–2003.

[6] A. Torralba, R. Fergus, and W. T. Freeman, “million tiny images: A large data set for nonparametric object and scene recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1958–1970, 2008.

[7] G. Griffin, A. Holub, and P. Perona, “Caltech-256 object category dataset,” California Institute of Technology, Tech. Rep. 7694, Tech. Rep., 2007.

[8] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” University of Massachusetts, Amherst, Tech. Rep. 07-49, Tech. Rep., 2007.

[9] A. Veeraraghavan, R. Chellappa, and A. K. Roy-Chowdhury, “The function space of an activity,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, vol. 1, 2006, pp. 959–968.

[10] D. Weinland, R. Ronfard, and E. Boyer, “Free viewpoint action recognition using motion history volumes,” *Comput. Vision Image Understanding*, vol. 104, no. 2-3, pp. 249–257, 2006.

[11] D. Tran and A. Sorokin, “Human activity recognition with metric learning,” in *Proc. 10th European Conf. Comput. Vision*, 2008, pp. 548–561.

[12] M. D. Rodriguez, J. Ahmed, and M. Shah, “Action mach: A spatio-temporal maximum average correlation height filter for action recognition,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2008, pp. 1–8.

[13] K. Mikolajczyk and H. Uemura, “Action recognition with motion-appearance vocabulary forest,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2008, pp. 1–8.

[14] L. Yeffet and L. Wolf, “Local trinary patterns for human action recognition,” in *IEEE 12th Int. Conf. Comput. Vision*, 2009, pp. 492–497.

[15] R. Messing, C. Pal, and H. Kautz, “Activity recognition using the velocity histories of tracked keypoints,” in *IEEE 12th Int. Conf. Comput. Vision*, 2009, pp. 104–111.

[16] A. Patron-Perez, M. Marszalek, A. Zisserman, and I. Reid, “High five: Recognising human interactions in tv shows,” in *Proc. British Mach. Vision Conf.*, 2010.

[17] J. C. Nieblesand, C.-W. Chen, and L. Fei-Fei, “Modeling temporal structure of decomposable motion segments for activity classification,” in *Proc. 11th European Conf. Comput. Vision*, 2010, pp. 392–405.

[18] G. Yo, J. Yuan, and Z. Liu, “Unsupervised random forest indexing for fast action search,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2011, pp. 865–872.

[19] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, “Behavior recognition via sparse spatio-temporal features,” in *VS-PETS*, 2005, pp. 65–72.

[20] P. Scovanner, S. Ali, and M. Shah, “A 3-dimensional sift descriptor and its application to action recognition,” in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 357–360.

[21] J. C. Niebles and L. Fei-Fei, “A hierarchical model of shape and appearance for human action classification,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2007, pp. 1–8.

[22] I. Junejo, E. Dexter, I. Laptev, and P. Pérez, “Cross-view action recognition from temporal self-similarities,” in *Proc. 10th European Conf. Comput. Vision*, 2008, pp. 293–306.

[23] K. Schindler and L. V. Gool, “Action snippets: How many frames does human action recognition require?” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2008, pp. 1–8.

[24] A. Kovashka and K. Grauman, “Learning a hierarchy of discriminative space-time neighborhood features for human action recognition,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2010, pp. 2046–2053.

[25] M. Raptis and S. Soatto, “Tracklet descriptors for action modeling and video analysis,” in *Proc. 11th European Conf. Comput. vision*, 2010, pp. 577–590.

[26] W. Kim, J. Lee, M. Kim, D. Oh, and C. Kim.

[27] D. Weinland, M. Ozuysal, and P. Fua, “Making action recognition robust to occlusions and viewpoint changes,” in *Proc. 11th European Conf. Comput. Vision*, 2010, pp. 635–648.

[28] X. Wu, C. W. Ngo, J. Li, and Y. Zhang, "Localizing volumetric motion for action recognition in realistic videos," in *Proc. 17th Int. Conf. Multimedia*, 2009, pp. 505–508.

[29] A. Gilbert, J. Illingworth, and R. Bowden, "Action recognition using mined hierarchical compound features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 883–897, 2011.

[30] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2011, pp. 3169–3176.

[31] A. Gaidon, M. Marszalek, and C. Schmid, "Mining visual actions from movies," in *Proc. British Mach. Vision Conf.*, 2009, p. 128.

[32] N. Ikizler and D. A. Forsyth, "Searching for complex human activities with no visual examples," *Int. J. Comput. Vision*, vol. 80, no. 3, pp. 337–357, 2008.

[33] S. Zanetti, L. Zelnik-Manor, and P. Perona, "A walk through the webs video clips," in *CVPRW*, 2008, pp. 1–8.

[34] Z. Wang, M. Zhao, Y. Song, S. Kumar, and B. Li, "Youtubecat: Learning to categorize wild web videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2010.

[35] L. Duan, D. Xu, I. W. Tsang, and J. Luo, "Visual event recognition in videos by learning from web data," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2010.

[36] T. S. Chua, S. Tang, R. Trichet, H. K. Tan, and Y. Song, "Moviebase: a movie database for event detection and behavioral analysis," in *Proc. 1st workshop Web-scale multimedia corpus*, 2009, pp. 41–48.

[37] N. Ikizler-Cinbis and S. Sclaroff, "Object, scene and actions: Combining multiple features for human action recognition," in *Proc. 11th European Conf. Comput. vision*, 2010, pp. 494–507.

[38] P. Matikainen, M. Hebert, and R. Sukthankar, "Representing pairwise spatial and temporal relations for action recognition," in *Proc. 11th European Conf. Comput. vision*, 2010, pp. 508–521.

[39] E. Shechtman and M. Irani, "Space-time behavior based correlation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2005, pp. 405–412.

[40] L. Zelnik-Manor and M. Irani, "Statistical analysis of dynamic actions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1530–1535, 2006.

[41] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2007, pp. 1–8.

[42] A. Farhadi and M. Tabrizi, "Learning to recognize activities from the wrong view point," in *Proc. 10th European Conf. Comput. Vision*, 2008, pp. 154–166.

[43] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2010, pp. 3485–3492.

[44] L. Wolf, R. Littman, N. Mayer, T. German, N. Dershowitz, R. Shweka, and Y. Choueka, "Identifying join candidates in the cairo genizah," *Int. J. Comput. Vision*, 2011.

[45] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, 2011.

[46] A. Ferencz, E. Learned-Miller, and J. Malik, "Building a classification cascade for visual identification from one example," in *Proc. 10th IEEE Int. Conf. Comput. Vision*, vol. 1, 2005, pp. 286–293.

[47] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Faces in Real-Life Images Workshop in European Conf. Comput. Vision*, 2008.

[48] M. Sargin, H. Aradhye, P. Moreno, and M. Zhao, "Audiovisual celebrity recognition in unconstrained web videos," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2009, pp. 1977–1980.

[49] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011, available: http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[50] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.