Win-Fail Action Recognition

Paritosh Parmar
University of British Columbia

Brendan Morris
University of Nevada, Las Vegas

Abstract

Current video/action understanding systems have demonstrated impressive performance on large recognition tasks. However, they might be limiting themselves to learning to recognize spatiotemporal patterns, rather than attempting to thoroughly understand the actions. To spur progress in the direction of a more comprehensive understanding of videos, we introduce the task of win-fail action recognition — differentiating between successful and failed attempts at various activities. We introduce a first of its kind paired win-fail action understanding dataset with samples from the following domains: "General Stunts," "Internet Wins-Fails," "Trick Shots," & "Party Games." Unlike existing action recognition datasets, intra-class variation is high making the task challenging, yet feasible. Using a battery of experiments, including a novel video retrieval test, we systematically analyze the characteristics of our win-fail task/dataset, and determine its suitability to serve as a video understanding problem benchmark. While current prototypical action recognition methods work well on our task/dataset, they still leave a large gap to achieve high performance. We hope to motivate more work towards the true understanding of actions/videos. Dataset will be available from: https://github.com/ParitoshParmar/Win-Fail-Action-Recognition.

1. Introduction

Action recognition, which can be defined as the task of identifying various action classes in videos, has thus far been used as a representative task for video understanding. Video action recognition involves the processing of spatiotemporal data, and extracting low-dimensional spatiotemporal signatures from video volumes. Based on these signatures, probabilities of action classes are determined.

We make two observations regarding the task of action recognition. Firstly, while current action recognition datasets, like UCF101, HMDB51, Kinetics, Sports1M, *etc*., have focused on increasing their dataset sizes and covering a larger number of classes, samples in those datasets

Figure 1: **Illustration of intra-class variance** (along columns, not rows) in a typical action recognition dataset vs. in our action understanding dataset. (Left) Samples from two randomly chosen classes, Basketball (BB) and TennisSwing (TS), from a typical action recognition dataset. (Right) Samples from ours newly compiled dataset, which has two action classes: *wins* and *fails*. As can be seen, the typical action recognition dataset does not have much intra-class variance, because of which action recognition is reduced to a pattern recognition problem. Please view in an Adobe Reader to play videos.

exhibit low intra-action-class variance in their spatiotemporal signatures. For example, all of the samples from action class BasketBall contain identical spatiotemporal signatures, like people holding a basketball with their hands, and trying to throw it into the basket (refer to Fig. 1); or as another example, all the samples from action class TennisSwing contain people holding a tennis racquet in their hand, and moving their arm. As a result, action recogni-

tion has, so far, been limited to cases where spatiotemporal signatures within individual action classes remain identical. Secondly, action sequences are not complex, although this trend is starting to increase with the introduction of datasets like Something-Something. However, overall, and as a consequence of the first shortcoming, the datasets do not require video understanding models to reason: *e.g.*, trying to draw logical inferences about the actors' goals by piecing together contextual (including human-object interactions) and human-movement cues as the video progresses, to determine whether the actor/s were able to accomplish what they set out to do.

This raises a question as to, if the current video/action understanding systems make an attempt to really understand the action, or if they limit themselves to identifying spatiotemporal patterns. We believe that current video understanding boils down to a pattern recognition problem. We are not conveying that action recognition is not needed or is unimportant, rather we view action recognition as a very important initial task. However, solely focusing on developing in the direction of action recognition might be limiting in nature. Therefore, in order to encourage action understanding systems to gain a holistic understanding of human actions, we slightly redefine the task of action recognition as it currently stands. Instead of differentiating among action classes, we propose to repurpose the task of (action) recognition to differentiating between the concepts of winning and failing. Winning can be defined as completing a task that the human set out to do, while failing can be defined when human is not able to complete the task. For example, successfully flipping a cup, putting a basketball in the basket, or being able to walk on one's hands, *etc*. are considered as wins, while not successfully flipping a cup, throwing a basketball that does not go into the basket, or trying to walk on one's hands but instead falling over, are considered as fails. As one can imagine, & see in Fig. 1, that the spatiotemporal signatures within the samples are very different, yet represent the same concepts. A naive way to increase intra-class variance would be to consolidate multiple action classes from current datasets under one super-class. For example, treat classes 1-50 from UCF101 [41] as super-class-1, and action classes 51-101 as super-class-2. However, in that way, the super-classes would not be connected to each other or hold meaning at a semantic level, and thus, this approach would not make much sense.

Humans, even as young as a year old, are able to infer and/or reason about the goals of others' actions from experience, contextual cues, kinematic cues, *etc*. [2,11,40,47,48]. They are able to perceive the difficulty of a task/action [10, 13, 16], and tend to put more value on actions that are more difficult or require perceivably more effort [21]. In fact, in Olympic events like diving and gymnastic vaulting, the scores are directly proportional to the degree of difficulty of the actions. It has been shown that the degree of difficulty can be measured from videos [26, 37]. Therefore, when observing competitive scenarios (where participants are trying to gain the maximum score), even when the game rules, or a task descriptions are not explicitly intimated to humans, they may still be able to figure them out by reasoning about what action sequence might be more difficult to execute, and consequently, decide if an action instance was a win or fail. We aim to give our models this kind of comprehensive understanding through learning general notions of win and fail actions in videos in simplified setting.

Our contributions can be summarized as:

- We first bring to light the problem with current video understanding works that they use only surface-level action recognition task as the benchmark task. To that end, we introduce a novel task of win-fail recognition, which requires models to understand videos comprehensively, linking together various contextual and human-movement cues from start to finish in a sequential manner.
- To facilitate our task, we introduce a new, carefully curated dataset, which contains *paired* samples of wins and fails from various domains.
- We found that using standalone recognition approach (typically used in action recognition) had very poor performance, only slightly better than random prediction. To that end, we proposed a pair-wise recognition approach which performed significantly better than standalone approach.

Applications: Our task/dataset can potentially be useful in scenarios like: 1) *Safety monitoring* for Children, Elderly, Factories/Workplace; 2) *anomaly detection*; 3) *scene understanding*; 4) *Video recommendation*; 5) *Video retrieval*.

2. Related Work

Action recognition datasets: Datasets can be divided into the following categories: 1) short-term temporal dynamics (UCF101 [41], HMDB51 [22], Kinetics [3]), where actions can be classified from a single or very few frames, or even the background; 2) long-term temporal dynamics (Something-Something [15], Diving48 [27], MTLAQA [35], Epic-Kitchens [5], Jester [31], *etc*.); 3) coarse-grained (UCF101, Kinetics, *etc*.); and 4) fine-grained (Diving48, MTLAQA, *etc*.). Unlike in coarse-grained action classification, actions in fine-grained classification category have very subtle differences between signature action patterns.

Regardless of how we group current action datasets, the task ultimately remains same – to learn the spatiotemporal signatures pertaining to each action class. Note that, longer temporal dynamics (for example, counting somersaults in [27, 35] or differentiating between pulling and

pushing in [15]) do not necessarily require comprehensive understanding of videos/actions. From earlier action recognition days to the present, the focus, thus far, has been to increase the dataset size and increase the number of action classes. We believe this limits the models from gaining holistic understanding of what is happening in front of a camera. Therefore, instead, we focus on increasing the intra-class variance.

Our work can also be considered closer in spirit to [14], which builds a video dataset with fully observable and controllable object and scene bias, and which truly requires spatiotemporal understanding in order to be solved. Our task and dataset are different in several ways, such as: ours is real world dataset; have humans performing actions/avtivities; objects in our dataset have purposes/meaning, *etc*.

Action recognition models: Unlike in image recognition, where a decision is made based on a single image, in a video understanding task, modeling temporal relationships is crucial. Some of the earlier deep learning based action recognition works, which considered multiple frames to do recognition include works like [7, 19, 52].

TSN [45] proposes a very simple, yet effective approach of sampling a few frames from the entire video and processing these frames individually to extract frame-wise features. Frame-level features are then combined using an aggregation scheme to get video-level representation. The authors found averaging to work the best. Averaging is actually temporal order agnostic, which indicates that action recognition tasks on datasets like UCF101, HMDB51, and Kinetics do not really demand temporal modeling.

The introduction of datasets like Something-Something, in which temporal order of frames matter (*e.g*. recognizing pushing vs. pulling something), motivated works like TRN [55], which proposed a temporal reasoning module. While TRN worked by modeling/discovering temporal relations from extracted features, TSM [29] aimed extracting temporal relations in the CNN backbone at a lower computational cost. Some works like [12, 49] propose approaches to combine short-term and long-term features.

Other works propose approaches that improve focus on the human actor, either by jointly estimating the pose of the actor [30, 54], or by modeling the relationship between the actor and objects [42].

We employ developments in designing our models, and then compare them to see what works and what does not work on our dataset.

AQA/skills assessment: AQA [24, 25, 32–35, 37, 39, 43, 44, 46, 50, 51, 53] is another action analysis task, which involves quantifying *how well* an action was performed. Similar in nature is the task of skills assessment [8, 9, 28, 36]. Like action recognition, the task still comes to learning and/or recognizing spatiotemporal patterns, although it is more fine-grained than action recognition. In addition to recognizing patterns, AQA/SA also involves valuation of those patterns.

Action domain	No. of pairs	Avg. len. (# fr.; s)
General stunts	122	96; 3.83
Internet wins-fails	258	112; 4.46
Trickshots	135	66; 2.62
Party games	302	62; 2.48
Overall	817	84; 3.33

Table 1: **Dataset details.**

In AQA, examples of these patterns include keeping legs straight in pike position, stable landing, tight form in tuck position, *etc*.; in SA, examples of these patterns include, not stretching tissues too much, handling them carefully, *etc*. We do note that there is an association between AQA/SA and win-fail recognition, in that higher skills-levels are generally associated with wins, and lower skills-levels are associated with fails. However, our work is different from these works, in that while these works propose action-specific approaches, our core idea is to increase intraclass variance among samples – our dataset contains four different domains – in order to encourage models to understand actions thoroughly, beyond surface-level pattern recognition; action sequences in our dataset are much more complex. Parmar *et al*. [33] have proposed learning a single AQA model across multiple actions, resulting in more intra-class variance, but the goal of their work was to learn shared action quality elements more efficiently, while our goal in considering multiple domains is to gain an actual understanding of the actions.

Visual concept learning: We find that works by Binder *et al*. [1], Zhou *et al*. [56], and Chesneau *et al*. [4] are closest to ours. While [1, 56] focus on recognizing more complex visual concepts, beyond objects in image domain, we introduce win-fail recognition in the video domain for comprehensive human action understanding. Chesneau *et al*. [4] address recognizing concepts like *'Birthday Party,' 'Grooming an Animal,'* and *'Unstuck a Vehicle'* in web videos. However, these concepts do not have large intra-class variance like ours, and are less complex and challenging than ours.

3. Win-Fail Action Recognition Dataset

To address the previously mentioned limitations of current action recognition datasets and to facilitate our new task of win-fail action recognition task, we introduce a novel Win-Fail action recognition dataset. The Win-Fail

dataset has the following characteristics: 1) a large variance in the structure of the task (both in action and context) and in the semantic definitions of wins and fails across samples; and 2) action sequences that are complex but at the same time win/fail recognition task is feasible. It is possible to identify winning/failing through reasoning on human movements and context (including actor-object interactions, *etc.*), without requiring external knowledge. For example, we do not include games of chess in our dataset since it requires knowledge of game mechanics. Since identifying wins and fails in standalone fashion may be overly difficult, we collect paired win and fail samples: *i.e*. for every win action instance, we have provided a fail version of that action instance. We collected data samples from following domains. Examples from all the domain are shown in **Supplementary Material**[1] in Fig. 3.

1. **General Stunts (GS):** Actions from this domain resemble stunts similar to those seen in movies or arbitrarily choreographed stunts. To collect data samples from this domain, we made use of paired compilations released by the stunt artists themselves. In these paired compilations, they include and specifically indicate their failed and successful attempts at various stunt routines. Failures can be attributed to factors like: miscalculation in placement of limbs, imbalance, erroneous landing, not able to securely grip handles, etc. In the samples from this domain, people can be seen working/interacting with large objects like truck tires, foam plyo boxes, chutes, ladders, etc. Action sequences are mainly comprised of a single actor.
2. **Internet Wins-Fails (IWF):** This is a popular category of videos on YouTube, where people attempt to do all sorts of things like walking on their hands, pole dancing, cycling at high speed through forests, skateboarding, hulahooping, etc. We collected pairs of wins and fails of people trying to these things. Note that these types of compilations many times include cases where mishaps happen because of some other person's mistake or some objects' failure (breaking off, falling, etc.). We did not include those kinds of samples; we only include samples where the wins and fails are outcomes of the efforts of the person under consideration. We also did not included cases where the person was affected due to factors outside of their control. Examples of samples omitted are: a fan unexpectedly falls on a person working at their desk; or a pole becomes loose and comes off, while a pole dancer is using it. These kinds of videos may not require the actual understanding of actions, and may simply be classified by detecting the sudden increase in the video speed/motion magnitude. Reasons for failure include errors in planning, aiming, perception/judgement, or execution; lack of skills/ability/strength (unlike in general stunts, actors are not always trained), etc. In this domain, actors can be seen interacting with large to medium sized objects such as skateboards, bicycles, hulahoops, skis, ropes, poles, exercise balls, etc. Action sequences mainly involves a single person.
3. **Trick-shots (TS):** This is another popular category of videos on YouTube, where people try to do things that are extremely difficult to perform. Examples include throwing compact disc into a very slim opening from a distance; generally, this requires many attempts before one succeeds. We compiled samples from failure footage and corresponding successful attempts. Objects used are medium to small sized such as, basketballs, spoons, bags, bottles, food items, cellphones, cups, etc. Unlike previous domains, the choreography of the action sequences in this domain is not limited to a single actor, and may involve the coordinated performance of two actors.
4. **Party Games (PG):** Parties/social gatherings/get-togethers generally have a series of games. We collected pairs of failed and successful attempts at numerous indoor party games. We only selected games that are short ($\sim$ 2.5 secs), and where win/fail can be recognized. Actors can be seen interacting with small sized objects like cups, pencils, ping-pong balls, etc. Action sequences mainly involve a single actor, although unlike other domains, human spectators can be seen in the background standing steadily or moving.

Excluding telltale signs: Sometimes, actors might be behaving joyously (after winning), or acting disappointed (after failing), which might give enough clue to the models to correctly predict win or fail without actually needing to understand the whole action sequence. There could be other signs as well. Therefore, during data collection, we made sure to not include any such signs in our action sequences.

All of the videos are of high resolution – 720p. Further specifications about our dataset are provided in Table 1.

4. Experiments

In this section, we systematically determine the characteristics of our dataset, then provide baselines and suggestions for future efforts.

Models [2]: We used a CNN (G) to compute spatial features, followed by a temporal modeling module[3] (TMM)

[1]With the permission of organizers, **Supplementary material** or `https://arxiv.org/abs/2102.07355`.

[2]For diagrams of all models, please refer to the **Supplementary material** or `https://arxiv.org/abs/2102.07355`.

[3]In this paper, we alternately use terms: temporal modeling module, aggregation scheme, and consensus scheme.

(F) to compute temporal relationships from spatial features from G. In particular, we considered temporal order agnostic (*averaging* activations from G, and further processed through `fc` layers) and temporal order respecting (LSTM [18] and TRN [55]) as our temporal models.

We used cross-entropy loss, $\mathcal{L}$ as the objective function to train the networks. Let x_i and y_i be the predicted and ground-truth labels, then,

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} y_i log x_i \quad (1)$$

We experimented with the following two approaches for the win-fail action recognition task:

1. **Individual/standalone analysis:** this is identical to any typical image/action classification, where we process the images/video-clip through a network, and it predicts the class (win or fail in our case). $x_i = H(F(G(V)))$, where, V is input video frames in the case G is a 2D-CNN, or it is video clips if G is a 3D-CNN; and H represents a linear layer. This is a binary classification problem: $x_i, y_i \in \{0, 1\}$.

2. **Pairwise analysis:** In this approach, we are able to leverage the pairwise nature of our dataset using a siamese setup. Let V_a and V_b represent two input videos, then, $x_i = H(C(F(G(V_a)), F(G(V_b))))$, where, C is the concatenation operation. We allow $V_a = V_b$ to incorporate the individual/standalone analysis of samples. We see pairwise loss as an aid to the learning process. In all, pairwise analysis is a four-way classification problem: $x_i, y_i \in \{00, 01, 10, 11\}$.

Implementation details: We used PyTorch [38] to implement all of our models. We used 2D ResNet-18 [17] pretrained on ImageNet [6] as our CNN, unless mentioned otherwise. We trained all of our models for 100 epochs using ADAM [20] as our optimizer, with a learning rate of 1e-4, and a batchsize of 5. This also helped in keeping hyperparameter tuning to a minimum. We used a LSTM module with a hidden state of size 256. For a fair comparison with LSTM, for the *averaging* case, we further add fully-connected layers after the averaging operation. Unless specified otherwise, we uniformly sampled 16 frames from entire video sample sequences and employed our pairwise approach. We resized all of the videos to a resolution of 320×240 pixels, and applied center cropping (224×224 pixels); during the training phase, we also applied horizontal flipping. Center-cropping also removes branding/watermarking, which may give out the win/fail class, and may allow the network to take shortcuts.

Metric: Unless specified otherwise, we report *overall* accuracy in percentage.

Temporal model	Train:Test ratios			
	10:90	**30:70**	**50:50**	**70:30**
AVG.	44.63	61.76	69.01	71.33
LSTM	**48.10**	**68.88**	**72.56**	**77.04**

(a)

GS	**IWF**	**TS**	**PG**
+13.00	+9.75	+0.25	+5.50

(b)

00	**01**	**10**	**11**
84.79	61.88	61.54	67.31

(c)

Table 2: (a) **Split ratios and aggregation methods**; (b) domain-wise gains of LSTM over AVG for 30:70 split; (c) class-wise accuracy of LSTM for 30:70 split.

4.1. Task feasibility, split ratios and aggregation schemes

First of all, we wanted to determine if our task is feasible. Secondly, video action recognition by definition is a task of spatiotemporal nature, which implies that, ideally, temporal order of frames/clips, and hence temporal modeling, plays a very important part. On current action datasets, averaging (which ignores temporal order) as the consensus scheme yields the best results [19, 35, 45]. Although some works incorporate local, short-term motion cues using 3D-CNN, optical flow, etc., the demand for actual long-term temporal modeling from current datasets is still limited. In this experiment, we wanted to determine which temporal modeling scheme is better suited for our dataset: temporal-order agnostic (averaging) or temporal-order sensitive (LSTM).

In order to focus only on temporal modeling, we pretrained our CNN backbone on ImageNet and then froze it, which acts as a general spatial feature extractor. Then, we learned only the parameters of the temporal model, which takes in features from the CNN backbone. We have decoupled the spatial learning aspect from temporal modeling – both temporal models are fed the same spatial features.

Thirdly, we wanted to determine a good train:test split ratio for our dataset. For that, we considered various train:test ratios. We compared Averaging (AVG) vs. LSTM for various train:test split ratios.

For this experiment, we employed pairwise comparative approach. Results are shown in Table. 2a. Random guessing would have an accuracy of 25%, since a pairwise comparative approach is a four-way classification problem. Both models performed significantly better than random chance across all split ratios, which suggests that our task is feasible. We observed that LSTM outperforms AVG for all split ratios. We also note that the LSTM's gain over AVG increases as the training pool increases. LSTM performing better than AVG clearly indicates that our dataset demands actual temporal modeling from models. This is

Method	Chance	Individual	Pairwise
Accuracy	50.00	58.65	**76.05**

Table 3: **Individual vs. Pairwise Assessment.**

because, even with a comparative approach, various contextual and human-movement cues from start to finish need to be strung together in a sequential manner to infer about the actor's goal and determine whether they achieved it.

Noting the trade-off between split ratios and performance, we chose 30:70 as our optimal split, which was used for the rest of the experiments.

4.2. Individual vs. Pairwise

In this experiment, we aimed to determine the correct approach to a win-fail action understanding problem: individual/standalone analysis or a pairwise comparative approach.

We used the LSTM aggregation and the same settings as Experiment 4.1. We compare the performances of individual and pairwise approaches in Table 3. Note that individual assessment is actually built into our pairwise approach as well. In Table 3, for the pairwise approach, we show the average accuracy of 00 and 11 (individual assessment), which is directly comparable to that of actual individual assessment. Since individual assessment is a two-way (win or fail) classification problem, a random guess would have an accuracy of 50%. We observe that the individual assessment model has quite a poor performance. On other hand, by learning through pairwise comparison, the model was able learn in a much better way, and was able to perform significantly better, with an accuracy of 76.05%. These results indicate that a pairwise comparative approach is much more suitable, at least for the model that we used. We suggest a comparative approach, but we also want to encourage future works to develop better standalone approaches.

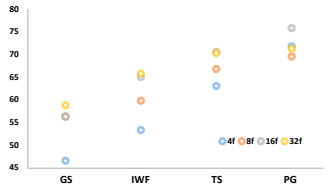
We could have altered the order of Experiments 4.1 and 4.2, but that would have resulted in performing an unnecessarily larger number of experiments. For the rest of the experiments, we use the LSTM based pairwise approach.

4.3. Rate of sampling input frames

In this experiment, we studied the effect of the rate of sampling input frames. In particular, we considered sampling uniformly spaced 4, 8, 16, and 32 frames as input. We also conducted the same experiment on a typical action recognition dataset. The results are in Table 4a. We observed that on UCF101 dataset, the performance saturated with just 4 frames, while on ours action understanding dataset, it saturates at 16 frames. This indicates that intermediate frames and the cues/details in those are important. We also show the effect of varying the number of input frames across individual domains in Table 4b.

# frs.	Accuracy	
	Ours	UCF101
4	60.97	**61.12**
8	64.20	60.06
16	**68.88**	57.60
32	67.66	58.00

(a)



(b)

Table 4: (a) **Effect of rate of sampling input frames**; (b) effect on individual domains.

CNN	Trained on	Accu.
R18-2D	ImageNet	**68.88**
R18-3D	Kinetics	65.65

(a)

GS	IWF	TS	PG
+5.75	-1.25	-2.00	-8.75

(b)

Table 5: (a) **Backbone Choice: 2DCNN vs. 3DCNN**; (b) effect of using 3DCNN compared to 2DCNN across all domains.

4.4. Typical 3DCNN as feature extractor

3DCNNs are known to extract much richer features than a 2DCNNs and as a result, obtain state-of-the results on action recognition tasks. In this experiment, we used a 3D counterpart (ResNet18-3D) of our 2DCNN. Both extract 512-dimensional features. ResNet18-3D extracts features from 16-frame clips. With 3DCNN as the backbone, we used 16 16-frame clips, in place of 16 frames, as input. Clips used with 3DCNN have a lower resolution (112×112 pixels), as compared to that of frames used with 2DCNN (224×224 pixels).

We compare the results in Table 5. Interestingly, we found that 3DCNN performed worse than 2DCNN. Only domain where 3D-CNN performed better is General Stunts, probably because Kinetics has similar action classes like Gymnastics. Potential reasons for poorer performance of 3D-CNN could be: 1) smaller resolution input might be hurting in our case because of the smaller sized objects and interactions involved with those; 2) actions patterns are different; 3) multiple humans present in the scene; or 4) ImageNet contains classes for many objects found in our dataset, while Kinetics does not. We believe explicitly modeling/finetuning 3D-CNN for human-object interactions would be beneficial.

4.5. End-to-End learning

So far, we have used spatial features extracted using an off-the-shelf CNN. In this experiment, we sought to determine if there is a utility in jointly learning spatial and temporal representations. We unfroze the CNN backbone and

Training	LSTM	TRN
Only TMM	68.88	71.24
End-to-End	**74.78**	**75.74**

(a)

GS	IWF	TS	PG
+20.00	+5.75	+0.75	+2.75

(b)

00	01	10	11
79.20	70.28	70.28	79.37

(c)

00	01	10	11
79.90	73.60	72.40	77.10

(d)

Table 6: (a) **End-to-End learning**. (b) Domain-wise improvements. (c, d) Class-wise accuracy of LSTM and TRN.

Shuffle type	Accu.
None	74.78
First $1/3$	74.16
Middle $1/3$	74.26
Last $1/3$	72.42
Full	61.19

Table 7: **Effect of shuffling.**

Observed seq.	Accu.
Full	74.78
First $1/4$	25.13
First $1/2$	35.62
First $3/4$	48.60
Last $1/4$	23.47
Last $1/2$	28.93
Last $3/4$	52.58

Table 8: **Partial observations.**

optimized the network end-to-end. Then, we again finetuned only the temporal modeling module. The results are presented in Table 6a. We also evaluated a multiscale TRN (16f) baseline. We observed a significant boost in performance, indicating that the dataset requires spatial representation learning as well. We observe highest improvement in General Stunts, probably because ImageNet does not have people in unusual, convoluted poses, and hence, benefits a lot from finetuning. Furthermore,class-wise accuracies after end-to-end optimization are much more balanced w.r.t. 00 and 11 (compare Tables 6c and 2c).

These performances also serve as baselines for future works. Since the end-to-end optimized model worked best, we use that in the rest of the experiments. For simplicity, we continue to use LSTM as our temporal modeling module.

4.6. Importance of temporal order

In this experiment, we wanted to confirm if the temporal order from various parts of sequences matter. For that, in the testphase, we perturbed the temporal order of only a part of the sequence, while keeping the temporal order of the other parts of the sequence intact. In particular, we shuffled: 1) one-third of the sequence from the start (first 5 of the 16 sampled frames); 2) the middle one-third of the sequence (middle 5 frames); and 3) the last one-third of the sequence. Additionally, we considered shuffling the entire sequence.

The results are shown in Table 7. We observed that perturbing the temporal order in all the parts affected the performance negatively. Furthermore, we observed that impact of perturbation increased as we moved the focus of the shuffling towards the end of the sequence. Shuffling the entire sequence had the most negative impact. These observations support the hypothesis that our dataset demands/requires the algorithms/models to temporally model from the beginning to the end of the sequences.

4.7. Where are the necessary cues?

In this experiment, we wanted to probe if "seeing" the entire sequence is needed, and/or if the model is able to make prediction just from a subsequence. During the testphase, we asked the model to classify based on partial sequences. Results are presented in Table 8. We found that predictions based only on the first or last one-fourth sequence are equal to random guessing (25% accuracy). We noticed that the accuracy of the model increased the further it observed the sequence, indicating that the necessary cues are present along the entire sequence.

4.8. Video Retrieval

To evaluate if our task yields a more comprehensive understanding, we devised a novel video retrieval experiment. We collected an additional, separate set of win and fail samples (from Internet Wins-Fails domain), which served as our queries. We also collect a set of baby and animal win-fails. Actors in our original win-fail dataset and the additional samples (queries) are adolescent and adult human beings. We use queries to retrieve videos from databases, where we changed the situations and actors. Particularly, we considered three different databases:

1. Activities of Daily Living (ADL)-Fall: We used the dataset released by [23] as our database. ADL include activities such as sitting down, standing up, getting things from floor, *etc*. Falls include person walking and falling down. This dataset was built to be used in monitoring elderly people. We consider ADL, and Fall as relevant to win and fail queries, respectively.

2. Win-Fail with Babies as actors: Fails in babies include babies trying to get up, crawl, walk but falling over since they have not yet acquired balance, falling while sitting due to lack of balance and control, *etc*. Movements of babies are a lot more jittery compared to adults. Wins include climbing into their cradles successfully, throwing balls into baskets, passing through narrow spaces, *etc*. We consider baby wins and fails as relevant to adult win and fail queries, respectively.

3. Win-Fail with Animals as actors: Wins include animals being able to play ping-pong; shoot basket/pass a

Model	*Win-Fail → Fall-ADL*			*Win-Fail → Baby Win-Fail*			*Win Fail → Animal Win-Fail*		
	R@1	**R@5**	**Sim Δ**	**R@1**	**R@5**	**Sim Δ**	**R@1**	**R@5**	**Sim Δ**
AR	0.015	0.073	0.11	0.016	0.094	0.06	0.042	0.193	0.05
WFR	**0.017** (↑13%)	**0.088** (↑20%)	**0.29**	**0.018** (↑13%)	0.094	**0.13**	**0.057** (↑36%)	**0.267** (↑38%)	**0.44**

Table 9: **Video retrieval results**. Higher is better. AR - action recognition model; WFR - win-fail recognition model.

ball; open a door/window and get out/in; *etc*. Fails include reaching out a take something, but falling; tumbling over while running; jumping but falling short; *etc*. Note that animals have different structure and movements than humans. We consider animal win and fail as relevant to human win and fail queries, resp.

Note that these queries and databases were not seen during training. We used cosine similarity as a similarity measure when retrieving, and recall at rank 1 and 5 (R@1, R@5) as metrics. We also noted average similarity difference from query to relevant and irrelevant samples in the databases (Sim Δ), which would show sensitivity of features towards wins/fails in unseen domains. We considered model trained UCF101 for action recognition as our baseline. Results are summarized in Tab. 9. We found that win-fail recognition model outperformed action recognition model across all the databases. Moreover, the gap in performances increased when retrieving from animal database. We also observed that win-fail recognition model was more sensitive to win-fail aspect of the query, retrieved samples. These results also suggest application of WFR in areas like elderly and children safety monitoring.

Qualitative results are presented in Fig. 2. For brevity, in the following we refer to individual samples using respective row (R), and column (C) numbers in Fig. 2. We observe that AR model retrieves considerably on the basis of color (*e.g.*: (R1,C6), (R1,C7), (R5,C5), (R5,C7)); low-level motion patterns (*e.g.* extended hand in (R3,C1)→(R3,C6); sliding pattern (R4,C1)→(R4,C5) – skater smoothly gliding down the road is good, while the baby is smoothly falling down the slide is bad, jumping pattern (R6,C5),(R6,C6), (R6,C7)). Compared to that WFR model retrieves while maintaining meaning (*e.g.* (R2,C2), (R2,C4) both exhibit compact body form of the diver in (R2,C1) necessary to sit on the chair and pass through swim rings; landing safely in (R6,C4) and (R6,C1); stunt involving multiple parties (R6,C1), (R6,C3); falling while reaching out (R5,C1),(R5,C2), (R5,C3)). Sometimes, like AR, WFR also puts more emphasis on motion patterns (*e.g.*, (R6,C2)).

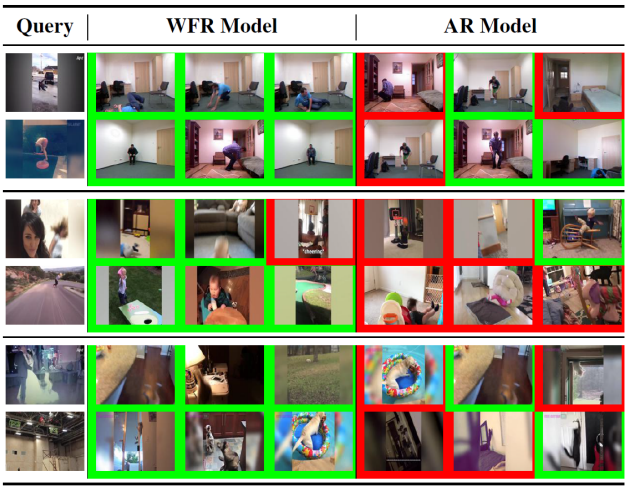


Figure 2: **Qualitative results**. Odd and even rows show 'Fails' and 'Wins' as queries, respectively. First, second, and third two rows: ADL-Fall, Baby Win-Fail, Animal Win-Fail as databases. Red and green indicate relevant and irrelevant retrievals w.r.t. Win/Fail aspect. Please view in an Adobe Reader to play videos.

5. Conclusion

As a step towards true, comprehensive video/action understanding, we proposed the task of differentiating between the concepts of wins & fails. To facilitate our task, we also introduced a new dataset, which contains 817 pairs of successful & failed attempts at various activities from 'General Stunts,' 'Internet Wins-Fails,' 'Trick Shots,' & 'Party Games' domains. The action sequences in our dataset are not overly long, yet are complex. We systematically analyzed our dataset & found that: 1) our dataset requires true temporal modeling; 2) pairwise approach worked better than individual/standalone assessment; 3) details/cues important for understanding video/action are present in intermediate frames along the entire sequence; 4) better performance (as compared to a 2DCNN) of an off-the-shelf 3DCNN did not translate well to our dataset/task; & 5) spatial modeling is equally important. All these characteristics are desirable in a video understanding dataset, which makes our dataset & task are suitable for comprehensive video understanding problem benchmark, & will help advance the field of video understanding. While current action recognition methods worked well on our task/dataset, they still leave a large gap to cover, indicating that there is a significant opportunity to improve on this task.

References

[1] Alexander Binder, Wojciech Samek, Klaus-Robert Müller, and Motoaki Kawanabe. Machine learning for visual concept recognition and ranking for images. In *Towards the Internet of Services: The THESEUS Research Program*, pages 211–223. Springer, 2014.

[2] Szilvia Biro and Alan M Leslie. Infants' perception of goal-directed actions: development through cue-based bootstrapping. *Developmental science*, 10(3):379–398, 2007.

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[4] Nicolas Chesneau, Karteek Alahari, and Cordelia Schmid. Learning from web videos for event classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):3019–3029, 2017.

[5] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[7] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[8] Hazel Doughty, Dima Damen, and Walterio Mayol-Cuevas. Who's better? who's best? pairwise deep ranking for skill determination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6057–6066, 2018.

[9] Hazel Doughty, Walterio Mayol-Cuevas, and Dima Damen. The pros and cons: Rank-aware temporal attention for skill determination in long videos. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7862–7871, 2019.

[10] Terry Eskenazi, Marc Grosjean, Glyn W Humphreys, and Guenther Knoblich. The role of motor simulation in action perception: a neuropsychological case study. *Psychological Research PRPF*, 73(4):477–485, 2009.

[11] Terje Falck-Ytter, Gustaf Gredebäck, and Claes von Hofsten. Infants predict other people's action goals. *Nature neuroscience*, 9(7):878–879, 2006.

[12] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 6202–6211, 2019.

[13] Paul M Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954.

[14] Rohit Girdhar and Deva Ramanan. Cater: A diagnostic dataset for compositional actions and temporal reasoning. *arXiv preprint arXiv:1910.04744*, 2019.

[15] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense.

[16] Marc Grosjean, Maggie Shiffrar, and Günther Knoblich. Fitts's law holds for action perception. *Psychological Science*, 18(2):95–99, 2007.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[18] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

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

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[21] Justin Kruger, Derrick Wirtz, Leaf Van Boven, and T William Altermatt. The effort heuristic. *Journal of Experimental Social Psychology*, 40(1):91–98, 2004.

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

[23] Bogdan Kwolek and Michal Kepski. Human fall detection on embedded platform using depth maps and wireless accelerometer. *Computer methods and programs in biomedicine*, 117(3):489–501, 2014.

[24] Qing Lei, Hong-Bo Zhang, Ji-Xiang Du, Tsung-Chih Hsiao, and Chih-Cheng Chen. Learning effective skeletal representations on rgb video for fine-grained human action quality assessment. *Electronics*, 9(4):568, 2020.

[25] Yongjun Li, Xiujuan Chai, and Xilin Chen. End-to-end learning for action quality assessment. In *Pacific Rim Conference on Multimedia*, pages 125–134. Springer, 2018.

[26] Yongjun Li, Xiujuan Chai, and Xilin Chen. Scoringnet: Learning key fragment for action quality assessment with ranking loss in skilled sports. In *Asian Conference on Computer Vision*, pages 149–164. Springer, 2018.

[27] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 513–528, 2018.

[28] Zhenqiang Li, Yifei Huang, Minjie Cai, and Yoichi Sato. Manipulation-skill assessment from videos with spatial attention network. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[29] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7083–7093, 2019.

[30] Diogo C Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5137–5146, 2018.

[31] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The jester dataset: A large-scale video dataset of human gestures. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[32] Jia-Hui Pan, Jibin Gao, and Wei-Shi Zheng. Action assessment by joint relation graphs. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6331–6340, 2019.

[33] Paritosh Parmar and Brendan Morris. Action quality assessment across multiple actions. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1468–1476. IEEE, 2019.

[34] Paritosh Parmar and Brendan Tran Morris. Measuring the quality of exercises. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2241–2244. IEEE, 2016.

[35] Paritosh Parmar and Brendan Tran Morris. What and how well you performed? a multitask learning approach to action quality assessment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–313, 2019.

[36] Paritosh Parmar, Jaiden Reddy, and Brendan Morris. Piano skills assessment. *arXiv preprint arXiv:2101.04884*, 2021.

[37] Paritosh Parmar and Brendan Tran Morris. Learning to score olympic events. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–28, 2017.

[38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8026–8037. 2019.

[39] Faegheh Sardari, Adeline Paiement, and Majid Mirmehdi. View-invariant pose analysis for human movement assessment from rgb data. In *International Conference on Image Analysis and Processing*, pages 237–248. Springer, 2019.

[40] Jessica A Sommerville and Amanda L Woodward. Pulling out the intentional structure of action: the relation between action processing and action production in infancy. *Cognition*, 95(1):1–30, 2005.

[41] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[42] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[43] Yansong Tang, Zanlin Ni, Jiahuan Zhou, Danyang Zhang, Jiwen Lu, Ying Wu, and Jie Zhou. Uncertainty-aware score distribution learning for action quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9839–9848, 2020.

[44] Jiahao Wang, Zhengyin Du, Annan Li, and Yunhong Wang. Assessing action quality via attentive spatio-temporal convolutional networks. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pages 3–16. Springer, 2020.

[45] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016.

[46] Tianyu Wang, Minhao Jin, Jingying Wang, Yijie Wang, and Mian Li. Towards a data-driven method for rgb video-based hand action quality assessment in real time. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pages 2117–2120, 2020.

[47] Amanda L Woodward and Sarah A Gerson. Mirroring and the development of action understanding. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1644):20130181, 2014.

[48] Amanda L Woodward and Jessica A Sommerville. Twelve-month-old infants interpret action in context. *Psychological Science*, 11(1):73–77, 2000.

[49] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 284–293, 2019.

[50] Xiang Xiang, Ye Tian, Austin Reiter, Gregory D Hager, and Trac D Tran. S3d: Stacking segmental p3d for action quality assessment. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 928–932. IEEE, 2018.

[51] Chengming Xu, Yanwei Fu, Bing Zhang, Zitian Chen, Yu-Gang Jiang, and Xiangyang Xue. Learning to score figure skating sport videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[52] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015.

[53] Ling-An Zeng, Fa-Ting Hong, Wei-Shi Zheng, Qi-Zhi Yu, Wei Zeng, Yao-Wei Wang, and Jian-Huang Lai. Hybrid dynamic-static context-aware attention network for action assessment in long videos. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2526–2534, 2020.

[54] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *Proceedings of the*

IEEE International Conference on Computer Vision, pages 2248–2255, 2013.

[55] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018.

[56] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.