Mimetics: Towards Understanding Human Actions Out of Context

Philippe Weinzaepfel · **Grégory Rogez**

Abstract Recent methods for video action recognition have reached outstanding performances on existing benchmarks. However, they tend to leverage context such as scenes or objects instead of focusing on understanding the human action itself. For instance, a tennis field leads to the prediction *playing tennis* irrespectively of the actions performed in the video. In contrast, humans have a more complete understanding of actions and can recognize them without context. The best example of out-of-context actions are mimes, that people can typically recognize despite missing relevant objects and scenes. In this paper, we propose to benchmark action recognition methods in such absence of context and introduce a novel dataset, *Mimetics*, consisting of mimed actions for a subset of 50 classes from the Kinetics benchmark. Our experiments show that (a) state-of-the-art 3D convolutional neural networks obtain disappointing results on such videos, highlighting the lack of true understanding of the human actions and (b) models leveraging body language via human pose are less prone to context biases. In particular, we show that applying a shallow neural network with a single temporal convolution over body pose features transferred to the action recognition problem performs surprisingly well compared to 3D action recognition methods.

Keywords Biases in Action Recognition, Mimes

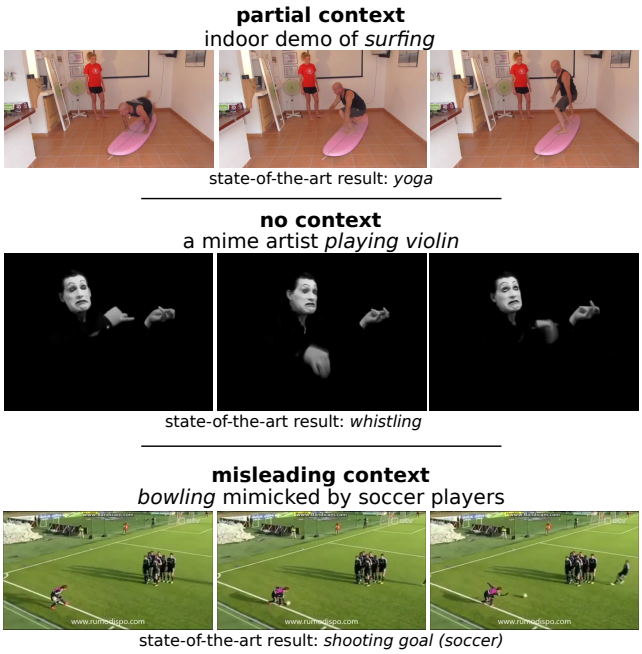

Fig. 1 Examples where context is partial (top), absent (middle), or misleading (bottom). The first row shows someone training indoor for *surfing*, with the right object but not in a standard place. The second row shows a mime artist mimicking someone *playing violin*, but the object and the scene are absent. The third example contain a misleading context: soccer players are mimicking a scene of *bowling* on a soccer field with a soccer ball. In all these cases, state-of-the-art 3D CNNs fail to recognize the actions.

1 Introduction

Action recognition has made remarkable progress over the past few years (Carreira and Zisserman 2017; Feichtenhofer et al. 2019; Simonyan and Zisserman 2014; Wang et al. 2016). Most state-of-the-art methods (Carreira and Zisserman 2017; Hara et al. 2018; Tran et al. 2018) are built upon deep spatio-temporal convolutional achitectures applied on short clips of RGB frames. These approaches achieved impressive classification performance, with a top-1 accuracy over 77% on the Kinetics dataset (Kay et al. 2017), and a top-5 accuracy of more than 93%. However, the explanations behind such perfor-

NAVER Labs Europe
6 chemin de Maupertuis, 38240 Meylan, France
Tel.: +33-476-615-050
E-mail: firstname.lastname@naverlabs.com

Fig. 2 Examples of training frames where humans are masked. The context clearly suffices to guess the action *parasailing* (left) and *driving tractor* (right).

mances remain unclear. In particular, recent works (Li et al. 2018; Li and Vasconcelos 2019; Jacquot et al. 2020) have shown that most datasets, and thus what Convolutional Neural Networks (CNNs) learn, are biased by static context such as scenes and objects. For instance, Figure 1 shows some examples where context is only partial, absent or misleading, and that are misclassified by state-of-the-art 3D CNNs. In particular, the last video taking place on a soccer field is classified as *shooting goal (soccer)*, regardless of the actual action performed in the video.

To further assess the bias of existing datasets towards scenes and objects, we retrain a model on Kinetics after masking out all the humans in the videos, see Figure 2. The performance of this model on the original test set is around 65%, which is extremely high for a model that has never seen any human at training. This shows that scenes and objects are often sufficient to correctly classify the actions.

While this contextual information is certainly useful to predict human actions, it is not sufficient to truly understand what is happening in a scene. Humans have a more complete understanding of actions and can even recognize them without any context, object or scene. The most obvious example is given by mime artists, see middle row of Figure 1, who can suggest emotions or actions to the audience using only facial expressions, gestures and movements, but without words or context. Mime as an art originates from ancient Greece and reached its heights with sixteen century Commedia dell'Arte, but it is considered one of the earliest mediums of expressions even before the appearance of spoken language. We claim that an intelligent system should also be able to understand mimed actions.

To understand action in out-of-context scenarios, *i.e.*, when object and scene are absent or misleading as shown in Figure 1, action recognition can only rely on *body language* captured by human pose and motion. Such a cue is leveraged in the well-established field of 3D skeleton-based action recognition, also called *3D action recognition* (Du et al. 2015b; Liu et al. 2016; Zhu et al. 2016), that take as input sequences of 3D pose skeletons. These methods have shown impressive results, validating that contextual information is not always necessary to recognize actions. However, they are usually trained and tested on accurate and scripted sequences of 3D human poses, captured with RGB-D sensor (Shahroudy et al. 2016) or Motion Capture systems (Du et al. 2015b; Zhu et al. 2016) in constrained and unrealistic environments. To the best of our knowledge, 3D action recognition has never been applied to real-world situations and videos captured in the wild. Another of our contributions is therefore to study whether such techniques generalize in-the-wild given current pose detectors and can be employed in out-of-context scenarios.

Recent human pose estimation methods (Mehta et al. 2017; Rogez et al. 2019), allow to estimate 3D poses of multiple people from a single image. In this paper, we employ LCR-Net++ (Rogez et al. 2019) to extract human 3D pose information from videos. It has shown robustness to challenging cases like occlusions and truncations by image boundary, estimating full-body 2D and 3D poses for every person in an image. We compare three different action recognition baselines based on these poses. The most intuitive pipeline is to detect 3D human poses in every frame, build 3D pose sequences by linking detections over time, and apply a state-of-the-art 3D action recognition algorithm. However, such a method is likely to be sensitive to the level of noise inherent to 3D pose estimation in the wild. The second baseline applies graph convolutions on 2D pose sequences, without 3D information, which might have the advantage to be more accurate. We finally study another approach where 1D temporal convolutions are applied on human-level intermediate pose feature representations from LCR-Net++. In other words, we transfer the features learned for 2D-3D pose estimation to action recognition: they typically contain information about the human poses without explicitly representing them as body keypoint coordinates.

Finally, to benchmark action recognition methods in out-of-context scenarios, we introduce the **Mimetics** dataset[1]. It contains over 700 video clips of mimed actions for a subset of 50 classes from the Kinetics dataset. Mimetics allows to evaluate on mimed actions models that have been trained on Kinetics. It is not meant to be used as training data. Our claim is that systems that supposedly try to reach human performance should be able to recognize actions out of context as humans do without seeing mimes at training. For further analysis, we additionally annotate for each clip whether an object gives clues on the action or not, and similarly for the scene. We also labelled the size of the objects

[1] https://europe.naverlabs.com/research/computer-vision/mimetics/

for a fine-grained analysis of the bias towards objects. We evaluate a state-of-the-art 3D convolutional network, and confirm that these models are biased towards scenes and objects. Pose-based action recognition provides a more interpretable output but can lack fine-grained pose details, *e.g.*, face and hands, for higher performance.

This paper is organized as follows. After reviewing related work in Section 2, we study the bias of state-of-the-art action recognition datasets and models in Section 3. Section 4 then presents various pose-based baselines and compares them on existing action recognition datasets. Finally, Section 5 introduces the Mimetics dataset and analyzes the performance on out-of-context action recognition.

2 Related work

We benchmark action recognition approaches, comparing standard CNNs on RGB clips with pose-based methods. This latter category can be further split into 2D pose-based approaches and 3D action recognition.

Action classification in real-world videos. Different strategies have been deployed to handle video processing with CNNs such as two-stream architectures (Feichtenhofer et al. 2016; Simonyan and Zisserman 2014), Recurrent Neural Networks (RNNs) (Donahue et al. 2015), or spatio-temporal 3D convolutions (Carreira and Zisserman 2017; Feichtenhofer et al. 2019; Tran et al. 2015). Simonyan and Zisserman (2014) introduced a two-stream architecture with 2D convolutions, in which one stream captures appearance information from RGB inputs while the second one operates on optical flow representation and models motion. While improvements of this approach have been proposed (Feichtenhofer et al. 2016), most state-of-the-art methods now use a 3D deep convolutional network (Carreira and Zisserman 2017; Tran et al. 2015, 2018; Xie et al. 2018), optionally in combination with a two-stream architecture. Compared to 2D convolutions, 3D convolutions allow to leverage spatio-temporal information at the cost of a higher number of parameters and higher computational cost. With recent very large-scale datasets such as Kinetics (Kay et al. 2017), it is possible to train such 3D CNNs effectively (Hara et al. 2018), and impressive performances can be obtained even on small datasets, thanks to pre-training on Kinetics (Carreira and Zisserman 2017). For instance, I3D (Carreira and Zisserman 2017) achieved state-of-the-art accuracy on HMDB51 (Kuehne et al. 2011) and UCF101 (Soomro et al. 2012) using a two-stream network with a 3D Inception backbone Szegedy et al. (2015). Tran et al. (2018) and Xie et al. (2018) replaced 3D convolutions with separate spatial and temporal convolutions, which reduces the number of parameters to learn. However, all these methods lack a clear understanding of their classification choices. In particular, recent studies (Li et al. 2018; Li and Vasconcelos 2019) suggest that they tend to leverage dataset biases instead of focusing on the human action.

2D pose for action classification in real-world videos. An insightful diagnostic to understand what affects the action recognition results most was provided by Jhuang et al. (2013), who found that high-level 2D pose features greatly outperform low/mid level features. This has motivated further research on incorporating 2D body poses information in real-world action recognition models (Angelini et al. 2018; Chéron et al. 2015; Du et al. 2017; Iqbal et al. 2017; McNally et al. 2019; Zhu et al. 2018). For instance, this can be done by pooling features (Cao et al. 2016; Chéron et al. 2015) or defining an attention mechanism (Du et al. 2017; Girdhar and Ramanan 2017). However, this leads to limited gain and often assumes that humans are fully-visible. Zolfaghari et al. (2017) trained a 3D CNN on human part segmentation inputs, and added a third stream to two-stream networks. Some other recent methods have shown improved action recognition performance by incorporating 2D pose information from off-the-shelf pose detectors (Choutas et al. 2018; Liu and Yuan 2018; Wang et al. 2018). For instance, Choutas et al. (2018) and Liu and Yuan (2018) extract joint heatmaps and encode their evolution over time. Wang et al. (2018) define a two-stream network: one stream encodes the evolution of the pose while the second one models relationship with objects. However, it remains limited to single-person action recognition. Luvizon et al. (2018) propose a multi-task architecture where 2D poses are predicted at the same time as appearance features are pooled over body joints for action recognition.

3D action recognition. Compared to 2D poses, 3D poses have the advantage to be unambiguous and to better handle motion dynamics. Recent attempts on 3D action recognition have employed RNNs to handle sequential data and to model the contextual dependencies in the temporal domain (Du et al. 2015b; Liu et al. 2016; Si et al. 2018; Weng et al. 2018; Zhu et al. 2016). Du et al. (2015b) propose a hierarchical RNN in which the human skeleton was divided into five parts (arms, legs and trunk) to feed five different subnets later fused hierarchically. Zhu et al. (2016) added a mixed-norm regularization term to a RNN cost function in order to learn the co-occurrence features of skeleton joints for action classification. More recently, simple CNN-based methods applied to the 2D or 3D joint coordinates have shown to outperform more complex RNN architectures (Du et al. 2015a). In a similar spirit, Yan

et al. (2018) represent the sequence of poses as a graph, and apply a spatio-temporal graph convolutional network (STGCN) to recognize actions. Most of these algorithms use 3D human poses obtained from a Motion Capture system (Du et al. 2015b; Zhu et al. 2016), a Kinect sensor (Liu et al. 2016) or a multi-camera setting (Yao et al. 2012), and none of them experimented on real-world videos with estimated 3D poses.

To the best of our knowledge, we are the first to analyze 3D action recognition in real-world videos. Yan et al. (2018) show that their STGCN method can also be applied in the wild, but they only use 2D poses in this scenario. More precisely, they extract 2D human poses with OpenPose (Cao et al. 2018), build a graph using the 2 highest-scored detections per frame, and apply their spatio-temporal graph network, replacing X,Y,Z coordinates of the 3D poses, by x, y, s, where x, y are the 2D coordinates of the joint, normalized into $[-0.5, 0.5]$ and s is the score for this keypoint. In the framework of Luvizon et al. (2018), the multi-task architecture can deal with 2D and 3D poses at the same time as action recognition. However, ground-truth keypoints are required for training, and the 3D component is disabled for datasets in-the-wild, *i.e.*, without 3D ground-truth poses.

3 Context biases in action recognition

To assess how much context is leveraged by current methods based on spatio-temporal CNNs, we consider videos where people are masked out. To do so, we extracted human tubes in all videos using LCR-Net++ (Rogez et al. 2019) detections linked over time (see Section 4.1 for a detailed description) and removed all the humans from the video frames by coloring the tubes content in grey, see Figure 2.

We performed this experiment on the standard Kinetics dataset (Kay et al. 2017) which consists of around 240k training videos, 20k for validation and 40k for testing for a total of 400 classes. As a state-of-the-art model, we use a 3D CNN model, *i.e.*, with spatio-temporal convolutions instead of 2D convolutions, using a ResNeXt-101 backbone (Xie et al. 2017). We first evaluate a 3D CNN trained on original videos and tested on masked videos, thus measuring the biases learned by the model. Mean top-1 accuracy on the validation set is reported in Table 1. It remains close to 40%, which is extremely high given that there is no human from which the action can be recognized in the test videos. This prediction is thus based on the remaining content of the video, *i.e.*, context such as objects or scenes.

To better measure the biases of the dataset itself, we have trained a 3D CNN model on the masked videos and obtain 65.7% on the original videos, down by only 8.8% compared to training on the original data. This performance is outstanding for a model that has not seen any human during training, and therefore has not really seen any action. To further analyze this aspect, we additionally show in Table 2 the classes with the most increase of accuracy. Masking the actors at training increases the accuracy for classes in which the scene context (*e.g. long jump*, *playing basketball*) or the presence of large objects (*e.g. driving tractor*) are sufficient to recognize the actions, see also Figure 2.

Such bias problem can be tackled by sampling over multiple datasets or reweighting samples, as shown for action (Li et al. 2018; Li and Vasconcelos 2019) or object recognition (Bahng et al. 2019; Khosla et al. 2012; Torralba et al. 2011). For action recognition, another direction is to leverage body language which is not affected by this context bias.

Table 1 Mean top-1 accuracy (in %) when training and testing a 3D CNN model on Kinetics using the original videos or videos where humans are masked

		test on	
		original	masked
train on	original	74.5	38.7
	masked	65.7	63.9

Table 2 Classes with the most increase in accuracy (in %) on Kinetics validation set when training on original videos or masked videos. The last column highlights the difference between these two settings.

class	original	masked	diff.
building shed	74.5	85.1	+10.6
long jump	62.0	72.0	+10.0
driving tractor	68.1	76.6	+8.5
riding elephant	86.0	94.0	+8.0
tying knot (not on a tie)	64.0	72.0	+8.0
playing basketball	66.0	74.0	+8.0
changing oil	85.7	91.8	+6.1
planting trees	77.6	83.7	+6.1
peeling potatoes	59.2	65.3	+6.1
parasailing	82.0	88.0	+6.0

4 Real-world 3D action recognition baselines

We benchmark three baselines, that all require the extraction of human tubes (Section 4.1). We present two different methods that employ a spatio-temporal graph convolutional network, on explicit 3D (Section 4.2) or 2D (Section 4.3) pose sequences respectively. Next, we introduce a third approach that consists of a single 1D temporal convolution applied on mid-level implicit pose features (Section 4.4). Finally, we present experimental results on existing benchmarks in Section 4.5.

4.1 Extracting human tubes

Overview of LCR-Net++. We build our tube extraction and pose estimation upon LCR-Net++ (Rogez et al. 2019), which leverages a Faster R-CNN like architecture (Ren et al. 2015) with a ResNet-50 backbone (He et al. 2016). A Region Proposal Network extracts candidate boxes around humans. These regions are then classified into different so-called 'anchor poses' that replace standard object classes: these key poses typically correspond to a person standing, a person sitting, *etc*. Poses are then refined using a regression branch, that takes as input the same features used for classification. Anchor-poses are defined jointly in 2D and 3D, and the refinement occurs in this joint 2D-3D pose space. The detection framework allows to handle multiple people in a scene. As the approach is holistic, it outputs full-body poses, even in case of occlusions or truncation by image boundaries. We use the real-time model released by the authors[2], allowing experiments on large-scale datasets.

Tube extraction. In order to leverage the evolution of poses over time, one needs to track each individual, *i.e.*, to obtain human tubes (Gkioxari and Malik 2015). We proceed by first running LCR-Net++ in every frame and follow standard procedures used in the spatio-temporal action localization literature (Kalogeiton et al. 2017; Singh et al. 2017) to link detections over time. Starting from the highest scored detection, we match it with the detections in the next frame based on the Intersection-over-Union (IoU) between boxes. We link it if the IoU is over 0.3. Otherwise, we match it to the frame after, and perform linear interpolation in the missing frames. We stop a tube if there was no match during 10 consecutive frames. This procedure is run forward and backward to obtain a human tube. We then delete all detections in this first link, and repeat the procedure for the remaining detections. At training, we label the tubes with the video class. At test time, for each video and for each class, we take the maximum score over all tubes.

4.2 Baseline based on explicit 3D pose

Figure 3 shows an overview of the most intuitive baseline. It is based on explicit 3D pose information. More precisely, given the human tubes, we extract the 3D poses estimated by LCR-Net++ for each box, thus building a 3D pose skeleton sequence for each tube. We finally run a state-of-the-art 3D action recognition method, namely STGCN, using the code released by Yan et al. (2018)[3]. The idea consists in building a graph in space and time from the pose sequence, on which spatio-temporal convolution are applied. We denote this first baseline as **STGCN3D**.

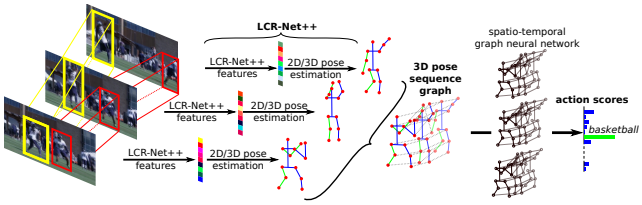

Fig. 3 Overview of the STGCN3D baseline. Given an input video, we run LCR-Net++ to detect human tubes (yellow and red boxes) and estimate 2D/3D poses (shown only for the yellow tube for readability). We build 3D pose sequences and run a state-of-the-art 3D action recognition method based on spatio-temporal graph neural network (Yan et al. 2018) to obtain action scores.

4.3 Variant based on explicit 2D pose

As the STGCN method of Yan et al. (2018) has also been applied to 2D poses, we use a variant of the previous pipeline, replacing the 3D poses estimated by LCR-Net++ by its 2D poses. On the one hand, this variant is likely to get worse performance, as 3D poses are more informative than 2D poses which are inherently ambiguous. But on the other hand, 2D poses extracted from images and videos tend to be more accurate than 3D poses which are more prone to noise. We call this second baseline **STGCN2D**.

4.4 Temporal convolution on implicit pose features

We finally study a baseline that transfers the implicit pose representation carried by mid-level features within LCR-Net++, without using explicit body keypoint coordinates, see Figure 4. We select the features used as input to the final layers for pose classification and refinement. These features have 2048 dimensions with a ResNet50 backbone and carry information about both 2D and 3D poses. The features are stacked over time along human tubes and a temporal convolution of kernel size T is applied on top of the resulting matrix. This convolution outputs action scores for the sequence.

At training, we sample random clips of T consecutive frames and use a cross-entropy loss. At test time, we use a fully-convolutional architecture and average the class probabilities by a softmax on the scores for all clips in the videos. We did experiment with deeper network on top of the stacked features but did not see

[2] http://thoth.inrialpes.fr/src/LCR-Net/

[3] https://github.com/yysijie/st-gcn

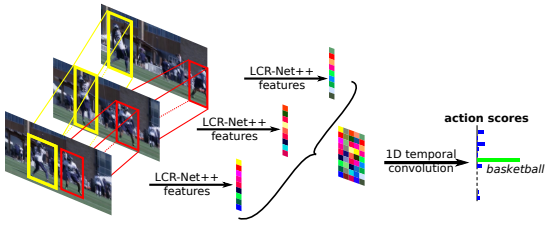

Fig. 4 Overview of the implicit pose baseline, SIP-Net. Given an input video, we run LCR-Net++ to detect humans tubes (yellow and red boxes) and extract mid-level pose features (shown only for the yellow tube for readability). We stack them over time and apply a single 1D temporal convolution to obtain action scores.

Table 3 Overview of the datasets used in our experiments

	#cls	#vid	#splits	in-the-wild	GT 2D	GT 3D
NTU	60	56,578	2	✗	✓	✓
JHMDB	21	928	3	✓	✓	
PennAction	15	2,326	1	✓	✓	
HMDB51	51	6,766	3	✓		
UCF101	101	13,320	3	✓		
Kinetics	400	306,245	1	✓		

any significant improvement. Due to GPU memory constraint, we freeze the weights of LCR-Net++ during training, allowing larger temporal windows to be considered. We denote this third baseline as **SIP-Net** for Stacked Implicit Pose Network.

4.5 Comparison on existing datasets

Before comparing these baselines on out-of-context actions (Section 5), we assess their performance for real-world action recognition on existing datasets, with various levels of ground-truth. Table 3 summarizes them in terms of number of videos, classes, splits, as well as frame-level ground-truths. For datasets with multiple splits, some results are reported on the first split only, denoted for instance as JHMDB-1 for the split 1 of JHMDB. While our goal is to perform action recognition in real-world videos, we validate the baselines on the constrained NTU 3D action recognition dataset (Shahroudy et al. 2016) that contains ground-truth poses in 2D and 3D, using the standard cross-subject (cs) split. We also experiment on the JHMDB (Jhuang et al. 2013) and PennAction (Zhang et al. 2013) datasets that have ground-truth 2D poses, but no 3D poses. Finally, we use HMDB51 (Kuehne et al. 2011), UCF101 (Soomro et al. 2012) and Kinetics (Kay et al. 2017) that contain no more information than the ground-truth label of each video. As metric, we report the standard mean accuracy, *i.e.*, the ratio of correctly classified videos per class, averaged over all classes.

In Appendix A, we report various experiments based on this various levels of ground-truth, allowing to study the impact of extracted tubes, extracted poses as well as the benefit of transferring pose features for SIP-Net. We also plot the performance of SIP-Net with varying T and use $T = 32$ in the remaining of this work.

Table 4 provides a comparison of the mean accuracy on all datasets (last three rows). The method based on implicit pose features (SIP-Net) significantly outperforms the baselines that employ explicit 2D and 3D poses, except on NTU. The gap is over 10% on HMDB51, UCF101 and Kinetics. This can be explained by the fact that explicitly extracting the poses lead to a significant level of noise in the body keypoint representations for in-the-wild videos. Using an implicit pose representation as in SIP-Net allows for more robustness. Interestingly, on HMDB51, UCF101 and Kinetics, the 2D pose baseline performs slightly better than the 3D, suggesting that 3D pose suffers from much more noise in unconstrained videos.

Finally, we compare our baselines to the state of the art among pose-based methods, see Table 4. SIP-Net obtains a higher accuracy than PoTion (Choutas et al. 2018) with a margin over 5% on JHMDB, HMDB51 and UCF101-1, and of 16% on Kinetics. Compared to the pose model only of Zolfaghari et al. (2017), we obtain a higher accuracy on JHMDB, HMDB51 and UCF101. On NTU and PennAction, Luvizon et al. (2018) obtain a higher accuracy because their approach also leverages appearance features. When combining SIP-Net with a standard RGB stream using 3D ResNeXt-101 backbone, we obtain 98.9% on PennAction. Finally, as in (Yan et al. 2018), we run STGCN code on 2D poses detected by OpenPose (Cao et al. 2018). We significantly outperform this approach on JHMDB, PennAction, HMDB51 and UCF101. On Kinetics, the gap is much smaller, with only 2%. This dataset contains many videos with very near close-ups on faces or captured from a first-person viewpoint, which leads to a large number of misdetections by LCR-Net++ that has not been trained in such conditions. For videos where only the face is visible, OpenPose that outputs 18 keypoints including 5 on the head (nose, two ears, two eyes) is able to detect a pose. In contrast, LCR-Net++ that estimates only 1 (out of 13) keypoint on the center of the head, fails to detect humans in such cases. Table 5 shows the 10 classes with the highest and lowest accuracy for SIP-Net. Classes with high top-1 accuracy can be clearly recognized from body pose only. In contrast, the classes at 0% are either actions often captured in first-person viewpoint where the poses are not detected (*making a cake*), or classes with no motion of the body keypoint as they mainly contain motion of the face (*sniffing*) or the hands (*drumming fingers*).

Table 4 Mean accuracies (in %) for our three baselines on all datasets, and for state-of-the-art pose-based approaches

	JHMDB-1	JHMDB	PennAction	NTU (cs)	HMB51-1	HMDB51	UCF101-1	UCF101	Kinetics
PoTion (Choutas et al. 2018)	59.1	57.0	-	-	46.3	43.7	60.5	65.2	16.6
Zolfaghari et al. (2017) (pose only)	45.5	-	-	67.8	36.0	-	56.9	-	-
MultiTask (Luvizon et al. 2018) (uses RGB)	-	-	**97.4**	74.3	-	-	-	-	-
STGCN (Yan et al. 2018) (OpenPose)	25.2	25.4	71.6	**79.8**	38.6	34.7	54.0	50.6	30.7
STGCN2D	23.2	23.2	85.5	69.4	36.5	32.7	49.2	44.4	11.9
STGCN3D	53.1	50.5	89.2	75.0	39.8	41.0	48.5	51.1	10.6
SIP-Net	**66.4**	**62.4**	93.5	64.8	**50.7**	**51.2**	**66.1**	**66.0**	**32.8**

Table 5 Classes with the highest/lowest accuracy (in %) for SIP-Net on Kinetics

highest top-1 accuracy		lowest top-1 accuracy	
crawling baby	91.8	rock scissors paper	0.0
presenting weather forecast	90.0	throwing ball	0.0
riding mechanical bull	89.8	eating chips	0.0
deadlifting	88.9	drumming fingers	0.0
surfing crowd	87.5	tossing coin	0.0
arm wrestling	87.5	sniffing	0.0
filling eyebrows	84.4	unloading truck	0.0
shearing sheep	83.7	holding snake	0.0
bench pressing	82.0	making a cake	0.0
front raises	81.6	ripping paper	0.0

5 Experiments on mimed actions

To assess the bias of action recognition algorithms towards scenes and objects, and evaluate their generalizability in absence of such visual context, we introduce **Mimetics**, a dataset of mimed actions.

5.1 The Mimetics dataset

Mimetics contains short YouTube video clips of mimed human actions that mostly consist in manipulations of, or interactions with certain objects. These include sport actions, such as *playing tennis* or *juggling a soccer ball*, daily activities such as *drinking*, personal hygiene, *e.g. brushing teeth*, or playing musical instruments including *bass guitar*, *accordion* or *violin*. These classes were selected from the action labels of the Kinetics dataset, allowing to evaluate models trained on Kinetics. Mimetics contains 713 video clips for a subset of 50 human action classes, *i.e.*, an average of 14.3 clips per class. As it is hard to find mimed actions on the web, we restrict Mimetics to testing purposes, not for training. These actions are performed on stage or on the street by mime artists (middle row of Figure 1) but also in everyday life of people, typically during mime games, or captured and shared for fun on social media. For instance, the top row of Figure 1 shows a video of someone training indoor for *surfing water* or the bottom row shows soccer players mimicking the action *bowling* to celebrate a goal.

Finding videos of mimed action on YouTube is a very difficult task. The clips for each class were obtained by searching for candidates through the use of key words such as *miming* or *imitating* followed by the desired action, or using query words such as *imaginary* and *invisible* followed by a certain object category. We queried these keywords using several different languages but this was not enough to ensure a sufficient number of instances for all the considered action classes. To complete the dataset, we also watched hours of videos looking for interesting mimed actions and identifying the clips of interest. In comparison, datasets like Kinetics use a semi-automatic process using a frame-level classifier to prune the videos, this was not possible for mimed actions as the classifiers fail. Some classes had to be dropped due to the lack of videos. The dataset was built making sure that a human observer was able to recognize the mimed actions. The videos have variable resolutions and frame rates and have been manually trimmed between 1 and 10 seconds, following the Kinetics dataset. The URLs of the original YouTube videos and the temporal intervals of the video clips have been shared to spur further research on this topic. The detailed list of classes with the number of videos per class is available in Appendix B.

5.2 Experimental results

We compare several approaches on the Mimetics dataset: our three pose-based baselines, a state-of-the-art 3D CNN method on RGB input or Flow input as well as their late fusion, in addition to STGCN (Yan et al. 2018) with OpenPose. For optical flow input, we use the TV-L1 algorithm (Zach et al. 2007). All methods were trained on the 400 classes of Kinetics. We then run them on the videos from the Mimetics dataset, and report top-1, top-5 accuracies as well as the mean average-precision (mAP). As each video has a single label, average-precision computes for each class the inverse of the rank of the ground-truth label, averaged over all videos of this class. Overall performances are reported in Table 6. We refer to Appendix B for per-class results. Figure 5 shows some qualitative examples.

We first observe that the performance is relatively low for all methods, below 15% top-1 accuracy and 25% mAP, showing that the recognition of mimed actions is challenging. In fact, all methods completely fail

Table 6 Mean top-k accuracies and mean average-precision (in %) on the Mimetics dataset when training on Kinetics

	top-1	top-5	mAP
RGB (3D-ResNeXt-101)	8.6	20.1	15.6
Flow (3D-ResNeXt-101)	11.8	29.6	21.1
RGB+Flow (late fusion)	10.5	26.9	19.1
STGCN (OpenPose)	12.6	27.4	20.7
STGCN2D	9.0	20.5	15.4
STGCN3D	5.8	13.8	11.3
SIP-Net	**14.2**	**32.0**	**22.7**

for a certain number of actions including *climbing a rope*, *reading newspaper*, *eating cake* or, more surprisingly, *sweeping floor*. One reason for this overall low accuracy is that some Kinetics actions are fine-grained (*e.g.* different classes correspond to *eating* various types of food) and are hard to distinguish, especially when mimed. Another difficulty is that mimed actions tend to be exaggerated, some in a comical way but also to virtually represent the object. This is particularly true when they are performed by mime artists. For instance, in the *reading newspaper* sequence of Figure 5, the artist exaggerates the movements of the head to make people understand that he/she is reading. These gestures are consequently not aligned with real performances of the actions as observed in the training videos. Interestingly, a person who has never seen a mime before is still capable of understanding what is happening and so should an intelligent system. We manually label a flag for each video whether the actor is a mime artist or not, and show the global top-1 accuracy in Table 7. For all approaches, the performance is significantly lower on videos where actions are performed by mime artists compared to standard people.

The best overall performance is achieved by SIP-Net which consists of a temporal convolution applied on pose features, reaching 14.2% top-1 accuracy and a mAP of 22.7%. Some failure cases occur when several people are present in the scene. The tubes can erroneously mix several individuals or other persons (*e.g.* spectators) sometimes obtain higher scores than the one miming the action of interest.

In comparison, state-of-the-art 3D CNN model trained on RGB clips performs more poorly, with 8.6% mean top-1 accuracy and 15.6 mAP. For some classes such as *archery*, *playing accordion*, *playing bass guitar*, *playing trumpet*, this state-of-the-art RGB model obtains 0% while SIP-Net performs decently. One key reason for that is the bias learned by the model: it focuses on the objects being manipulated or the scenes where the video is captured more than on the performed actions. For instance, in the second row of Figure 5, someone mimics *playing piano* on a console table covered with a tablecloth, which looks like a massage table. As a consequence, the RGB model predicts the action *massage back* without considering what the person is really doing. To further verify the bias towards object and scene, we manually label for each video if there is any relevant object or not, and if the scene is relevant for the action. We report the global top-1 accuracy (as some classes have no video or just a few, global accuracy is better suited than mean per-class accuracy) in Table 7 for the subset of videos where there is no relevant object, where the scene is not relevant or both. On these videos, the performance of the state-of-the-art RGB 3D CNN significantly drops while the SIP-Net baseline is more robust. RGB 3D CNN still performs better than SIP-Net on classes such as *brushing teeth*, *catching or throwing baseball*, or *juggling balls*. This corresponds to classes in which the object is barely visible in most training videos, either too small (*e.g.* cigarette for *smoking*) or mostly occluded by hands (baseball ball, toothbrush, hair brush). In such cases, 3D CNN model focuses on face and hands (for *brushing teeth*, *smoking*) or on the body (*throwing baseball*) and therefore performs reasonably well on these mimed actions. To further verify this, we manually annotate for each of the 50 classes of Mimetics whether there is an object being manipulated or not, and if it is small or large. We report the global top-1 accuracy in Table 8. RGB performs better than SIP-Net on actions with no object or with small objects, while SIP-Net clearly outperforms RGB in case of large objects.

We then also evaluate a similar 3D CNN that takes as input optical flow clips instead of RGB clips. The overall performance is higher than RGB, with 11.8% top-1 accuracy and 21.1% mAP. This suggests that this flow model learns less biases than RGB, because it does not see the appearance of the scenes and objects. For instance, *playing piano* is correctly predicted in the example of the second row of Figure 5, because from the optical flow, a piano and a covered table roughly look the same. Sevilla-Lara et al. (2018) suggest that flow

Table 7 Global top-1 accuracy (in %) on various subsets of Mimetics for models trained on Kinetics

	(#vid.)	RGB	Flow	SIP-Net
all videos	(713)	8.4	11.5	**14.3**
mime artist	(203)	4.9	**6.4**	5.4
not a mime artist	(510)	9.8	13.5	**17.8**
no object is relevant	(644)	6.8	9.8	**13.4**
scene is not relevant	(644)	6.4	9.8	**13.5**
no object is relevant, and scene is not relevant	(584)	4.5	8.0	**12.7**

Table 8 Global top-1 accuracy (in %) for classes with no/small object and with large objects

	#cls.	(#vid.)	RGB	Flow	SIP-Net
all classes	50	(713)	8.4	11.5	**14.3**
no/small object	19	(268)	11.2	**12.3**	9.0
large object	31	(445)	6.7	11.0	**17.5**

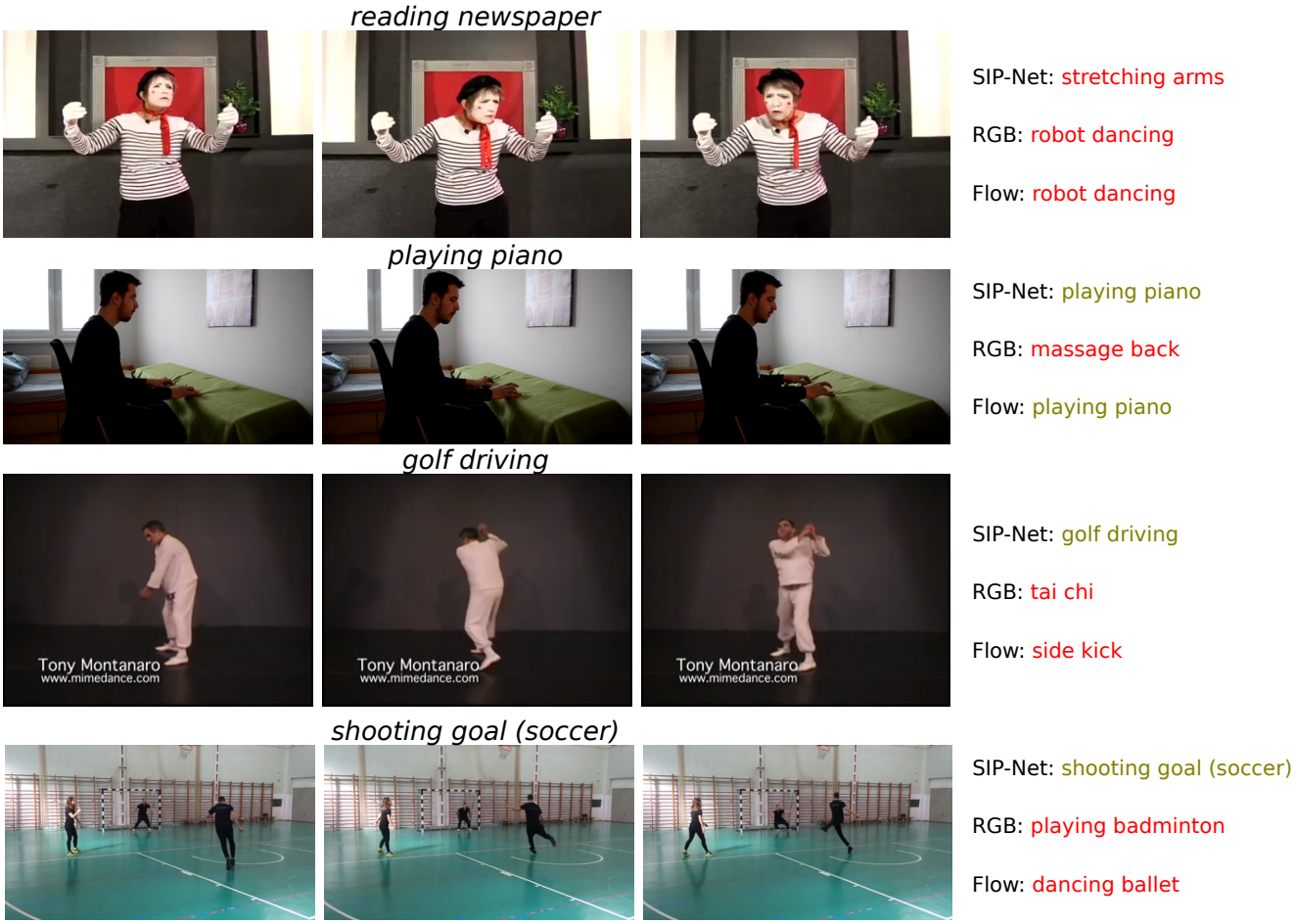

Fig. 5 Four video examples from Mimetics with the highest scored class for the SIP-Net, RGB 3D CNN and Flow 3D CNN

may still capture global shape of the actor or objects. This explains why flow performs better on classes without object or with small objects compared to larger objects, see Table 8, as RGB does: when the subject is manipulating small objects, the network is not able to capture these details and it focuses on bigger structure like the person, thus generalizing better to out-of-context actions. We evaluated in Table 6 a late fusion of RGB and Flow, *i.e.*, a two-stream model (Simonyan and Zisserman 2014), and observe a small decrease of performance as both models tend to perform well on the same kind of classes and videos.

Next, we also benchmark other pose-based approaches. Our two baselines based on explicit 2D or 3D poses perform quite poorly, comparably to their respective performance on the Kinetics dataset. This can be explained by the difficulty to extract accurate body keypoint coordinates for videos in-the-wild with abrupt camera and actor motion, blur, and occlusions. In particular, the low performance on Kinetics itself suggests this occurs also in the training set, leading to a poor model. We also compare to STGCN (Yan et al. 2018) that uses OpenPose to estimate the pose, *i.e.*, with more keypoint on the head than LCR-Net++. The performance is higher with 12.6% top-1 accuracy but remains lower than the SIP-Net baseline that does not explicitly compute poses but transfers the learned pose features to action recognition.

To explain the relatively poor performance of all methods, we argued that Kinetics classes might be too fine-grained and too difficult to distinguish when mimed. This is illustrated by the significantly higher top-5 accuracy (32.0%) than top-1 accuracy (14.2%), see Table 6. To further verify this statement, we trained a SIP-Net model on the Kinetics training videos from the 50 classes of Mimetics and report the results in Table 9. Top-1 accuracy increases to more than 25% and top-5 accuracy to more than 50%.

6 Conclusion

In this paper, we have highlighted the context biases of existing action recognition datasets and 3D CNN models. To benchmark performances on out-of-context ac-

Table 9 Mean top-k accuracies and mAP (in %) of SIP-Net on the Mimetics dataset when training on the full Kinetics training set, or on the subset of classes from Mimetics

training set	top-1	top-5	mAP
Kinetics (400 classes)	14.2	32.0	22.7
Kinetics subset (50 classes)	**25.1**	**51.4**	**38.3**

tions, we have introduced the Mimetics dataset. Our experiments show that models leveraging body language via human pose are less prone to the context biases. Applying a shallow neural network such as a single convolution over features transferred from human poses performs surprisingly well compared to 3D action recognition applied in-the-wild. Our analysis shows that using a sparse set of keypoints might not be sufficient to distinguish some fine-grained actions. Using a more complete representation of human poses including full-body, hands, and face dense pose information, as predicted by recent works in human pose/shape estimation could significantly increase the performance.

We think that our new Mimetics benchmark will allow to better understand what action recognition models learn and is a step towards designing more intelligent systems. We hope it will stimulate research into the particular challenges of out-of-context action recognition. Estimating the performance of future action recognition methods on our Mimetics dataset could help bringing additional analysis on their similarity to human performance but it could also help evaluate their capability to detect/ignore mimes. Ideally, an action recognition system should solve both the problems of recognizing a human action, and identifying whether it is mimicked or not (fake or real). Our work also allows to make a step toward this goal by showing how much state-of-the-art action recognition systems can be fooled by mimes. This particularly occurs when the context is partial, see the second case in Figure 5 classified as 'massage back'. The mistakes made by these models in such scenarios are harmful for their real-life deployments.

References

Angelini F, Fu Z, Long Y, Shao L, Naqvi SM (2018) ActionXPose: A Novel 2D Multi-view Pose-based Algorithm for Real-time Human Action Recognition. arXiv preprint arXiv:181012126 3

Bahng H, Chun S, Yun S, Choo J, Oh SJ (2019) Learning debiased representations with biased representations. arXiv 4

Cao C, Zhang Y, Zhang C, Lu H (2016) Action recognition with joints-pooled 3D deep convolutional descriptors. In: IJCAI 3

Cao Z, Hidalgo G, Simon T, Wei SE, Sheikh Y (2018) OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In: arXiv preprint arXiv:1812.08008 4, 6

Carreira J, Zisserman A (2017) Quo vadis, action recognition? A new model and the Kinetics dataset. In: CVPR 1, 3

Chéron G, Laptev I, Schmid C (2015) P-CNN: Pose-based CNN features for action recognition. In: ICCV 3

Choutas V, Weinzaepfel P, Revaud J, Schmid C (2018) Potion: Pose motion representation for action recognition. In: CVPR 3, 6, 7

Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T (2015) Long-term recurrent convolutional networks for visual recognition and description. In: CVPR 3

Du W, Wang Y, Qiao Y (2017) RPAN: An end-to-end recurrent pose-attention network for action recognition in videos. In: ICCV 3

Du Y, Fu Y, Wang L (2015a) Skeleton based action recognition with convolutional neural network. In: ACPR 3

Du Y, Wang W, Wang L (2015b) Hierarchical recurrent neural network for skeleton based action recognition. In: CVPR 2, 3, 4

Feichtenhofer C, Pinz A, Zisserman A (2016) Convolutional two-stream network fusion for video action recognition. In: CVPR 3

Feichtenhofer C, Fan H, Malik J, He K (2019) Slowfast networks for video recognition. In: ICCV 1, 3

Girdhar R, Ramanan D (2017) Attentional pooling for action recognition. In: NIPS 3

Gkioxari G, Malik J (2015) Finding action tubes. In: CVPR 5

Hara K, Kataoka H, Satoh Y (2018) Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet. In: CVPR 1, 3

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: CVPR 5

Iqbal U, Garbade M, Gall J (2017) Pose for action-action for pose. In: International Conference on Automatic Face & Gesture Recognition (FG) 3

Jacquot V, Ying Z, Kreiman G (2020) Can deep learning recognize subtle human activities. In: CVPR 2

Jhuang H, Gall J, Zuffi S, Schmid C, Black MJ (2013) Towards understanding action recognition. In: ICCV 3, 6

Kalogeiton V, Weinzaepfel P, Ferrari V, Schmid C (2017) Action tubelet detector for spatio-temporal action localization. In: ICCV 5

Kay W, Carreira J, Simonyan K, Zhang B, Hillier C, Vijayanarasimhan S, Viola F, Green T, Back T, Natsev P, et al. (2017) The Kinetics human action video dataset. arXiv preprint arXiv:170506950 1, 3, 4, 6

Khosla A, Zhou T, Malisiewicz T, Efros AA, Torralba A (2012) Undoing the damage of dataset bias. In: ECCV 4

Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T (2011) HMDB: a large video database for human motion recognition. In: ICCV 3, 6

Li Y, Vasconcelos N (2019) Repair: Removing representation bias by dataset resampling. In: CVPR 2, 3, 4

Li Y, Li Y, Vasconcelos N (2018) Resound: Towards action recognition without representation bias. In: ECCV 2, 3, 4

Liu J, Shahroudy A, Xu D, Wang G (2016) Spatio-temporal LSTM with trust gates for 3D human action recognition. In: ECCV 2, 3, 4

Liu M, Yuan J (2018) Recognizing human actions as the evolution of pose estimation maps. In: CVPR 3

Luvizon DC, Picard D, Tabia H (2018) 2D/3D pose estimation and action recognition using multitask deep learning. In: CVPR 3, 4, 6, 7

McNally W, Wong A, McPhee J (2019) STAR-Net: Action recognition using spatio-temporal activation reprojection. In: CRV 3

Mehta D, Sridhar S, Sotnychenko O, Rhodin H, Shafiei M, Seidel HP, Xu W, Casas D, Theobalt C (2017) VNect: Real-time 3D human pose estimation with a single RGB camera. ACM Transactions on Graphics 2

Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: Towards real-time object detection with region proposal networks. In: NIPS 5

Rogez G, Weinzaepfel P, Schmid C (2019) LCR-Net++: Multi-person 2D and 3D pose detection in natural images. IEEE trans PAMI 2, 4, 5

Saha S, Singh G, Sapienza M, Torr PH, Cuzzolin F (2016) Deep learning for detecting multiple space-time action tubes in videos. In: BMVC 12

Sevilla-Lara L, Liao Y, Güney F, Jampani V, Geiger A, Black MJ (2018) On the integration of optical flow and action recognition. In: GCPR 8

Shahroudy A, Liu J, Ng TT, Wang G (2016) NTU RGB+D: A large scale dataset for 3D human activity analysis. In: CVPR 2, 6

Si C, Jing Y, Wang W, Wang L, Tan T (2018) Skeleton-based action recognition with spatial reasoning and temporal stack learning. In: ECCV 3

Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. In: NIPS 1, 3, 9

Singh G, Saha S, Sapienza M, Torr PH, Cuzzolin F (2017) Online real-time multiple spatiotemporal action localisation and prediction. In: ICCV 5

Soomro K, Zamir AR, Shah M (2012) UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. In: CRCV-TR-12-01 3, 6

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: CVPR 3

Torralba A, Efros AA, et al. (2011) Unbiased look at dataset bias. In: CVPR 4

Tran D, Bourdev L, Fergus R, Torresani L, Paluri M (2015) Learning spatiotemporal features with 3D convolutional networks. In: ICCV 3

Tran D, Wang H, Torresani L, Ray J, LeCun Y, Paluri M (2018) A closer look at spatiotemporal convolutions for action recognition. In: CVPR 1, 3

Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L (2016) Temporal segment networks: Towards good practices for deep action recognition. In: ECCV 1

Wang W, Zhang J, Si C, Wang L (2018) Pose-based two-stream relational networks for action recognition in videos. arXiv preprint arXiv:180508484 3

Weinzaepfel P, Harchaoui Z, Schmid C (2015) Learning to track for spatio-temporal action localization. In: ICCV 12

Weng J, Liu M, Jiang X, Yuan J (2018) Deformable pose traversal convolution for 3d action and gesture recognition. In: ECCV 3

Xie S, Girshick R, Dollár P, Tu Z, He K (2017) Aggregated residual transformations for deep neural networks. In: CVPR 4

Xie S, Sun C, Huang J, Tu Z, Murphy K (2018) Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In: ECCV 3

Yan S, Xiong Y, Lin D (2018) Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI 3, 4, 5, 6, 7, 9, 13, 14

Yao A, Gall J, van Gool L (2012) Coupled action recognition and pose estimation from multiple views. IJCV 4

Zach C, Pock T, Bischof H (2007) A duality based approach for realtime TV-L1 optical flow. In: Joint pattern recognition symposium 7

Zhang W, Zhu M, Derpanis KG (2013) From actemes to action: A strongly-supervised representation for detailed action understanding. In: ICCV 6

Zhu J, Zou W, Xu L, Hu Y, Zhu Z, Chang M, Huang J, Huang G, Du D (2018) Action machine: Rethinking action recognition in trimmed videos. arXiv preprint arXiv:181205770 3

Zhu W, Lan C, Xing J, Li Y, Shen L, Zeng W, Xie X (2016) Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In: AAAI 2, 3, 4

Zolfaghari M, Oliveira GL, Sedaghat N, Brox T (2017) Chained multi-stream networks exploiting pose, motion, and appearance for action classification and detection. In: ICCV 3, 6, 7

A Extended experiments on existing datasets

In this section, we provide more analysis about the performance of the three pose-based baselines on existing action recognition datasets. We first perform a parametric study of SIP-Net in Section A.1. We then use the various levels of ground-truth (see Table 3 of the main paper) to study the impact of using ground-truth or extracted tubes and poses (Section A.2).

Tubes. For datasets with ground-truth 2D poses, we compare the performance when using ground-truth tubes (GT Tubes) obtained from GT 2D poses, or estimated tubes (LCR Tubes) built from estimated 2D poses, see Section 4.1 of the main paper. In the latter case, tubes are labeled positive if the spatio-temporal IoU with a GT tube is over 0.5, and negative otherwise. When there is no tube annotation, we assume that all tubes are labeled with the video class label. Note that in some videos, no tube is extracted, in which case the videos are ignored when training, and considered as wrongly classified for test videos. In particular, this happens when only the head is visible, as well as for many clips with first person viewpoint, where only one hand or the main manipulated object is visible. We obtain no tube for 0.1% of the videos on PennAction, 2.5% on JHMDB, 2.7% on HMDB51, 6.7% on UCF101 and 15.3% on Kinetics.

A.1 SIP-Net baseline

We first present the results for the SIP-Net baseline with GT tubes (blue curve 'GT tubes, Pose Feats') and LCR tubes (green curve 'LCR Tubes, Pose Feats') on all datasets for varying clip length T, see Figure 6. Overall, a larger clip size T leads to a higher classification accuracy. This is in particular the case for datasets with longer videos such as NTU and Kinetics. This holds both when using GT tubes (blue curve) and LCR tubes (green curve). We keep T=32 in the remaining of this paper.

Next, we measure the impact of applying transfer learning from the pose domain to action recognition. To this end, we compare the temporal convolution on LCR pose features (blue curve, 'Pose Feats'), to features extracted from a Faster R-CNN model with ResNet50 backbone trained to classify actions (red curve, 'Action Feats'). This latter method is not supposed to be state-of-the-art in action recognition, but it allows to fairly compare the pose features to action features, keeping the network architecture exactly the same, simply changing the learned weights. Note that such a frame-level action detector has been used in the spatio-temporal action detection literature (Saha et al. 2016; Weinzaepfel et al. 2015), before the rise of 3D CNNs. Results in Figure 6 show a clear drop of accuracy when using action features instead of pose features: about 20% on JHMDB-1 and PennAction, and around 5% on NTU for T=32. Interestingly, this holds for T=1 on HMDB-1 and PennAction, *i.e.*, without temporal integration, showing that 'Pose feats' are more powerful. To better understand why using pose features considerably increases performance compared to action features, we visualize the distances between features inside tubes in Figure 7. When training a per-frame detector specifically for actions, most features of a given tube are correlated. It is therefore hard to leverage temporal information from them. In contrast, LCR-Net++ pose features considerably change over time, as does the pose, deriving greater benefit from temporal integration. Figure 8 shows confusion matrices on PennAction when

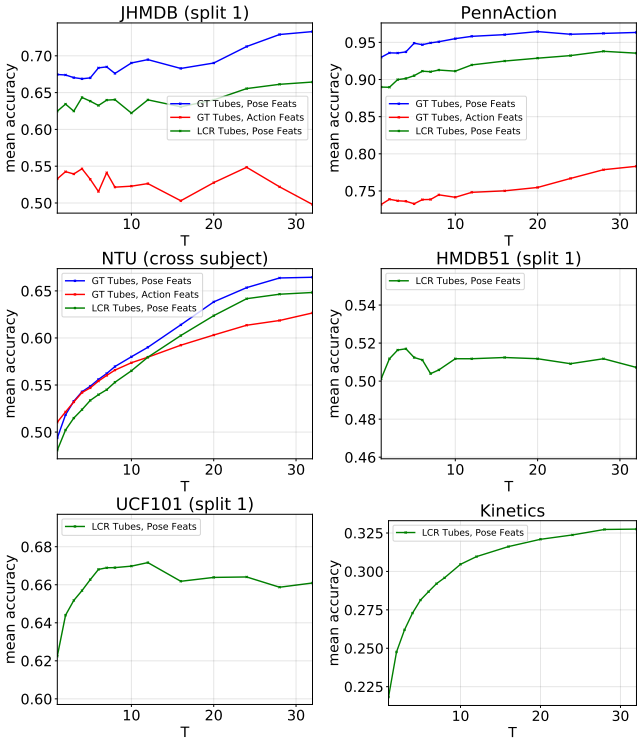

Fig. 6 Mean-accuracy of SIP-Net for varying T on all datasets, for different tubes (GT or LCR) and features (Pose or Action).

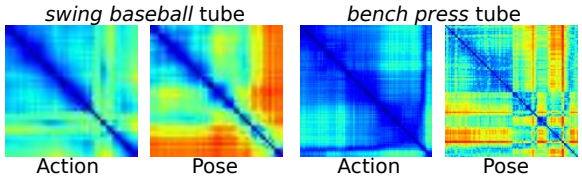

Fig. 7 Feature correlation for two different videos of PennAction (action *swing baseball* on the left and *bench press* on the right). For each sequence, we show the distances between features along the tube when using Faster R-CNN action or LCR pose features. Blue/red color indicates low/high distances and therefore high/low correlation. Implicit pose features clearly show more variation inside a tube.

using 'Pose feats' (left) *vs*. 'Action feats' (right). With 'Action feats', confusions happen between the two *tennis* or the two *baseball* actions, while this is disambiguated with 'Pose feats'.

A.2 Comparison between baselines

We compare the performance of the baselines using GT and LCR tubes, on the JHMDB-1, PennAction and NTU datasets in Table 10. On JHMDB-1 and PennAction, despite being a much simpler architecture, the SIP-Net baseline outperforms the methods based on explicit 2D-3D pose representations, both with GT and LCR tubes. Estimated 3D pose sequences are usually noisy and may lack temporal consistency. We also observe that the STGCN3D approach significantly outperforms its 2D counterpart (STGCN2D), confirming that 2D

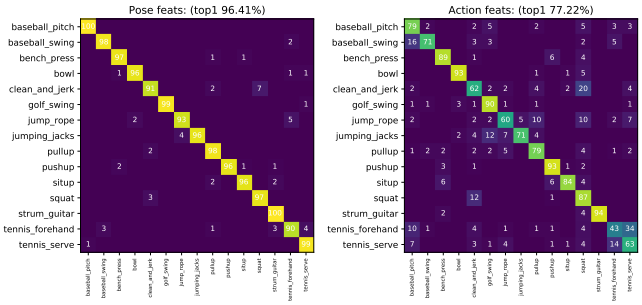

Fig. 8 Confusion matrices on PennAction when using action features (left) and pose features (right) in SIP-Net.

Table 10 Mean accuracies (in %) for our three baselines on datasets with GT tubes, or when using LCR tubes

Method	Tubes	JHMDB-1	PennAction	NTU (cs)
STGCN2D	GT	34.9	90.8	66.3
STGCN3D		57.9	94.7	**74.8**
SIP-Net		**73.3**	**96.3**	66.4
STGCN2D	LCR	23.2	85.5	69.4
STGCN3D		53.1	89.2	**75.0**
SIP-Net		**66.4**	**93.5**	64.8
STGCN3D (GT 3D poses)		-	-	81.5

poses contain less discriminative and more ambiguous information.

On the NTU dataset, the 3D pose baseline obtains 74.8% accuracy when using GT tubes and estimated poses (STGCN3D on GT Tubes), compared to 81.5% reported in (Yan et al. 2018) when using ground-truth 3D poses. This gap of 7% in a constrained environment is likely to increase for videos captured in the wild. The performance of the features-based baseline (SIP-Net) is lower, 66.4% on GT tubes, suggesting than SIP-Net performs better only in unconstrained scenarios.

B Per-class results on Mimetics

In Table 11, we present for each class the top-1 accuracy and the AP of the different methods. For the top-1 accuracy metric, SIP-Net obtains the best performance for 19 out of 50 classes, with a mean accuracy of 14.2%. The RGB 3D CNN baseline obtains the highest AP for 8 classes, which often correspond to classes in which manipulated objects are small, making the network less bias towards context (*e.g.* the ball for the action *catching of throwing baseball*). Table 11 also highlights that the recognition of mimed actions is a very challenging and open task, as none of the videos are correctly classified (*i.e.* 0% top-1 accuracy) by the 5 baselines for 5 out of the 50 classes.

Table 11 Per-class results on the Kinetics for the RGB and Flow 3D CNN baselines, for STGCN (Yan et al. 2018) (2D with OpenPose), as well as our three baselines (STGCN2D, STGCN3D, SIP-Net). In each column, the first number is the top-1 accuracy per class (in %), and the number in parenthesis is the AP (in %).

class	#vid	RGB		Flow		STGCN		STGCN2D		STGCN3D		SIP-Net	
archery	19	0.0	(3.4)	0.0	(2.6)	5.3	(11.3)	0.0	(9.5)	0.0	(10.7)	**36.8**	**(42.0)**
bowling	13	**15.4**	(16.8)	**15.4**	**(21.1)**	0.0	(2.4)	0.0	(4.9)	0.0	(3.4)	7.7	(15.9)
brushing hair	20	15.0	(23.6)	**25.0**	**(39.6)**	0.0	(8.6)	0.0	(2.7)	0.0	(1.0)	0.0	(7.5)
brushing teeth	15	40.0	(45.4)	**53.3**	**(62.8)**	13.3	(24.4)	0.0	(1.7)	0.0	(1.9)	6.7	(25.3)
canoeing or kayaking	14	0.0	(1.5)	0.0	(5.3)	0.0	(2.6)	0.0	(2.8)	0.0	**(8.2)**	0.0	(3.9)
catching or throwing baseball	14	**21.4**	**(27.0)**	0.0	(22.9)	0.0	(9.2)	0.0	(5.9)	0.0	(2.6)	0.0	(17.6)
catching or throwing frisbee	14	**21.4**	(31.5)	**21.4**	**(42.7)**	7.1	(28.1)	0.0	(10.3)	0.0	(6.7)	**21.4**	(39.5)
clean and jerk	13	15.4	(25.3)	38.5	(47.7)	**46.2**	**(52.3)**	23.1	(43.0)	30.8	(47.5)	**46.2**	(50.1)
cleaning windows	16	**12.5**	**(17.0)**	6.2	(11.0)	6.2	(8.6)	0.0	(1.2)	0.0	(1.2)	0.0	(3.6)
climbing a rope	14	0.0	(1.2)	0.0	(1.1)	0.0	**(9.5)**	0.0	(6.2)	0.0	(4.8)	0.0	(5.1)
climbing ladder	13	0.0	(1.8)	0.0	(5.4)	**7.7**	**(11.4)**	0.0	(1.2)	0.0	(1.8)	0.0	(2.1)
deadlifting	11	36.4	(52.8)	45.5	(64.9)	36.4	(55.2)	54.5	(69.2)	45.5	(67.1)	**63.6**	**(75.5)**
dribbling basketball	18	5.6	(11.6)	22.2	(31.5)	50.0	(60.6)	44.4	(49.4)	**61.1**	**(67.9)**	27.8	(46.9)
drinking	27	3.7	(10.4)	0.0	(13.6)	0.0	**(13.9)**	0.0	(0.9)	0.0	(0.9)	**7.4**	(10.3)
driving car	16	0.0	(2.9)	0.0	(3.7)	**6.2**	(8.8)	0.0	(2.1)	0.0	(0.7)	**6.2**	**(9.4)**
dunking basketball	10	40.0	(55.3)	**60.0**	**(64.3)**	20.0	(28.9)	30.0	(41.8)	0.0	(6.3)	40.0	(47.9)
eating cake	19	0.0	(2.0)	0.0	**(2.9)**	0.0	(1.3)	0.0	(0.6)	0.0	(1.4)	0.0	(0.9)
eating ice cream	11	0.0	(4.7)	0.0	(11.3)	0.0	(4.4)	0.0	(2.4)	0.0	(1.8)	**18.2**	**(21.5)**
flying kite	10	**10.0**	**(14.4)**	0.0	(3.7)	0.0	(3.2)	0.0	(1.6)	0.0	(1.2)	0.0	(6.6)
golf driving	16	12.5	(19.7)	31.2	(44.0)	**62.5**	(69.5)	50.0	(57.8)	37.5	(47.5)	**62.5**	**(70.5)**
hitting baseball	15	6.7	(18.5)	13.3	(23.4)	0.0	(17.7)	**20.0**	**(34.9)**	6.7	(16.0)	**20.0**	(34.1)
hurdling	10	0.0	(13.5)	**20.0**	**(29.7)**	**20.0**	(29.2)	0.0	(9.8)	0.0	(11.0)	10.0	(23.3)
juggling balls	12	33.3	(40.9)	25.0	(39.7)	33.3	(53.0)	**58.3**	**(60.3)**	25.0	(35.5)	16.7	(32.6)
juggling soccer ball	18	11.1	(23.9)	5.6	(25.5)	**50.0**	**(61.6)**	0.0	(12.1)	27.8	(41.8)	44.4	(57.5)
opening bottle	9	0.0	(1.7)	0.0	(4.7)	**11.1**	**(13.8)**	0.0	(1.0)	0.0	(0.8)	0.0	(6.9)
playing accordion	11	0.0	(4.7)	9.1	(18.5)	9.1	(11.2)	0.0	(6.7)	0.0	(5.0)	**27.3**	**(36.1)**
playing basketball	14	7.1	(21.6)	**14.3**	**(35.5)**	0.0	(27.9)	7.1	(23.6)	0.0	(7.3)	0.0	(10.5)
playing bass guitar	13	0.0	(5.2)	7.7	(12.4)	7.7	(20.3)	0.0	(6.0)	0.0	(3.2)	**15.4**	**(27.7)**
playing guitar	18	**5.6**	(9.2)	**5.6**	(12.8)	**5.6**	(14.4)	0.0	(3.1)	0.0	(1.1)	**5.6**	**(14.9)**
playing piano	17	0.0	(9.6)	11.8	(18.7)	**17.6**	**(19.6)**	0.0	(6.8)	5.9	(11.2)	11.8	(13.5)
playing saxophone	13	0.0	(2.7)	0.0	(6.3)	**7.7**	(9.1)	0.0	(3.7)	0.0	(4.0)	0.0	**(14.2)**
playing tennis	19	5.3	(7.9)	10.5	(15.1)	21.1	(35.0)	**31.6**	**(45.5)**	5.3	(20.4)	21.1	(34.5)
playing trumpet	14	0.0	(8.0)	21.4	(25.1)	7.1	(14.0)	0.0	(14.2)	0.0	(12.7)	**35.7**	**(47.2)**
playing violin	20	10.0	(15.3)	10.0	(26.0)	5.0	(15.2)	**25.0**	**(37.5)**	**25.0**	(34.7)	**25.0**	(36.5)
playing volleyball	13	30.8	(44.4)	7.7	(28.3)	**38.5**	**(52.9)**	0.0	(5.3)	0.0	(4.5)	7.7	(18.4)
punching person (boxing)	16	12.5	(22.8)	18.8	(30.3)	**25.0**	**(31.3)**	6.2	(19.8)	0.0	(8.8)	12.5	(20.3)
reading book	10	0.0	(1.8)	0.0	(6.0)	0.0	(3.5)	0.0	(2.1)	0.0	(2.3)	**10.0**	**(17.9)**
reading newspaper	10	0.0	(2.3)	0.0	(1.1)	0.0	(1.2)	0.0	(0.7)	0.0	(0.5)	0.0	**(3.1)**
shooting basketball	19	**5.3**	(15.4)	**5.3**	**(20.2)**	**5.3**	(11.3)	**5.3**	(19.2)	0.0	(3.4)	**5.3**	(13.4)
shooting goal (soccer)	14	7.1	(23.9)	0.0	(21.2)	7.1	(22.6)	7.1	(24.3)	0.0	(10.0)	**14.3**	**(29.8)**
skiing (not slalom or crosscountry)	10	0.0	(4.1)	**20.0**	**(23.0)**	0.0	(1.5)	0.0	(1.1)	0.0	(1.4)	0.0	(2.0)
skiing slalom	10	0.0	(5.5)	0.0	(1.5)	0.0	(0.6)	10.0	(13.3)	**20.0**	**(20.8)**	10.0	(15.8)
skipping rope	12	41.7	(53.6)	41.7	(58.5)	**75.0**	**(83.3)**	**75.0**	(81.3)	0.0	(8.8)	50.0	(61.6)
smoking	19	0.0	(8.5)	**5.3**	**(14.8)**	0.0	(7.2)	0.0	(2.0)	0.0	(1.5)	**5.3**	(13.8)
surfing water	10	0.0	**(6.9)**	0.0	(2.9)	0.0	(6.6)	0.0	(4.2)	0.0	(3.6)	0.0	(2.6)
sweeping floor	11	0.0	(1.9)	0.0	(1.7)	0.0	(1.4)	0.0	(1.0)	0.0	**(3.0)**	0.0	(0.9)
sword fighting	17	0.0	(15.2)	**17.6**	**(36.1)**	11.8	(25.2)	0.0	(7.2)	0.0	(5.4)	0.0	(10.2)
tying tie	8	0.0	(7.3)	0.0	(7.4)	**12.5**	**(21.5)**	0.0	(6.2)	0.0	(0.8)	0.0	(13.4)
walking the dog	15	**6.7**	**(11.7)**	0.0	(4.9)	0.0	(2.8)	0.0	(1.6)	0.0	(2.0)	0.0	(3.7)
writing	13	0.0	(1.8)	0.0	(3.0)	0.0	(2.9)	0.0	(2.6)	0.0	(0.9)	**15.4**	**(18.5)**
avg (50 classes)	713	8.6	(15.6)	11.8	(21.1)	12.6	(20.7)	9.0	(15.4)	5.8	(11.3)	**14.2**	**(22.7)**