https://doi.org/10.2352/EI.2026.38.16.AVM-100

A Comparative Analysis of Video- and Pose-Based Action Recognition for In-Cabin Driver Monitoring

Lukas Brunner and Dominik Schörkhuber
Faculty of Informatics, TU Wien, 1040 Vienna, Austria

Abstract

We present a comparative study of pose-based vs. video-based Human Action Recognition (HAR) methods for driver monitoring in car cockpits. In this context, comparisons of neural network architectures from the field of deep learning-based video understanding are scarce. However, pose- and video-based HAR has significant potential for advanced driver-assistance systems in semi-autonomous driving on public roads. We compare prediction performance, per-class false-negative rate, model size, computational requirements, and inference latency on the established Drive&Act and the proprietary Driver Action Insight datasets. While the diversity and scale of available datasets make comparisons challenging, results suggest that both approaches benefit from pretraining, but pose- and video-based techniques perform differently for specific action classes, such as those that depend on body motion or the appearance of objects.

Introduction

Advanced Driver-Assistance Systems (ADAS) have reduced damage to life and property in everyday road traffic [1]. Nevertheless, current ADAS matching Level 3 of the *SAE Levels of Driving Automation*[1] still require constant human supervision for reasons of safety and liability. Attentional fatigue is a common problem among vehicle operators [2], and Human Action Recognition (HAR) methods can provide risk detection and prevention systems with a signal that encodes the driver's attention state. However, the application in the automotive environment poses unique challenges, such as resource and latency constraints. Unlike the recent trend to scale transformer architectures to very large models, our field of study is concerned with models that can potentially run on embedded hardware.

Within the field of deep learning-based Computer Vision (CV) techniques, video-based and pose-based methods have emerged as main paradigms for the HAR task. But general reviews of HAR techniques [3]–[6] do not focus on the specific considerations relevant to this particular area of application. Comparisons between pose-based and end-to-end video-based methods seem rare in publications on in-cabin driver monitoring and are scattered throughout papers on new concrete techniques. We attempt to close this gap by exploring the following aspects:

- We compare a selection of several pose- and video-based models for HAR and report on their performance for two domain-specific in-cabin action recognition datasets.
- For the pose modality, we introduce a new classification model based on the transformer architecture.
- With respect to specific action classes, we analyze the strengths and weaknesses of each model and modality.
- In addition to classification performance, we compute each model's efficiency on the basis of Floating Point Operations (FLOPs) and prediction latency.

[1] `https://www.sae.org/blog/sae-j3016-update`

Related Work

Distracted driver recognition using neural networks has been envisioned at least since 1997 [2]. Developments in the field of HAR within the wider CV context contained valuable contributions to practical implementations of such systems. Surveys on HAR exist with varying key focus areas and levels of detail [3]–[6]. Wanli et al. [7], however, provide a broad introduction into the applied topic of distracted driver recognition. Applications in the automotive space benefit most from model architectures and pretraining done within the general HAR context. But general action recognition datasets like Kinetics-400 [8] do not sufficiently cover the automotive space.

Action Recognition Promising precursors of modern architectures for action recognition first appeared within the space of Convolutional Neural Networks (CNNs), when image models were extended to 3D input tensors [9], [10]. The transformer block gained wide recognition within the field of Natural Language Processing (NLP) [11]. The multi-head self-attention mechanism at its heart had first been employed on 2D image data (e.g. Vision Transformer (ViT) [12] and Swin Transformer [13]) before causing a spike of proposed transformer-based architectures for video understanding at the beginning of the 2020s [14]–[21]. Attempts to deal with the computational complexity of the self-attention mechanism include localized self-attention and hierarchical aggregation of features [13], [21], factorized self-attention [14], [15], and sparse matrix calculations [22]. Similarly to the transformer, the Masked Auto Encoder (MAE) framework spread from the NLP domain via the image- [23] to the video-domain [24], [25], improving computational scalability and effectiveness of the pretraining process through self-supervised learning with pretext tasks. Recently, transformer architectures have demonstrated strong results when parameter sizes are scaled to the giga range [26], leading to what is now often referred to as *foundation models* [27]. But this scaling is not feasible in the automotive space due to safety, hardware, and latency constraints. Research in our target domain rather focuses on small and efficient models. Accordingly, techniques based on Graph Convolutional Networks (GCNs) like ST-GCN [28], which consume a time series of spatial graphs that encode body poses, have received increased attention. AS-GCN [29] and STGCN++ [30] take higher-order joint neigh-

bors into account, while STGCN++ optimizes for computational efficiency. PoseConv3D [31] uses pose input, but transforms it back into a spatiotemporal tensor as a stack of Gaussian maps and then uses a 3D CNN for classification. Recently, transformer-based architectures for pose-based action recognition have been proposed [32], [33], including pretraining using a pretext task [34].

Driver Monitoring Eraqi et al. [35] used an ensemble of deep CNNs for distracted driver detection in car interiors based on single images. In the domain of timeseries data, Martin et al. [36] applied RNNs using LSTM [37] to pose-based driver action classification, focusing on adding surrounding context to the driver pose. They compare results for video data [38] using *Pseudo-3D Residual Network* [39] and *I3D* [10]. Within this line of research, Martin et al. [40] later proposed a method based on ST-GCN [28]. Furthermore, Martin et al. [41] showed the benefit of using 3D poses as a scene representation that can be decoupled from the underlying sensor platform. For single images, Ma et al. [42] devised a framework for training ViT for distraction detection under direct supervision with an additional supervision signal from a teacher model for facial expressions.

Datasets Large-scale datasets for HAR with video and pose data, such as the widely used NTU-RGB+D [43], NTU-RGB+D 120 [44], and the Kinetics datasets [8], [10], [45], are fundamentally important for establishing benchmarks for a wide variety of methods. However, they do not adequately cover domain-specific actions, which exhibit greater visual similarity. For example, Kinetics-400 [8] includes the conceptually similar classes "drawing", "brush-painting", and "spray-painting", all of which imply different motion patterns and framing of the subject. However, car interiors restrict movement, and sensor positioning remains constant, leading to visually similar scenes with only subtle differences in body movement between action classes. Wanli et al. [7] review open datasets targeted at vision-based driver-distraction analysis, of which only three (Drive&Act [38], DMD [46], and 3MDAD [47]) provide annotated video sequences for multiple distraction-related action classes in the Near Infrared (NIR) spectrum. The recent Driver Action Insight (DAI) dataset, first used by Panadero et al. [48], features a similar lighting setup as Martin et al. [38] with active illumination and an optical bandpass filter, as well as comparable views from above the middle console and co-driver side of the cockpit. However, in general, HAR datasets for driver monitoring differ in sensor positioning, video modality, or action classes and do not reach amounts comparable to large-scale datasets for general purpose HAR [46].

Method

In the following, we describe the design of our study. Consisting of the used datasets, models, and evaluation metrics. We base our experiments on Drive&Act [38] and the DAI dataset [48] due to their similar modality and views. We use two views from each dataset and unify their naming for the present work: "inner-mirror" (further referred to as "center") and "a-column co-driver" (further referred to as "right") from Drive&Act, and "center" and "right" from the DAI dataset (see Figure 1 for examples). Table 1 lists the chosen model variants for the study. Model architecture implementations and pretrained weights were obtained from the MMAction2[2] model zoo, except for VideoMAE V2 weights, which were sourced from the official implementation[3] by the original authors [25]. To complement the established ST-GCN methods [28], [30] with a transformer-based architecture, we implement our pose transformer as detailed in the respective subsection below. We analyze classification performance under the maximum likelihood rule as the decision criterion. We use Average Recall (AR) and the class-specific False Negative Rate (FNR). The class-specific FNR is calculated per tested model variant as in Equation 1, where c and c' are respectively a true and a false class, and S_c is the set of pairs of y (the true label) and $\hat{y}$ (the prediction) from the test set, where $y = c$.

$$\begin{aligned} \mathrm{FNR}(c,c') &= \frac{|\mathrm{FN}_{cc'}|}{|\mathrm{S}_c|}, \\ \mathrm{FN}_{cc'} &= \{(y,\hat{y}) \in \mathrm{S}_c | \hat{y} = c'\} \end{aligned} \tag{1}$$

For all model variants, we further report the model size by the number of parameters, the FLOPs required to process a single clip, and the GPU latency for batch sizes up to the 11GB VRAM limit of our hardware (NVIDIA® RTX™ 2080 Ti).

(a) (b) (c) (d)

Figure 1. *Examples of the DAI dataset. Center view: (a) "Adjust-radio"; Right view: (b) "Steering", (c) "Read-paper", (d) "Put-onoff-jacket". Censored for privacy reasons.*

Table 1. *List of the tested model variants.*

	Model Variant	Pretraining
video-based	VideoMAE V2 (ViT-s) [25]	distilled from VideoMAE V2 pretrained ViT-g (see [25] for details on ViT-g training)
video-based	Video Swin (small) [21]	supervised on Kinetics-400 [8] initialized from ImageNet 22k pretrained Swin [13]
video-based	Video Swin (small) [21]	—
video-based	I3D (RGB) [10]	supervised on Kinetics-400 [8]
pose-based	ST-GCN [28]	supervised on NTU RGB+D 120 (cross-subject) [44]
pose-based	ST-GCN [28]	—
pose-based	ST-GCN++ [30]	supervised on NTU RGB+D 60 (cross-subject) [43]
pose-based	ST-GCN++ [30]	—
pose-based	Pose Transformer (ours)	—

[2] `https://github.com/open-mmlab/mmaction2`

[3] Distilled ViT-small from `https://github.com/OpenGVLab/VideoMAEv2/blob/master/docs/MODEL_ZOO.md`

Table 2. *Comparison of key properties of Drive&Act and the DAI dataset. Sequences in Drive&Act are preprocessed into clips, while DAI contains long annotations.*

Property	Drive&Act [38]	DAI [48]
classes	34	20
subjects per split (train/validation/test)	10/2/3	14/3/3
sequences	10,294	1,302
sequences per class ($\mu \pm \sigma$)	302.8±503.9	65.1±39.9
frames per second	30	20
frames per sequence ($\mu \pm \sigma$)	76±26	354±274

Datasets and Preprocessing

The footage in DAI contains 20 actors performing scripted scenarios according to distraction- and normal driving-related classes. Annotations span whole scenarios that match an action class rather than atomic actions. The "Eating-drinking" scenario, for instance, may encompass fetching, opening, consuming, closing, and storing an object. The *"fine-grained activities"* annotations of Drive&Act [38], however, are pre-processed into slices of at most three seconds in length. Table 2 summarizes the key properties of both datasets.

As a prerequisite for training pose-based models, we extract pose data from all annotated NIR video footage using RTM Pose [49]. This data contains multiple pose candidates per frame, along with a confidence value for each keypoint. We remove false-positive poses based on their accumulated keypoint confidence and reject poses with bounding boxes that are too small or too far from the known location of the driver seat. The resulting pose dataset contains at most one pose per video frame. In cases where no valid pose has been detected, we interpolate pose keypoints between the two temporally closest valid poses. For our experiments, we train, validate, and test on each dataset and view separately. On Drive&Act, we use the predefined "0" fold. The splits in both datasets are cross-subject. Input preprocessing standardizes brightness values within each dataset and applies a rescaling of pose data such that the frame is rescaled to $[-1, 1]$ for both axes. Frames are rescaled to a smaller side length of 256px. We apply data augmentation per clip, comprised of a 224×224 random crop and random horizontal flipping, equally for pose- and image data. Pose keypoints are retained even outside of crop boundaries.

The Pose Transformer

Our pose transformer architecture is designed based on the fundamental ideas of [11], [12], and [14] with minimal changes to adapt the concepts for the targeted purpose. Each pose frame consists of $K = 17$ keypoints with 3 values each: x, y, and the confidence value from the pose estimation stage. For a clip with T frames, we generate T tokens with $C_{\text{hidden}} = 128$ dimensions by flattening the values of each pose frame and projecting the resulting vectors into latent space using a learnable embedding and adding an also learnable position embedding. The tokens then pass through 6 transformer blocks, after which we project the first token to a vector of C_{out} logits that are used for classification. The transformer blocks use multi-head self-attention with $h = 8$ heads and a feed-forward network with a hidden layer dimension of 1024. We use the GELU activation function and apply layer normalization and dropout ($p = 0.1$) for regularization. The detailed architecture is illustrated in Figure 2.

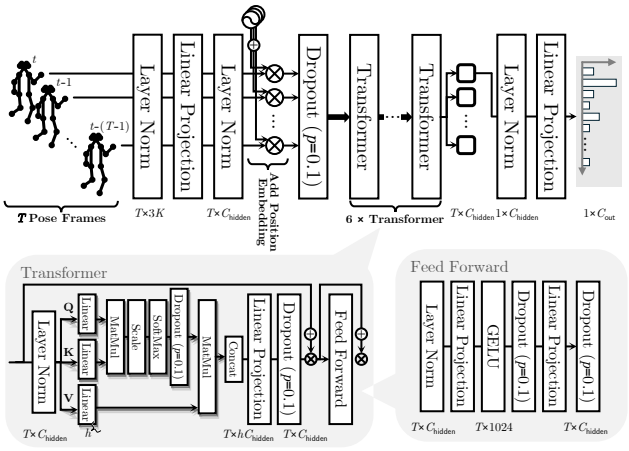


Figure 2. *The pose transformer architecture.*

Table 3. *Measurements for each model and dataset (2 best results in bold). Each measurement is the mean AR over 7 repeated trials ($\bar{x}$). $\bar{x}$ is given for individual views and as a mean per dataset. Models where pretrained and non-pretrained variants are evaluated are labeled respectively.*

Model Variant (Pretrained dataset)	Input	DAI Dataset			Drive&Act		
		$\bar{x}_{\text{center}}$	$\bar{x}_{\text{right}}$	mean	$\bar{x}_{\text{center}}$	$\bar{x}_{\text{right}}$	mean
ST-GCN (NTU RGB+D 120) [28]	P	**0.81**	**0.75**	**0.78**	**0.62**	0.52	**0.57**
ST-GCN++ (NTU RGB+D 60) [30]	P	**0.78**	**0.77**	**0.77**	0.56	0.47	0.53
VideoMAE V2 [25]	V	0.75	0.75	0.75	**0.65**	**0.64**	**0.64**
Pose Transformer (ours)	P	0.72	0.74	0.73	0.56	0.43	0.49
Video Swin (K400) [21]	V	0.70	0.65	0.68	0.56	**0.57**	0.57
ST-GCN [28]	P	0.65	0.65	0.66	0.56	0.44	0.50
ST-GCN++ [30]	P	0.68	0.64	0.66	0.60	0.49	0.55
Video Swin [21]	V	0.61	0.60	0.60	0.45	0.48	0.46
I3D [10]	V	0.61	0.58	0.59	0.51	0.44	0.47

Training

To stabilize our measurements, we repeat experiments 7 times and oversample both datasets. The training set of DAI is oversampled 8 times, the validation and test sets of DAI 24 times, and the validation and test sets of Drive&Act 3 times, resulting overall in 30.882 samples from Drive&Act and 31.248 samples from DAI. Each time the same annotation is revisited, we sample a new clip from a random location within the annotation. If the annotation is shorter than the clip, the clip is centered around the annotation. Data for the frames of each clip is read from the original footage (or the corresponding pose sequence). Training uses a fixed 36 epochs cosine annealing learning rate schedule with a linear warm-up for 2.5 epochs. A hyperparameter sweep on the DAI dataset was used to find Learning Rate (LR) settings for video-based model variants, and settings for LR, clip size (number of frames in a clip), and frame interval (distance of two frames in a clip in the original footage) for pose-based variants.

Evaluation and Results

Table 3 shows measurements for each model variant on different sets of data: the individual views and both views combined per each dataset. Table 4 shows results on Drive&Act when predictions are aggregated into scenarios that correspond more closely to the scenarios of the DAI dataset. These post-

Table 4. *Analogous to Table 3, but for predictions on the Drive&Act test set with mapped classes.*

Model Variant	Input	$\bar{x}_{\text{center}}$	$\bar{x}_{\text{right}}$
VideoMAE V2	V	**0.79**	**0.74**
Video Swin (K400)	V	**0.72**	**0.69**
I3D	V	0.69	0.60
Video Swin	V	0.67	0.60
ST-GCN (NTU120)	P	0.67	0.57
ST-GCN++ (NTU60)	P	0.65	0.56
ST-GCN++	P	0.64	0.57
Pose Transformer (ours)	P	0.67	0.53
ST-GCN	P	0.64	0.57

processed results are produced by grouping predictions for specific original Drive&Act classes into scenario-level classes, e.g. "reading magazine" and "reading newspaper" into "reading" or "opening laptop", "working on laptop", and "closing laptop" into "laptop". These results do not show the performance one could expect when training with consolidated Drive&Act classes – the mapped results rather show, how much confusion exists among classes within their scenario-level bins.

Table 5 compares wrong predictions of one class c when confused with another class c' for video- and posed-based variants. The comparison uses the difference between the mean FNR values within each sample. The p-values for the FNR means being different on a particular dataset are computed by a precise two-sided permutation test for independent samples, where each observation in a sample averages $\text{FNR}(c,c')$ over 7 repeated trials on both camera views for a particular model. Table 5 only lists rows where the difference between pose- and video-based models is significant and $c' \notin \{\text{fetching an object}, \text{placing an object}\}$, because these would occupy a large portion of the table while not adding substantial insight beyond the explanations in our discussion. Figure 3 compares the size and computational requirements

Table 5. *A list of pairs of true and wrongly predicted classes whose class-specific FNR is significantly different ($p <= 0.05$) between pose-based and video-based models. These are all such pairs detected among the 5 true classes with the worst recall and their 3 classes that were predicted most instead.* † *Rows where $c' \in \{\text{fetching an object}, \text{placing an object}\}$ were disregarded for Drive&Act due to limited space and additional informative value.*

True Class (c)	Predicted Class (c')	$\text{FNR}(c,c')$ in %		p
		Pose	Video	
DAI Dataset				
Blow-nose	Eating-drinking	30.28	8.11	0.032
Talking	Texting-on-phone	15.30	3.93	0.016
Eating-drinking	Sunglasses	15.26	33.02	0.016
Texting-on-phone	Steering	12.74	0.57	0.016
Drowsy	Sunglasses	12.00	4.76	0.048
Eating-drinking	Put-onoff-jacket	0.19	4.82	0.032
Drive&Act†				
preparing food	eating	39.29	66.48	0.016
looking or moving around	working on laptop	4.54	0.00	0.016
	pressing automation button	2.02	15.55	0.016
taking laptop from backpack	opening backpack	0.00	13.10	0.016

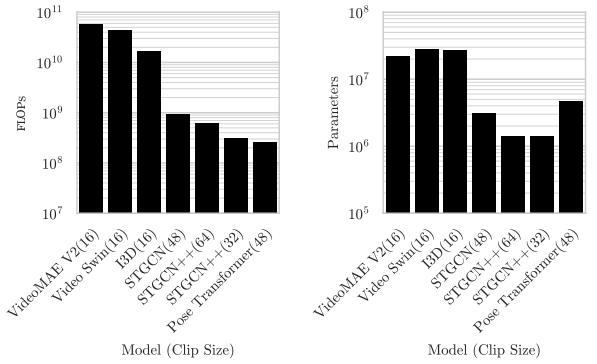


Figure 3. *FLOPs and parameter count.*

of the evaluated models. The plot in Figure 4 shows the GPU latency measured on our hardware platform. In addition to the model architectures listed in Table 1, we evaluate the latency of two *"null"* models – one for pose- and one for video-input. These are empty network implementations for measuring the overhead of running a batch through the inference pipeline. Among others, this accounts for the GPU-side preprocessing pipeline, the framework, and instrumentation. In our results, we subtract the measured timings of the respective null model from the measurements of the actual models, resulting in values corresponding to pure GPU-side latency.

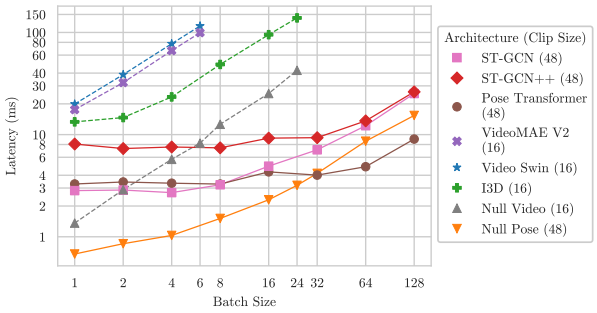


Figure 4. *GPU inference latency in milliseconds depending on batch size.*

Discussion

Table 3 shows pretrained model variants significantly outperforming respective non-pretrained model variants. Pose Transformer is not pretrained but performs better or almost as good as other non-pretrained GCNs. Cross-view performance appears to be more consistent for video-based methods – especially for the pretrained ones –, the mean AR diverges more between the two views for pose-based methods.

On average, results on Drive&Act are worse than on the DAI dataset by 16.11%, but only by 5.53% when Drive&Act results are binned into scenario-level classes (see Table 4). This suggests that the increased number of wrong predictions is mostly due to confusion within each consolidated scenario-level action group. Our original results on Drive&Act in Table 3 are mixed, with a pose- (ST-GCN) and a video-based model variant (VideoMAE V2) leading (both pretrained). When confusion within scenario-level action groups are resolved by a postprocessing step, all

video-based model variants show a better mean AR than any pose-based method (see Table 4).

Table 5 presents details on which pairs of classes are confused the most. Within Drive&Act, we see that the pose-based models perform on average better at separating some challenging actions with only different limb movement while involving either the same or no visible object (e.g. "preparing food" vs. "eating", "taking laptop from backpack" vs. "opening backpack", or "looking or moving around" vs. "pressing automation button"). Inversely, in these cases, the tested video-based models underperformed, despite having access to a superset of the information that was accessible to pose-based methods. However in return, we see on the DAI dataset, that video-based models may be preferable when limb movement is similar but involved objects are visibly different (e.g. "looking or moving around" vs. "working on laptop, "Texting-on-phone" vs. "Steering", "Blow-nose" vs."Eating-drinking", and "Drowsy" vs. "Sunglasses"). Exceptions are:

1. "Eating-drinking" vs. "Put-onoff-jacket": The lower FNR of pose-based models may be due to occlusion by the jacket, causing the upstream pose detection to fail for extended frame ranges. The pose-based model may not be aware of the jacket due to the lack of visual context, but it can separate the action based on the amount of missing input data.
2. "Eating-drinking" vs. "Sunglasses": Although both classes involve similar movements of hands towards the head, video-based models underperformed pose-based models. This is likely an artifact that arises from the fact that some actors used an eyeglass case for the "eating-drinking" scenario instead of a lunch box, due to a lack thereof.

It is generally notable that the confusion between many Drive&Act action classes and "fetching an object" or "placing an object" was such striking that we excluded them from Table 5 to free up space for possibly otherwise interesting results. These two classes accounted on average for 7% and 8%, respectively, of the FNR of other classes[4]. We argue that some of the remaining issues in the original Drive&Act dataset that may cause the residual 5.53% reduction in mean AR as compared to results on the DAI dataset may be due to this problematic choice of classes, which can not be resolved by post-processing the results as in Table 4. "fetching an object" and "placing an object" occur in conjunction with otherwise separate actions that belong to different scenarios, such as "eating" or "taking off sunglasses" as can be seen in the supplementary material of [38]. This ambiguity poses similar problems to both video- and pose-based models.

Figure 3 shows the general correlation between model size and computational requirements. Yet, there are two exceptions: VideoMAE V2 requires more FLOPs on fewer parameters than Video Swin, and our pose transformer requires the least computation using the most parameters among pose-based models. Furthermore, in Tables 3 and 4 ST-GCN++ performed similarly to ST-GCN using fewer parameters and FLOPs (see Figure 3), owing to its optimizations. However, this did not translate into faster GPU inference for batch sizes < 128 as shown in Figure 4. This indicates that factors other than model size and FLOPs can have a significant influence on GPU inference latency, as seen in this case, where the increased number of operations in ST-GCN++ compared to the simpler ST-GCN architecture plays a significant role. Similarly, VideoMAE V2 infers consistently quicker than Video Swin despite a higher FLOP count. Overall, from a resource-focused view, pose-based models seem preferrable. Yet, we have to note that resource usage of pose estimation models are not included in Figures 3 and 4. While the separation of concerns between pose estimation and pose-based HAR has its advantages, it needs to be considered together within concrete practical implementations. In summary, it is worth pointing out that these values may all be negligible compared to the overall latency between recording an action and its prediction. This kind of latency commonly resides in the range of seconds, as it depends on the temporal receptive field covered by a clip and the nature and duration of the action.

[4]Mean across $\{\text{FNR}(c,c') | c' \neq c \wedge c \in C\}$, where C is the set of all valid classes and $c' \in \{\text{fetching an object}, \text{placing an object}\}$

Limitations and Conclusions

In this paper, we compared the performance of 4 video-based and 5 pose-based models on two datasets. Although this is not an exhaustive list of models and only provides for a p-value granularity of 1/126 = 0.00794 in Table 5, we still observed evidence that both approaches show their strengths with specific types of actions. The results demonstrate the benefit of using scenario-level classes as in the DAI dataset, which does not suffer from additional ambiguities. We observed the competitive performance of pose-based models. However, video-based methods have greater potential to surpass the observed results by extending hyperparameter search to more dimensions. The analysis of model sizes and computational requirements shows that this requires a least an order of magnitude more resources for video-based models compared to pose-based ones.

We hope that the results presented stimulate further research into the details of HAR methods for in-cabin driver monitoring that may not be covered by general-purpose HAR benchmarks.

Acknowledgments

This work was supported in parts by the project SyntheticCabin (no. 884336) and in parts by the project Uniscope-3D (no. F0999923852), which are funded through the Austrian Research Promotion Agency (FFG) on behalf of the Austrian Ministry of Climate Action (BMK).

References

[1] L. Masello, G. Castignani, B. Sheehan, F. Murphy, and K. McDonnell, "On the road safety benefits of advanced driver assistance systems in different driving contexts," *Transportation Research Interdisciplinary Perspectives*, vol. 15, p. 100 670, 2022.

[2] P. A. Hancock and W. B. Verwey, "Fatigue, workload and adaptive driver systems," *Accident Analysis & Prevention*, vol. 29, pp. 495–506, 4 1997.

[3] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, "Vision-based human activity recognition: a survey," *Multimedia Tools and Applications*, vol. 79, no. 41, pp. 30 509–30 555, 2020.

[4] L. Song, G. Yu, J. Yuan, and Z. Liu, "Human pose estimation and its application to action recognition: A survey," *Journal of Visual Communication and Image Representation*, vol. 76, C 2021.

[5] R. Kumar and S. Kumar, "A survey on intelligent human action recognition techniques," *Multimedia Tools and Applications*, vol. 83, pp. 52 653–52 709, 2023.

[6] G. Morshed, T. Sultana, A. Alam, and Y.-K. Lee, "Human Action Recognition: A Taxonomy-Based Survey, Updates, and Opportunities," *Sensors*, vol. 23, no. 4, 2023.

[7] W. Li, J. Huang, G. Xie, F. Karray, and R. Li, "A survey on vision-based driver distraction analysis," *Journal of Systems Architecture*, vol. 121, p. 102 319, 2021.

[8] W. Kay, J. Carreira, K. Simonyan, *et al.*, "The kinetics human action video dataset," *arXiv*, 2017.

[9] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning Spatiotemporal Features with 3D Convolutional Networks," in *ICCV*, 2015, pp. 4489–4497.

[10] J. Carreira and A. Zisserman, "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset," in *CVPR*, 2017, pp. 4724–4733.

[11] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is All you Need," in *NeurIPS*, 2017, pp. 6000–6010.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *ICLR*, 2021.

[13] Z. Liu, Y. Lin, Y. Cao, *et al.*, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in *ICCV*, 2021, pp. 9992–10 002.

[14] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "ViViT: A Video Vision Transformer," in *ICCV*, 2021, pp. 6836–6846.

[15] G. Bertasius, H. Wang, and L. Torresani, "Is Space-Time Attention All You Need for Video Understanding?" In *ICML*, 2021, pp. 813–824.

[16] H. Fan, B. Xiong, K. Mangalam, *et al.*, "Multiscale Vision Transformers," in *ICCV*, 2021, pp. 6804–6815.

[17] Y. Zhang, X. Li, C. Liu, *et al.*, "VidTr: Video Transformer Without Convolutions," in *ICCV*, 2021, pp. 13 557–13 567.

[18] M. Patrick, D. Campbell, Y. M. Asano, *et al.*, "Keeping Your Eye on the Ball: Trajectory Attention in Video Transformers," in *NeurIPS*, 2021, pp. 12 493–12 506.

[19] J. Chen and C. M. Ho, "MM-ViT: Multi-Modal Video Transformer for Compressed Video Action Recognition," in *WACV*, 2022, pp. 1910–1921.

[20] Y. Li, C.-Y. Wu, H. Fan, *et al.*, "MViTv2: Improved Multiscale Vision Transformers for Classification and Detection," in *CVPR*, 2022, pp. 4804–4814.

[21] Z. Liu, J. Ning, Y. Cao, *et al.*, "Video Swin Transformer," in *CVPR*, 2022, pp. 3202–3211.

[22] D. Neimark, O. Bar, M. Zohar, and D. Asselmann, "Video Transformer Network," in *ICCVW*, 2021, pp. 3163–3172.

[23] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked Autoencoders Are Scalable Vision Learners," in *CVPR*, 2022, pp. 15 979–15 988.

[24] Z. Tong, Y. Song, J. Wang, and L. Wang, "VideoMAE: Masked Autoencoders are Data-Efficient Learners for Self-Supervised Video Pre-Training," in *NeurIPS*, 2022, pp. 10 078–10 093.

[25] L. Wang, B. Huang, Z. Zhao, *et al.*, "VideoMAE V2: Scaling Video Masked Autoencoders With Dual Masking," in *CVPR*, 2023, pp. 14 549–14 560.

[26] Y. Wang, K. Li, Y. Li, *et al.*, "InternVideo: General Video Foundation Models via Generative and Discriminative Learning," *ArXiv:2212.03191*, 2022.

[27] R. Bommasani, D. A. Hudson, E. Adeli, *et al.*, "On the Opportunities and Risks of Foundation Models," *ArXiv:2108.07258*, 2021.

[28] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI/AI*, 2018, pp. 7444–7452.

[29] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-Structural Graph Convolutional Networks for Skeleton-Based Action Recognition," in *CVPR*, 2019, pp. 3590–3598.

[30] H. Duan, J. Wang, K. Chen, and D. Lin, "Pyskl: Towards good practices for skeleton action recognition," in *ACM MM*, 2022, pp. 7351–7354.

[31] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting Skeleton-based Action Recognition," in *CVPR*, 2022, pp. 2959–2968.

[32] S. Sun, Z. Jia, Y. Zhu, G. Liu, and Z. Yu, "Decoupled spatio-temporal grouping transformer for skeleton-based action recognition," *The Visual Computer*, 2023.

[33] L. Wang and P. Koniusz, "3Mformer: Multi-Order Multi-Mode Transformer for Skeletal Action Recognition," in *CVPR*, 2023, pp. 5620–5631.

[34] W. Zhu, X. Ma, Z. Liu, L. Liu, W. Wu, and Y. Wang, "MotionBERT: A Unified Perspective on Learning Human Motion Representations," in *ICCV*, 2023, pp. 15 085–15 099.

[35] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, "Driver Distraction Identification with an Ensemble of Convolutional Neural Networks," *Journal of Advanced Transportation*, p. 4 125 865, 2019.

[36] M. Martin, J. Popp, M. Anneken, M. Voit, and R. Stiefelhagen, "Body Pose and Context Information for Driver Secondary Task Detection," in *IV*, 2018, pp. 2015–2021.

[37] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[38] M. Martin, A. Roitberg, M. Haurilet, *et al.*, "Drive&Act: A Multi-Modal Dataset for Fine-Grained Driver Behavior Recognition in Autonomous Vehicles," in *ICCV*, 2019, pp. 2801–2810.

[39] Z. Qiu, T. Yao, and T. Mei, "Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks," in *ICCV*, 2017, pp. 5534–5542.

[40] M. Martin, M. Voit, and R. Stiefelhagen, "Dynamic Interaction Graphs for Driver Activity Recognition," in *IEEE International Conference on Intelligent Transportation Systems*, 2020.

[41] M. Martin, D. Lerch, and M. Voit, "Viewpoint Invariant 3D Driver Body Pose-Based Activity Recognition," in *IV*, 2023.

[42] Y. Ma and Z. Wang, "Vit-dd: Multi-task vision transformer for semi-supervised driver distraction detection," in *IV*, 2024, pp. 417–423.

[43] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis," in *CVPR*, 2016, pp. 1010–1019.

[44] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. Kot, "NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding," *TPAMI*, vol. 42, no. 10, pp. 2684–2701, 2020.

[45] J. Carreira, E. Noland, C. Hillier, and A. Zisserman, "A Short Note on the Kinetics-700 Human Action Dataset," *arXiv:1907.06987*, 2022.

[46] J. D. Ortega, N. Kose, P. Cañas, *et al.*, "DMD: A Large-Scale Multi-modal Driver Monitoring Dataset for Attention and Alertness Analysis," in *ECCVW*, 2020, pp. 387–405.

[47] I. Jegham, A. Ben Khalifa, I. Alouani, and M. A. Mahjoub, "A novel public dataset for multimodal multiview and multispectral driver distraction analysis: 3MDAD," *Signal Processing: Image Communication*, vol. 88, p. 115 960, 2020.

[48] R. Panadero, D. Schörkhuber, and M. Gelautz, "Importance-guided interpretability and pruning for video transformers in driver action recognition," in *WACV*, 2025, pp. 5295–5304.

[49] T. Jiang, P. Lu, L. Zhang, *et al.*, "RTMPose: Real-Time Multi-Person Pose Estimation based on MMPose," *ArXiv:2303.07399*, 2023.

Author Biography

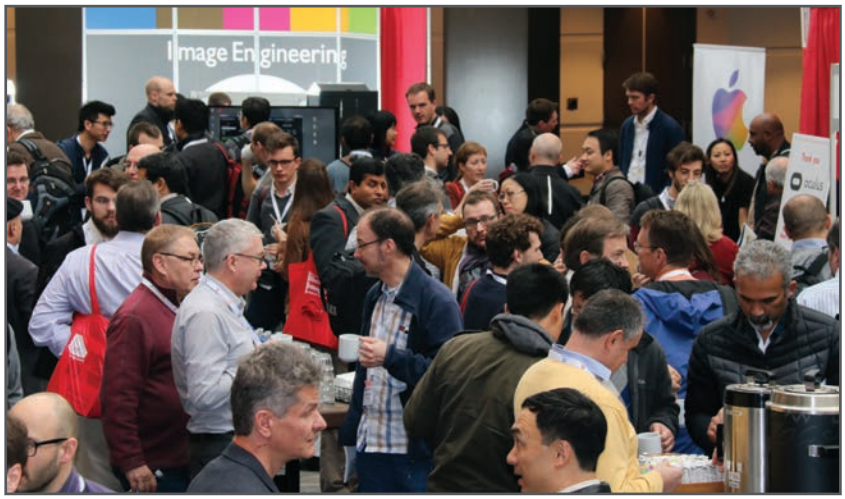
Lukas Brunner is a Master's Student at TU Wien, Austria. He received his bachelor's degree in 2017 and is currently completing his master's degree. He focuses on real-time methods for scene understanding and industrial applications.

Dominik Schörkhuber is a PhD student and research assistant at TU Wien, Austria. He received his bachelor's and master's degree in 2016 and 2018, respectively. His research is centered around video analysis in application to autonomous and assisted driving.

JOIN US AT THE NEXT EI!

electronic IMAGING

Imaging across applications . . . Where industry and academia meet!



• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

ADVANCING IMAGING SCIENCE 75 1947 IS&T 2022 imaging.org