POLITECNICO DI TORINO
Repository ISTITUZIONALE

Action Transformer: A Self-Attention Model for Short-Time Pose-Based Human Action Recognition

Original
Action Transformer: A Self-Attention Model for Short-Time Pose-Based Human Action Recognition / Mazzia, V., Angarano, S., Salvetti, F., Angelini, F., Chiaberge, M.. - In: PATTERN RECOGNITION. - ISSN 0031-3203. - ELETTRONICO. - 124:(2022), p. 108487. [10.1016/j.patcog.2021.108487]

Availability:
This version is available at: 11583/2946512 since: 2021-12-20T10:29:13Z

Publisher:
Elsevier

Published
DOI:10.1016/j.patcog.2021.108487

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
Elsevier postprint/Author's Accepted Manuscript

© 2022. This manuscript version is made available under the CC-BY-NC-ND 4.0 license
http://creativecommons.org/licenses/by-nc-nd/4.0/.The final authenticated version is available online at:
http://dx.doi.org/10.1016/j.patcog.2021.108487

(Article begins on next page)

18 November 2026

Action Transformer: A Self-Attention Model for Short-Time Pose-Based Human Action Recognition

Vittorio Mazzia[a,b,c], Simone Angarano[a,b], Francesco Salvetti[a,b,c], Federico Angelini[d] and Marcello Chiaberge[a,b]

[a]Department of Electronics and Telecommunications, Politecnico di Torino, 10124 Turin, Italy
[b]PIC4SeR, Politecnico di Torino Interdepartmental Center for Service Robotics, Turin, Italy
[c]SmartData@PoliTo, Big Data and Data Science Laboratory, Turin, Italy
[d]School of Computing, Newcastle University, 1 Science Square, Newcastle, NE45TG, UK

ARTICLE INFO

Keywords:
Human Action Recognition
Deep Learning
Computer Vision
Transformer

ABSTRACT

Deep neural networks based purely on attention have been successful across several domains, relying on minimal architectural priors from the designer. In Human Action Recognition (HAR), attention mechanisms have been primarily adopted on top of standard convolutional or recurrent layers, improving the overall generalization capability. In this work, we introduce Action Transformer (AcT), a simple, fully, self-attentional architecture that consistently outperforms more elaborated networks that mix convolutional, recurrent, and attentive layers. In order to limit computational and energy requests, building on previous human action recognition research, the proposed approach exploits 2D pose representations over small temporal windows, providing a low latency solution for accurate and effective real-time performance. Moreover, we open-source MPOSE2021, a new large-scale dataset, as an attempt to build a formal training and evaluation benchmark for real-time, short-time HAR. The proposed methodology was extensively tested on MPOSE2021 and compared to several state-of-the-art architectures, proving the effectiveness of the AcT model and laying the foundations for future work on HAR.

1. Introduction

Human Action Recognition (HAR) is a problem in computer vision and pattern recognition that aims to detect and classify human actions. The ability to recognize people inside a scene and predict their behavior is fundamental for several applications, such as robotics [36], surveillance and security [46, 50], autonomous vehicles [33, 7], and automatic video captioning [55, 45, 49]. Most previous works dealing with HAR adopted datasets characterized by samples with a long temporal duration [25, 47, 32]. Thus, HAR has been mainly treated as a post-processing operation, classifying complex and long-lasting human actions by exploiting past and future information. Conversely, in this work, we focus on short-time HAR, which aims at continuously classifying actions within short past time steps (up to a second). This approach is fundamental to target real-time applications: in robotics, for example, HAR problem should be solved promptly to react to sudden behavioral changes, only relying on near past information.

In this paper, we propose a new model for HAR called the Action Transformer (AcT), schematized in Fig.1, inspired by the simple and prior-free architecture of the Vision Transformer [18]. The Transformer architecture [48] has been one of the most important deep learning advances of the last years in natural language processing (NLP). Moreover, multi-head self-attention has proven to be effective for a wide range of tasks besides NLP, e.g. image classification

ORCID(s): 0000-0002-7624-1850 (V. Mazzia); 0000-0002-4445-9783 (S. Angarano); 0000-0003-4744-4349 (F. Salvetti); 0000-0001-8333-656X (F. Angelini); 0000-0002-1921-0126 (M. Chiaberge)

[18, 44, 15], image super-resolution [54, 56], and speech recognition [17]. Furthermore, optimized versions of the Transformer have been developed for real-time and embedded applications [8], proving that this architecture is also suitable for Edge AI purposes. Recently, many models for Human Action Recognition have proposed the integration of attention mechanisms with convolutional and recurrent blocks to improve the accuracy of models. However, solutions that rely exclusively on self-attention blocks have not been investigated for this task yet.

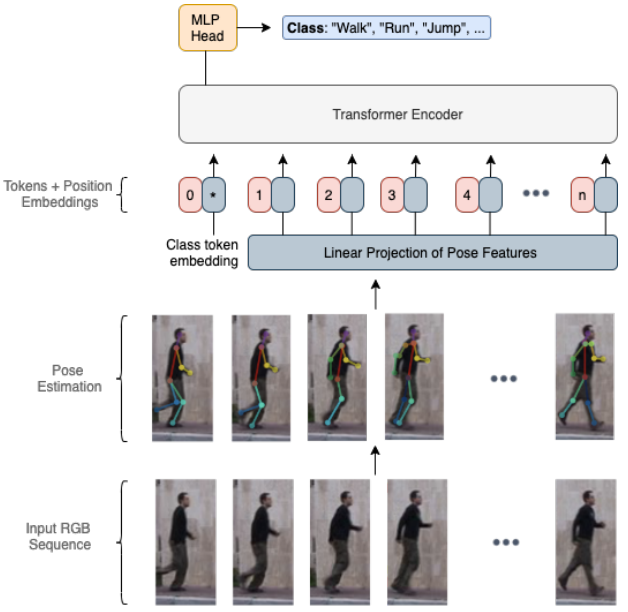


Figure 1: Overview of the Action Transformer architecture. Pose estimations are linearly projected to the dimension of the model, and together with the class token, they form the input tokens of the transformer encoder. As for Vision Transformer models [18, 44, 15], a learnable positional embedding is added to each input token. Then, only the output class token is passed through a multi-layer perceptron head to obtain the final class prediction.

With AcT, we apply a pure Transformer encoder derived architecture to action recognition obtaining an accurate and low-latency model for real-time applications. We study the model at different scales to investigate the impact of the number of parameters and attention heads. We use the new single-person, short-time action recognition dataset MPOSE2021 as a benchmark and exploit 2D human pose representations provided by two existing detectors: OpenPose [9] and PoseNet [34]. Moreover, we compare AcT with other state-of-the-art baselines to highlight the advantages of the proposed approach. To highlight the effectiveness of self-attention, we conduct a model introspection providing visual insights of the results and study how a reduction of input temporal sequence length affects accuracy. We also conduct extensive experimentation on model latency on low-power devices to verify the suitability of AcT for real-time applications.

The main contributions of this work can be summarized as follows:

- We study the application of the Transformer encoder to 2D pose-based HAR and propose the novel AcT model, proving that fully self-attentional architectures can outperform existing convolutional and recurrent models for

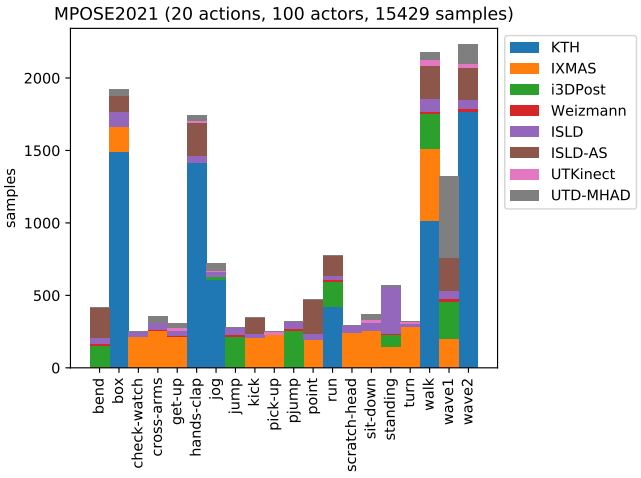


Figure 2: The number of samples of MPOSE2021 divided by action. The colors show the distribution of precursor datasets among classes, highlighting the unbalanced nature of the data. The final dataset contains 15429 samples where each sample represents one of 100 actors performing one of 20 actions.

pose-based HAR.

- We introduce MPOSE2021, a dataset for real-time short-time HAR, suitable for both pose-based and RGB-based methodologies. It includes 15429 sequences from 100 actors and different scenarios, with limited frames per scene (between 20 and 30). In contrast to other publicly available datasets, the peculiarity of having a constrained number of time steps stimulates the development of real-time methodologies that perform HAR with low latency and high throughput.

The rest of the paper is organized as follows. Section 2 briefly reports the relevant research on HAR and self-attention deep learning methodologies. Section 3 presents the MPOSE2021 dataset, highlighting features and elements of novelty with respect to other existing datasets. Then, Section 4 describes the architecture of the proposed AcT model, shortly recalling the main features of the Transformer encoder. Section 5 summarizes the experimentation conducted to verify the effectiveness of the proposed approach, gives a visual insight into the model functioning, studies its behavior under temporal information reduction, and measures its latency for real-time applications. Section 6 draws some conclusions for this work and indicates future research directions.0

2. Related Works

Human Action Recognition algorithms aim at detecting and classifying human behaviors based on a different source of information. Works in this field can be mainly subdivided into two broad categories: video-based and depth-based methodologies [42]. In both, the input consists of a sequence of points representing body joints, extracted from the corresponding RGB frame [28], in the case of video-based methods, or a point cloud, in the case of depth-based

ones. In the latter, body joints are provided as 3D coordinates (skeletal data), such as those captured by Kinect sensors [38, 29], that can be possibly projected onto the 2D space [11, 53]. On the opposite, in this work, we focus on a video-based analysis, where body joints are already provided as 2D coordinates (pose data) by pose detection algorithms such as OpenPose [9] and PoseNet [34]. This characteristic makes 2D HAR methodologies applicable in a great range of applications using a simple RGB camera. Conversely, skeletal data require particular sensors to be acquired, such as Kinect or other stereo-cameras. That raises substantial limitations, such as availability, cost, limited working range (up to 5-6 meters in the case of Kinect [27]), and performance degradation in outdoor environments.

Angelini et al. first collected the MPOSE dataset for video-based short-time HAR [2], obtaining 2D poses by processing several popular HAR video datasets with OpenPose. Moreover, the authors proposed ActionXPose, a methodology that increases model robustness to occlusions. The sequences were classified using MLSTM-FCN [26], which exploits a combination of 1D convolutions, LSTM [23], and Squeeze-and-excitation attention [24]. The same authors successively applied their approach to anomaly detection [4, 3] and expanded MPOSE with the novel ISLD and ISLD-Additional-Sequences datasets. Conversely, Yan et al. [53] first applied OpenPose to extract 2D poses from the Kinetics-400 RGB dataset [11] and used graph convolutions to capture spatial and temporal information. Similarly, Shi et al. [39, 40] applied graph convolutions to pose information using two streams that extract information from both joints and bones to represent the skeleton. Hao et al. [21]introduced knowledge distillation and dense-connectivity to explore the spatiotemporal interactions between appearance and motion streams along different hierarchies. At the same time, Si et al. [41] put together a hierarchical spatial reasoning network and temporal stack learning network to extract discriminative spatial and temporal features. On the other hand, Liu et al. [30] proposed an ensemble of two independent bone-based and joint-based models using a unified spatial-temporal graph convolutional operator. Conversely, Cho et al. [14] first applied self-attention [48] to the skeletal-based HAR problem. More recently, Plizzari et al. [35], inspired by Bello et al. [6], employed self-attention to overcome the locality of the convolutions, again adopting a two-stream ensemble method, where self-attention is applied on both temporal and spatial information.

Unlike previous methodologies, this paper presents an architecture for HAR entirely based on the Transformer encoder, without any convolutional or recurrent layer. Moreover, we focus on verifying the suitability of our method for low-latency and real-time applications. For this reason, we introduce a new 2D pose-based dataset specifically designed for short-time HAR. Traditional datasets used by previous works, such as 3D NTU RGB+D [38, 29], or Kinetics-Skeleton [11, 53], include long temporal sequences which must be entirely scanned to make the correct classification. In contrast, the proposed MPOSE2021 dataset includes samples with a temporal duration of at most 30 frames. That makes it a new and more suitable benchmark to test the short-time and low-latency performance of HAR models.

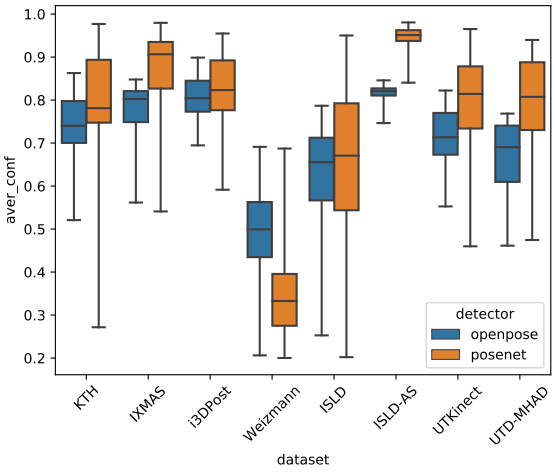


Figure 3: Comparison between OpenPose and PoseNet average confidence (aver_conf) in the different MPOSE2021 sub-datasets. The detectors achieve different average confidence based on the considered precursor dataset.

3. The MPOSE2021 Dataset

In this section, MPOSE2021 is presented as an RGB-based dataset designed for short-time, pose-based HAR. As in [2, 4, 3], video data have been previously collected from popular HAR datasets (precursors), i.e. Weizmann [20], i3DPost [19], IXMAS [51], KTH [37], UTKinetic-Action3D (RGB only) [52], UTD-MHAD (RGB only) [12], ISLD, and ISLD-Additional-Sequences [2].

Due to the heterogeneity of actions across different datasets, labels are remapped to a list of 20 common classes. Actions that cannot be remapped accordingly are discarded. Therefore, precursor videos are divided into non-overlapping samples (clips) of 30 frames each whenever possible and retaining tail samples with more than 20 frames. A visual representation of the number of frames per sample for each sub-dataset of MPOSE2021 is shown in Fig. 4. The peculiarity of a reduced number of time steps contrasts with other publicly available datasets and stimulates the development of methodologies that require low latency to perform a prediction. That would largely benefit many real-world applications requiring real-time perception of the actions performed by humans nearby.

Subsequently, clips not containing a single action are discarded. Moreover, ambiguous clips are relabelled whenever possible or discarded otherwise. This process leads to 15429 samples, where each sample represents a single actor performing a single action. The total number of distinct actors in MPOSE2021 is 100, and the number of samples for each action is reported in Fig.2, which also shows the distribution of precursor datasets.

OpenPose [9] and PoseNet [34] are used to extract landmarks from MPOSE2021 samples. The average confidence is computed for each sample as the mean across landmarks and frames. It turns out that the two detectors achieve different average confidences based on the considered precursor dataset. The box plot of Fig.3 describes the comparison statistics.

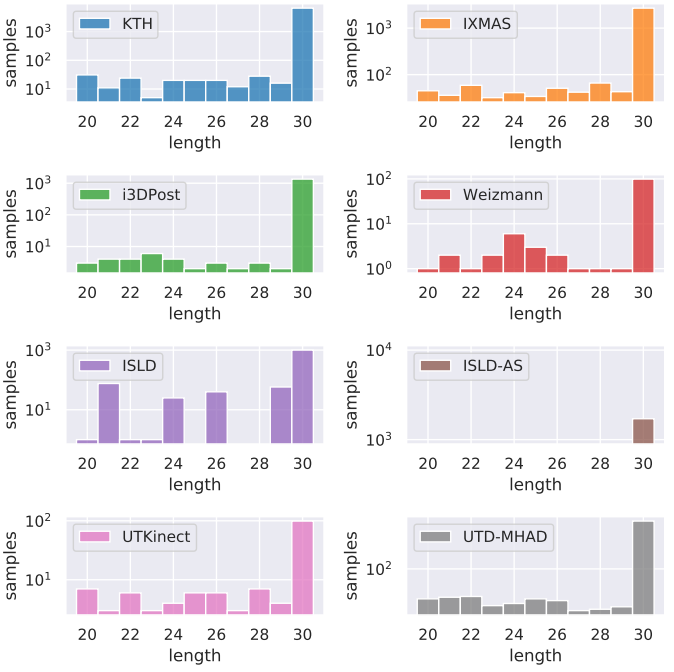


Figure 4: MPOSE2021 sequence length distribution for the different sub-datasets. It is possible to notice how most of them present a mode of 30 frames per scene.

Due to the significant sample heterogeneity and the high number of actors, three different training/testing splits are defined for MPOSE2021, i.e. Split1, Split2, and Split3, by randomly selecting 21 actors for testing and using the rest of them for training. This division makes the proposed dataset a challenging benchmark to effectively assess and compare the accuracy and robustness of different methodologies. Moreover, the suggested evaluation procedure requires testing a target model on each split using ten validation folds and averaging the obtained results. That makes it possible to produce statistics and reduces the possibility of overfitting the split testing set with an accurate choice of hyperparameters.

With MPOSE2021, we aim at providing an end-to-end and easy-to-use benchmark to robustly compare state-of-the-art methodologies for the short-time human action recognition task. We thus release a code repository[1] to access the different levels of the dataset (video, RGB frames, 2D poses). Moreover, we open source a practical Python package to access, visualize and preprocess the poses with standard functions. The Python package can be easily installed with the command `pip install mpose`.

4. Action Transformer

In this section, we describe the architecture of the AcT network (Fig.1) briefly recalling some preliminary concepts associated with the Transformer model[48].

[1] https://github.com/PIC4SeR/MPOSE2021_Dataset

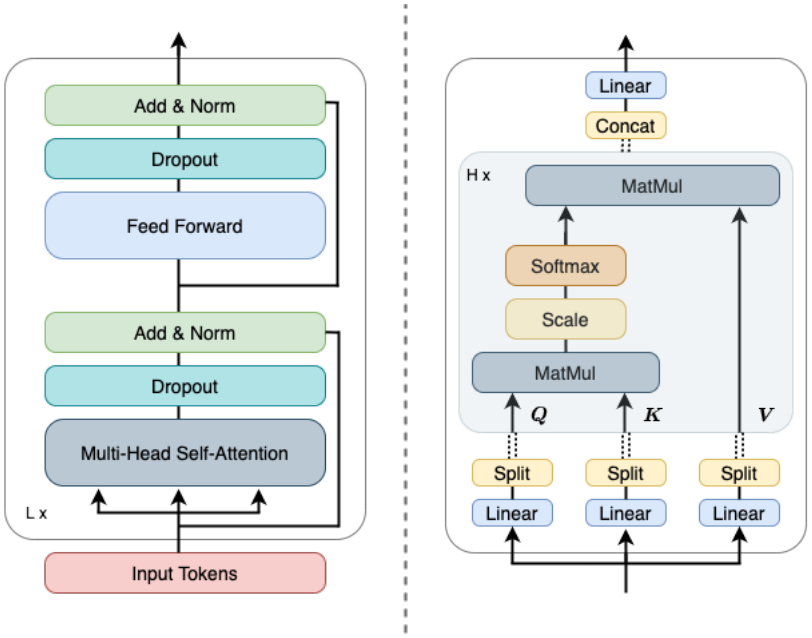


Figure 5: Transformer encoder layer architecture (left) and schematic overview of a multi-head self-attention block (right). Input tokens go through L encoder layers and H self-attention heads.

4.1. AcT Architecture

A video input sequence with T frames of dimension $H \times W$ and C channels $\mathbf{X}_{rgb} \in \mathbb{R}^{T \times H \times W \times C}$ is pre-processed by a multi-person 2D pose estimation network

$$\mathbf{X}_{2Dpose} = F_{2Dpose}(\mathbf{X}_{rgb}) \tag{1}$$

that extracts 2D poses of dimension $N \times T \times P$, where N is the number of human subjects present in the frame and P is the number of keypoints predicted by the network. The Transformer architecture receives as input a 1D sequence of token embeddings, so each of the N sequences of pose matrices $\boldsymbol{X}_{2Dpose} \in \mathbb{R}^{T \times P}$ is separately processed by the AcT network. Nevertheless, at inference time, all detected poses in the video frame can be batch-processed by the AcT model, simultaneously producing a prediction for all N subjects.

Firstly, the T poses are mapped to a higher dimension D_{model} using a linear projection map $\boldsymbol{W}^{l_0} \in \mathbb{R}^{P \times D_{model}}$. As in BERT [16] and Vision Transformers [18, 44, 8, 10], a trainable vector of dimension D_{model} is added to the input T sequence. This class token [CLS] forces the self-attention to aggregate information into a compact high-dimensional representation that separates the different action classes. Moreover, positional information is provided to the sequence with a learnable positional embedding matrix $\boldsymbol{X}_{pos} \in \mathbb{R}^{(T+1) \times D_{model}}$ added to all tokens.

The linearly projected tokens and [CLS] are fed to a standard Transformer encoder F_{Enc} of L layers with a post-

norm layer normalization [48, 13], obtaining

$$\boldsymbol{X}^L = F_{Enc}(\boldsymbol{X}^{l_0}) = F_{Enc}([\boldsymbol{x}_{cls}^{l_0}; \boldsymbol{X}_{2Dpose}] + \boldsymbol{X}_{pos}) \tag{2}$$

where $\boldsymbol{X}^L \in \mathbb{R}^{(T+1)\times D_{model}}$ is the overall representation produced by the Transformer encoder at its last layer. Finally, only the [CLS] token $\boldsymbol{x}_{cls}$ is fed into a linear classification head MLP_{head} that performs the final class prediction

$$\hat{z} = \text{MLP}_{head}(\boldsymbol{x}_{cls}^L) \tag{3}$$

where $\hat{z}$ is the output logit vector of the model. At training time, the supervision signal comes only from the [CLS] token, while all remaining T tokens are the only input of the model. It is important to notice how the nature of the network makes it possible to accept a reduced number of frames as input even if trained with a fixed T. That gives an additional degree of freedom at inference time, making AcT more adaptive than other existing models.

The resulting network is a lightweight solution capable of predicting actions for multiple people in a video stream with high accuracy. The advantage of building on 2D pose estimations enables effective real-time performance with low latency and energy consumption.

4.2. Transformer Architecture

The Transformer encoder [48] is made of L layers with alternating H multi-head self-attention and feed-forward blocks. Dropout [43], Layernorm [5], and residual connections are applied after every block. The overall sequence of blocks of a Transformer encoder is summarized on the left of Fig.5.

Each feed-forward block is a multi-layer perceptron with two layers and GeLu [22] non-linearity. The first layer expands the dimension from D_{model} to $D_{mlp} = 4 \cdot D_{model}$ and applies the activation function. On the other hand, the second layer reduces the dimension back from D_{mlp} to D_{model}.

Instead, the multi-head $\boldsymbol{QKV}$ self-attention mechanism (MSA) is based on a trainable associative memory with key-value vector pairs. For the l-th layer of the Transformer encoder and the h-th head, queries ($\boldsymbol{Q}$), keys ($\boldsymbol{K}$) and values ($\boldsymbol{V}$) are computed as $\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}_Q$, $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}_K$ and $\boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}_V$ respectively, where $\boldsymbol{W}_Q$, $\boldsymbol{W}_K$ and $\boldsymbol{W}_V$ belong to $\mathbb{R}^{D_{model}\times D_h}$, being D_h the dimension of the attention head. So, for each self-attention head (SA) and element in the input sequence $\boldsymbol{X} \in \mathbb{R}^{T\times D_{model}}$ we compute a weighted sum over all values $\boldsymbol{V}$. The attention weights A_{ij} are derived from the pairwise similarity between two elements of the sequence and their respective query Q_i and key K_j representations. Therefore, it is possible to compute the attention weights $\boldsymbol{A} \in \mathbb{R}^{T\times T}$ for the l-th layer as

$$\boldsymbol{A} = \text{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{D_h}}\right) \tag{4}$$

Table 1
Action Transformer parameters for the four version sizes. We fix $D_{model}/\mathrm{H} = 64$, linearly increasing H, D_{mlp}, and L in order to obtain different versions of the AcT network.

Model	H	D_{model}	D_{mlp}	L	Parameters
AcT-μ	1	64	256	4	227k
AcT-S	2	128	256	5	1,040k
AcT-M	3	192	256	6	2,740k
AcT-L	4	256	512	6	4,902k

and the final weighted sum SA as

$$\mathrm{SA}(\boldsymbol{X}) = \boldsymbol{A}\boldsymbol{V} \tag{5}$$

We perform this SA operation for all the H heads of the l-th layer. We concatenate the results and linearly project the output tensor to the original dimension model as

$$\mathrm{MSA}(\boldsymbol{X}) = [SA_1(\boldsymbol{X}); SA_2(\boldsymbol{X}); ...; SA_{\mathrm{H}}(\boldsymbol{X})]\boldsymbol{W}_{MSA} \tag{6}$$

where $\boldsymbol{W}_{MSA} \in \mathbb{R}^{\mathrm{H}\cdot D_h \times D_{model}}$. All operations are schematized on the right side of Fig. 5. The input tensor $\boldsymbol{X}$ is firstly projected to $T \times \mathrm{H} \cdot D_h$, and then a SA operation is performed to all H splits of the resulting projected tensor. Finally, all head outputs are concatenated and linearly projected back to D_{model}. So, the attention mechanism takes place in the time-domain, allowing a global class embedding representation by correlating different time windows.

In order to reduce the number of hyperparameters and linearly scale the dimension of AcT, we fix $D_{model}/\mathrm{H} = 64$, varying H, D_{mlp}, and L to obtain different versions of the network. A simple grid search using train and validation sets is performed to determine lower and upper bounds for the four parameters. In particular, in Table 1, we summarize the four AcT versions with their respective number of parameters. The four models (micro, small, medium, and large) differ for their increasing number of heads and layers, substantially impacting the number of trainable parameters.

5. Experiments

This section describes the main experiments conducted to study the advantages of using a fully self-attentional model for 2D pose-based HAR. First, the four variants of AcT described in Section 4 are compared to existing state-of-the-art methodologies and baselines on MPOSE2021. Only for a specific comparison with ST-TR [35] and MS-G3D [30], we use additional ensemble versions of our model named AcT-μ (x n), n being the number of ensembled instances. Then, we further analyze the behavior of the network in order to get a visual insight of the attention mechanism and study the performance under a reduction of temporal information. Finally, we study model latency for all the designed architectures with two different CPU types, proving that AcT can easily be used for real-time applications.

Table 2
Hyperparameters used in AcT experiments.

Training		Regularization	
Training epochs	350	Weight decay	1e-4
Batch size	512	Label smoothing	0.1
Optimizer	AdamW	Dropout	0.3
Warmup epochs	40%	Random flip	50%
Step epochs	80%	Random noise σ	0.03

Table 3
Benchmark of different models for short-time HAR on MPOSE2021 splits using OpenPose 2D skeletal representations.

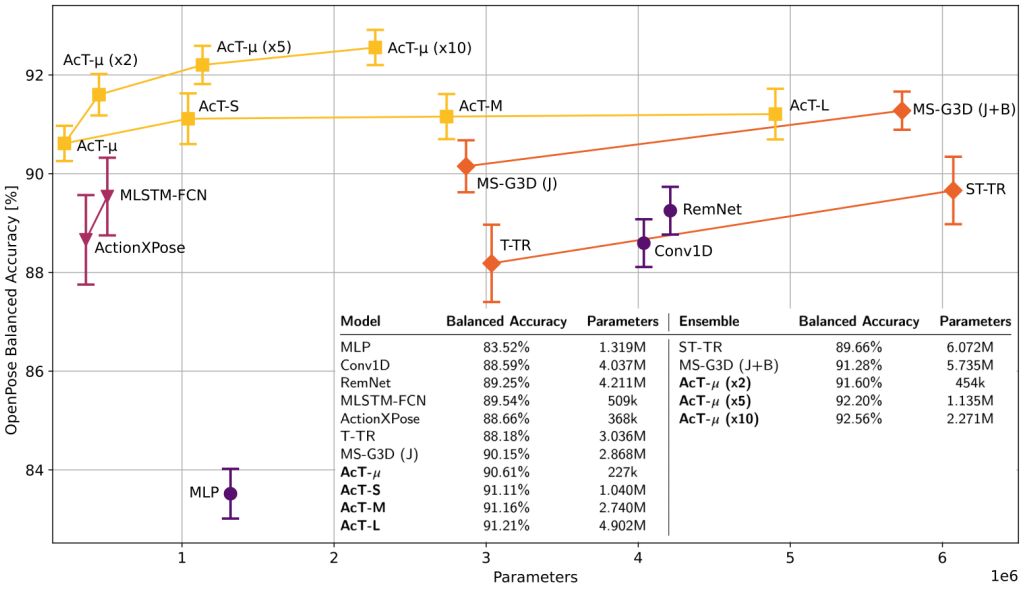
MPOSE2021 Split		OpenPose 1		OpenPose 2		OpenPose 3	
Model	**Parameters**	**Accuracy [%]**	**Balanced [%]**	**Accuracy [%]**	**Balanced [%]**	**Accuracy [%]**	**Balanced [%]**
MLP	1,319k	82.66 ± 0.33	74.56 ± 0.56	84.41 ± 0.60	74.58 ± 1.0	83.48 ± 0.58	76.60 ± 0.77
Conv1D	4,037k	88.18 ± 0.64	81.97 ± 1.4	88.93 ± 0.43	80.49 ± 0.95	88.67 ± 0.38	83.93 ± 0.58
REMNet [1]	4,211k	89.18 ± 0.51	84.20 ± 0.84	88.77 ± 0.35	80.29 ± 0.88	89.80 ± 0.59	86.18 ± 0.40
ActionXPose [2]	509k	87.60 ± 0.98	82.13 ± 1.5	88.42 ± 0.70	81.28 ± 1.4	89.96 ± 1.0	86.65 ± 1.6
MLSTM-FCN [26]	368k	88.62 ± 0.74	83.55 ± 0.88	90.19 ± 0.68	83.84 ± 1.2	89.80 ± 0.94	87.33 ± 0.67
T-TR [35]	3,036k	87.72 ± 0.87	81.99 ± 1.64	88.14 ± 0.53	80.23 ± 1.19	88.69 ± 0.95	85.03 ± 1.60
MS-G3D (J) [30]	2,868k	89.90 ± 0.50	85.29 ± 0.98	90.16 ± 0.64	83.08 ± 1.1	90.39 ± 0.44	87.48 ± 1.2
AcT-μ	227k	90.86 ± 0.36	86.86 ± 0.50	91.00 ± 0.24	85.01 ± 0.51	89.98 ± 0.47	87.63 ± 0.54
AcT-S	1,040k	91.21 ± 0.48	87.48 ± 0.76	91.23 ± 0.19	85.66 ± 0.58	90.90 ± 0.87	88.61 ± 0.73
AcT-M	2,740k	**91.38 ± 0.32**	**87.70 ± 0.47**	91.08 ± 0.48	85.18 ± 0.80	91.01 ± 0.57	88.63 ± 0.51
AcT-L	4,902k	91.11 ± 0.32	87.27 ± 0.46	**91.46 ± 0.42**	**85.92 ± 0.63**	**91.05 ± 0.80**	**89.00 ± 0.74**
ST-TR [35]	6,072k	89.20 ± 0.71	83.95 ± 1.11	89.29 ± 0.81	81.53 ± 1.39	90.49 ± 0.53	87.06 ± 0.70
MS-G3D (J+B) [30]	5,735k	91.13 ± 0.33	87.25 ± 0.50	91.28 ± 0.29	85.10 ± 0.50	91.42 ± 0.54	89.66 ± 0.55
AcT-μ (x2)	454k	91.76 ± 0.29	88.27 ± 0.37	91.34 ± 0.40	86.88 ± 0.48	91.70 ± 0.57	88.87 ± 0.37
AcT-μ (x5)	1,135k	92.43 ± 0.24	89.33 ± 0.31	91.55 ± 0.37	87.80 ± 0.39	92.63 ± 0.55	89.77 ± 0.35
AcT-μ (x10)	2,271k	**92.54 ± 0.21**	**89.79 ± 0.34**	**92.03 ± 0.33**	**88.02 ± 0.31**	**93.10 ± 0.53**	**90.22 ± 0.31**

5.1. Experimental Settings

In the following experiments, we employ both the OpenPose and PoseNet versions of the MPOSE2021 dataset. Either set of data has $T = 30$ and $P = 52$ or 68 features, respectively. In particular, for OpenPose we follow the same preprocessing of [2] obtaining 13 keypoints with four parameters each (position x, y and velocities v_x, v_y), while PoseNet samples contain 17 keypoints with the same information. The training set is composed of 9421 samples and 2867 for testing. The remaining 3141 instances are used as validation to find the most promising hyperparameters with a grid search analysis. All results, training, and testing code for the AcT model are open source and publicly available[2].

In Table 1 are summarized all the hyperparameters for the four versions of the AcT architecture and in Table 2 all settings related to the training procedure. The AdamW optimization algorithm [31] is employed for all training with the same scheduling proposed in [48], but with a step drop of the learning rate λ to 1e-4 at a fixed percentage (80%) of the total number of epochs. We employ the TensorFlow 2[3] framework to train the proposed network on a PC with 32-GB RAM, an Intel i7-9700K CPU, and an Nvidia 2080 Super GP-GPU. Following the previously defined benchmark strategy, the total training procedure for the four versions takes approximately 32 hours over the three different splits.

[2] https://github.com/PIC4SeR/AcT
[3] https://www.tensorflow.org



Model	Balanced Accuracy	Parameters
MLP	83.52%	1.319M
Conv1D	88.59%	4.037M
RemNet	89.25%	4.211M
MLSTM-FCN	89.54%	509k
ActionXPose	88.66%	368k
T-TR	88.18%	3.036M
MS-G3D (J)	90.15%	2.868M
AcT-μ	90.61%	227k
AcT-S	91.11%	1.040M
AcT-M	91.16%	2.740M
AcT-L	91.21%	4.902M

Ensemble	Balanced Accuracy	Parameters
ST-TR	89.66%	6.072M
MS-G3D (J+B)	91.28%	5.735M
AcT-μ (x2)	91.60%	454k
AcT-μ (x5)	92.20%	1.135M
AcT-μ (x10)	92.56%	2.271M

Figure 6: Visual representation of the benchmark of different models for short-time HAR on MPOSE2021 splits using OpenPose 2D skeletal representations. For brevity and clearness, the average balanced accuracy on the three splits of MPOSE2021 is reported. The lines connect models that use the same methodology.

Table 4
Benchmark of different models for short-time HAR on MPOSE2021 splits using PoseNet 2D skeletal representations.

MPOSE2021 Split		PoseNet 1		PoseNet 2		PoseNet 3	
Model	**Parameters**	**Accuracy [%]**	**Balanced [%]**	**Accuracy [%]**	**Balanced [%]**	**Accuracy [%]**	**Balanced [%]**
Conv1D	4,062k	85.83 ± 0.71	79.96 ± 1.1	87.47 ± 0.35	78.51 ± 0.78	87.46 ± 0.67	81.31 ± 0.58
REMNet [1]	4,269k	84.75 ± 0.65	77.23 ± 0.94	86.17 ± 0.68	75.79 ± 1.3	86.31 ± 0.60	79.20 ± 0.79
ActionXPose [2]	509k	75.98 ± 0.72	64.47 ± 1.1	79.94 ± 1.1	67.05 ± 1.4	77.34 ± 1.4	66.86 ± 1.4
MLSTM-FCN [26]	368k	76.17 ± 0.84	64.75 ± 1.1	79.04 ± 0.72	65.62 ± 1.4	77.84 ± 1.3	67.05 ± 1.2
AcT-μ	228k	86.66 ± 1,10	81.56 ± 1.60	87.21 ± 0.99	79.21 ± 1.60	87.75 ± 0.53	82.99 ± 0,87
AcT-S	1,042k	**87.63 ± 0.52**	**82.54 ± 0.87**	88.48 ± 0.57	81.53 ± 0.68	88.49 ± 0.65	83.63 ± 0.99
AcT-M	2,743k	87.23 ± 0.48	82.10 ± 0.66	**88.50 ± 0.51**	**81.79 ± 0.44**	**88.70 ± 0.57**	**83.92 ± 0.96**

We exploit publicly available code for what concerns other state-of-the-art models and use the same hyperparameters and optimizer settings described by the authors in almost all the cases. The only exception is made for learning rate, epochs number, and batch size to adapt the methodologies to our dataset and obtain better learning curves.

5.2. Action Recognition on MPOSE2021

We extensively experiment on MPOSE2021 considering some baselines, common HAR architectures, and our proposed AcT models. We report the mean and standard deviation of 10 models trained using different validation

splits to obtain statistically relevant results. The validation splits are constant for all the models and correspond to 10% of the train set, maintaining the same class distribution. The benchmark is executed for both OpenPose and PoseNet data and repeated for all the three train/test splits provided by MPOSE2021. The baselines chosen for the benchmark are a Multilayer Perceptron (MLP), a fully convolutional model (Conv1D), and REMNet, which is a more sophisticated convolutional network with attention and residual blocks proposed in [1] for time series feature extraction. In particular, the MLP is designed as a stack of three fully connected (FC) layers with 512 neurons each, followed by a dropout layer and a final FC layer with as many output nodes as classes. Instead, the Conv1D model is built concatenating five 1D convolutional layers with 512 filters alternated with batch normalization stages and followed by a global average pooling operator, a dropout layer, and an FC output stage as in the MLP. Finally, the configuration used for REMNet consists of two Residual Reduction Modules (RRM) with a filter number of 512, followed by dropout and the same FC output layer as in the other baselines.

Regarding state-of-the-art comparisons, four popular models used for multivariate time series classification, and in particular HAR, are reproduced and tested. Among those, MLSTM-FCN [26] combines convolutions, spacial attention, and an LSTM block, and its improved version ActionXPose [2] uses additional preprocessing, leading the model to exploit more correlations in data and, hence, be more robust against noisy or missing pose detections. On the other hand, MS-G3D [30] uses spatial-temporal graph convolutions to make the model aware of spatial relations between skeleton keypoints, while ST-TR [35] joins graph convolutions with Transformer-based self-attention applied to both space and time. As the last two solutions also propose a model ensemble, these results are further compared to AcT ensembles made of 2, 5, and 10 single-shot models. We also report the achieved balanced accuracy for each model and use it as the primary evaluation metric to account for the uneven distribution of classes.

The results of the experimentation for OpenPose are reported in Table 3 and, in synthesis, in Fig.6. The fully convolutional baseline strongly outperforms the MLP, while REMNet proves that introducing attention and residual blocks further increases accuracy. As regards MLSTM-FCN and ActionXPose, it is evident that explicitly modeling both spatial and temporal correlations, made possible by the two separate branches, slightly improves the understanding of actions with respect to models like REMNet. MS-G3D, in its joint-only (J) version, brings further accuracy improvement by exploiting graph convolutions and giving the network information on the spatial relationship between keypoints. On the other hand, ST-TR shows performance comparable to all other single-shot models, despite, as MS-G3D, taking advantage of graph information.

The proposed AcT model demonstrates the potential of pure Transformer-based architectures, as all four versions outperform other methodologies while also showing smaller standard deviations. Moreover, even the smallest AcT-μ (227k parameters) is able to extract general and robust features from temporal correlations in sequences. Increasing the number of parameters, a constant improvement in balanced accuracy can be observed for splits 3, while split 1

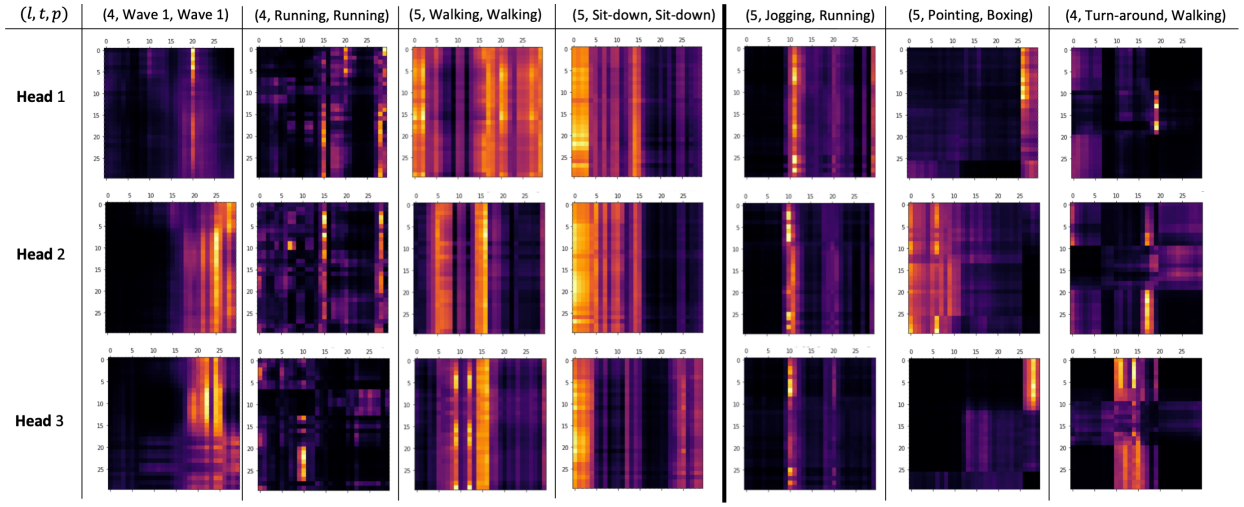


Figure 7: Self-attention weights $\boldsymbol{A}$ of MPOSE2021 test samples. (l, t, p) represents the AcT-M l-th layer, the true label and the prediction respectively. The three rightmost columns show three attention maps of a failed prediction and the other columns are from correct classifications. It is clear from all examples how the model focuses on certain particular frames of the series in order to extract a global representation of the scene.

and 2 present oscillations. The difference between splits reflects how much information is conveyed by training sets and how much the model can learn from that. So, it is evident that AcT scales best on split three because it presents complex correlations that a bigger model learns more easily. On the contrary, it seems AcT-μ is able to extract almost all the information relevant for generalization from split 2, as the accuracy only slightly increases going towards more complex models.

As explained in Section 2, ST-TR exploits an ensemble of two networks modeling spatial and temporal sequence correlations, respectively. Moreover, MS-G3D leverages further information such as skeleton graph connections and the position of bones in one of the ensembled networks. Ensembles are very effective in reducing model variance and enhancing performance by exploiting independent representations learned by each network, so it is unfair to compare them with single architectures. For this reason, we create three ensemble versions of AcT-μ to have an even confront, with 2, 5, and 10 instances, respectively. To compute ensemble predictions, we average the output logits of the network instances before passing through a softmax function. The results reported at the bottom of Table 3 show that AcT-μ (x2) outperforms MS-G3D (J+B) in all the benchmarks except for balanced accuracy in split 3, despite having less than one-tenth of its parameters. Finally, the ensembles AcT-μ (x5) and AcT-μ (x10), made of 5 and 10 instances respectively, achieve even higher accuracy on all the splits with only around 1 to 2 million parameters. That proves how the balancing effect of the ensemble enhances model predictions even without feeding the network additional information.

As PoseNet data is mainly dedicated to real-time and Edge AI applications, only the models designed for this purpose have been considered in the benchmark, excluding MS-G3D, ST-TR, and AcT-L. In general, the results give

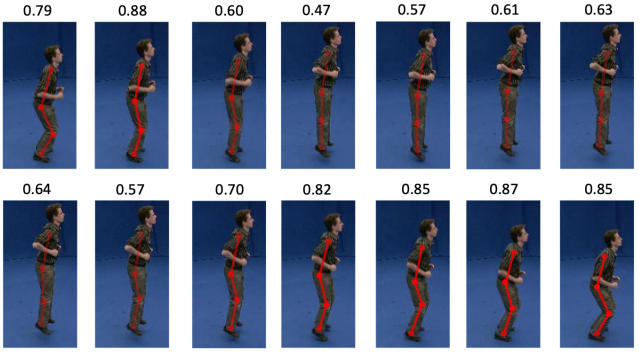


Figure 8: Self-attention of the [CLS] token, computed as the normalized sum of the last layer of the different heads. Scores give a direct insight into the frames exploited by AcT to produce the classification output. The example clearly shows how bending positions are more insightful for the network to predict the jumping-in-place action. In the image, the attention score defines the skeleton alpha channel.

similar insights. The tested models are the same as the previous case, with all the necessary modifications given by the different input formats. In the MLP case, however, performance seriously degrades as networks strongly tend to overfit input data after a small number of epochs, so the results are not included in Table 4. That is caused by the fact that PoseNet is a lighter methodology developed for Edge AI, and hence noisy and even missing keypoint detections are more frequent. That results in less informative data and emphasizes the difference between sequences belonging to different sub-datasets, confusing the model and inducing it to learn very specific but unusable features. The MLP is too simple and particularly prone to this kind of problem. Naturally, all the models are affected by the same problem, and the balanced accuracy on PoseNet is generally lower. The same considerations made for OpenPose apply in this case, where AcT outperforms all the other architectures and demonstrates its ability to give an accurate and robust representation of temporal correlations. Also, it is interesting to notice that Conv1D performs better than REMNet, proving to be less prone to overfitting, and that standard deviations are more significant than in the OpenPose case.

5.3. Model Introspection

In order to have an insight into the frames of the sequence the AcT model attends to, we extract the self-attention weights at different stages of the network. In Fig.7, MPOSE2021 test samples are propagated through the AcT-M model, and attention weights $\boldsymbol{A}$ of the three distinct heads are presented. It can be seen that the model pays attention to specific frames of the sequence when a specific gesture defines the action. On the other hand, attention is much more spread across the different frames for more distributed actions such as walking and running. Moreover, it is clear how the three heads mostly focus on diverse frames of the sequence. Finally, the rightmost columns show three attention maps of failed predictions. In these last cases, attention weights are less coherent, and the model cannot extract a valid global representation of the scene.

Instead, in Fig.8, the last-layer self-attention scores of the [CLS] token are shown together with the RGB and

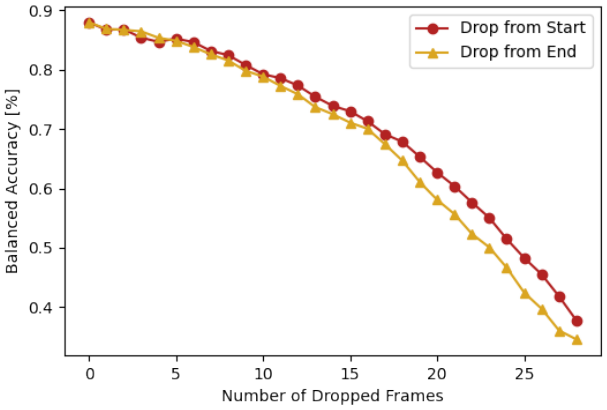


Figure 9: AcT-M balanced accuracy with an incremental reduction of temporal information. Due to the intrinsic nature of the network, it is possible to reduce the number of temporal steps without a retraining or any kind of explicit adaptation.

skeleton representations of the scene. The scores are computed as the normalized sum of the three attention heads and give a direct insight into the frames exploited by AcT to produce the classification output. It can be seen that bent poses are much more informative for the model to predict the jumping-in-place action.

Moreover, we analyze the behavior of the network under a progressive reduction of temporal information. That can be easily done without retraining due to the intrinsic nature of AcT. In Fig.9, we present how the test-set balanced accuracy is affected by frame dropping. The two curves show a reduction starting from the beginning and the end of the temporal sequence, respectively. It is interesting to notice how the performance of AcT degrades with an almost linear trend. That highlights the robustness of the proposed methodology and demonstrates the possibility of adapting the model to applications with different temporal constraints.

Finally, we also study the positional embeddings of the AcT-M model by analyzing their cosine similarity, as shown in Fig.10. Very nearby position embeddings demonstrate a high level of similarity, and distant ones are orthogonal or in the opposite direction. This pattern is constant for all T frames of the sequence, highlighting how actions are not particularly localized and that relative positions are essential for all the frames.

5.4. Latency Measurements

We test the performance of all the considered models for real-time applications. To do so, we use the TFLite Benchmark[4] tool, which allows running TensorFlow Lite models on different computing systems and collecting statistical results on latency and memory usage. In our case, two CPUs are employed to measure model speed both on a PC and a mobile phone: an Intel i7-9700K for the former and the ARM-based HiSilicon Kirin 970 for the latter. In both experiments, the benchmark executes 10 warm-up runs followed by 100 consecutive forward passes, using 8 threads.

[4]https://www.tensorflow.org/lite/performance/measurement

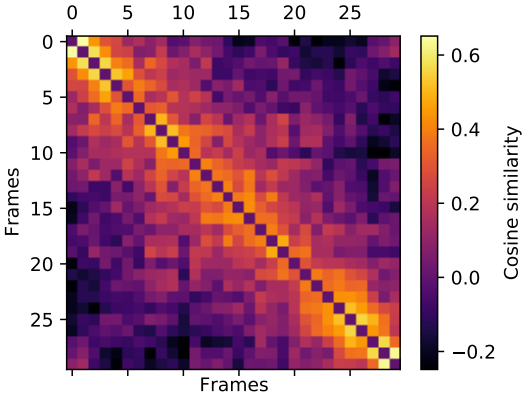


Figure 10: Cosine similarities of the learned T position embeddings of AcT-M model.

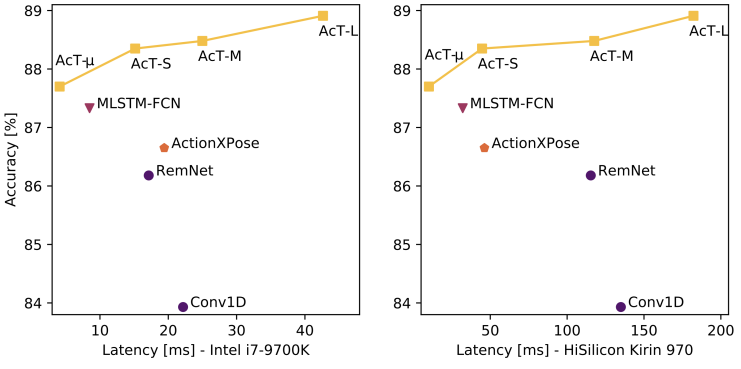


Figure 11: Study of the latency of different tested models on a high-performance Intel CPU and on a mobile phone equipped with an ARM-based CPU.

The results of both tests are reported in Fig. 11, where only the MLP has been ignored because, despite being the fastest-running model, its accuracy results are much lower than its competitors. The graph shows the great computational efficiency of Transformer-based architectures, whereas convolutional and recurrent networks result in heavier CPU usage. Indeed, in the Intel i7 case, REMNet achieves almost the same speed as AcT-S, but its accuracy is 2% lower. Moreover, AcT-μ is able to outperform REMNet, running at over four times its speed. MLSTM-FCN and ActionXPose, being smaller models, achieve lower latencies than the baselines: the former stays between AcT-μ and Act-S, while the latter performs similarly to AcT-S. Those results are remarkable but still outperformed by AcT-μ both on accuracy and speed.

The difference with the baselines is even more evident on the ARM-based chip, as convolutional architectures seem to perform poorly on this kind of hardware. Indeed, REMNet and Conv1D run as fast as AcT-M with significantly lower accuracies, and AcT-μ is ten times quicker. Nothing changes for what concerns MLSTM-FCN and ActionXPose, less accurate and three times slower than AcT-μ.

6. Conclusion and Future Work

In this paper, we explored the direct application of a purely Transformer-based network to human action recognition. We introduced the AcT network, which significantly outperforms commonly adopted models for HAR with a simple and fully self-attentional architecture. In order to limit computational and power requests, building on previous HAR and pose estimation research, the proposed methodology exploits 2D skeletal representations of short time sequences, providing an accurate and low latency solution for real-time applications. Moreover, we introduced MPOSE2021, a large-scale open-source dataset for short-time human action recognition, as an attempt to build a formal benchmark for future research on the topic. Extensive experimentation with our methodology clearly demonstrates the effectiveness of AcT and poses the basis for meaningful impact on many practical computer vision applications. In particular, the remarkable efficiency of AcT could be exploited for Edge AI, achieving good performance even on computationally limited devices. Future work may further investigate the AcT architecture with 3D skeletons and long sequence inputs. Moreover, relational information between points of the estimated pose are currently scarcely exploited by our methodology. So, future investigations may attempt to include skeleton graphs as prior knowledge in the positional embedding of the network without impacting latency.

Acknowledgements

This work has been developed with the contribution of the Politecnico di Torino Interdepartmental Center for Service Robotics PIC4SeR[5] and SmartData@Polito[6].

References

[1] Angarano, S., Mazzia, V., Salvetti, F., Fantin, G., Chiaberge, M., 2021. Robust ultra-wideband range error mitigation with deep learning at the edge. Engineering Applications of Artificial Intelligence 102, 104278.

[2] Angelini, F., Fu, Z., Long, Y., Shao, L., Naqvi, S.M., 2020. 2d pose-based real-time human action recognition with occlusion-handling. IEEE Transactions on Multimedia 22, 1433–1446.

[3] Angelini, F., Naqvi, S.M., 2019. Joint rgb-pose based human action recognition for anomaly detection applications, in: 2019 22th International Conference on Information Fusion (FUSION), IEEE. pp. 1–7.

[4] Angelini, F., Yan, J., Naqvi, S.M., 2019. Privacy-preserving online human behaviour anomaly detection based on body movements and objects positions, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE. pp. 8444–8448.

[5] Ba, L.J., Kiros, J.R., Hinton, J.E., 2016. Layer normalization. CoRR abs/1607.06450.

[6] Bello, I., Zoph, B., Vaswani, A., Shlens, J., Le, Q.V., 2019. Attention augmented convolutional networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3286–3295.

[5] https://pic4ser.polito.it
[6] https://smartdata.polito.it

[7] Ben-Younes, H., Zablocki, É., Pérez, P., Cord, M., 2021. Driving behavior explanation with multi-level fusion. Pattern Recognition , 108421.

[8] Berg, A., O'Connor, M., Cruz, M.T., 2021. Keyword Transformer: A Self-Attention Model for Keyword Spotting, in: Proc. Interspeech 2021, pp. 4249–4253.

[9] Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y., 2019. Openpose: realtime multi-person 2d pose estimation using part affinity fields. IEEE transactions on pattern analysis and machine intelligence 43, 172–186.

[10] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A., 2021. Emerging properties in self-supervised vision transformers. CoRR abs/2104.14294.

[11] Carreira, J., Zisserman, A., 2017. Quo vadis, action recognition? a new model and the kinetics dataset, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6299–6308.

[12] Chen, C., Jafari, R., Kehtarnavaz, N., 2015. Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor, in: 2015 IEEE International conference on image processing (ICIP), IEEE. pp. 168–172.

[13] Chen, M.X., Firat, O., Bapna, A., Johnson, M., Macherey, W., Foster, G.F., Jones, L., Parmar, N., Schuster, M., Chen, Z., Wu, Y., Hughes, M., 2018. The best of both worlds: Combining recent advances in neural machine translation. CoRR abs/1804.09849.

[14] Cho, S., Maqbool, M., Liu, F., Foroosh, H., 2020. Self-attention network for skeleton-based human action recognition, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 635–644.

[15] D'Ascoli, S., Touvron, H., Leavitt, M.L., Morcos, A.S., Biroli, G., Sagun, L., 2021. Convit: Improving vision transformers with soft convolutional inductive biases, in: Meila, M., Zhang, T. (Eds.), Proceedings of the 38th International Conference on Machine Learning, PMLR. pp. 2286–2296.

[16] Devlin, J., Chang, M., Lee, K., Toutanova, K., 2019. BERT: pre-training of deep bidirectional transformers for language understanding, in: Burstein, J., Doran, C., Solorio, T. (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics. pp. 4171–4186.

[17] Dong, L., Xu, S., Xu, B., 2018. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition, in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5884–5888. doi:10.1109/ICASSP.2018.8462506.

[18] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An image is worth 16x16 words: Transformers for image recognition at scale, in: International Conference on Learning Representations.

[19] Gkalelis, N., Kim, H., Hilton, A., Nikolaidis, N., Pitas, I., 2009. The i3dpost multi-view and 3d human action/interaction database, in: 2009 Conference for Visual Media Production, IEEE. pp. 159–168.

[20] Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R., 2007. Actions as space-time shapes. IEEE transactions on pattern analysis and machine intelligence 29, 2247–2253.

[21] Hao, W., Zhang, Z., 2019. Spatiotemporal distilled dense-connectivity network for video action recognition. Pattern Recognition 92, 13–24.

[22] Hendrycks, D., Gimpel, K., 2016. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 .

[23] Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation 9, 1735–1780.

[24] Hu, J., Shen, L., Sun, G., 2018. Squeeze-and-excitation networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141.

[25] Huang, L., Huang, Y., Ouyang, W., Wang, L., 2019. Part-aligned pose-guided recurrent network for action recognition. Pattern Recognition 92, 165–176.

[26] Karim, F., Majumdar, S., Darabi, H., Harford, S., 2019. Multivariate lstm-fcns for time series classification. Neural Networks 116, 237–245.

[27] Langmann, B., Hartmann, K., Loffeld, O., 2012. Depth camera technology comparison and performance evaluation., in: ICPRAM (2), pp. 438–444.

[28] Li, J., Liu, X., Zhang, M., Wang, D., 2020. Spatio-temporal deformable 3d convnets with attention for action recognition. Pattern Recognition 98, 107037.

[29] Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C., 2019. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence 42, 2684–2701.

[30] Liu, Z., Zhang, H., Chen, Z., Wang, Z., Ouyang, W., 2020. Disentangling and unifying graph convolutions for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 143–152.

[31] Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization, in: International Conference on Learning Representations.

[32] Luvizon, D., Picard, D., Tabia, H., 2020. Multi-task deep learning for real-time 3d human pose estimation and action recognition. IEEE transactions on pattern analysis and machine intelligence .

[33] Martin, M., Roitberg, A., Haurilet, M., Horne, M., Reiß, S., Voit, M., Stiefelhagen, R., 2019. Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2801–2810.

[34] Papandreou, G., Zhu, T., Chen, L.C., Gidaris, S., Tompson, J., Murphy, K., 2018. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model, in: Proceedings of the European Conference on Computer Vision (ECCV), pp. 269–286.

[35] Plizzari, C., Cannici, M., Matteucci, M., 2021. Skeleton-based action recognition via spatial and temporal transformer networks. Computer Vision and Image Understanding 208, 103219.

[36] Rodríguez-Moreno, I., Martínez-Otzeta, J.M., Goienetxea, I., Rodriguez-Rodriguez, I., Sierra, B., 2020. Shedding light on people action recognition in social robotics by means of common spatial patterns. Sensors 20, 2436.

[37] Schuldt, C., Laptev, I., Caputo, B., 2004. Recognizing human actions: a local svm approach, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., IEEE. pp. 32–36.

[38] Shahroudy, A., Liu, J., Ng, T.T., Wang, G., 2016. Ntu rgb+ d: A large scale dataset for 3d human activity analysis, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1010–1019.

[39] Shi, L., Zhang, Y., Cheng, J., Lu, H., 2019a. Skeleton-based action recognition with directed graph neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7912–7921.

[40] Shi, L., Zhang, Y., Cheng, J., Lu, H., 2019b. Two-stream adaptive graph convolutional networks for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12026–12035.

[41] Si, C., Jing, Y., Wang, W., Wang, L., Tan, T., 2020. Skeleton-based action recognition with hierarchical spatial reasoning and temporal stack learning network. Pattern Recognition 107, 107511.

[42] Song, L., Yu, G., Yuan, J., Liu, Z., 2021. Human pose estimation and its application to action recognition: A survey. Journal of Visual Communication and Image Representation , 103055.

[43] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research 15, 1929–1958.

[44] Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jegou, H., 2021. Training data-efficient image transformers & distillation through attention, in: Meila, M., Zhang, T. (Eds.), Proceedings of the 38th International Conference on Machine Learning, PMLR. pp. 10347–

10357.

[45] Tu, Y., Zhou, C., Guo, J., Gao, S., Yu, Z., 2021. Enhancing the alignment between target words and corresponding frames for video captioning. Pattern Recognition 111, 107702.

[46] Vallathan, G., John, A., Thirumalai, C., Mohan, S., Srivastava, G., Lin, J.C.W., 2021. Suspicious activity detection using deep learning in secure assisted living iot environments. The Journal of Supercomputing 77, 3242–3260.

[47] Varol, G., Laptev, I., Schmid, C., 2018. Long-term temporal convolutions for action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 40, 1510–1517.

[48] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I., 2017. Attention is all you need, in: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc.

[49] Wan, B., Jiang, W., Fang, Y.M., Zhu, M., Li, Q., Liu, Y., 2022. Revisiting image captioning via maximum discrepancy competition. Pattern Recognition 122, 108358.

[50] Wang, X., Srivastava, G., 2021. The security of vulnerable senior citizens through dynamically sensed signal acquisition. Transactions on Emerging Telecommunications Technologies , e4037.

[51] Weinland, D., Ronfard, R., Boyer, E., 2006. Free viewpoint action recognition using motion history volumes. Computer vision and image understanding 104, 249–257.

[52] Xia, L., Chen, C.C., Aggarwal, J.K., 2012. View invariant human action recognition using histograms of 3d joints, in: 2012 IEEE computer society conference on computer vision and pattern recognition workshops, IEEE. pp. 20–27.

[53] Yan, S., Xiong, Y., Lin, D., 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition, in: Proceedings of the AAAI conference on artificial intelligence, pp. 1113–1122.

[54] Yang, F., Yang, H., Fu, J., Lu, H., Guo, B., 2020. Learning texture transformer network for image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5791–5800.

[55] Zhou, L., Zhou, Y., Corso, J.J., Socher, R., Xiong, C., 2018. End-to-end dense video captioning with masked transformer, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8739–8748.

[56] Zhu, H., Liu, H., Zhu, C., Deng, Z., Sun, X., 2020. Learning spatial-temporal deformable networks for unconstrained face alignment and tracking in videos. Pattern Recognition 107, 107354.



Vittorio Mazzia is a Ph.D. student in Electrical, Electronics, and Communications Engineering working with the two Interdepartmental Centres PIC4SeR (https://pic4ser.polito.it/) and SmartData (https://smartdata.polito.it/). He received a master's degree in Mechatronic Engineering from Politecnico di Torino, presenting a thesis entitled "Use of deep learning for low-cost automatic detection of cracks in tunnels", developed in collaboration with the California State University. His current research interests involve deep learning applied to different computer vision tasks, autonomous navigation for service robotics, and reinforcement learning. Moreover, he is currently working on machine learning algorithms and their embedded implementation for AI at the edge using neural compute devices (like Jetson Xavier, Jetson Nano, Movidius Neural Stick) for hardware acceleration.



Simone Angarano is a Ph.D. student in Electrical, Electronics, and Communications Engineering and member of the Interdepartmental Center for Service Robotics at Politecnico di Torino (https://pic4ser.polito.it/). At the same university, he achieved a Bachelor's Degree in Electronic Engineering in 2018 and a Master's Degree in Mechatronic Engineering in 2020, presenting the thesis "Deep Learning Methodologies for UWB Ranging Error Compensation". His research topic is Deep Learning for service robotics applications, spanning computer vision, navigation, and control. Moreover, he also focuses on model optimization for real-time applications and edge AI.



Francesco Salvetti is currently a Ph.D. student in Electrical, Electronics and Communications Engineering in collaboration with the interdepartmental centers PIC4SeR (https://pic4ser.polito.it/) and Smart Data (https://smartdata.polito.it/) at Politecnico di Torino, Italy. He received his Bachelor's Degree in Electronic Engineering in 2017 and his Master's Degree in Mechatronic Engineering in 2019 at Politecnico di Torino. He is currently working on Machine Learning applied to Computer Vision and Image Processing in robotics applications.



Federico Angelini is currently a Research Associate in the ICOS Group, School of Computing, Newcastle University, UK. He works on machine learning methods for data analysis applied to Osteoarthritis disease stratification and progression. He received his PhD in Electrical and Electronic Engineering within the Intelligent Sensing and Communications (ISC) Research Group, School of Engineering, Newcastle University, UK. His PhD work focused on deep learning methods for video-based human action recognition and anomaly detection. He collaborated with the National Research Council (CNR), Rome, Italy, on the image denoising methods projects in 2015–2016. His research interests include data analysis of medical data, human action recognition via multimodal sensors, machine learning, deep learning, and signal processing.



Marcello Chiaberge is currently an Associate Professor within the Department of Electronics and Telecommunications, Politecnico di Torino, Turin, Italy. He is also the Co-Director of the Mechatronics Lab, Politecnico di Torino (www.lim.polito.it), Turin, and the Director and the Principal Investigator of the Interdepartmental Center for Service Robotics (PIC4SeR, https://pic4ser.polito.it/), Turin. He has authored more than 100 articles accepted in international conferences and journals, and he is the co-author of nine international patents. His research interests include hardware implementation of neural networks and fuzzy systems and the design and implementation of reconfigurable real-time computing architectures.