UNIK: A Unified Framework for Real-world Skeleton-based Action Recognition

Di Yang, Yaohui Wang, Antitza Dantcheva, Lorenzo Garattoni, Gianpiero Francesca, Francois F Bremond

UNIK: A Unified Framework for Real-world Skeleton-based Action Recognition

Di Yang*[1]
di.yang@inria.fr
Yaohui Wang*[1]
yaohui.wang@inria.fr
Antitza Dantcheva[1]
antitza.dantcheva@inria.fr
Lorenzo Garattoni[2]
lorenzo.garattoni@toyota-europe.com
Gianpiero Francesca[2]
gianpiero.francesca@toyota-europe.com
François Brémond[1]
francois.bremond@inria.fr

[1] Inria
Université Côte d'Azur
Valbonne, France

[2] Toyota Motor Europe
Brussels, Belgium

Abstract

Action recognition based on skeleton data has recently witnessed increasing attention and progress. State-of-the-art approaches adopting Graph Convolutional networks (GCNs) can effectively extract features on human skeletons relying on the pre-defined human topology. Despite associated progress, GCN-based methods have difficulties to generalize across domains, especially with different human topological structures. In this context, we introduce UNIK, a novel topology-free skeleton-based action recognition method that is not only effective to learn spatio-temporal features on human skeleton sequences but also able to generalize across datasets. This is achieved by learning an optimal dependency matrix from the uniform distribution based on a multi-head attention mechanism. Subsequently, to study the cross-domain generalizability of skeleton-based action recognition in real-world videos, we re-evaluate state-of-the-art approaches as well as the proposed UNIK in light of a novel Posetics dataset. This dataset is created from Kinetics-400 videos by estimating, refining and filtering poses. We provide an analysis on how much performance improves on the smaller benchmark datasets after pre-training on Posetics for the action classification task. Experimental results show that the proposed UNIK, with pre-training on Posetics, generalizes well and outperforms state-of-the-art when transferred onto four target action classification datasets: Toyota Smarthome, Penn Action, NTU-RGB+D 60 and NTU-RGB+D 120.

1 Introduction

As skeleton-based human action recognition methods rely on 2D or 3D positions of human key joints only, they are able to filter out noise caused, for instance, by background clutter, changing light conditions, and to focus on the action being performed [1, 10, 18, 21, 24, 27, 32, 33, 34, 36, 38, 40, 42, 43, 45, 47]. Recent approaches, namely Graph Convolutional Networks (GCNs) [43], models human joints, as well as their natural connections

* indicates equal contribution. Project website: https://yangdi666.github.io/UNIK-project/

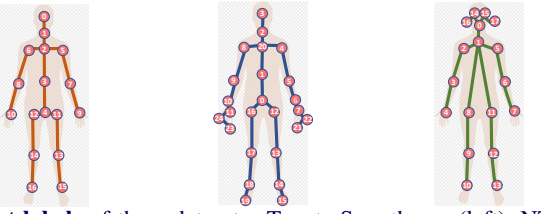

Figure 1: **Human joint labels** of three datasets: Toyota Smarthome (left), NTU-RGB+D (middle) and Kinetics-Skeleton (right). We note the different numbers, orders and locations of joints.

(*i.e.*, bones) in skeleton spatio-temporal graphs to carry both, spatial and temporal inferences. Consequently, several successors, namely Adaptive GCNs (AGCNs), with optimized graph construction strategies to extract multi-scale structural features and long-range dependencies have been proposed and shown encouraging results. Promising examples are graph convolutions with learnable *adjacency matrix* [33], higher-order polynomials of *adjacency matrix* [21] and separate multi-scale subsets of *adjacency matrix* [24]. All these *adjacency matrices* are manually pre-defined to represent the relationships between joints according to human topology. Nevertheless, compared with RGB-based methods such as spatio-temporal Convolutional Neural Networks (CNNs) [3, 12] that are pre-trained on Kinetics [3] to boost accuracy in downstream datasets and tasks, GCN-based models are limited because they are always trained individually on the target dataset (often small) from scratch. Our insight is that, the generalization abilities of these approaches are hindered by the need for different adaptive adjacency matrices when different topological human structures are used (*e.g.*, joints number, joints order, bones), as in the case of the three datasets of Fig. 1. However, we note that such adaptive sparse *adjacency matrices* are transformed to fully dense matrices in deeper layers in order to capture long-range dependencies between joints. This new structure contradicts the initial and original topological skeleton structure.

Based on these considerations and as the human-intrinsic graph representation is deeply modified during training, we hypothesize that there should be a more optimized and generic initialization strategy that can replace the *adjacency matrix*. To validate this hypothesis, we introduce UNIK, a novel unified framework for skeleton-based action recognition. In UNIK, the *adjacency matrix* is initialized into a uniformly distributed *dependency matrix* where each element represents the dependency weight between the corresponding pair of joints. Subsequently, a multi-head aggregation is performed to learn and aggregate multiple dependency matrices by different attention maps. This mechanism jointly leverages information from several representation sub-spaces at different positions of the *dependency matrix* to effectively learn the spatio-temporal features on skeletons. The proposed UNIK does not rely on any topology related to the human skeleton, makes it much easier to transfer onto other skeleton datasets. This opens up a great design space to further improve the recognition performance by transferring a model pre-trained on a sufficiently large dataset.

In addition, another reason for poor generalization abilities is that many skeleton datasets have been captured in lab environments with RGBD sensors (*e.g.*, NTU-RGB+D [23, 31]). Then, the action recognition accuracy significantly decreases, when the pre-trained models on the sensor data are transferred to the real-world videos, where skeleton data encounters a number of occlusions and truncations of the body. To address this, we create Posetics dataset by estimating and refining poses, as well as filtering, purifying and categorizing videos and annotations based on the real-world Kinetics-400 [3] dataset. To this aim, we apply multi-expert pose estimators [2, 6, 29] and a refinement algorithm [14]. Our experimental analysis confirms: pre-training on Posetics improves state-of-the-art skeleton-based action recognition methods, when transferred and fine-tuned on all evaluated datasets [4, 23, 31, 46].

In summary, the contributions of this paper are: (i) we go beyond GCN-based architec-

tures by proposing UNIK with a novel design strategy by adopting dependency matrices and a multi-head attention mechanism for skeleton-based action recognition. (ii) We revisit real-world skeleton-based action recognition focusing on cross-domain transfer learning. The study is conducted on four target datasets with pre-training on Posetics, a novel and large-scale action classification dataset that features higher quality skeleton detections based on Kinetics-400. (iii) We demonstrate that pre-training UNIK on Posetics and fine-tuning it on the target real-world datasets (*e.g.*, Toyota Smarthome [4] and Penn Action [46]) can be a generic and effective methodology for skeleton-based action classification.

2 Related Work

Human Action Recognition. Human action recognition approaches could be mainly categorized into three types. (i) 3D-CNNs [5, 7, 9, 11, 15, 30, 39] and their variants [19, 41] have become the mainstream approach as the models can effectively extract spatio-temporal features for RGB videos and can be pre-trained on a large-scale dataset Kinetics [3] to facilitate transfer learning. (ii) Two-stream CNNs [8, 16] use two inputs of RGB and optical flow to separately model appearance and motion information in videos with a late fusion. Unlike RGB-based methods, (iii) skeleton-based approaches [24, 33, 36, 43] can learn good video representation with less amounts of parameters and are more robust to changes in appearances, environments, and view-points. In this work, we specifically focus on improving the skeleton-based action recognition performance and the model generalization ability.

Skeleton-Based Action Recognition. Early skeleton-based approaches using Recurrent Neural Networks (RNNs) [35, 38, 42, 45, 47] or Temporal Convolutional Networks (TCNs) [17] were proposed due to their high representation capacity. However, these approaches ignore the spatial semantic connectivity of the human body. Subsequently, [1, 18, 45] proposed to map the skeleton as a pseudo-image (*i.e.*, in a 2D grid structure to represent the spatial-temporal features) based on manually designed transformation rules and to leverage 2D CNNs to process the spatio-temporal local dependencies within the skeleton sequence by considering a partial human-intrinsic connectivity. ST-GCN [43] used spatial graph convolutions along with interleaving temporal convolutions for skeleton-based action recognition. This work considered the topology of the human skeleton, however ignored the important long-range dependencies between the joints. In contrast, recent AGCN-based approaches [10, 21, 24, 27, 32, 33, 34, 36] have seen significant performance boost, by the advantage of improving the representation of human skeleton topology to process long-range dependencies for action recognition. Specifically, 2s-AGCN [33] introduced an adaptive graph convolutional network to adaptively learn the topology of the graph with self-attention, which was shown beneficial in action recognition and hierarchical structure of GCNs. Associated extension, MS-AAGCN [34] incorporated multi-stream adaptive graph convolutional networks that used attention modules and 4-stream ensemble based on 2s-AGCN [33]. These approaches primarily focused on spatial modeling. Consequently, MS-G3D Net [24] presented a unified approach for capturing complex joint correlations directly across space and time. However, the accuracy depends on the scale of the temporal segments, which should be carefully tuned for different datasets, preventing transfer learning. Thus, these previous approaches [24, 33, 34] learn adaptive adjacency matrices from the sub-optimal initialized human topology. In contrast, our work proposes an optimized and unified dependency matrix that can be learned from a *uniform distribution* by a multi-head attention process without the constraint of human topology and a limited number of attention maps in order to improve performance, as well as generalization capacity for skeleton-based action recognition.

Model Generalization for Skeletons. Previous methods [24, 33, 34, 43] were only evaluated on the target datasets, trained from scratch without taking advantages of fine-tuning on

a pre-trained model. To explore the transfer ability for action recognition using human skeleton, recent research [20, 37] proposed view-invariant 2D or 3D pose embedding algorithms with pre-training performed on lab datasets [14, 23] that do not correspond to real-world and thus these techniques struggle to improve the action recognition performance on downstream tasks with large-scale real-world videos [4, 22]. To the best of our knowledge, we are the first to explore the skeleton-based pre-training and fine-tuning strategies for real-world videos.

3 Proposed Approach

3.1 Unified Architecture (UNIK)

In this section we present UNIK, a unified spatio-temporal dependencies learning network for skeleton-based action recognition.

Skeleton Sequence Modeling. As shown in Fig. 3 (a), the sequence of the input skeletons is modeled by a 3D spatio-temporal matrix, noted as $\mathbf{f_{in}}$. For each frame, the 2D or 3D body joint coordinates are arranged in a vector within the spatial dimension in any order as long as the order is consistent with other frames in the same video. For the temporal dimension, the same body joints in two consecutive frames are connected. T, V, and C_{in} represent the length of the video, the number of joints of the skeleton in one frame, as well as the input channels (2D or 3D at the beginning and expanded within the building blocks), respectively. The input $\mathbf{f_{in}}$ and the output $\mathbf{f_{out}}$ for each building block (see 3.1) are represented by a matrix in $\mathbb{R}^{C_{in} \times T \times V}$ and a matrix in $\mathbb{R}^{C_{out} \times T \times V}$, respectively.

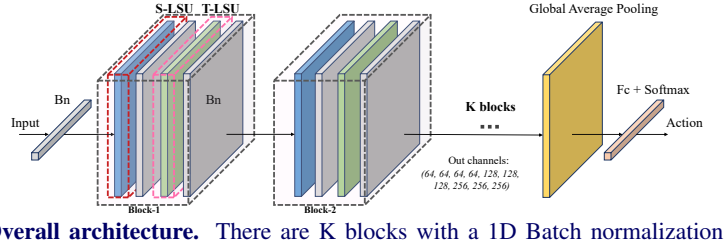

Figure 2: **Overall architecture.** There are K blocks with a 1D Batch normalization layer at the beginning, a global average pooling layer and a fully connected classifier at the end. Each block contains a Spatial Long-short dependency Unit **(S-LSU)**, a Temporal Long-short dependency Unit **(T-LSU)** and two Batch normalization layers.

Overall Architecture. The overall architecture is composed of K building blocks (see Fig. 2). Key components of each block constitute the Spatial Long-short Dependency learning Unit (S-LSU), as well as the Temporal Long-short Dependency learning Unit (T-LSU) that extract both spatial and temporal multi-scale features on skeletons over a large receptive field. The building block ST-LS$_{\text{block}}$ is formulated as follows:

$$\mathbf{f_{out}} = \text{ST-LS}_{\text{block}}(\mathbf{f_{in}}) = \text{T-LSU}\left(\text{S-LSU}(\mathbf{f_{in}})\right) \tag{1}$$

S-LSU and T-LSU are followed by a 2D Batch normalization layer respectively. A 1D Batch normalization layer is added in the beginning for normalizing the flattened input data. Given a skeleton sequence, the modeled data is fed into the building blocks. After the last block, global average pooling is performed to pool feature maps of different samples to the same size. Finally, the fully connected classifier outputs the prediction of the human action. The number of blocks K and the number of output channels should be adaptive to the size of the training set, as a large network cannot be trained with a small dataset. However, in this work, we do not need to adjust K, as we propose to pre-train the model on a large, generic dataset (see 4). We set $K = 10$ with the number of output channels

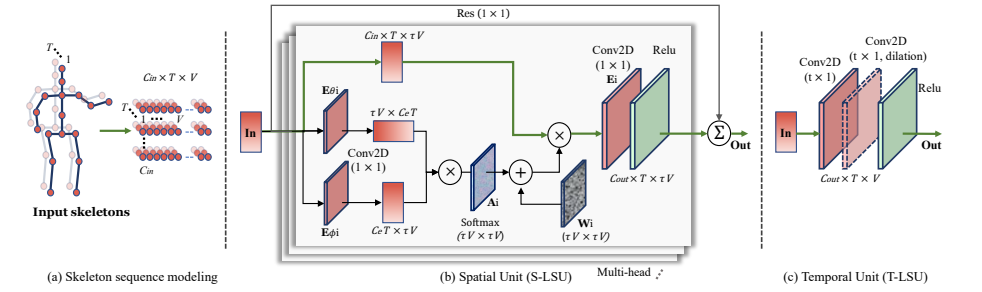

Figure 3: **Unified Spatial-temporal Network.** (a) The input skeleton sequence is modeled into a matrix with C_{in} channels $\times$ T frames $\times$ V joints. (b) In each head of the S-LSU, the input data over a temporal sliding window (τ) is multiplied by a dependency matrix which are obtained from the unified, uniformly initialized $\mathbf{W_i}$ and the self-attention based $\mathbf{A_i}$. $\mathbf{E}_i$, $\mathbf{E}_{\theta i}$ and $\mathbf{E}_{\phi i}$ are for the channel embedding from C_{in} to C_{out}/C_e respectively by (1×1) convolutions. The final output is the sum of the outputs from all the heads. (c) The T-LSU is composed of convolutional layers with $(t \times 1)$ kernels . d denotes the dilation coefficient which can be different in each block.

$64, 64, 64, 64, 128, 128, 128, 256, 256, 256$ (see Fig. 2). In order to stabilize the training and ease the gradient propagation, a residual connection is added for each block.

Spatial Long-short Dependency Unit (S-LSU). To aggregate the information from a larger spatial-temporal receptive field, a sliding temporal window of size τ is set over the input matrix. At each step, the input $\mathbf{f_{in}}$ across τ frames in the window becomes a matrix in $\mathbb{R}^{C_{in} \times T \times \tau V}$. For the purpose of spatial modeling, we use a multi-head and residual based S-LSU (see Fig. 3 (b)) and formulated as follows:

$$\mathbf{f_{out}} = \text{S-LSU}(\mathbf{f_{in}}) = \sum_{i=1}^{N} \mathbf{E}_i \cdot \big(\mathbf{f_{in}} \times (\mathbf{W}_i + \mathbf{A}_i)\big), \tag{2}$$

where N represents the number of heads. $\mathbf{E}_i \in \mathbb{R}^{C_{out} \times C_{in} \times 1 \times 1}$ denotes the 2D convolutional weight matrix with 1×1 kernel size, which embeds the features from C_{in} to C_{out} by the dot product. $\mathbf{W}_i \in \mathbb{R}^{\tau V \times \tau V}$ is the "dependency matri" mentioned in Sec. 1 to process the dependencies for every pair of spatial features. Inspired by [13], $\mathbf{W}_i$ is learnable and uniformly initialized as random values within bounds (Eq. 3).

$$\mathbf{W}_i = \text{Uniform}(-bound, bound), \quad \text{where} \quad bound = \sqrt{\frac{6}{(1+a^2)V}}, \tag{3}$$

where a denotes a constant indicating the negative slope of the rectifier [13]. In this work, we take $a = \sqrt{5}$ as the standard initialization strategy of the fully connected layers for $\mathbf{W}_i$, in order to efficiently reach the optimal dependencies.

Self-attention Mechanism. The matrix $\mathbf{A}_i$ in Eq. 2 represents the non-local self attention map that adapts the dependency matrix $\mathbf{W}_i$ dynamically to the target action. This adaptive attention map is learned end-to-end with the action label. In more details, given the input feature map $\mathbf{f_{in}} \in \mathbb{R}^{C_{in} \times T \times \tau V}$, we first embed it into the space $\mathbb{R}^{C_e \times T \times \tau V}$ by two convolutional layers with 1×1 kernel size. The convolutional weights are denoted as $\mathbf{E}_{\theta i} \in \mathbb{R}^{C_e \times C_{in} \times 1 \times 1}$ and $\mathbf{E}_{\phi i} \in \mathbb{R}^{C_e \times C_{in} \times 1 \times 1}$, respectively. The two embedded feature maps are reshaped to $\tau V \times C_e T$ and $C_e T \times \tau V$ dimensions. They are then multiplied to obtain the attention map $\mathbf{A}_i \in \mathbb{R}^{\tau V \times \tau V}$, whose elements represent the attention weights between each two joints adapted to different actions. The value of the matrix is normalized to $0 \sim 1$ using a softmax function. We can formulate $\mathbf{A}_i$ as:

$$\mathbf{A}_i = \text{Softmax}\big((\mathbf{E}_{\theta i}^{\mathbf{T}} \cdot \mathbf{f}_{\mathbf{in}}^{\mathbf{T}}) \times (\mathbf{E}_{\phi i} \cdot \mathbf{f_{in}})\big). \tag{4}$$

Temporal Long-short Dependency Unit (T-LSU). For the temporal dimension, the video length is generally large. If we use the same method as spatial dimension, *i.e.*, establishing dependencies by $T \times T$ weights for every pair of frames, it will consume too much calculation. Therefore, we leverage multiple 2D convolutional layers with kernels of different dilation coefficient d and temporal size t on the $C_{out} \times T \times N$ feature maps to learn the multi-scale long-short term dependencies (see Fig. 3 (c)). The T-LSU can be formulated as:

$$\mathbf{f_{out}} = \text{T-LSU}(\mathbf{f_{in}}) = \text{Conv}_{2D(t\times 1,d)}(\mathbf{f_{in}}). \tag{5}$$

Joint-bone Two-stream Fusion. Inspired by the two-stream methods [24, 32, 33], we use a two-stream framework where a separate model with identical architecture is trained using the bone features initialized as vector differences of adjacent joints directed away from the body center. The softmax scores from the joint and bone models are summed to obtain final prediction scores.

3.2 Design Strategy

In this section, we present our design strategy that goes beyond GCNs by using a generic dependency matrix $\mathbf{W}_i$ (see Eq. 2) and the attention mechanism $\mathbf{A}_i$ to model the relations between joints in our unified formulation.

Dependency Matrix. For many human actions, the natural connectivity between joints are not the most appropriate to be used to extract features on skeletons (*e.g.*, for "drinking", the connectivity between the head and the hand should be considered, but the original human topology does not include this connectivity). Hence, it is still an open question what kind of adjacency matrix can represent the optimal dependencies between joints for effective feature extraction. Recent works [21, 24, 33] aim at optimizing the adjacency matrices to increase the receptive field of graph convolutions, by higher-order polynomials to make distant neighbors reachable [21] or leveraging an attention mechanism to guide the learning process of the adjacency matrix [24, 33]. Specifically, they decompose the adjacency matrix into a certain number of subsets according to the distances between joints [24] or according to the orientation of joints to the gravity (*i.e.*, body center) [33], so that each subset is learned individually by the self-attention. The learned feature maps are then aggregated together for the action classification. However, the number of subsets is constrained by the body structure. Moreover, we note that the manually pre-defined subsets of the adjacency matrix with prior knowledge (*i.e.*, pre-defined body topology) are all sparse. At the initial learning stage, this spatial convolution relies on a graph-representation, while at the deeper stage, the relations coded within the adjacency matrix are no more sparse and the joint connections are represented by a complete-graph, which corresponds to a fully connected layer in the narrow sense. Finally, the dependencies converge to a sparse representation again, which is locally optimal but completely different from the original topological connectivity of the human body (see Fig. 4). This motivates us, in this work, to revise the *adjacency matrix* by a generic *dependency matrix* that is prospectively initialized with a fully dense and uniform distribution (Eq. 3) to better reach the globally optimal representation.

Multi-head Aggregation. With our proposed initialization strategy, we can repeat the self-attention mechanism by leveraging multiple dependency matrices and sum the outputs to automatically aggregate the features focusing on different body joints (Eq. 2). As the number of attention maps (*i.e.*, heads) N is no longer limited by the human topology, we can use it as a flexible hyper-parameter to improve the model. In the ablation study (see Fig. 4 and Tab. 1), our insight has been verified. Overall, our design strategy makes the architecture more flexible, effective and generic, which facilitates the study of cross-domain transfer learning in this field for datasets using different joint distributions (see Fig. 1).

4 Posetics Skeleton Dataset

In this section, we introduce Posetics, a novel large-scale pre-training dataset for skeleton-based action recognition. The Posetics dataset is created to study the transfer learning on skeleton-based action recognition. It contains 142,000 real-world video clips with the corresponding 2D and 3D poses including 17 body joints. All video clips in Posetics dataset are filtered from Kinetics-400 [3], to contain at least one human pose over 50% of frames.

Motivation and Data Collection. Recent skeleton-based action recognition methods on NTU-RGB+D [23, 31] can perform similarly or better compared to RGB-based methods. However, as laboratory indoor datasets may not contain occlusions, it is difficult to use such datasets to pre-train a generic model that can be transferred onto real-world videos, where skeleton data encounters a number of occlusions and truncations of the body. On the other hand, the accuracy based on skeleton data on the most popular real-world pre-training dataset, Kinetics [3], is still far below the accuracy on other datasets. The main problems are: (i) the skeleton data is hard to obtain by pose estimators as Kinetics is not human-centric. Human body may be missing or truncated by the image boundary in many frames. (ii) Many action categories are highly related to objects rather than human motion (*e.g.*, "making cakes", "making sushi" and "making pizza"). These make it difficult to effectively learn the human skeleton representation for recognizing actions. Hence, recent datasets [23, 43] are unable to significantly boost the action recognition performance when applied to different datasets. In order to better study the generalizability of skeleton-based models in the real-world, we extract the pose (*i.e.*, skeleton) data on Kinetics-400 [3] videos. Specifically, we compare the recent pose estimators and extract pose data from RGB videos through multiple pose estimation systems. Then we apply SSTA-PRS [44], a pose refinement system, for obtaining higher quality pose data in real-world videos. This system aggregates the poses of three off-the-shelf pose estimators [2, 6, 29], as pseudo ground-truth and retrain LCR-Net++ [29] to improve the estimation performance. Moreover, for the problem (i), we filter out the videos where no body detected, and for the problem (ii), we slightly and manually modify the video category labels of Kinetics-400, and place emphasis on relating verbs to poses. (*e.g.*, For "making cakes", "making sushi" and "making pizza", we collectively chose the label "making food", whereas "washing clothes", "washing feet", and "washing hair" remain with the original labels). All in one, we organize 320 action categories for Posetics and this dataset can be more appropriately used for studying the real-world generalizability of skeleton-based action recognition models across datasets by transfer learning.

5 Experiments and Analysis

5.1 Experimental Settings

Extensive experiments are conducted on 5 action classification datasets: **Toyota Smarthome (Smarthome) [9], Penn Action [46], NTU-RGB+D 60 (NTU-60) [31], NTU RGB+D 120 (NTU-120) [23]** and the proposed **Posetics**. See the supplementary material (SM) for complete datasets and implementation details pertaining to all experiments. Firstly, we perform (i) exhaustive ablation study on Smarthome and NTU-60 without pre-training to verify the effectiveness of our proposed *dependency matrix* and *multi-head attention*. Then we (ii) re-evaluate state-of-the-art models [24, 33, 43], as well as our model on the proposed Posetics dataset (baselines are shown in Tab. 3), proceed to provide an analysis on how much performance improves on target datasets: Smarthome, Penn Action, NTU-60 and NTU-120, after pre-training on Posetics in order to demonstrate that our model generalizes well and benefits the most from pre-training. (iii) Final fine-tuned model is evaluated on all datasets

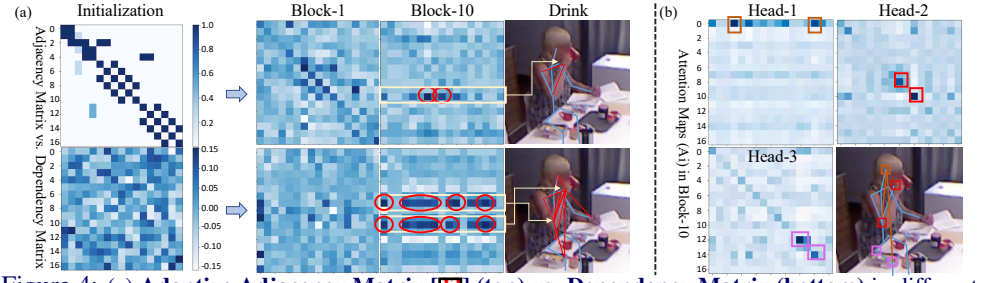

Figure 4: (a) **Adaptive Adjacency Matrix [33] (top) vs. Dependency Matrix (bottom)** in different blocks for action "Drink" of Smarthome (right). They have different initial distributions. During training, the dependencies will become optimized representations, that are salient and more sparse in the deeper blocks, while our proposed matrix represents longer range dependencies (indicated by the red circles and red lines). (b) **Multi-head attention maps** in Block-10. Similar to dependency matrices, attention maps are salient and sparse in the deep block. The different heads automatically learn the relationships between the different body joints (as shown in the boxes and lines with different colors) to process long-range dependencies between joints instead of using pre-defined adjacency matrices.

to compare with the other state-of-the-art approaches for action recognition.

Evaluation Protocols. For Posetics, we split the dataset into 131,268 training clips and 10,669 test clips. We use Top-1 and Top-5 accuracy as evaluation metrics [43]. With respect to real-word settings, 2D poses extracted from images and videos tend to be more accurate than 3D poses, which are more prone to noise. Therefore, we only use 2D data for evaluation and comparison of the models on Posetics. We note that for pre-training, both can be used, 2D and 3D data, in order to obtain different models that can be transferred to datasets with different skeleton data. For the other datasets, we evaluate cross-subject (CS on Smarthome, NTU-60 and 120), cross-view (CV1 and CV2 on Smarthome and CV on NTU-60), cross-setup (CSet on NTU-120) protocols and the standard protocol (on Penn Action). Unless otherwise stated, we use 17 (2D) joints on Smarthome and Penn Action, 25 (3D) joints on NTU-60 and 120.

5.2 Ablation Study of UNIK

Impact of Dependency Matrix. Here we compare the dependency matrices with the adaptive adjacency matrices. In order to verify our analysis in Sec. 3.2, we visualize the adjacency matrices [33] before and after learning. As shown in Fig. 4 (a) (top), we find that the previous learned graph [33] becomes a complete-graph, whose relationships are represented by weights that are well distributed over the feature maps. In contrast, our method is able to explore longer range dependencies, while being based on a dependency matrix with self-attention, which freely searches for dependencies of the skeleton from the beginning without graph-representation (see Fig. 4 (a)-bottom). Quantitatively, results in Tab. 1 show the effectiveness of the Dependency Matrix. Overall, we conclude that, both our method and AGCN-based methods are fully connected layers with different initialization strategies and attention mechanisms in the spatial dimension, both are better than using a fixed graph [43]. It becomes evident that for skeleton-based tasks, where the number of nodes (*i.e.*, spatial body joints) is not large, multi-head attention based dependency matrix learning along with temporal convolutions can be a more generic and effective way to learn spatio-temporal dependencies compared with graph convolution.

Impact of Multi-head Attention. In this section, we visualize the multi-head attention maps and analyze the impact of the number of heads N for UNIK with $N = 1, 3, 6, 9, 12, 16$. As shown in Fig. 4, our multi-head aggregation mechanism can automatically learn the relationships between different positions of body joints by conducting the spatial processing (see

Datasets (J)	Matrix ($N=3,\tau=1$)			#Heads-N ($\tau=1$)						TW-τ ($N=3$)				TD ($N=3,\tau=1$)	
	FM	AM	DM	0	1	3	6	9	12	1	3	6	9	×	✓
SH(%)	50.4	55.7	**58.5**	56.8	58.1	**58.5**	57.9	56.3	58.1	**58.5**	56.6	56.2	55.5	58.5	**58.9**
NTU-60(%)	84.3	86.1	**87.3**	86.8	87.0	87.3	87.1	85.8	**88.0**	87.3	86.8	**87.8**	85.0	87.3	**87.8**

Table 1: Ablation study on Smarthome (SH) CS and NTU-60 CS using joint (J) data only. FM: Fixed Adjacency Matrix (ST-GCN), AM: Adaptive Adjacency Matrix (AGCNs), DM: Dependency Matrix (Ours). TW: Temporal window size. TD: Temporal dilation.

Methods	Pre-training	Smarthome (J)			Penn Action (J)	*NTU-60 (J+B)		*NTU-120 (J+B)	
		CS (%)	CV1 (%)	CV2 (%)	Top-1 Acc. (%)	CS (%)	CV (%)	CS (%)	CSet (%)
2s-AGCN [63]	Scratch	55.7	21.6	53.3	89.5	84.2	93.0	78.2	82.9
MS-G3D [24]	Scratch	55.9	17.4	56.7	88.5	**86.0**	**94.1**	**80.2**	**86.1**
UNIK (Ours)	Scratch	**58.9**	**21.9**	**58.7**	**90.1**	85.1	93.6	79.1	83.5
2s-AGCN [63]	Posetics	58.8	32.2	57.9	96.4	85.8	93.4	79.7	85.0
MS-G3D [24]	Posetics	59.1	26.6	60.1	92.2	86.2	94.1	80.6	86.4
UNIK (Ours)	Posetics	**62.1**	**33.4**	**63.6**	**97.2**	**86.8**	**94.4**	**80.8**	**86.5**

Table 2: Generalizability study of state-of-the-art by comparing the impact of transfer learning on Smarthome, Penn Action, NTU-60 and 120 datasets. The blue values indicate the best generalizabilities that can take the most advantage of pre-training on Posetics. "*" indicates that we only use 17 main joints adapted to the pre-trained model on Posetics.

Eq. 3) using the unified dependency matrices with a uniform initialization. Quantitative results in Tab. 1 show that obtaining a correct number of heads N is instrumental in improving the accuracy in a given dataset, but weakens the generalization ability across datasets with different types of actions (*e.g.*, the model benefits predominantly from $N=12$ for NTU-60, and $N=3$ for Smarthome). Consequently, we set $N=3$ as a unified setting for all experiments and all datasets in order to balance the efficiency and performance of the model, as well as the generalization ability.

Other Ablations. For further analysis, results in Tab. 1 also show that (i) similar to [24], the size of the sliding window (see 3.1) τ can help to improve the performance, however weakening the generalizability of the model as it is sensitive to the number of frames in the video clip. (ii) Temporal dilated convolution contributes to minor boosts. See SM for more ablation study about initialization of Dependency Matrix and multi-stream fusion.

5.3 Impact of Pre-training.

In this section, we pre-train [24, 63] and our proposed UNIK in a unified setting, ($N=3, K=10, \tau=1$). Note that for pre-training GCN-based models [24, 63], we need to manually calibrate the different human topological structures in different datasets to keep the predefined graphs unified. For evaluation, we report the classification results on all the four datasets to demonstrate the impact of pre-training and compare the generalization capacities *i.e.*, benefits compared to training from scratch. Note that unless otherwise stated, we use the consistent skeleton data (2D on Smarthome, Penn Action and 3D on NTU-60, 120), number of joints (17 main joints) for fair comparison of all models. On NTU-60 and 120, we use both joint (J) and bone (B) data to compare the full models with two-stream fusion.

Generalizability Study. The results suggest that pre-training consistently boosts all models, see Tab. 2, in particular, small benchmarks (*e.g.*, Smarthome CV and Penn Action with $5\% \sim 12\%$ improvement), as we do not have sufficiently large training data. Previous work [24] has a weak transfer capacity, due to the dataset-specific model settings (*e.g.*, the number of GCN scales and G3D scales) not always being able to adapt to the transferred datasets. On NTU-60, we take the main 17 joints for fine-tuning as we estimate and refine the main 17 joints on Posetics, and our pre-trained model outperforms state-of-the-art model [24]. Therefore, we conclude that our pre-trained model is the most generic-applicable

Methods	RGB	Pose	Pre-training	Posetics		Smarthome			Penn Action
				Top-1(%)	Top-5(%)	CS(%)	CV1(%)	CV2(%)	Accuracy(%)
I3D [3]	✓		Kinetics-400	46.4	60.1	53.4	34.9	45.1	96.3
AssembleNet++ [30]	✓		Kinetics-400	-	-	63.6	-	-	-
NPL [28]	✓		Kinetics-400	-	-	-	39.6	54.6	-
Separable STA [4]	✓	✓	Kinetics-400	-	-	54.2	35.2	50.3	-
VPN [5]	✓	✓	Kinetics-400	-	-	60.8	43.8	53.5	-
Multi-task [25]	✓	✓	Scratch	-	-	-	-	-	97.4
LSTM [26]		✓	Scratch	-	-	42.5	13.4	17.2	-
ST-GCN [43]		✓	Scratch	43.3	67.3	53.8	15.5	51.1	89.6
2s-AGCN [33]		✓	Scratch	47.0	70.8	60.9	22.5	53.5	93.1
Res-GCN [36]		✓	Scratch	46.7	70.6	61.5	-	-	93.4
MS-G3D Net [24]		✓	Scratch	47.1	70.0	61.1	17.5	59.4	92.7
UNIK (Ours)		✓	Scratch	**47.6**	**71.3**	**63.1**	**22.9**	**61.2**	**94.0**
Pr-ViPE [37]		✓	Human3.6M	-	-	-	-	-	97.5
UNIK (Ours)		✓	Posetics(Ours)	-	-	**64.3**	**36.1**	**65.0**	**97.9**

Table 3: Comparison with state-of-the-art methods on the Posetics, Toyota Smarthome and Penn Action dataset. The best results using RGB data are marked in blue for reference.

especially for real-world scenarios. We provide further analysis in SM on (i) the pre-training on Posetics using 25 joints including the additional 8 joints on fingers and feet derived from linear interpolation for transferring on NTU-60 with full 25 joints and (ii) the evaluation of pre-trained features by linear classification on smaller datasets with the fixed backbone.

5.4 Comparison with State-of-the-art

We compare our full model (*i.e.*, Joint+Bone fusion) with and without pre-training to state-of-the-art methods, reporting results in Tab. 3 (Posetics, Smarthome and Penn Action). Note that for fair comparison, we use the same skeleton data (2D and 17 joints) for all models. For real-world benchmarks using estimated skeleton data (*e.g.*, Posetics, Smarthome and Penn Action), our model without pre-training outperforms all state-of-the-art methods [24, 26, 33, 36, 43] in skeleton (*i.e.*, pose) stream and with pre-training, outperforms the embedding-based method [37] that pre-trained on Human3.6M [14]. On NTU-60 and 120 (see Tab. 2), we compare the most impressive two-stream graph-based methods [24, 33] and our model performs competitively without pre-training. We argue that, we simplify our model as generically as possible without data-specific settings, which can improve the performance but weaken the transfer behavior (*e.g.*, the setting of N and τ). Subsequently, we further compare RGB-based methods [3, 4, 5, 25, 28, 30] for reference, that can be pre-trained on Kinetics-400 [3]. It suggests that previous skeleton-based methods [24, 26, 33, 43] without leveraging the pre-training are limited by the poor generalizability and the paucity of pre-training data. In contrast, our proposed framework, UNIK with pre-training on the Posetics dataset, outperforms state-of-the-art using RGB and even both RGB and pose data on the downstream tasks (*e.g.*, Smarthome and Penn Action).

6 Conclusion

In this paper, we have proposed UNIK, a unified framework for real-world skeleton-based action recognition. Our experimental analysis shows that UNIK is effective and has a strong generalization ability to transfer across datasets. In addition, we have introduced Posetics, a large-scale real-world skeleton-based action recognition dataset featuring high quality skeleton annotations. Our experimental results demonstrate that pre-training on Posetics improves performance of the action recognition approaches. Future work involves an analysis of our framework for additional tasks involving skeleton sequences (*e.g.*, 2D-to-3D pose estimation).

Acknowledgement

This work has been supported by Toyota Motor Europe (TME) and the French government, through the 3IA Cote d'Azur Investmentsin the Future project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002. We are grateful to Inria-Sophia Antipolis "NEF" computation cluster for providing resources and support.

References

[1] C. Caetano, F. Brémond, and W. Schwartz. Skeleton image representation for 3D action recognition based on tree structure and reference joints. *SIBGRAPI*, 2019.

[2] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE TPAMI*, 2019.

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[4] Srijan Das, Rui Dai, Michal Koperski, Luca Minciullo, Lorenzo Garattoni, Francois Bremond, and Gianpiero Francesca. Toyota smarthome: Real-world activities of daily living. In *ICCV*, 2019.

[5] Srijan Das, Saurav Sharma, Rui Dai, Francois Bremond, and Monique Thonnat. Vpn: Learning video-pose embedding for activities of daily living. *ECCV*, 2020.

[6] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[7] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *CVPR*, 2020.

[8] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, 2016.

[9] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019.

[10] X. Gao, W. Hu, Jiaxiang Tang, J. Liu, and Zongming Guo. Optimized skeleton-based action recognition via sparsified graph regression. *ACM MM*, 2019.

[11] K. Hara, H. Kataoka, and Y. Satoh. Learning spatio-temporal features with 3D residual networks for actio recognition. In *ICCVW*, 2017.

[12] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3D cnns retrace the history of 2D cnns and imagenet? In *CVPR*, 2018.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015.

[14] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE TPAMI*, 2014.

[15] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE TPAMI*, 2013.

[16] Simonyan Karen and Zisserman Andrew. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014.

[17] T. S. Kim and A. Reiter. Interpretable 3D human action analysis with temporal convolutional networks. In *CVPRW*, 2017.

[18] Chao Li, Qiaoyong Zhong, Di Xie, and S. Pu. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. In *IJCAI*, 2018.

[19] Kunchang Li, Xianhang Li, Yali Wang, Jun Wang, and Yu Qiao. Ct-net: Channel tensorization network for video classification. In *ICLR*, 2021.

[20] Linguo Li, Minsi Wang, Bingbing Ni, Hang Wang, Jiancheng Yang, and Wenjun Zhang. 3d human action representation learning via cross-view consistency pursuit. In *CVPR*, 2021.

[21] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019.

[22] Tianjiao Li, Jun Liu, Wei Zhang, Yun Ni, Wenqian Wang, and Zhiheng Li. Uav-human: A large benchmark for human behavior understanding with unmanned aerial vehicles. In *CVPR*, 2021.

[23] J. Liu, A. Shahroudy, M. Perez, G. Wang, L. Y. Duan, and A. C. Kot. Ntu rgb+d 120: A large-scale benchmark for 3D human activity understanding. *IEEE TPAMI*, 2020.

[24] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *CVPR*, 2020.

[25] D. Luvizon, D. Picard, and H. Tabia. Multi-task deep learning for real-time 3D human pose estimation and action recognition. *IEEE TPAMI*, 2020.

[26] Behrooz Mahasseni and S. Todorovic. Regularizing long short term memory with 3D human-skeleton sequences for action recognition. *CVPR*, 2016.

[27] W. Peng, Xiaopeng Hong, H. Chen, and G. Zhao. Learning graph convolutional network for skeleton-based human action recognition by neural searching. In *AAAI*, 2020.

[28] AJ Piergiovanni and Michael S. Ryoo. Recognizing actions in videos from unseen viewpoints. In *CVPR*, 2021.

[29] Grégory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images. *IEEE TPAMI*, 2019.

[30] M. Ryoo, A. Piergiovanni, Juhana Kangaspunta, and A. Angelova. Assemblenet++: Assembling modality representations via attention connections. *ECCV*, 2020.

[31] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3D human activity analysis. *CVPR*, 2016.

[32] L. Shi, Yifan Zhang, Jian Cheng, and H. Lu. Skeleton-based action recognition with directed graph neural networks. *CVPR*, 2019.

[33] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019.

[34] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing LU. Skeleton-Based Action Recognition with Multi-Stream Adaptive Graph Convolutional Networks. *IEEE TIP*, 2020.

[35] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *AAAI*, 2017.

[36] Yi-Fan Song, Zhang Zhang, Caifeng Shan, and Liang Wang. Stronger, faster and more explainable: A graph convolutional baseline for skeleton-based action recognition. In *ACMMM*, 2020.

[37] Jennifer J. Sun, Jiaping Zhao, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, and Ting Liu. View-invariant probabilistic embedding for human pose. In *ECCV*, 2020.

[38] Amor Ben Tanfous, Hassen Drira, and Boulbaba Ben Amor. Sparse coding of shape trajectories for facial expression and action recognition. *IEEE TPAMI*, 2019.

[39] Du Tran, Heng Wang, Matt Feiszli, and Lorenzo Torresani. Video classification with channel-separated convolutional networks. In *ICCV*, 2019.

[40] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3D skeletons as points in a lie group. *CVPR*, 2014.

[41] Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. Tdn: Temporal difference networks for efficient action recognition. In *CVPR*, 2021.

[42] Chunyu Xie, Ce Li, B. Zhang, Chen Chen, Jungong Han, Changqing Zou, and Jianzhuang Liu. Memory attention networks for skeleton-based action recognition. *IJCAI*, 2018.

[43] S. Yan, Yuanjun Xiong, and D. Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. *AAAI*, 2018.

[44] Di Yang, Rui Dai, Yaohui Wang, Rupayan Mallick, Luca Minciullo, Gianpiero Francesca, and Francois Bremond. Selective spatio-temporal aggregation based pose refinement system: Towards understanding human activities in real-world videos. In *WACV*, 2021.

[45] Pengfei Zhang, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, and Nanning Zheng. View adaptive neural networks for high performance skeleton-based human action recognition. *IEEE TPAMI*, 2019.

[46] W. Zhang, M. Zhu, and K. G. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *ICCV*, 2013.

[47] W. Zheng, L. Li, Z. Zhang, Y. Huang, and L. Wang. Relational network for skeleton-based action recognition. In *ICME*, 2019.