Signal, Image and Video Processing
https://doi.org/10.1007/s11760-024-03510-9

ORIGINAL PAPER

Human action recognition using ST-GCNs for blind accessible theatre performances

Leyla Benhamida[1] · **Slimane Larabi**[1]

Received: 13 June 2024 / Revised: 26 July 2024 / Accepted: 7 August 2024

Abstract
Audio descriptions present a tool that helps blind audience members assist theater performances by conveying visual information, such as actors' gestures. However, its high production process cost and effort limit its availability. To address this, we propose a computer vision based system for automated actor gestures recognition, using the state-of-the-art spatio-temporal graph convolution networks (ST-GCNs) for skeleton-based action recognition via transfer learning technique. Hence, we evaluated the transferability of three pre-trained ST-GCNs: the first proposed spatio-temporal graph convolution network (ST-GCN), convolution network of two-stream adaptive graphs (2s-AGCN), and the multi-scale disentangled unified graph convolution network (MS-G3D). We used NTU-RGBD action benchmark as the source domain and collected a novel dataset: TS-RGBD, to serve as the target domain. We then proposed two configurations to accommodate the diversity between the source and target domains. Results showed that ST-GCNs exhibit positive transferability enhancing the models' recognition performance in theatre contexts, promoting automated system for gesture accessibility in theaters.

Keywords Human action recognition · Transfer learning · Graph convolution network · Skeleton data

1 Introduction

Numerous computer vision based aid systems have been developed to assist people with visual impairments in their daily tasks, including scene understanding [1–5], visual positioning [6–8], navigating and avoiding obstacles [9, 10], understanding actions of their surroundings [11]...etc. Despite these advancements, there is a dearth of systems designed for entertainment accessibility. Theaters, in particular, remain inaccessible to people with visual disabilities due to the multiple visual elements, such us actors' gestures, that are crucial for the proper understanding of the ongoing scenes. Very few theaters provide audio descriptions due to its complex production process which requires a team of professionals to go through the scripts and provide verbal descriptions. This results in high costs that small theaters cannot afford limiting accessibility in these venues. Therefore, this research aims to enhance and automate gesture accessibility in theaters using computer vision based human action recognition methods.

Human action recognition, a rapidly advancing field within computer vision, has found applications in various domains such as surveillance [12, 13], healthcare [14, 15], sports [16], and human-computer interaction [17]. However, deploying these models in real-life scenarios, like theatres, presents challenges due to the disparity between training data and new environments. Existing models may struggle to accurately recognize and classify actors' gestures in theaters, necessitating specialized frameworks tailored for theatrical scenes.

In this research, we present a comprehensive framework for recognizing human actions in theatrical scenes using computer vision and deep learning models. Our framework leverages input data, specifically actors actions, captured by depth sensor which provides three types of data: RGB, depth, and skeleton data. In this study, we focus on utilizing skeleton data, representing 3D positions of human body joints. (Fig. 1). One widely-used deep learning model for skeleton-based Human Action Recognition (HAR) is the Spatio-Temporal Graph Convolution Network (ST-GCN), known for its success in challenging benchmarks such as NTU-RGBD [18,

✉ Leyla Benhamida
lbenhamida@usthb.dz

Slimane Larabi
slarabi@usthb.dz

[1] RIIMA Laboratory, Computer Science Faculty, USTHB, BP 32 El Alia, 16111 Algiers, Algeria

Published online: 04 September 2024

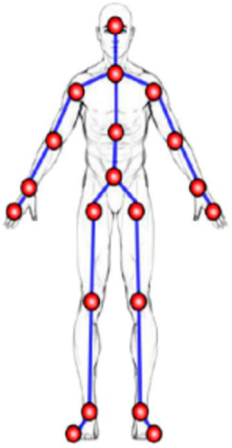


Fig. 1 Skeleton representation provided by the depth sensor. Red points represent different human body key joints captured by the sensor

19]. Our goal is to explore the applicability of ST-GCN and similar models for action recognition in theatre scenes by employing transfer learning technique to enhance their performance. To accomplish this, we curate a new dataset of human action sequences specifically recorded in a theatre environment: TS-RGBD dataset[1].

Through this research, we investigate and analyze the transferability of ST-GCNS for skeleton-based HAR in the context of theatre actions-a topic that has not been previously explored in existing literature.

The paper is structured as follows: In Sect. 2, we commence with an extensive review of the skeleton-based human action recognition approach utilizing ST-GCNs. This review encompasses in-depth explanations of three ST-GCN variants [20–22] providing a comprehensive understanding of their functionalities. Subsequently, we delve into the concept of transfer learning, offering a precise definition and discussing its relevance to our research. Moving forward to Sect. 3, we elucidate the objectives pursued in this study and provide a well-founded justification for the choices made during the course of our research. Section 4 entails a detailed description of theater human action dataset we have collected: TS-RGBD, and we introduce our proposed method for human action recognition. Lastly, in Sect. 5, we present the experimental setup conducted to evaluate the effectiveness of our approach, along with an in-depth discussion of the results obtained. Finally, Sect. 6, summarizes the key findings and the contributions of our research.

2 Related works

Human action recognition (HAR) is a complex research field that finds application in numerous domains. In the past years, various techniques were developed, with the majority relying on computer vision approaches with video sequences captured by camera sensors. One of the most frequently used sensors in HAR systems today is the Microsoft Kinect. It captures both RGB and depth information, providing a richer representation of the scene than traditional RGB cameras. The depth information can be used to extract features, such as joint positions and orientations, which helps boost the performance of the action recognition system. The Microsoft Kinect sensor generates three types of data: RGB images, depth maps that gather the distances of different scene objects from the viewpoint, and Skeleton data that is represented by a set of 3D positions of different human body key joints.

Several approaches were proposed based on each modality: depth approach by extracting features from depth maps and exploring different points [23, 24], skeleton approach that involves the skeleton representation of human body movement where the positions and orientations of the joints are used to describe the human poses and actions [25, 26], and hybrid approach that extract features by combining the two types of data [27]. In this work, we focus on skeleton based approach due to its several advantages over the other modalities such as robustness to view-point and illumination changes, and its compact representation resulting in a reduced computation cost.

The initial deep learning skeleton-based HAR systems employ traditional deep learning models like Convolution Neural Networks(CNN) and Recurrent Neural Networks(RNN). To make use of CNNs, researchers convert sequences of skeleton data into pseudo-images in order to have a Euclidean representation, which is then fed to the CNN [28, 29]. RNN-based methods, on the other hand, also need a transformation by representing each joint by a sequence of coordinate vectors [30, 31]. In fact, skeleton data are embedded as graph-structured data where the joints are the nodes and the bones are the edges linking different joints. Recently, Deep learning methods have been generalized to treat graph-based problems using Graph Neural Network (GNN) to capture both local and global structural information [32]. Several variants of GNN were proposed with different architectures that offer a range of ways to handle information propagation and aggregation on the graph. One popular GNN is the Graph Convolution Network (GCN)[33] which uses a spectral convolution operator to aggregate information from a node's neighbors in the graph. GCNs have been shown to be effective for skeleton-based human action recognition. In [20], Yan et al. proposed a novel model for skeleton based action recognition: Spatio-Temporal GCN (ST-GCN). It can capture spatial patterns from joints distributions as well as their temporal dynamics. Following the same concept, numerous ST-GCN variants were emerged within the past few years [21, 22, 34, 35] that achieved significant performance on different benchmarks.

[1] GitHub Repository

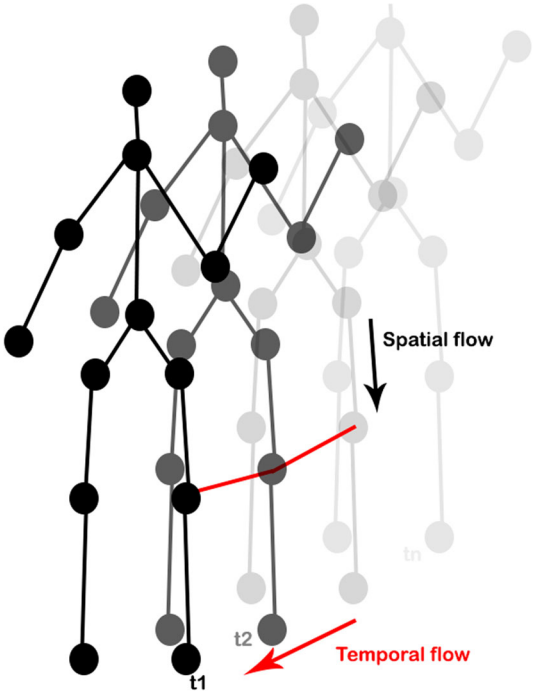


Fig. 2 Spatio-temporal skeleton representation where black lines refer to spatial edges and red lines refer to temporal edges

2.1 Skeleton-based human action recognition with spatio-temporal graph convolution networks

GCN-based methods showed better ability to capture actions' patterns than CNN-based and RNN-based methods because of the non-Euclidian nature of skeleton representation. Also, it represents a simpler method by eliminating the pre-step of manual data transformation as needed for CNN and RNN. Recently, variant ST-GCNs [21, 22, 34] were proposed following the framework introduced in [20] that can learn both spatial and temporal patterns from the skeleton sequences by extracting features from spatial edges that express connectivity between human joints, and temporal edges that connect the same joints across time steps (Fig. 2).

2.1.1 Graph-based representation of action skeleton sequence

ST-GCN models take, as input, a spatio-temporal representation of the skeleton action sequence defined by a set of undirected graphs $\{G_1,\ldots,G_n\}$ where G_i is the skeleton graph having spatial edges of each frame, and n is the number of frames of the input sequence. The skeleton graph is represented as $G = \{V, E\}$ where $V = \{v_1, \ldots, v_m\}$ is the set of nodes referring to the human body joints. m is the total number of nodes where each has a set of features X. E is the set of edges connecting different joints including spatial edges that are bones representing natural connections

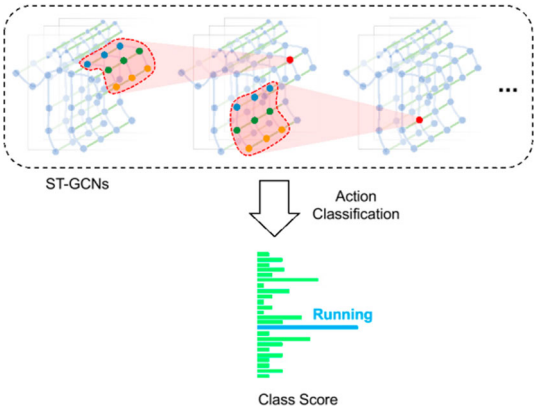


Fig. 3 Spatio-temporal Graph Convolution Network architecture [20]: multiple Spatio-temporal Convolution layers stacked to extract features followed by a SoftMax classifier to predict the action label

between the joints in the human body, and temporal edges connecting the joints of two adjacent frames. The graph is denoted by an adjacency matrix $A \in \{0, 1\}$ of $m \times m$ dimension, and $A_{i,j} = 1$ if v_i and v_j are adjacent and $A_{i,j} = 0$ otherwise.

2.1.2 Graph convolution network

The goal of GCN is to learn a new set of features of a given input graph with features X, by capturing information from both the node's own features and the features of its neighbors. GCN falls under the category of message-passing neural networks. It can be designed by stacking multiple layers on top of each other, where the output of one layer serves as the input of the next one. The layer-wise update rule applied to features X at time t can be defined as follows:

$$X_t^{l+1} = \sigma(\tilde{D}^{\frac{1}{2}}(A + I)\tilde{D}^{\frac{1}{2}}X_t^l W^l)$$

$(A+I)$ represents the addition of self-loops to add the current node's features. $\tilde{D}$ is the diagonal degree matrix of $(A + I)$, W is the network weights and σ is the activation function.

2.1.3 ST-GCN

The concept of GCN in the skeleton-based human action recognition was first introduced by Yan et al. [20]. They presented the graph convolution method on spatio-temporal representation of skeleton sequences. This model uses multiple layers (ST-GCN blocks), that calculate spatial and temporal convolutions simultaneously (Fig. 3). Mathematically, ST-GCN can be defined as follows:

$$X^{l+1} = \sum \check{A} X^l W^l$$

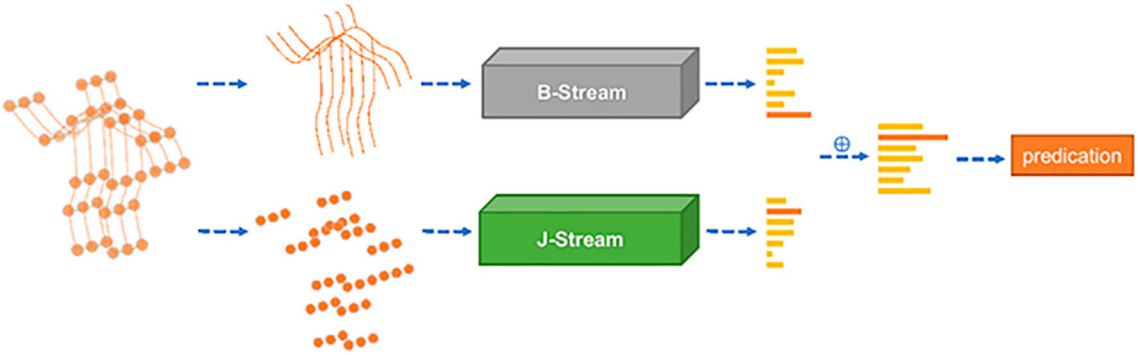


Fig. 4 The two-stream AGCN design [21]: B-stream and J-stream are the Adaptive Graph Convolution Neural Networks for joints and bones respectively

$\breve{A}$ refers to the normalized form of the adjacency matrix A: $\breve{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$. Then, a global pooling and softMax function are used to recognize the action of the resulting tensor.

2.1.4 Two-stream adaptive graph convolution network (2s-AGCN)

Shi et al. [21] introduced a new variant of ST-GCN: Two-stream adaptive graph convolution networks. In contrast to the first ST-GCN where the topology of the graph is fixed and set manually, 2s-AGCN learns the human graph topology adaptively during the training process to increase the flexibility of the model in representing skeleton sequences. Moreover, a two-stream framework is used to capture both first-order information: the joint dependencies, and second-order information: the directions and the length of bones (Fig. 4), in order to boost the performance. It showed a notable improvement in the recognition accuracy on benchmark datasets, and it surpasses the first proposed ST-GCN. The layer-wise update rule of the 2s-AGCN can be defined as:

$$X^{l+1} = \sum X^l W^l (\breve{A} + \breve{B} + \breve{C})$$

A is the normal adjacency matrix, B is the learnt adjacency matrix during the training and C is the node similarity matrix. $\breve{A}$, $\breve{B}$, and $\breve{C}$ are the normalized form of A, B and C respectively.

2.1.5 MS-G3D

Multi-scale spatio-temporal GCN is another variant of GCNs for skeleton based action recognition task [36, 37]. It is based on a feature aggregation from long-range neighbor nodes using higher polynomials of the graph adjacency matrix. However, this aggregation with adjacency powering can make it ineffective to capture long-range joint dependencies because of the weighting bias problem where the aggregated feature will be dominated by signals from local body parts. Liu et al. [22] proposed a disentangled and unifying GCN framework to encounter this problem. They introduced a disentangled multi-scale aggregator that obtains direct information from farther nodes and removes redundant dependencies between node features, using a k-adjacency matrix as:

$$[\tilde{A}_k]_{i,j} = \begin{cases} 1, & \text{if } d(v_i, v_j) = k \\ 1, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}$$

where $\tilde{A} = A + I$, k is the number of scales to aggregate, and $d(v_i, v_j)$ returns the shortest distance in number of hops between v_i and v_j. Moreover, this model uses a unified spatial-temporal graph convolution operator to facilitate direct information flow across space and time. The combination of these two methods results in a powerful feature extraction across both spatial and temporal dimensions. It significantly outperforms the state-of-the-art methods.

2.2 Transfer learning

Transfer learning is a deep learning strategy that has been successfully applied to a wide range of applications. It consists of taking past learnt knowledge, in a specific context: *source domain*, and reusing it to solve a new problem in a related context: *target domain*. Usually, deep learning models are trained from a random weights initialization, for a specific task using a large number of training data labels. If the task changes, the model must be retrained from scratch. In transfer learning, the retraining is applied using a previously-trained network's weights, rather than retraining from scratch. This technique can be especially useful when there are limited training data labels of the new task, as the pre-trained model can help to enhance the performance with fewer samples. Additionally, transfer learning can help to speed up the training process and reduce the computational resources needed

for training a new model from scratch. There are two ways of applying transfer learning:

- Fixed-weights transfer using the pre-trained model's weights as fixed features, where all the weights of different layers, except the last one, are frozen.
- Fine-tuned transfer using the pre-trained model's weights as the starting point, where all network's weights can change during the retraining.

With the recent explosion in the number of HAR applications and the success achieved by transfer learning technique, many researchers have been investigating transfer learning for HAR task [38–40]. However, most of the proposed works were based on RGB images using Euclidean Neural Networks. For instance, in [38], CNN and LSTM (Long Short Term Memory) were used to develop an HAR framework based on the transfer learning technique using RGB images. To our knowledge, no works have been interested in exploring transfer learning for 3D skeleton-based HAR. Additionally, there has been little insight into the transferability of GNNs and not much research exit that investigates transfer learning for GNNs. Although, in [41], the effectiveness of transfer learning with GNNs has been demonstrated and the experiments on real-world datasets showed that the transfer is most effective when the source and target graphs are similar.

The main contributions of this work are:

- We thoroughly investigate and analyze the transferability of three widely-used variants of ST-GCN for the skeleton-based HAR to new environments that are different from the training environment for real-life applications. Specifically, we examine ST-GCN [20], 2s-AGCN [21], and MS-G3D [22]. Through our experiments, we observe positive transfer effects as all the models demonstrate improved performance compared to the baseline approach without transfer learning.
- We present a novel dataset comprising of 3D skeleton sequences of human actions recorded within a theatre scene environment, utilizing depth sensor. This dataset serves as the foundation for developing the first-ever theatre Human Action Recognition (HAR) framework.
- We introduce a framework that enhances the performance of transfer learning methods by effectively addressing the differences between the target domain (our dataset) and the source domain (NTU-RGBD dataset[18, 19]: a benchmark for human action recognition). This framework leads to significant recognition performance improvements.

3 Problem positioning

As previously mentioned, we aim to leverage the power of computer vision to contribute to the development of accessible theater performances by providing a HAR system that recognizes the gestures of actors during a play that will be communicated to the BVI audience. To achieve this, we investigate the transferability of ST-GCNs and utilize the 3D skeleton representation of human gestures provided by the Microsoft Kinect sensor as input to the recognition system (as shown in Fig. 5).

We choose to employ the skeleton approach with ST-GCN models using transfer learning due to the following reasons:

- The first choice is justified by the fact that skeleton approach has achieved considerable success over the other approaches due to its several advantages:
  - It is robust to occlusion: it uses only the joint positions, which can be estimated even when some body parts are occluded.
  - It is invariant to appearance changes: the skeleton-based approach is invariant to changes in clothing, illumination conditions, and camera viewpoint, due to the use of joint positions only which are relatively stable under these changes.
  - It is computationally efficient: the compact skeleton representation of the human body helps reduce the computation time which is crucial in the case of real-time applications.

  In addition, the skeleton representation provides better both spatial and temporal information with the strong intra-frame and inter-frame correlations between the joints allowing the recognition algorithms to capture more expressive and significant features. The skeleton data can be acquired using different techniques, each has its advantages and disadvantages. In Table 1, we list these techniques and provide a comparison between them after analysing their characteristics in the context of theatre. According to Table 1, the Kinect sensor is the most suitable device for the theatre HAR system, which provides rich information and does not affect the actors' performance. The limitation of distance can be encountered by placing it in a near position to the stage.
- The second choice is justified by the fact that the GCN-based approach provides the ability to model and learn joint dependencies across space-time implicitly, unlike RNN-based and CNN-based approaches where a data transformation step is necessary to represent skeletons as

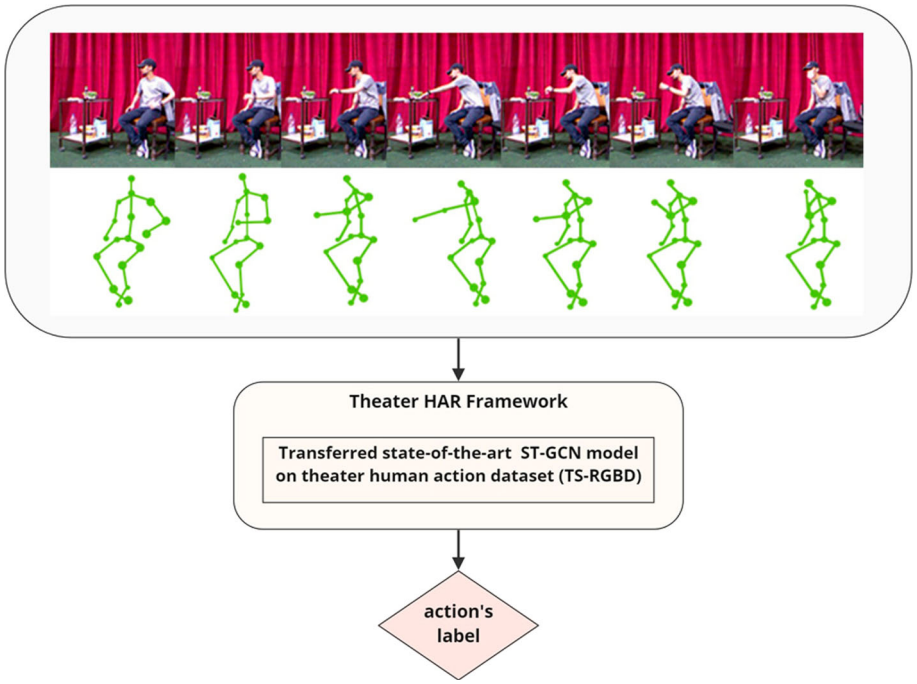


Fig. 5 The proposed system to recognize actions of theater actors during a performance: the actor's gesture is captured via a depth sensor resulting in a sequence of skeleton representation. Then, the skeleton data is fed to a framework that exploits transferred ST-GCN models to recognize the performed action and provides, as output, the action label to be communicated to the blind audience members

Table 1 Comparison between the skeleton data generation techniques in the context of theatres

Technique	Advantages	Disadvantages
Motion capture technologies using sensing devices.	It provides very precise annotations of skeleton data.	It is hard to implement because the actors will have to wear sensing devices which are expensive and can hamper the actors' performance on the stage.
Collecting the skeleton data using Pose Estimation method[42] with RGB images.	It is simple to implement using only an RGB camera. Not expensive.	Its precision can be affected by several factors such as illumination variation, clothing colors, and complex backgrounds.
		It is impossible to control these factors in a theatre environment.
		It does not provide the depth information which can significantly boost the performance of the HAR system.
Using Microsot Kinect sensor.	It captures both RGB and depth information. This combination allows it to produce very accurate joints estimation.	It is limited by the distance: It cannot capture depth information from a large distance.
	Not expensive.	

2D or 3D Euclidean grids to be fed into the model (RNN or CNN). In addition, GCN has outperformed CNN and RNN in many recent studies on HAR. Most recent HAR research works follow the framework introduced by Yan et al.[20] using the spatio-temporal modeling of skeleton data.

- We use transfer learning despite the success achieved by the ST-GCN models on challenging human action benchmarks, because the use of one of the trained models with new users can decrease its performance if the action patterns of the new user are different from those in the training data. In our case, actions' patterns in theatre

scene environment can be different from the patterns of the existing human action benchmarks. A typical solution is to collect a new dataset containing theatre human action samples and train the model from scratch. This task is not feasible due to the following reasons:

- Deep learning models require large amount of training samples to learn intricate patterns and generalize well to new domains, which makes the task of manually collecting a new dataset very hard and time-consuming especially in the case of theatre actions. It is not convenient to get access to theatres and collect a large number of samples that can take several days to accomplish.
- The training of a deep learning model is time-consuming and requires high computation energy which in turn contributes to carbon emissions. This energy demand has seen immense growth in recent years and deep learning may become a significant contributor to climate change if this trend continues. As a result, the excitement over Deep learning success has shifted to warning and many recent studies showed interest in the environmental impact of deep learning, encouraging research into energy-efficient approaches by taking simple steps to reduce carbon emissions [43, 44].

Transfer learning technique on ST-GCNs presents a potential solution to encounter these problems. It helps tackling the problem of limited training data labels and it can also significantly reduce training time leading to decrease the energy and the carbon emissions. Thus, we adopt the transfer learning technique, not only because of the training data scarcity problem, but also to promote responsible computing and to avoid the cost of training models for extensive periods on specialized hardware accelerators. In addition, while conventional deep learning models, such as CNNs, have demonstrated remarkable transferability, research analyzing transfer learning technique within the graph-based field and their ability to transfer learnt knowledge with GNNs is limited. Furthermore, to our knowledge there are currently no studies examining transfer learning via spatio-temporal GCNs for skeleton-based human action recognition.

4 The proposed method

4.1 Our theatre human actions dataset

We collected a new dataset that contains skeleton action sequences at the auditorium of our university where two depth sensors were positioned at the same height and in two different view angles (front view and side view) resulting in a variety of the obtained data (Fig. 6).

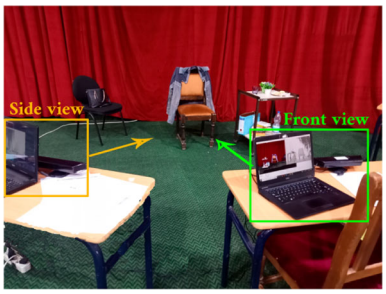


Fig. 6 The setup deployed while collecting the dataset and the placement of the two cameras: one camera from the front view and a second camera from the side view

Each action was performed by 3 individuals with different speed. We collected a total of 388 sequences from both viewpoint over 36 action classes, with a rate of 25 frames per second and an average of 170 frames per sequence.
We collected action classes that are common in theatre scenes including solo actions such as *walking*, *sitting down*, *jumping*, *throwing*, *falling down*, and two-person interactions such as *hugging*, *kicking a person*, *shaking hands*, *high five*. We also focused on capturing actions that are hard to discern from auditory cues only, and require visual interpretation, which impacts on the blind audience's proper understanding of the ongoing scene. Moreover, our study investigates the integration of ST-GCNs in new environments, particularly theaters, through transfer learning, thus, we concentrated on collecting actions from a theatrical environment rather than introducing new action classes distinct from those in NTU dataset. This approach evaluates how different environments affect the performance of pre-trained models.

A detailed description of number of samples per action class is presented in Table 2. Some samples from our dataset are illustrated in Fig. 7.

The provided skeleton information consists of 3D positions of 20 body key joints for each tracked human body. The configuration of the 20 joints is illustrated in Fig. 8.

4.2 Transfer learning

Limited research has been conducted on the transferability of GCNs despite the growing interest in them. The present study aims to explore the transferability of spatio-temporal GCNs for recognizing human action based on skeletons. However, the performance achieved by transfer learning relies on the choice of the pre-trained model and the source domain. This task remains difficult due to the diversity of state-of-the-art architectures. Thus, we selected three of ST-GCN common models (defined in section ***2***): the first proposed ST-GCN [20], the two-stream adaptive GCN [21], and the multi-

Table 2 Number of samples per action class

Action class	Number of samples	Action class	Number of samples	Action class	Number of samples
Standing up	14	Bowing	6	Sitting down	12
Drinking	14	Eating	10	Dropping something	9
Picking up something	14	Throwing something	16	Clapping	6
Reading	9	Writing	11	Tearing up paper	12
Putting on a jacket	6	Taking off a jacket	8	Putting on shoes	6
Taking off on shoes	12	A person walking	7	Handwaving	7
Touch head	12	Phone call	12	Jumpping	20
Kicking something or someone	22	Cheking time on a wristwatch	12	Wipe face	10
Salute	10	Putting palms together	20	Falling down	13
Fan self	16	Pushing a person	8	Punch/slap a person	14
Two persons hugging	4	Giving something to someone	8	Shaking hands	6
High five	8	Two persons walking towards each other	8	Two persons walking apart from each other	8

Fig. 7 Different skeleton samples from our TS-RGBD dataset

scale disentangled unified GCN [22], because they showed a significant performance and are open source and their pre-trained models are available. Furthermore, according to [45], the choice of the source domain can be made based on its size and its similarity with the target domain. Since we are focusing on a skeleton-based approach, the NTU-RGBD dataset represents a leading option due to its popularity and data diversity. In addition, its structure is quite similar to our collected dataset since both were captured using depth sensors and both contain indoor actions.

Figure 9 illustrates different inputs and parameters of our transfer learning investigation.

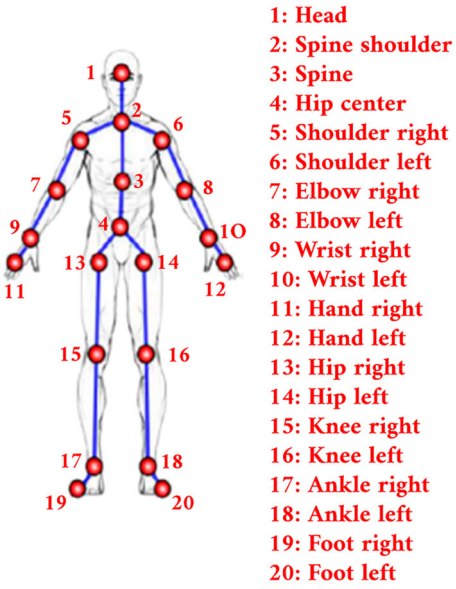
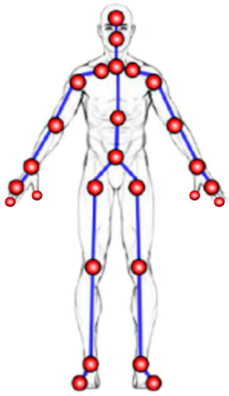


Fig. 8 Configuration of skeleton joints captured by Kinect v1

4.2.1 Source domain: NTU-RGBD dataset

It represents one of the most challenging and largest benchmarks due to the diversity of its data. It was captured using Kinect v2 which provides RGB, Depth, and skeleton sequences. The provided skeleton data consists of 25 human body joints (Fig. 10). NTU60-RGBD [18] was first introduced containing 60 indoor human action classes and a total of 56880 samples performed by 40 subjects with 80 different camera setups. Then, the extended version NTU120-RGBD [19] was introduced which contains additional 57.367 sequences of 60 extra indoor action classes with a total of 113.945 samples over 120 classes captured from 32 different camera setups and 106 subjects. The action classes are divided into three categories: daily actions (e.g. walking, reading, phone call ...etc), medical conditions (e.g. sneezing/coughing, staggering, falling down ...etc), and two-person interactions (e.g. pushing, punching, hugging ...etc).

Fig. 10 Skeleton joints provided by Kinect v2



The authors of this dataset proposed two protocols for recognition evaluation: xview protocol and xsub protocol:

> The xview protocol it focuses on cross-view action recognition. Its goal is to train a model on one set of viewpoints and evaluate its performance on a different set of viewpoints. Hence, the dataset is divided into two parts: *i) training set* that contains samples captured from one set of camera viewpoints, and *ii) testing set* that contains samples from a different set of camera viewpoints. The xsub protocol it focuses on cross-subject recognition where the objective is to train a model on a specific set of subjects and evaluate its performance on unseen subjects. The training set includes sequences of certain subjects, while the testing set consists of sequences of different subjects.

The three models: ST-GCN, 2s-AGCN, and MS-G3D achieved good results on this benchmark as shown in Table 3. Their high obtained performances on both protocols (xsub and xview) demonstrate their ability to achieve robust and generalized human action recognition across different subjects and camera viewpoints.

4.2.2 Diversity and similarity between source and target domains

Our dataset, was captured using depth sensor: Kinect v1, which provides 3D positions of 20 body joints. Unlike the

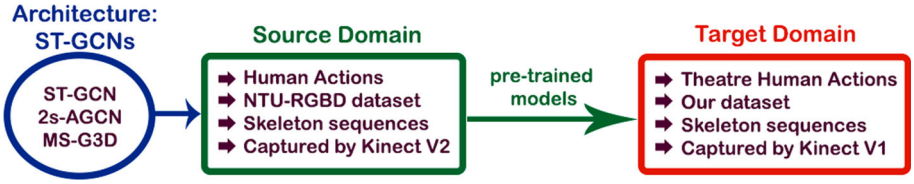


Fig. 9 Our transfer learning investigation parameters: The chosen architectures of the ST-GCN models (blue circle) : ST-GCN, 2S-AGCN, and MS-G3D. They are trained on a source domain (green rectangle): Human action skeleton sequences of NTU-RGBD captured by Kinect v2. Then, they are transferred to a target domain (red rectangle): Theatre human skeleton sequences of our dataset that are captured by Kinect v1

Table 3 The obtained accuracies by the selected ST-GCN models on NTU60-RGBD following the two evaluation protocols: xsub and xview

	NTU60-RGBD	
	xsub	xview
ST-GCN [20]	81.5%	88.3%
2s-AGCN [21]	88.5%	95.1%
MS-G3D [22]	**91.5%**	**96.6%**

The bold texts highlight the highest accuracies among the results

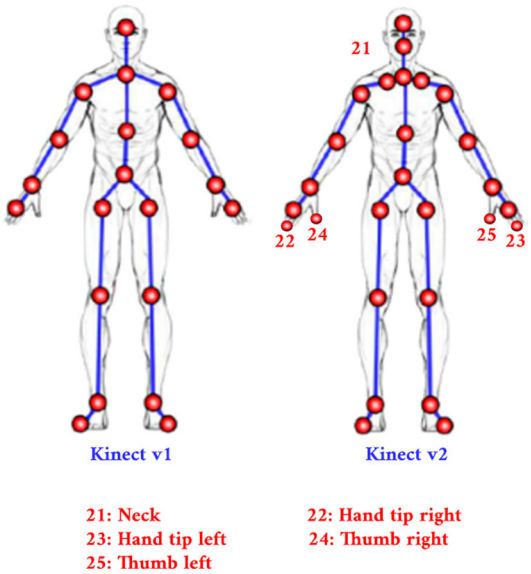


Fig. 11 The five additional joints captured by Kinect v2

NTU-RGBD dataset, it was captured using Kinect v2 which provides 3D positions of 25 joints. The additional five joints are: *neck*, *hand tip right*, *hand tip left*, *thumb right* and *thumb left*, as shown in Fig. 11.

Since these joints are not distant from the joints: *head*, *hand right*, *hand left*, *hand right* and *hand left* respectively, and the links between them do not represent body bones, we duplicated the positions of their neighboring joints in order to transform our dataset to 25 joints dataset and add the positions of the five missing joints (as shown in Fig. 12). This transformation allows us to save the same graph structure used in the training of the pre-trained models.

In addition, after we examined both datasets, it was evident that the actions in our dataset were executed at a slower pace when compared to the NTU action sequences. The average number of frames per sequence in our dataset is around 170 frames, while the average in NTU sequences is around 95 frames. Hence, we propose a framework for transfer learning application in section ***5.2.4*** to adapt the models to the new temporal correlations.

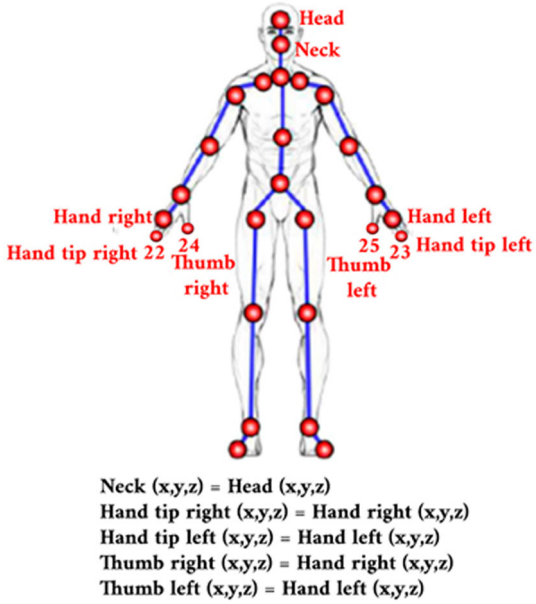


Fig. 12 The transformation of 20 joints skeleton to 25 joints skeleton

5 Validation and discussion

5.1 Training

We first trained the selected models on our dataset in order to calculate the performance of transfer learning and be able to compare it with the baseline. We utilized the identical architectures and parameters used while their training on NTU-RGBD:

- ST-GCN is comprised of 9 layers of ST-GCN units (spatial temporal graph convolution operators) followed by a global pooling to get a feature vector for each sequence and feed it to a SoftMax classifier [20]. We use the stochastic gradient descent: Adam optimizer, for the learning process of the model with a learning rate of 0.1 which is decayed by 0.1 after every 20 epochs.
- 2s-AGCN is composed of two adaptive graph convolution networks: J-stream and B-stream representing the networks of joints and bones, respectively (as shown in section *2.1.1*). Each network has a total of 9 blocks followed by a global average pooling layer to pool feature maps of different sequences to the same size and feed it to a SoftMax layer. Then, the scores of the SoftMax classifiers of both J-stream and B-stream are fused to predict the action label [21]. The learning rate is fixed at 0.1 and decayed by 0.1 every 20 epochs starting from epoch number 30.
- MS-G3D contains a stack of 3 spatial-temporal graph convolutional (STGC) blocks followed by a global average pooling layer and a SoftMax classifier. Each STGC

Table 4 The obtained accuracies by the ST-GCN models after training on our dataset

	Accuracy
ST-GCN	34.48%
2s-AGCN	39.66%
MS-G3D	41.38%

Table 5 Recognition performance of pre-trained ST-GCN, 2s-AGCN, and MS-G3D on theater data

	Accuracy.
ST-GCN	50.01%
2s-AGCN	55.73%
MS-G3D	**60.96%**

The bold text highlight the highest accuracies among the results

block is composed of two types of pathways to simultaneously capture long-range spatial and temporal dependencies using multi-scale convolutional layers, as well as regional spatial-temporal joint correlations by performing disentangled multi-scale convolutions. Then, the outputs from all pathways are aggregated as the STGC block output [22]. The learning rate is initiated at 0.5 and it is divided by 10 after every 30 epochs starting from epoch number 10.

Since the max number of frames in each action sequence in our dataset is 300 frames, all the samples with less than 300 frames are padded by replaying the sequences until they reach 300 frames. Then, we apply normalization and translation on the samples following [21, 46].

All the experiments in this research were repeated numerous times in order to ensure the validity of our findings, and the performance averages are reported.

The obtained results, illustrated in Table 4, show a low performance on our dataset which was expected due to: (1) the training data scarcity problem that prevents the models from capturing sufficient spatial and temporal patterns from the new dataset, (2) the low precision of the provided joint positions by Kinect v1 compared to Kinect v2.

5.2 Pre-trained models' evaluation on our dataset

Before we proceed with transfer learning, we test the pre-trained models on our dataset to compare their performance with the training data and the new data gathered in theater environment. We loaded the models' weights and applied the same data pre-processing steps presented in the training phase. We obtained the following outcomes:

We observe that the obtained accuracies on theater scenarios (Table 5) are relatively low compared to the accuracies obtained by the models on training data (presented in Table 3). This demonstrates the fact that the models struggle to accurately classify the new data due to its divergence with training data.

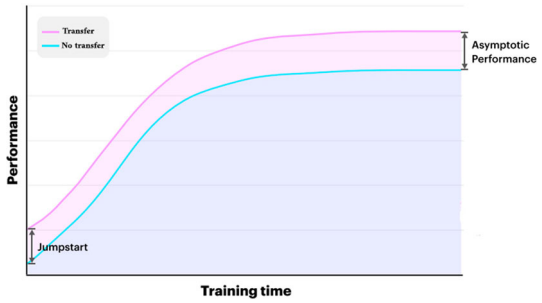


Fig. 13 Illustration of the jumpstart and the asymptotic performance[41]: The blue curve represents the baseline without transfer learning and the purple curve refers to the transferred model from a source domain

5.3 Transfer learning

In this section, we present our study of the effectiveness of transfer learning technique with the selected ST-GCNs using two frameworks: *configuration1* and *configuration2*. *Configuration1* represents the implementation of the fixed-weights transfer defined in section ***2.2***. Next, in order to adapt the pre-trained models to the target domain's temporal correlations, we propose a framework: *configuration2*, that combines the two transfer learning approaches: fixed-weights and fine-tuned transfer. The results are reported and discussed.

5.3.1 Evaluation metrics

In order to evaluate the performance of the transfer learning technique, the jumpstart and asymptotic metrics, introduced by Taylor and Stone[47], were employed to assess the transferability of the pre-trained models performance. **Jumpstart** is defined as the difference between a model's initial performance in a target task and its initial performance after being transferred from a source task. **Asymptotic performance**, on the other hand, measures the improvement in the final performance achieved in the target task through transfer learning, compared to the performance achieved by the baseline (as shown in Fig. 13).

5.3.2 Configuration1

As a first attempt, we implemented the fixed-weights transfer approach due to the similarity between the graph structure of both source and target domains after applying the data pre-processing method detailed in section ***4.2.2***. This graph-structure similarity results in a similarity of the spatial patterns that can be captured by the models from both datasets. Therefore, we saved the same architectures and parameters of the pre-trained models listed in section ***5.1***. We loaded the weights of the pre-trained models while keeping

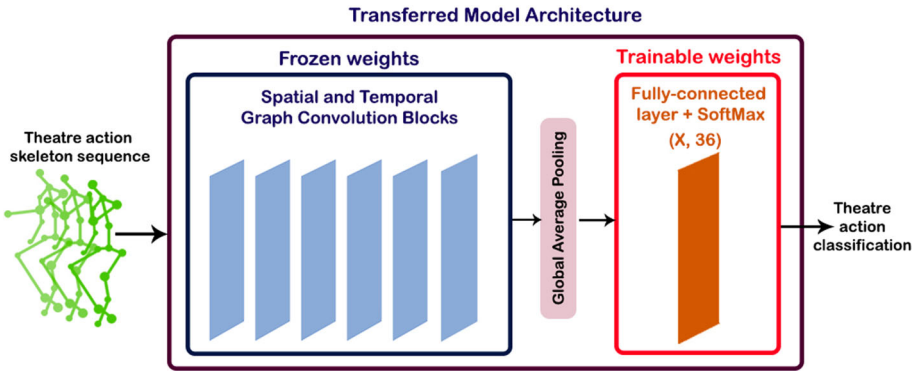


Fig. 14 An illustration of the framework of configuration1 on the models showing the trainable and the frozen blocks, as well as the added fully-connected layer with X representing the output feature tensor and 36 referring to the number of labels in the target domain

Table 6 The performances (final achieved accuracy, jumpstart, and the asymptotic performance) obtained by the ST-GCN models after implementing transfer learning following configuration1

	Final achieved performance	Jumpstart	Asymptotic performance
ST-GCN	0.67	0.10	0.33
2s-AGCN	0.68	0.12	0.29
MS-G3D	0.82	0.17	0.41

the same input layer and setting all spatio-temporal convolution blocks to non-trainable in order to preserve the weights learned from the source domain. Then, we modified the output layer of the models by adding a new fully-connected layer to match the 36 action classes present in our dataset (as shown in Fig. 14). Since all the models' layers are frozen and we are training only the fully-connected layer, we avoid to update the weights with large learning rates to prevent overfitting. Thus, we reduced the learning rate for each model by ten times with a decay of 0.1 after every 10 epochs for a stable training. Finally, we initiated the retraining process on our dataset.

5.3.3 Discussion1

Table 6 illustrates the calculated metrics and the final achieved performances after applying the transfer learning on each of the three models. From these results and the charts in Fig. 15, we observe that the three models obtained significant jumpstarts and asymptotic performances which prove their ability to transfer past knowledge and demonstrate that spatio-temporal graph neural networks can leverage strong properties in the source task for effective transfer, even with a less precise target dataset. The MS-G3D achieved the best transfer and converged faster compared to the other models, according to the charts (Fig. 15), which indicates that its architecture allows it to learn powerful transferable spatial and temporal patterns from the source domain resulting in the best transferability. On the other hand, ST-GCN achieved lower performance while training compared to 2s-AGCN as indicated in Table 6 , although, it achieved better asymptotic performance. This shows that ST-GCN benefits better than 2s-AGCN from sharing knowledge from the source domain and that its output classifier could adapt better to the target domain. In addition, the training time has been significantly reduced by the fact that all the models achieved their best accuracies before epoch number 100 compared to their baselines where they took more than 200 epochs. Furthermore, saving time results in the reduction of energy and carbon footprints of the models when training on a new large dataset.

5.3.4 Configuration2

In order to adapt the transfer learning technique to the temporal diversity between source and target domains introduced in Sect. 4.2.2, we proceeded with fine-tuning the temporal convolution blocks of each model by enabling them to be trainable so as to enhance the models' ability to learn more temporal features from the target domain. Therefore, we propose a framework that combines the two approaches of transfer learning by applying the fixed-weights approach with the spatial convolution blocks and fine-tuning the temporal convolution blocks (as shown in Fig. 16). We first load the weights of the pre-trained models, then, we freeze the weights of the spatial graph convolution layers of each model. We also change the output layers to match the number of labels in our dataset (36 labels), same as in configuration1. We preserve the same parameters of the last experiments except for the learning rate. A low learning rate in this case is crucial as we are retraining a larger number of weights (temporal convolution blocks) on our dataset which is typically very small. This can lead to an overfitting if we apply large weight updates. To overcome the overfitting, we lower the learning rates by 0.01 with a decay of 0.1 every 10 epochs. Finally, we lunch the retraining of the temporal convolution blocks as well as the output layer.

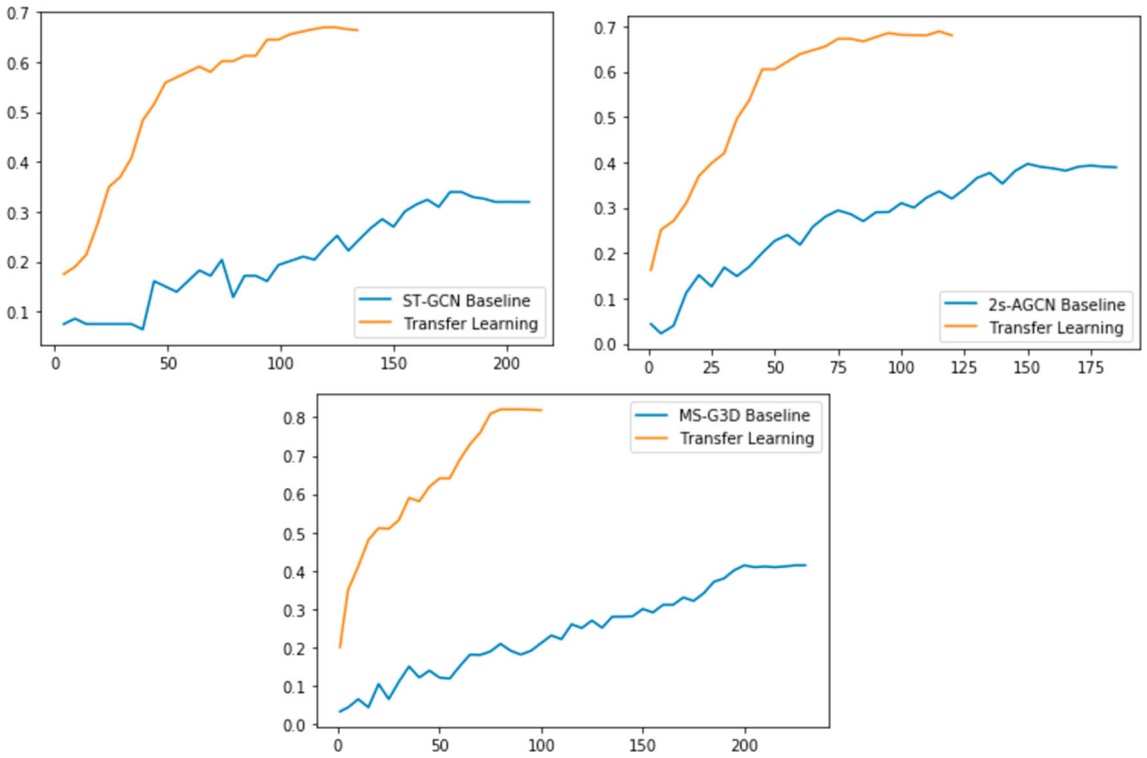


Fig. 15 The charts illustrate the accuracy progressions of different transferred models after applying configuration1 and their baselines without transfer learning

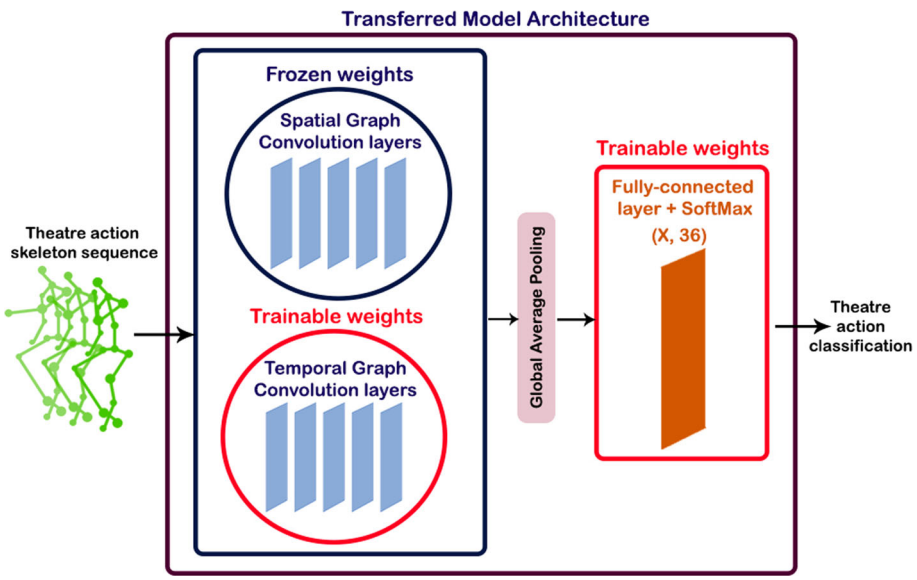


Fig. 16 An illustration of the configuration2 on the models showing the trainable blocks: temporal graph convolution layers + fully-connected layer, and the frozen blocks: spatial convolution layers

5.3.5 Discussion2

From Table 7, the charts in Fig. 17, and the bar charts in Fig. 18, we notice an improvement in the asymptotic performances obtained by all three models while the jumpstart values remain quite similar compared to the previous configuration. This indicates that the models were capable of acquiring new temporal properties from the target domain. It also demonstrates the vital role of selecting the right transfer learning approach for enhanced performance. The understanding of the similarity and diversity among the source and target domains is a key clue to determining which is the most suitable transfer learning approach to implement for better recognition performance. Although, we distinguish that the learning process of this second configuration takes more time to converge. As we can see from the charts in Fig. 17, during

Table 7 The performances (final achieved accuracy, jumpstart, and the asymptotic performance) obtained by the ST-GCN models after implementing transfer learning following configuration2

	Final achieved performance	Jumpstart	Asymptotic performance
ST-GCN	0.75	0.09	0.41
2s-AGCN	0.73	0.11	0.35
MS-G3D	0.88	0.19	0.47

the first epochs, the green curves (configuration2) of the three models are under orange curves (configuration1). This is due to the training process of temporal graph convolution layers as they are learning new temporal patterns from the target domain which results in a decreased training time compared to configuration1 where we only trained the fully-connected layer.

In summary, configuration2 significantly enhanced the recognition performance of the models. Furthermore, from the obtained accuracy (Table 7) and confusion matrices (Fig. 19), MS-G3D showed highest recognition performance representing a potential model for integration in theaters for actor gestures recognition.

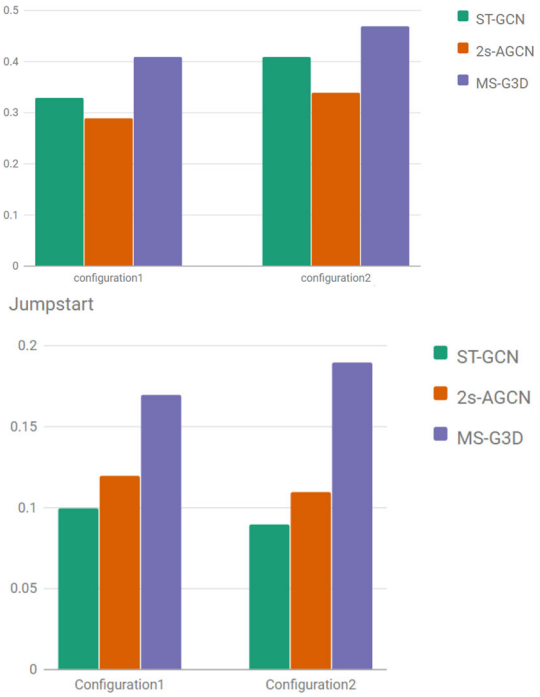


Fig. 18 Bar charts to compare the achieved jumpstart values and the asymptotic performances after applying configuration1 and configuration2

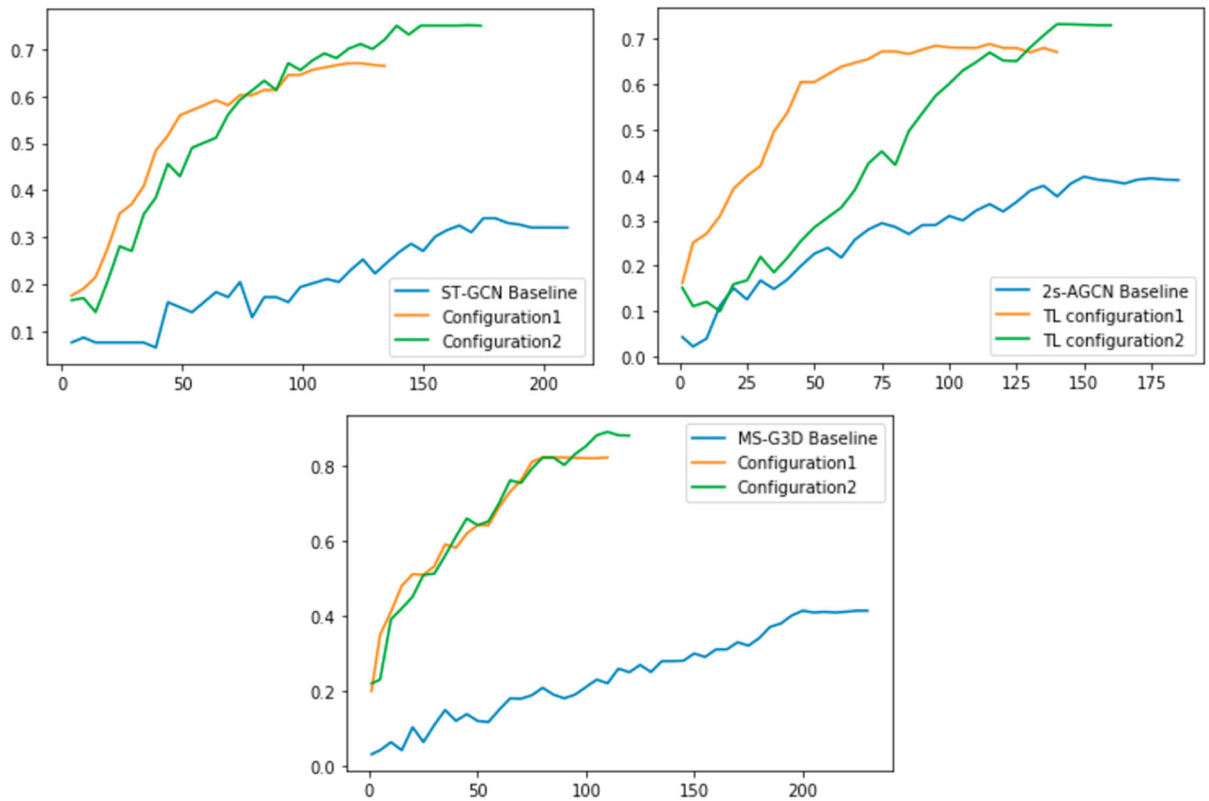


Fig. 17 For a clear comparison between the results obtained by configuration1 and configuration2, these charts illustrate the accuracy progressions of different transferred models of both configuration as well as their baselines

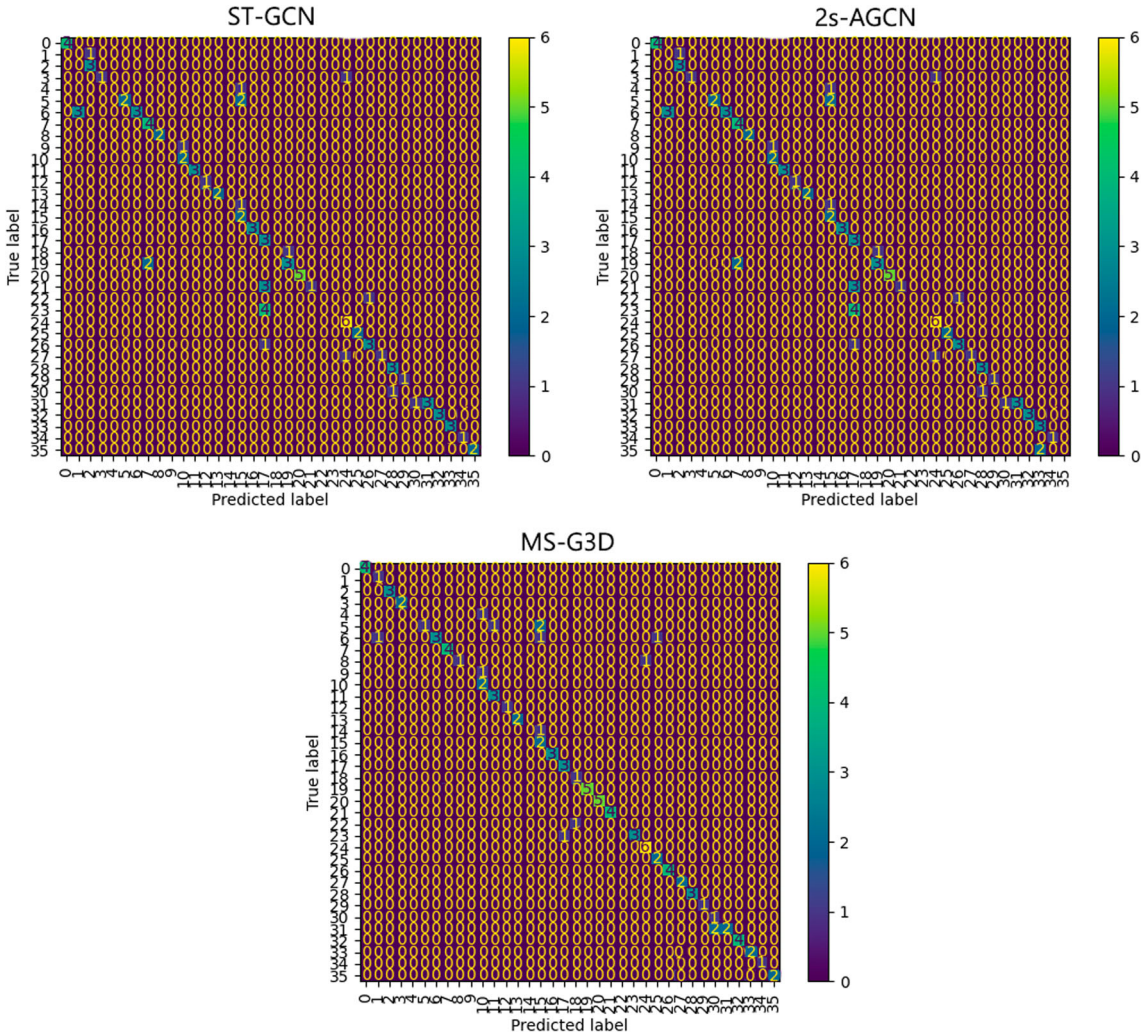


Fig. 19 Confusion matrices of the three models after transfer learning with configuration2

We also distinguish from the obtained confusion matrices in Fig. 19, that MS-G3D could surpass some misclassifications made by ST-GCN and 2s-AGCN with actions that have similar sequences of motions causing confusions such us the action picking up something with the action bowing. Other confusions were noticed with actions of type human-object interaction where objects are involved, such as eating and writing where both of them have similar hand motions which prevented the models from accurately classifying them.

Tackling the limitation of object information exclusion in the skeleton-based approach could significantly enhance the recognition of human-object interactions, improving the overall effectiveness of the system. For future work, we propose merging object detection models with ST-GCNs and other methods to cover complex actions involving objects, thus provide a better theater experience for visually impaired and blind people.

6 Conclusion

In this work, we introduced a Human Action Recognition (HAR) framework to recognize actor gestures in theatre performances, utilizing transfer learning technique. This framework addresses the limited availability of audio descriptions by presenting an automated solution that provides information about actor gestures to the BVI audience, enabling them to enjoy theatre performances.

To support our framework, we collected a new dataset of theatre human actions using a Kinect sensor in a theatre setting. We then employed three spatio-temporal graph convolution networks (ST-GCN, 2s-AGCN, and MS-G3D) for the recognition task, utilizing transfer learning to not only achieve better performance but also to avoid the need for training from scratch, which demands a large number of samples, significant computational resources, and is time-consuming.

This study represents the first to investigate transfer learning on ST-GCN models for skeleton-based human action recognition. The findings demonstrated the capability of these models to transfer previous knowledge, as well as learned spatial and temporal patterns from a source domain to new tasks. Through our experiments, we adapted the transfer learning process on the target domain based on its divergence from the source domain. The improvements we achieved validate the effectiveness of our proposed framework and highlight the importance of selecting appropriate transfer learning configurations based on the diversity and similarity between the source and target domains.

Ultimately, our efforts have significantly improved the recognition performance of pre-trained models in the targeted theatre environment, thereby promoting their integration within theatres to enhance accessibility.

Author Contributions Leyla Benhamida developed the methodology, collected the data, and wrote the main manuscript text. Slimane Larabi supervised the project and reviewed the manuscript. All authors conceptualized the study and approved the final manuscript.

Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data availibility The data collected and used during the current study are part of our TS-RGBD dataset that is publicly available on Github repository: GitHub Repository.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

1. Zatout, C., Larabi, S.: Semantic scene synthesis: application to assistive systems. Vis. Comput. **38**(8), 2691–2705 (2022)
2. Zatout, C., Larabi, S., Mendili, I., Ablam Edoh Barnabe, S.: Ego-semantic labeling of scene from depth image for visually impaired and blind people. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, p 0 (2019)
3. Zatout, C., Larabi, S.: A novel output device for visually impaired and blind people's aid systems. In: 2020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP), pp. 119–124 (2020). IEEE
4. Delloul, K., Larabi, S.: Egocentric scene description for the blind and visually impaired. In: 2022 5th International Symposium on Informatics and Its Applications (ISIA), pp. 1–6 (2022). IEEE
5. Delloul, K., Larabi, S.: Image captioning state-of-the-art: Is it enough for the guidance of visually impaired in an environment?. In: International Conference on Computing Systems and Applications, pp. 385–394 (2022). Springer
6. Ibelaiden, F., Larabi, S.: Visual place representation and recognition from depth images. Optik **260**, 169109 (2022)
7. Ibelaiden, F., Sayah, B., Larabi, S.: Scene description from depth images for visually positioning. In: 2020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP), pp. 101–106 (2020). IEEE
8. Ibelaiden, F., Larabi, S.: A benchmark for visual positioning from depth images. In: 2020 4th International Symposium on Informatics and Its Applications (ISIA), pp. 1–6 (2020). IEEE
9. Hegde, P., Devathraj, N., Sushma, S., Aishwarya, P.: Smart glasses for visually disabled person. Int. J. Res. Eng. Sci. (IJRES) **9**(7), 62–68 (2021)
10. Kandalan, R.N., Namuduri, K.: Techniques for constructing indoor navigation systems for the visually impaired: a review. IEEE Trans. Hum. Mach. Syst. **50**(6), 492–506 (2020)
11. Benhamida, L., Larabi, S.: Human action recognition and coding based on skeleton data for visually impaired and blind people aid system. In: 2022 First International Conference on Computer Communications and Intelligent Systems (I3CIS), pp. 49–54 (2022). IEEE
12. Khan, M.A., Javed, K., Khan, S.A., Saba, T., Habib, U., Khan, J.A., Abbasi, A.A.: Human action recognition using fusion of multiview and deep features: an application to video surveillance. Multimed. Tools Appl. **83**(5), 14885–14911 (2024)
13. Elharrouss, O., Almaadeed, N., Al-Maadeed, S., Bouridane, A., Beghdadi, A.: A combined multiple action recognition and summarization for surveillance video sequences. Appl. Intell. **51**, 690–712 (2021)
14. Zhou, X., Liang, W., Wang, K.I.-K., Wang, H., Yang, L.T., Jin, Q.: Deep-learning-enhanced human activity recognition for internet of healthcare things. IEEE Internet Things J. **7**(7), 6429–6438 (2020). https://doi.org/10.1109/JIOT.2020.2985082
15. Htet, Y., Zin, T.T., Tin, P., Tamura, H., Kondo, K., Chosa, E.: Hmm-based action recognition system for elderly healthcare by colorizing depth map. Int. J. Environ. Res. Public Health **19**(19), 12055 (2022)
16. Host, K., Ivašić-Kos, M.: An overview of human action recognition in sports based on computer vision. Heliyon **8**(6) (2022)
17. Lou, M., Li, J., Wang, G., He, G.: AR-C3D: Action recognition accelerator for human-computer interaction on FPGA. In: 2019 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–4 (2019). IEEE
18. Shahroudy, A., Liu, J., Ng, T.-T., Wang, G.: NTU-RGBD+ 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 (2016)
19. Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.-Y., Kot, A.C.: NTU-RGBD+ D 120: a large-scale benchmark for 3D human activity understanding. IEEE Trans. Pattern Anal. Mach. Intell. **42**(10), 2684–2701 (2019)
20. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
21. Shi, L., Zhang, Y., Cheng, J., Lu, H.: 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 (2019)
22. Liu, Z., Zhang, H., Chen, Z., Wang, Z., Ouyang, W.: 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 (2020)
23. Li, S., Li, W., Cook, C., Zhu, C., Gao, Y.: Independently recurrent neural network (indrnn): Building a longer and deeper rnn. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5457–5466 (2018)
24. Zhang, C., Tian, Y., Guo, X., Liu, J.: Daal: deep activation-based attribute learning for action recognition in depth videos. Comput. Vis. Image Underst. **167**, 37–49 (2018)
25. Li, M., Chen, S., Chen, X., Zhang, Y., Wang, Y., Tian, Q.: Actional-structural graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3595–3603 (2019)

26. Liu, J., Shahroudy, A., Xu, D., Kot, A.C., Wang, G.: Skeleton-based action recognition using spatio-temporal lstm network with trust gates. IEEE Trans. Pattern Anal. Mach. Intell. **40**(12), 3007–3021 (2017)
27. Yang, X., Tian, Y.: Super normal vector for activity recognition using depth sequences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 804–811 (2014)
28. Li, Y., Xia, R., Liu, X., Huang, Q.: Learning shape-motion representations from geometric algebra spatio-temporal model for skeleton-based action recognition. In: 2019 IEEE International Conference on Multimedia and Expo (ICME), pp. 1066–1071 (2019). IEEE
29. Xu, Y., Cheng, J., Wang, L., Xia, H., Liu, F., Tao, D.: Ensemble one-dimensional convolution neural networks for skeleton-based action recognition. IEEE Signal Process. Lett. **25**(7), 1044–1048 (2018)
30. Kim, T.S., Reiter, A.: Interpretable 3d human action analysis with temporal convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1623–1631 (2017). IEEE
31. Li, S., Li, W., Cook, C., Zhu, C., Gao, Y.: Independently recurrent neural network (indrnn): Building a longer and deeper RNN. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5457–5466 (2018)
32. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE Trans. Neural Netw. Learning Syst. **32**(1), 4–24 (2020)
33. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
34. Li, M., Chen, S., Chen, X., Zhang, Y., Wang, Y., Tian, Q.: Actional-structural graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3595–3603 (2019)
35. Huang, Z., Shen, X., Tian, X., Li, H., Huang, J., Hua, X.-S.: Spatio-temporal inception graph convolutional networks for skeleton-based action recognition. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 2122–2130 (2020)
36. Liao, R., Zhao, Z., Urtasun, R., Zemel, R.S.: Lanczosnet: Multi-scale deep graph convolutional networks. arXiv preprint arXiv:1901.01484 (2019)
37. Luan, S., Zhao, M., Chang, X.-W., Pre- cup, D.: Break the ceiling: stronger multi-scale deep graph convolutional networks. In: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 32, pp. 10943–10953. Curran Associates, Inc. arXiv:1906.02174 (2019)
38. Abdulazeem, Y., Balaha, H.M., Bahgat, W.M., Badawy, M.: Human action recognition based on transfer learning approach. IEEE Access **9**, 82058–82069 (2021)
39. Ray, A., Kolekar, M.H., Balasubramanian, R., Hafiane, A.: Transfer learning enhanced vision-based human activity recognition: a decade-long analysis. Int. J. Inform. Manag. Data Insights **3**(1), 100142 (2023)
40. Wang, J., Zheng, V.W., Chen, Y., Huang, M.: Deep transfer learning for cross-domain activity recognition. In: Proceedings of the 3rd International Conference on Crowd Science and Engineering, pp. 1–8 (2018)
41. Kooverjee, N., James, S., Van Zyl, T.: Investigating transfer learning in graph neural networks. Electronics **11**(8), 1202 (2022)
42. Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., Sheikh, Y.: Openpose: realtime multi-person 2D pose estimation using part affinity fields. IEEE Trans. Pattern Anal. Mach. Intell. **43**(1), 172–186 (2021)
43. Lacoste, A., Luccioni, A., Schmidt, V., Dandres, T.: Quantifying the carbon emissions of machine learning. arXiv preprint arXiv:1910.09700 (2019)
44. Henderson, P., Hu, J., Romoff, J., Brunskill, E., Jurafsky, D., Pineau, J.: Towards the systematic reporting of the energy and carbon footprints of machine learning. J. Mach. Learning Res. **21**(1), 10039–10081 (2020)
45. Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P.-A.: Transfer learning for time series classification. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 1367–1376 (2018). IEEE
46. Shi, L., Zhang, Y., Cheng, J., Lu, H.: 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 (2019)
47. Taylor, M.E., Stone, P.: Transfer learning for reinforcement learning domains: a survey. J. Mach. Learn. Res. **10**(56), 1633–1685 (2009)

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com