Cross-Skeleton Interaction Graph Aggregation Network for Representation Learning of Mouse Social Behavior

Feixiang Zhou, Xinyu Yang, Graduate Student Member, IEEE, Fang Chen, Long Chen, Zheheng Jiang, Hui Zhu, Reiko Heckel, Haikuan Wang, Minrui Fei, and Huiyu Zhou

Abstract—Automated social behaviour analysis of mice has become an increasingly popular research area in behavioural neuroscience. Recently, pose information (i.e., locations of keypoints or skeleton) has been used to interpret social behaviours of mice. Nevertheless, effective encoding and decoding of social interaction information underlying the keypoints of mice has been rarely investigated in the existing methods. In particular, it is challenging to model complex social interactions between mice due to highly deformable body shapes and ambiguous movement patterns. To deal with the interaction modelling problem, we hereby propose a Cross-Skeleton Interaction Graph Aggregation Network (CS-IGANet) to learn abundant dynamics of freely interacting mice, where a Cross-Skeleton Node-level Interaction module (CS-NLI) is used to model multi-level interactions (i.e., intra-, inter- and cross-skeleton interactions). Furthermore, we design a novel Interaction-Aware Transformer (IAT) to dynamically learn the graph-level representation of social behaviours and update the node-level representation, guided by our proposed interaction-aware self-attention mechanism. Finally, to enhance the representation ability of our model, an auxiliary self-supervised learning task is proposed for measuring the similarity between cross-skeleton nodes. Experimental results on the standard CRMH13-Skeleton and our PDMB-Skeleton datasets show that our proposed model outperforms several state-of-the-art approaches.

Index Terms—Social behavior recognition, graph neural network, self-attention, self-supervision, cross-skeleton.

I. INTRODUCTION

THE analysis of rodent social behaviour is an interesting issue in neuroscience and pharmacology. Laboratory mice provide a valuable platform to study psychiatric and neurological disorders such as Huntington’s [1], Alzheimer’s [2], schizophrenia [3], as well as Parkinson’s disease (PD) [4] because mice offer several advantages, including their genetic similarity to humans and the ability to manipulate and control experimental conditions. Traditionally, social behaviour identification is performed by manually annotating hours of video recordings of interactions between mice with pre-defined behaviour labels. Unfortunately, this human labelling practice is time-consuming, error-prone and highly subjective. Recent advances in computer vision and pattern recognition have facilitated automated analysis of mouse behaviours [5], [6], [7], [8], [9], [10], which provides another dimension to understand the relationship between neural activities and behavioural phenotypes in neuroscience research.

Mouse social behaviour recognition is non-trivial due to the intricate nature of not just individual behaviours but also the interactions of mice. Compared to human behaviours, mouse social interactions exhibit ambiguous movement patterns as these interactions can involve subtle cues and rapid movements, making it difficult to recognise and interpret specific behaviours. This motivates us to design a novel computer vision solution to analyse intricate movement patterns and social interactions of mice, which is valuable in fields like biobehavioral research [9], [11]. Additionally, our study has potential implications in studying neurological diseases, e.g., PD [4], [6], [12]. By investigating how PD affects social behaviours of mice, we can gain insights into its neurological underpinnings, potentially contributing to the early diagnosis of human diseases and the development of treatments or interventions. We believe that our research has the potential to yield valuable insights in the domains of image processing, neuroscience and biobehavioral research.

As more and more accurate results have been provided by deep learning based pose estimation models [13], [14], [15], [16], researchers have started to directly recognise mouse social behaviours using pose information (i.e., the locations of keypoints generated by pose estimators) [7], [17]. Compared with RGB features, pose information includes only 2D or 3D positions of keypoints, which may be free of environmental noise (e.g. complex background and illumination changes) [18]. However, the features extracted by most of the existing systems are hand-crafted, based on pre-defined keypoints. For instance, in [17], distance relations between two noses (i.e., distance feature) are represented by the distance between the noses of two mice. Actually, such hand-crafted shallow
features are insufficient to describe the dependency between the corresponding keypoints. To this end, we need to develop an effective way to automatically model the spatio-temporal interactions between keypoints.

Graph convolutional networks (GCNs) [19], which generalise convolution from images to graphs, have been successfully adopted in many areas to model graph-structured data, especially in skeleton based human action recognition approaches [20], [21], [22], [23], [24]. Nevertheless, most of GCN-based methods have been designed for action recognition of single objects rather than multiple interacting subjects. In most of these established models, a standard human skeleton with all joints is utilised to model the potential spatio-temporal dependencies between the joints. To capture discriminative action features, multi-view solutions [25] consisting of two ensemble models with different skeleton typologies are developed to utilise comprehensive information. Although such an approach can significantly improve the discriminative capability, the two sub-models need to be trained independently - how to select a new and effective skeleton topology is difficult to determine. Moreover, to obtain the graph-level representation\(^1\) that represents a specific action, global average pooling (GAP) [26] is normally used to aggregate node-level representation from the final stage of the network. However, this operation processes all node features equally without considering the importance of different nodes, structural constraints and dependencies between them. This limitation results in a constrained ability to globally represent complex social interactions due to the ambiguous motion patterns of mice. Recently, GCN [27], [28] and Transformer [29], [30] have been introduced for interactive action recognition. Although they consider intra-body and inter-body relations, the overall representation ability of the spatio-temporal interactions is still limited because of the above issue. Hence, how to effectively model complex and diverse social interactions of mice remains an open problem.

We here propose a Cross-Skeleton Interaction Graph Aggregation Network (CS-IGANet) to effectively learn abundant interaction relationships of mice, shown in Fig. 1. Our work is also based on GCN owing to its advantages in graph-structured data modelling. However, different from the above methods, we integrate GCN and Transformer to handle two aspects of social interactions (i.e., node-level and graph-level representation learning) by proposing a novel multi-level interaction module and a hierarchical interaction aggregation module.

Inspired by multi-view [25] and multi-scale [31], [32] skeleton structures for individual action modelling, we propose a novel Cross-Skeleton Node-level Interaction (CS-NLI) module, shown in Fig. 1(b), to model the intra- (between keypoints of each mouse), inter- (between keypoints of different mice) and cross-skeleton (between keypoints of different skeletons) interactions of mice in a unified way. Unlike most existing methods [30] for interactive action recognition, we consider the three types of interactions simultaneously to better learn multi-scale social interactions. Thus, we first introduce dense and sparse skeletons to describe multi-scale spatial structures of a mouse, shown in Fig. 1(a). Here, mouse skeleton refers to a list of keypoint connections [33]. For each skeleton branch, a GCN-based module [26] is first adopted to model the intra-skeleton interaction of each mouse, before we fuse dense geometric properties and velocity information. Afterwards, we model the social interactions of mice, where an adaptive inter-skeleton interaction matrix is formulated to integrate the motion information from two or more interactive mice. Similarly, we further explore the cross-skeleton interactions of mice. With the proposed multi-level interactions, our CS-NLI can discover abundant dynamic relations of social interactions, leading to more informative node-level representation.

Different from existing works [26], [30], [31] using GAP to directly generate graph-level representation, we propose to learn graph-level representation hierarchically while keeping crucial interaction information. This is achieved by a novel Interaction-Aware Transformer (IAT) guided by the proposed interaction-aware self-attention unit, shown in Fig. 1(c). The encoder aims at mining behaviour-related interaction saliency (i.e., conspicuous nodes) based on intra- and inter-skeleton interactions, where the node-level representation is used to generate multiple subgraphs and the last one denotes the graph-level representation of social behaviour. Afterwards, such graph-level representations from different skeletons are integrated with representations generated by trivial pooling methods (e.g., average [26], max [34]) to enhance the graph-level representation. Moreover, a decoder is designed to adaptively update the node-level representation via the proposed interaction-aware self-attention.

We believe that there exists meaningful similarity between the dense and sparse skeletons that both describe spatial configurations of a mouse. To better preserve these attributes within the cross-skeleton pairwise nodes, we design an auxiliary self-supervised learning module. By jointly optimising the self-supervised objective function and the traditional classification loss function (i.e., cross-entropy loss), our proposed model can yield more discriminative representation.

The main contributions can be summarised as follows:

- We propose a novel Cross-Skeleton Interaction Graph Aggregation Network (CS-IGANet) to learn mouse social behaviour representation, where dense and sparse skeletons cooperatively explore the spatio-temporal dynamics of social interactions.
- The proposed Cross-Skeleton Node-level Interaction (CS-NLI) module is able to engender powerful node-level representation by modelling multi-level interactions of mice, i.e., intra-, inter- and cross-skeleton interactions, where multi-order dense information are fused for inferring corresponding interaction patterns.
- The proposed Interaction-Aware Transformer (IAT) allows for dynamic updating of graph- and node-level representation. This can be achieved by designing an encoder-decoder architecture, where the former hierarchically aggregates node-level representation for graph-level representation learning whilst the latter adaptively

\(^1\)In this paper, node-level representation refers to the features of each node provided in a graph. Graph-level representation refers to the overall features of the whole graph.
update node-level representation for extracting higher-level features.

- We introduce an auxiliary self-supervised learning strategy to enable the proposed model to focus on the similarity between pairwise nodes from different skeletons, so as to enhance the representation ability of our model.

II. RELATED WORK

A. Pose-Based Mouse Social Behaviour Recognition

Mouse behaviour recognition can be divided into two main categories: methods relying on RGB features and those utilising pose features. While most existing works [5], [6], [9], [10] focus on extracting RGB features from videos, there is a limited exploration of mouse social behaviours through pose analysis. Giancardo et al. [35] constructed a spatio-temporal feature vector composed of 13 measurements (e.g., relative position, shape and movement) based on the tracking results of the proposed tracker, and then applied random decision trees to classify those extracted features. Similarly, Arac et al. [7] detected the nose, head, body and tail of each mouse using the standard YOLOv3 network, based on the extracting features such as distance between the body centers. However, these extracted features are shallow with limited spatio-temporal representation.

Thanks to pose estimation models [13], [14], [15], people directly adopted the results of pose estimators to conduct downstream tasks such as behaviour recognition. Nilsson et al. [17] reported SimBa that analyses mouse social behaviours based on the pose estimation tracking results, where a random forest algorithm was leveraged to classify behavioural patterns. However, the 490 features (e.g. area of mouse convex hull, distance between part1 and part2) in their system are still shallow. Similar to SimBa, Segalin et al. [36] also introduced a system called MARS for the analysis of social behaviours, whereas 270 keypoint based spatio-temporal features were generated. However, these hand-crafted features cannot capture robust spatio-temporal relationships of keypoints, especially for complex social interactions.

B. Skeleton-Based Human Action Recognition

1) GCN-Based Methods: Graph Convolutional Networks (GCNs) [26], [31], [37], [38], [39], [40], [41], [42] are prevalent for processing skeleton data due to their strong ability of capturing structural dependencies of joints. The construction of GCNs generally follows two principles: spectral perspective [43], [44] and spatial perspective [45], [46]. Spectral methods leverage the eigenvalues and eigenvectors of the graph Laplace matrices, and they operate in the Fourier domain. In contrast to spectral methods, spatial approaches, akin to traditional CNNs, perform convolutions directly in the spatial domain by aggregating information from local neighbourhoods. However, leveraging multi-head attention in GCNs improves accuracy but often leads to overparameterization and computational complexity [47]. To alleviate these issues, pruning-based methods [48], [49], [50] have emerged as mainstream, aiming to remove connections that have minimal impact on classification performance. Unlike these methods, our work integrates GCN and Transformer to handle two aspects of social interactions (i.e., node- and graph-level representation learning). This work follows the spatial methods.

Yan et al. [37] exploited GCN for skeleton-based action recognition, and utilised the spatial temporal graph convolutional network (ST-GCN) to model the skeleton data as the graph structure. However, it uses a fixed skeleton graph and represents only the physical structure of the human body. Shi et al. [26] delineated a two-stream GCN model,
i.e., 2s-AGCN to learn an adaptive graph where both the joint and bone information is explicitly utilised, significantly improving the model performance. Liu et al. [31] introduced a sophisticated feature extractor named MS-G3D, in which the disentangled multi-scale aggregator and G3D are used to eliminate redundant dependencies between neighbourhoods and model spatio-temporal information interaction, respectively. Wang et al. [25] proposed a MV-IGNet network to formulate complementary action representations by adopting two pre-defined skeleton topologies. As we discussed above, it is difficult to determine a new and effective skeleton topology. Chen et al. [39] proposed to dynamically learn different topologies and effectively aggregate joint features in each channel. Chi et al. [23] designed a novel learning objective to learn compact latent representations. In [24], ST-GCN has been reformulated as a continual inference network, enabling online frame-by-frame predictions in a highly efficient manner. Although most of the aforementioned approaches have produced promising results in skeleton-based human action recognition, they mainly focus on single-object action without modelling the interactions between subjects, and hence lack the ability to generalise social representations.

2) Transformer-Based Methods: Transformer [51] using self-attention has also been applied to graph-structured data modelling due to its powerful ability of modelling long-range dependencies [52], [53]. Recent studies have extended the Transformer model for skeleton-based action recognition [38], [54], [55], [56]. Plizzari et al. [38] proposed a spatio-temporal transformer network (i.e., ST-TR) for skeleton-based action recognition, where a spatial self-attention module was used to explore intra-frame interactions between different joints and a temporal self-attention module to model inter-frame correlations. Zhang et al. [54] introduced a transformer network, where the spatial and temporal dimensions are parallelly separated. Nevertheless, this attention learning neglects the influence of different individual body joints on spatio-temporal action feature representations. Huu et al. [56] designed a hybrid architecture that combines GCN and Transformer to learn joint and body-part correlations using different cross-attention blocks. However, these transformer-based networks are normally constrained by relatively high computational complexity.

C. Interactive Action Recognition

Recently proposed interactive action recognition methods [27], [28], [29] aim to capture spatio-temporal interactive features. Zhu et al. [27] employed a GCN with separate graphs and proposed inter-body graph convolution with a dynamic relational adjacency matrix to capture interactions. Different from this work, Li et al. [28] introduced a novel two-person graph topology to represent inter-body and intra-body correlations. Other works [29], [30] have adopted the self-attention mechanism for human interaction modelling. Pang et al. [29] proposed to model the interactive relationship of subjects from both semantic and distance levels via an interaction graph Transformer. Wen et al. [30] designed an interactive Spatio-temporal network to jointly model entity, temporal and spatial relations between interacting entities by fusing three-dimensional interactive spatio-temporal features. However, these approaches still face challenges in exploring mouse social interactions with ambiguous movement patterns.

D. Mouse Pose Estimation

Mouse pose estimation provides useful information for ethologically relevant behaviours. In recent years, deep learning-based methods [9], [13], [15], [57], [58] have been proposed for mouse or other animal pose estimation. Mathis et al. [13] proposed an animal pose estimation system called DeepLabCut, which adopts the feature detectors of DeeperCut with readout layers for markerless pose estimation. The system is trained with transfer learning, and it has been widely adopted in the behavioural research community. Similarly, LEAP [57] was developed to estimate poses in videos of individual mice and fruit flies, which provides a graphical interface for labelling body parts. However, its preprocessing is computationally expensive, thus limiting the application of their system in other environments. Pereira et al. [58] further designed a general framework called SLEAP for multi-animal pose estimation, which achieves comparable or improved accuracy compared to other systems for single-animal pose estimation with faster inference speed.

III. PROPOSED METHODS

The skeleton sequence of $K$ mice with $T$ frames and $N$ keypoints can be represented as a spatio-temporal graph $G = (V, E, X)$. $V = \{(v_{t,n,k} \mid t, n, k \in Z, 1 \leq t \leq T, 1 \leq n \leq N, 1 \leq k \leq K)\}$ is the set of all the nodes $v_{t,n,k}$ of the mouse skeleton graph, i.e., keypoints of the skeleton over all the time sequence. $E$ represents the edge set consisting of two subsets, i.e., spatial topology $E_S = \{(v_{t,n,k}, v_{i,m,k}) \mid 1 \leq t \leq T, 1 \leq n, m \leq N, 1 \leq k \leq K\}$ that describes the relationship between any pair of keypoints $(v_t, v_{nm})$ of mouse $k$ at time $t$, and temporal topology $E_T = \{(v_{t,n,k}, v_{t+1,n,k}) \mid 1 \leq t \leq T, 1 \leq n \leq N, 1 \leq k \leq K\}$ indicating the relationship between keypoints along consecutive time frames. $E_S$ of each mouse at time $t$ can be formulated as an adjacency matrix $A \in \mathbb{R}^{N \times N}$ where initial element $a_{nm} \in [0, 1]$ reflects the correlation strength between $v_n$ and $v_m$. $X = \{v_{t,n,k} \mid 1 \leq t \leq T, 1 \leq n \leq N, 1 \leq k \leq K\}$ is a node features set, which is represented as a matrix $X \in \mathbb{R}^{C \times T \times N \times K}$ where $x_{t,n,k} = X(c, t, n, k) \in \mathbb{R}^C$ is the $C$ dimensional feature vector for node $v_{t,n,k}$. In this work, we focus on skeleton-based mouse social behaviour recognition in long videos. During training, we wish to obtain a continuous behaviour sequence by the sliding window over the long video, where each window centered at a certain frame only contains one specific behaviour [6]. Hence, the behaviour sequence is represented as $X = \{X^{(1)}, X^{(2)}, \ldots, X^{(B)}\} \in \mathbb{R}^{B \times C \times T \times N \times K}$, where $B$ is the total number of sliding windows and $X^{(B)} \in \mathbb{R}^{C \times T \times N \times K}$ is the feature set of the $B$-th window in the long video. Consequently, given $X$, we aim to learn a non-linear prediction function to model the relationship between a sequence of
the predicted labels (i.e., $Y = [Y^{(1)}, Y^{(2)}, \ldots, Y^{(B)}]$) and $X$. In experiments, following the standard formulations [26], [37], we reshape the input sequence to $X \in \mathbb{R}^{K \times B \times C \times T \times N}$ by moving $K$ to the batch dimension. Normally, for each sliding window, one behaviour is described as $A$ and $X \in \mathbb{R}^{C \times T \times N}$, with $X_i \in \mathbb{R}^{C \times N}$ being the node features at time $t$.

In this section, we will fully explain our proposed CS-NLI module that jointly models intra-, inter- and cross-skeleton interactions, our proposed IAT that dynamically creates graph-level representation and updates node-level representation, and the proposed auxiliary self-supervised learning strategy that encourages the proposed model to focus on the similarity between cross-skeleton pairwise nodes. The overview of our proposed framework is illustrated in Fig. 1.

A. Cross-Skeleton Node-Level Interaction

Unlike traditional behaviour recognition tasks that focus solely on individual behaviours [5], understanding mouse social behaviour requires capturing the nuanced dynamics arising from the collective movements and interactions within a group of mice. Although the behavioural representation of each mouse on the both spatial and temporal domains can be interpreted by existing GCN-based network [25], [26], [31], they ignore the interaction between mice, which is crucial for fully learning the social behaviour representation. Therefore, in this section, we aim to explore the interaction between the keypoints of each mouse (i.e., intra-skeleton interaction) as well as interaction patterns between mice (i.e., inter-skeleton interaction) simultaneously. As aforementioned, we further model the spatio-temporal relationship between dense and sparse skeletons (i.e., cross-skeleton interaction) to learn skeleton-shared representations. The architecture of our proposed CS-NLI module is shown in Fig. 2.

1) Intra-Skeleton Interaction Modelling: Similar to [25] and [59], we first construct $I$ types of skeleton sequences to learn behavioural information of mice. Following [33], we define the dense physical connections of all the keypoints to form the dense skeleton structure of each mouse, as shown in Fig. 1(a). Then, we further design a sparse structure where keypoints in the same body part are aggregated into one keypoint. The transition from dense skeleton structure to sparse skeleton structure is inspired by the established human body parts structure [21], [32], [56]. In these works, the average operation is normally utilised to aggregate neighbouring joints to obtain part-based skeleton structure. While acknowledging the structural differences between mouse and human bodies, we can apply a similar aggregation method to constructing the sparse skeleton structure consisting of three mouse parts, i.e., head, body and tail. The multi-scale skeleton graphs facilitate the extraction of behaviour-relevant features across different levels of granularity (a comparison with CS-NLI using two dense graphs is provided in Tab. S2). This is because some behaviours such as ‘approach’ can be identified based on the movements of keypoints from the sparse skeleton without knowing the exact locations of each keypoint (e.g., left and right ears).

To model the intra-skeleton interaction of mice (without loss of generality, we use two mice in this paper), we adopt the GCN-TCN structure shown in [26] to encode the spatio-temporal representation (more details about this model can be found in Supplementary A). We adopt the standard GCN to extract spatial features from the structural node connections due to its flexibility on skeleton modelling. TCN is then used to extract temporal features from skeleton sequences. A residual connection is also added for both GCN and TCN. Mathematically, given the skeleton sequence $X_{si}^l \in \mathbb{R}^{C_i \times T_i \times N_{si}}$, $\forall i \in \{1, 2, \ldots, I\}$, where $s_i$ and $N_{si}$ denote the $i$-th skeleton and the number of the nodes in this skeleton, we define such interaction as follows:

$$X_{si}^{l+1} = \Gamma(\Phi(X_{si}^l)) + X_{si}^l$$

(1)

where $\Phi(\cdot)$ and $\Gamma(\cdot)$ represent spatial and temporal modelling, respectively. $l$ represents the $l$-th layer. The input $X_{si}^l$ in $\Phi(\cdot)$ is reshaped to $\mathbb{R}^{C_i \times T_i \times N_{si}}$ by assigning $T_i$ as the channel dimension, and is then projected back to $\mathbb{R}^{C_i \times T_i \times N_{si}}$ for temporal convolution. We can obtain the intra-skeleton interaction by stacking multiple residual GCN and TCN modules. The output of such interaction in the $l$-th layer of our network is represented as $X_{si}^{l \text{intra}}$.

2) Inter-Skeleton Interaction Modelling: Based on the high-level features extracted using a sequence of residual GCN and TCN modules defined in Eq. (1), we then explore the interaction pattern between mice where the inter-skeleton interaction matrix, i.e., $A_{ski}^{l \text{k1} \to k2}$, needs to be derived (for two mice, we have $A_{ski}^{l \text{k1} \to k2}$ and $A_{ski}^{l \text{k2} \to k1}$). We first explicitly embed both geometric distance and velocity information into the representation encoded by keypoint information because they carry behaviour-related features [21], [25], [26]. In particular, mouse body is highly deformable and most mouse behaviours have a relatively short duration. Unlike these approaches, we here extract dense geometric distance (Eq. (2)) and velocity information (Eq. (3)) simultaneously. For the node $v_m$ at the $l$-th layer of our network, we consider the relative positions between it and all the remaining

$$d_{mk} = \sqrt{(x_m - x_k)^2 + (y_m - y_k)^2 + (z_m - z_k)^2}$$

(2)

$$v_{mk} = \frac{v_m - v_k}{d_{mk}}$$

(3)
nodes to construct a dense geometric distance set \( R_{s_j}^{l,b} = \left\{ r_{s_j}^{l,b}(n, m) \mid n = 1, 2, \cdots, N_{s_j} \right\} \), where
\[
  r_{s_j}^{l,b}(n, m) = ||X_{s_j,n}^{l,b} - X_{s_j,m}^{l,b}||
\]
(2)

\( X_{s_j,m}^{l,b} = X_{s_j,\cdots,m}^{l,b} \in \mathbb{R}^{C \times N_{s_j}} \) is the feature of node \( v_m \) across the temporal domain. Similarly, we produce a dense velocity set of time \( t \), i.e., \( R_{s_j}^{l,v} = \left\{ r_{s_j}^{l,v}(p, t) \mid p = 1, 2, \cdots, T \right\} \) with the following definition:
\[
  r_{s_j}^{l,v}(p, t) = ||X_{s_j,p}^{l,v} - X_{s_j,t}^{l,v}||
\]
(3)

where \( X_{s_j,\cdot}^{l,v} = X_{s_j,\cdot,\cdot}^{l,v} \in \mathbb{R}^{C \times N_{s_j}} \) is the feature of all the nodes at time \( t \).

We integrate the features of node \( v_m \) over the temporal domain (i.e., \( X_{s_j,m}^{l,b} \)) with its dense geometric distance, and the features of all the nodes at time \( t \) (i.e., \( X_{s_j,\cdot}^{l,v} \)) with its dense velocity. For the former, \( X_{s_j,m}^{l,b} \) and one of the elements of set \( R_{s_j}^{l,b} \) are fused by concatenation, followed by performing dimensionality reduction on features. All the features are then stacked together, and we also add a residual connection in order to obtain the features of node \( v_m \) (i.e., \( H_{s_j,m}^{l,b} \)), using the dense geometric distance. Similarly, we can obtain the features of all the nodes at time \( t \) (i.e., \( H_{s_j,\cdot}^{l,v} \)), using the dense velocity information. We have the following expression:
\[
  H_{s_j,m}^{l,b} = \left( \sum_{n=1}^{N_{s_j}} g_{s_j}^{l,b}(X_{s_j,m}^{l,b}, R_{s_j,n,m}^{l,b}) \right) + X_{s_j,m}^{l,b}
\]
(4)

\[
  H_{s_j,\cdot}^{l,v} = \left( \sum_{p=1}^{T} g_{s_j}^{l,v}(X_{s_j,\cdot,p}^{l,v}, R_{s_j,p}^{l,v}) \right) + X_{s_j,\cdot}^{l,v}
\]
(5)

where \( X_{s_j,m}^{l,b} \) and \( X_{s_j,\cdot}^{l,v} \) are reshaped to \( \mathbb{R}^{C \times T_l} \) and \( \mathbb{R}^{C \times N_{s_j}} \), respectively. \( g_{s_j}^{l,b}(\cdot) \) and \( g_{s_j}^{l,v}(\cdot) \) denote the Multi-Layer Perceptrons (MLPs). \([ \cdot ] \) represents the concatenation operation. We then obtain the multi-order dense information embedded with keypoints, geometric distance as well as velocity by fusing \( H_{s_j,m}^{l,b} \) and \( H_{s_j,\cdot}^{l,v} \), as follows:
\[
  H_{s_j,m}^{l,v} = f_{s_j}(H_{s_j,m}^{l,b}, \varepsilon(H_{s_j,\cdot}^{l,v})) \in \mathbb{R}^{C_t \times T_l}
\]
(6)

where \( f_{s_j}(\cdot) \) denotes the MLPs. \( \varepsilon(\cdot) \) reshapes \( H_{s_j}^{l,v} \) from \( \mathbb{R}^{C_t \times C_s \times N_{s_j}} \) to \( \mathbb{R}^{C_t \times N_{s_j}} \), where the \( N_{s_j} \) dimension has been moved to the first position so that the geometric distance and velocity can be fused by the concatenation operation. Here, the information of each mouse can be represented as \( (H_{s_j,m}^{l,v})_{k_1} \) and \( (H_{s_j,\cdot}^{l,v})_{k_2} \).

Given the aggregated representation of two mice \( H_{s_j}^{l,v} \), our target is to learn an inter-skeleton interaction pattern. Therefore, the interaction between node \( v_m \) of mouse \( k_1 \) with representation \( (H_{s_j,m})_{k_1} \) and node \( v_n \) of mouse \( k_2 \) with representation \( (H_{s_j,n})_{k_2} \) can be expressed as:
\[
  E_{k_1 \rightarrow k_2}^{l}(m, n) = \sigma \left( \bar{\sigma} \left( (H_{s_j,m}^{l,v})_{k_1}; (H_{s_j,n}^{l,v})_{k_2} \right)^T \right) + \beta \cdot \Lambda_{k_1 \rightarrow k_2}(m, n)
\]
(7)

where \( \sigma \) is the activation function as ReLU, \( \bar{\sigma} \in \mathbb{R}^{1 \times 2C_t \times T_l} \) denotes a learnable weight vector. \( \Lambda_{k_1 \rightarrow k_2}(m, n) \) denotes the pre-defined physical connections describing the inter-skeleton interaction between mice, where \( \Lambda_{k_1 \rightarrow k_2} = 1 \) if node \( v_m \) of mouse \( k_1 \) and \( v_n \) of mouse \( k_2 \) is connected. Specifically, these connections link corresponding keypoints of different mice, such as the nose of mouse \( k_1 \) corresponding to the nose of mouse \( k_2 \). Similar to AGCN [26], our inter-skeleton interaction matrix is composed of both fixed and learnable matrices. \( \Lambda_{k_1 \rightarrow k_2} \) is fixed, and it is combined with a learnable matrix to generate the final interaction matrix \( E_{k_1 \rightarrow k_2}^{l} \). In particular, we introduce a trade-off parameter \( \beta \) to balance the potential effect of the pre-defined interaction pattern. Thus, the generated \( E_{k_1 \rightarrow k_2}^{l}(m, n) \) represents the correlation degree between the two nodes, and it is also dynamically updated to learn behaviour-specific inter-skeleton interaction. Besides, we normalise the results in Eq. (7) by the Softmax function to allow the correlation degree to be comparable, as follows:
\[
  \Lambda_{k_1 \rightarrow k_2}^{l}(m, n) = \frac{\exp \left( E_{k_1 \rightarrow k_2}^{l}(m, n) \right)}{\sum_{v=1}^{N_{s_j}} \exp \left( E_{k_1 \rightarrow k_2}^{l}(m, v) \right)}
\]
(8)

In this paper, we design a bidirectional interaction model, i.e., impact of mouse \( k_1 \) on \( k_2 \) and that of mouse \( k_2 \) on \( k_1 \), to fully explore potential inter-skeleton interaction. Thus, the interaction from \( k_2 \) to \( k_1 \), i.e., \( \Lambda_{k_2 \rightarrow k_1}^{l} \), can also be inferred using the same method shown in Eqs. (7) and (8). Afterwards, we generate the node-level representation integrated with the inter-skeleton interaction. Given the spatio-temporal representation of a mouse, after intra-skeleton interaction modelling, the behavioural representation of another mouse is updated as:
\[
  (X_{s_j}^{n,\text{inter}})_{k_1} = \Lambda_{k_1 \rightarrow k_2}^{l} (X_{s_j}^{n,\text{intra}})_{k_2} W_{k_2, \cdot \rightarrow k_1}^{l} + (X_{s_j}^{n,\text{intra}})_{k_1}
\]
\[
  (X_{s_j}^{n,\text{inter}})_{k_2} = \Lambda_{k_2 \rightarrow k_1}^{l} (X_{s_j}^{n,\text{intra}})_{k_1} W_{k_1, \cdot \rightarrow k_2}^{l} + (X_{s_j}^{n,\text{intra}})_{k_2}
\]
(9)

where \( W_{k_2, \cdot \rightarrow k_1}^{l} \) and \( W_{k_1, \cdot \rightarrow k_2}^{l} \in \mathbb{R}^{C_t \times C_s \times N_{s_j}} \) are trainable weight matrices. Then, we compose the representations of the mice to generate the node-level representation embedded with inter-skeleton interaction \( X_{s_j}^{\text{inter}} \) by concatenation on the batch dimension, as shown in Eq. (10):
\[
  X_{s_j}^{\text{inter}} = \Psi(X_{s_j}^{n,\text{inter}}) = [(X_{s_j}^{n,\text{inter}})_{k_1}; (X_{s_j}^{n,\text{inter}})_{k_2}] \in \mathbb{R}^{N_{s_j} \times C_t \times T_l}
\]
(10)

3) Cross-Skeleton Interaction Modelling: In this subsection, we aim to model the cross-skeleton interaction within the same mouse, and that between different mice at the same time. Based on \( X_{s_j}^{n,\text{inter}} \) shown in Eq. (10), we first learn skeleton-shared representation within the same mouse. Similar to the inter-skeleton interaction, the relation degree between the \( n \)-th keypoint of one mouse and the \( m \)-th body part of the same mouse needs to be derived. Thus, to integrate information from \( s_1 \) (i.e., dense skeleton) to \( s_2 \) (i.e., sparse skeleton), we rewrite Eq. (7) by combining Eq. (8) as follows:
\[
  A_{s_1 \rightarrow s_2}^{l}(m, n) = \text{Softmax} \left( \sigma \left( \bar{\beta} \left( H_{s_2,m}^{l,v}; H_{s_1,n}^{l,v} \right)^T \right) + \beta \cdot \Lambda_{s_1 \rightarrow s_2}(m, n) \right)
\]
(11)
an interaction-aware self-attention mechanism using a self-attention block, followed by a recurrent transition for hierarchical spatio-temporal representation learning. Regarding node $v_{mj} \in N_{S_i}^j$ in the $j$-th subgraph (the first subgraph represents the node-level representation generated by the CS-NLI module), we have:

$$H_{S_{ij+1}, m_{j+1}}^{cross,l} = LN\left(Q_{S_{ij+1}, m_{j+1}}^{l} + \gamma_{m_{j+1}}(X_{S_{ij+1}, m_{j}}^{cross,l})\right)$$

(13)

$$X_{S_{ij+1}, m_{j+1}}^{cross,l} = LN\left(H_{S_{ij+1}, m_{j+1}}^{cross,l} + \Gamma(H_{S_{ij+1}, m_{j+1}}^{cross,l})\right)$$

(14)

where $LN(\cdot)$ is to normalise the inputs across the entire feature dimensions. The transition function shown in Eq. (14) is defined as a TCN that models the temporal relations between nodes. $\gamma(\cdot)$ denotes the self-attention network to dynamically learn the graph-level representation, which can be formulated as:

$$\gamma_{m_{j+1}}(X_{S_{ij+1}, m_{j}}^{cross,l}) = \sum_{m_j \in N_{S_i}^j} \alpha_{m_{j+1}, m_j}^{l} v_{S_{ij}, m_{j}}^{l}$$

(15)

where $\alpha_{m_{j+1}, m_j}^{l}$ represents the strength of the correlations between nodes $v_{mj+1}$ and $v_{mj}$, based on the query and key vectors. $v_{S_{ij}, m_{j}}^{l}$ is the value vector of $m_j$, and the score $\alpha_{m_{j+1}, m_j}^{l}$ is used to weight each node’s key vector.

The query, key and value vectors in the Transformer architecture are used to establish a self-attention mechanism [51]. Different from most self-attention methods of applying linear transformations [60], [61] or convolution [38] to the node features, we propose an interaction-aware self-attention approach to construct the vectors, based on the structural intra- and inter-skeleton interactions. Particularly, we focus on the behaviour-related interaction saliency by condensing $N_{S_i}^j$ nodes into a sub-graph with $N_{S_i}^{j+1}$ nodes. Hence, the query vector is defined as:

$$Q_{S_{ij}, m_{j+1}}^{l} = \sum_{m_j \in N_{S_i}^j} W_{S_{ij}, m_j}^{l} (m_{j+1}, m_j) X_{S_{ij}, m_{j}}^{cross,l}$$

(16)

where $W_{S_{ij}, m_j}^{l}(m_{j+1}, m_j)$ are the elements of a trainable matrix $W_{S_{ij}}^{l} \in \mathbb{R}^{N_{S_i}^{j+1} \times N_{S_i}^{j}}$ at the $m_{j+1}$-th row and $m_j$-th column. The output $Q_{S_{ij}, m_{j+1}}^{l}$ denotes the feature vector of node $v_{mj+1}$. We next compute the key and value vectors for node $v_{mj}$, which are jointly encoded by different interaction patterns, i.e., intra- and inter-skeleton interactions. The key vector $K_{S_{ij}, m_j}^{l}$ can be computed as follows:

$$K_{S_{ij}, m_j}^{l} = Conv_{1 \times 1}\left(\Phi(X_{S_{ij}, m_j}^{cross,l}); \psi(X_{S_{ij}, m_j}^{cross,l})\right)$$

(17)

where $\Phi(X_{S_{ij}, m_j}^{cross,l})$ representing the intra-skeleton interaction on the spatial domain for node $v_{mj}$ can be formed using Eq. (1). $\psi(X_{S_{ij}, m_j}^{cross,l})$ denotes the corresponding inter-skeleton interaction that is calculated by combining Eqs. (9) and (10). The concatenation is performed on the channel dimension, and $Conv_{1 \times 1}(\cdot)$ denotes $1 \times 1$ convolution to reduce the channel dimension.

Similarly, we formulate the value vector $V_{S_{ij}, m_j}^{l}$ in Eq. (15) according to Eq. (17). Then, the attention weight $\alpha_{m_{j+1}, m_j}^{l}$ can

where $\alpha_{m_{j+1}, m_j}^{l}$ denotes the learnable weight vector. $A_{s_1 \rightarrow s_2}$ is the predefined connections across the overall skeletons (e.g., nose, left ear and right ear in $s_1$ correspond to head in $s_2$). Similarly, we model the cross-skeleton interaction of different mice. We exchange the orders of $k_1$ and $k_2$ in Eq. (10) to generate $X_{inter,l}^{\inter}$, and the order of the mouse in $H_{S_i,n}$ used in Eq. (11) to yield $A_{s_2 \rightarrow s_1}(m,n)$.

Finally, we have the corresponding node-level representation of $s_2$ (i.e., $X_{inter,l}^{\inter} \in \mathbb{R}^{N_{S_i} \times C_T \times T}$ ) by fusing the information from $s_1$, including four parts, i.e., the initial intra-skeleton interaction information, inter-skeleton interaction information, cross-skeleton interaction information of the same mouse and cross-skeleton interaction information of different mice. Hence, we can have:

$$X_{inter,l}^{\inter} = X_{intra,l}^{\inter} + X_{inter,l}^{\inter} + A_{s_1 \rightarrow s_2} X_{inter,l}^{\inter} W_{s_1 \rightarrow s_2}^{l} + A_{s_2 \rightarrow s_1} X_{inter,l}^{\inter} W_{s_2 \rightarrow s_1}^{l}$$

(12)

where $W_{s_1 \rightarrow s_2}^{l}, W_{s_2 \rightarrow s_1}^{l} \in \mathbb{R}^{C_T \times C_T}$ are trainable weight matrices. The fusion from $s_2$ to $s_1$ can also be made using the same way as mentioned above.

B. Interaction-Aware Transformer

In this section, we aim to generate graph-level representation of mouse social behaviour for classification from the CS-NLI module reported in Section III-A, and further update node-level representation used as the input to the next layer for capturing higher-level features. The architecture of our proposed IAT is shown in Fig. 3.

1) Encoder: As aforementioned in Section I, significant interaction information of mice may be lost if we adopt the pooling operation such as the global average pooling to produce graph-level representation. Hence, to learn a discriminative graph-level representation, we design a novel Interaction-Aware Transformer based on the self-attention mechanism [51], [60]. Given the node-level representation of skeleton graph $s_i$ on the $l$-th layer (i.e., $X_{s_i}^{cross,l} \in \mathbb{R}^{N_{S_i} \times C_T \times T}$), we sequentially generate $J$ subgraphs, i.e., $X_{s_i}^{cross,l} \in \mathbb{R}^{N_{S_i}^j \times C_T \times T}$, with $N_{S_i}^j, \forall j \in \{1, 2, \ldots, J\}$ nodes. Inspired by the universal transformer model [60], we design
be defined by applying the Softmax function to the scaled dot products \([51]\) between \(m_{j+1}\) and \(m_j\):

\[
\alpha_{m_{j+1},m_j} = \text{Softmax}_{m_j}(\frac{Q_{s_{j+1},m_{j+1}}(K_{s_{j+1},m_j})^T}{\sqrt{CTf}})
\]  

To this end, we can hierarchically generate multiple sub-graphs through Eqs. (13) and (14), in which the final one, i.e., \(X_{s_{ij},m_j}\) denotes the behaviour-related graph-level representation with \(N_f^2 = 1\).

To generate the graph-level representation from the skeleton graph, GAP \([26]\) or max pooling \([34]\) has been used to merge the information of all the keypoints or frames. Intuitively, different graph-level representations carry different semantic information describing social interactions. Thus, we also explore the relations between different graph-level representations by using our proposed interaction-aware self-attention defined in Eqs. (13) and (14) to enhance the graph-level representation. More details can be found in Supplementary A.

2) Decoder: In most prior works like \([26]\) and \([37]\), the node-level representation is directly passed between blocks in a GCN-TCN architecture. In contrast, we add a decoder at the end of the encoder to update node-level representation before passing it to the next layer. This update is supported by our interaction-aware self-attention mechanism presented in Section III-B.1. Supplementary A provides more explanations and our proposed IAT is summarised in Algorithm S1.

C. Self-Supervision for Cross-Skeleton Node Similarity Learning

As aforementioned, there potentially exists important similarity between the two skeletons (dense and sparse skeletons) because they describe the spatial structure of the mouse from different scales. In other words, the similarity is naturally embedded into the node-level representations of the two skeletons, which may play a crucial role in social behaviour representation learning. Inspired by the attribute based self-supervision \([62]\) on graphs, we design an auxiliary self-supervised learning task to better preserve these attributes between the cross-skeleton pairwise nodes.

For the \(i\)-th sliding window, given the initial spatial-temporal feature of the node \(v_m \) in skeleton \(s_1\) (i.e., \(X^{(i)}_{s_1,m}\)) and that of \(v_n \) in skeleton \(s_2\) (i.e., \(X^{(i)}_{s_2,n}\)), we first compute the node feature similarity between them according to the cosine similarity:

\[
S^{(i)}_{mn} = \frac{X^{(i)}_{s_1,m} \cdot X^{(i)}_{s_2,n}}{\|X^{(i)}_{s_1,m}\|_2 \cdot \|X^{(i)}_{s_2,n}\|_2 + \epsilon}
\]  

where \(\epsilon\) is a small constant avoiding divide-by-zero. Then, the self-supervised learning task can be formulated as a regression problem and the corresponding loss can be defined as:

\[
L_{\text{self}} = \frac{1}{B} \sum_{i=1}^{B} \sum_{(v_m, v_n) \in P} \left| \frac{1}{|P|} \sum_{(v_m, v_n) \in P} f_s (X^{(i)}_{s_1,m} - X^{(i)}_{s_2,n} - S^{(i)}_{mn})^2 \right|
\]  

where \(P\) denotes the set of node pairs consisting of nodes from different skeletons, and \(|P|\) is the number of the nodes. \(f_s (\cdot)\) is a fully connected layer with the output dimension of 1, \(X^{(i)}_{s_1,m}\) is the node-level representation of node \(v_m\) in the \(i\)-th layer of our network.

Finally, we can obtain the overall behaviour recognition loss by combining the self-supervised loss defined in Eq. (20) and cross-entropy loss \(L_{\text{class}}\) (see Supplementary A), which is defined as \(L = L_{\text{class}} + \lambda L_{\text{self}}\) where \(\lambda\) is a hyper-parameter adjusting the contribution of the self-supervised loss. By jointly optimising the self-supervised objective function and the traditional classification loss function, our proposed model can lead to more discriminative representation.

IV. EXPERIMENTS

A. Datasets and Experimental Setup

1) CRIM13-Skeleton Dataset: In this paper, we validate the proposed framework using videos of two mice. The Caltech Resident-Intruder Mouse dataset \([63]\) (CRIM13) consists of 237×2 videos of two mice (see Table S1 for the description of social behaviour), which was used to study neurophysiological mechanisms involved in aggression and courtship in mice. It was recorded with synchronized top- and side-view cameras with the resolution of 640×480 pixels and the frame rate of 25Hz. Each video lasts about 10 minutes and was annotated frame by frame. 13 social behaviours was defined in this dataset, including 12 specific behaviours (shown in Table S1) and one otherwise unspecified behaviour (i.e., ‘other’). In this paper, we use the public CRIM13 dataset with pose annotations (CRIM13-Skeleton) in \([17]\). It contains 64 top-view videos where each frame has 16 keypoints (each mouse has 8 keypoints), as shown in Fig. 4(a) and S1(a). For each keypoint, it is represented by a tuple of \((X, Y, C)\), in which \((X, Y)\) is the 2D coordinates and \(C\) denotes the confidence score. In our experiments, we only use 7 keypoints (0-6 in Fig. 4(a)) of each mouse due to low confidence on the tail end. Different from the original dataset \([63]\), CRIM13-Skeleton dataset is categorised into 12 behaviours where the behaviour ‘human’ is deleted. In our experiments, we randomly split the dataset into a training set of 51 videos and a test set of 13 videos.

2) PDMB-Skeleton Dataset: Our PDMB dataset was collected in collaboration with the biologists of Queen’s University Belfast of United Kingdom, for a study on motion recordings of mice with Parkinson’s disease (PD) \([6]\). The neurotoxin 1-methyl-4-phenyl-1,2,3,6-tetrahydropyridine (MPTP) is used as a model of PD, which has become an invaluable aid to produce experimental parkinsonism since its discovery in 1983 \([64]\). By utilising MPTP-induced models, we aim to establish a direct link between changes in mouse social behaviours and the neurodegenerative processes associated with PD. The MPTP model allows us to mimic key aspects of PD pathology in mice, facilitating the study of behavioural changes that parallel the symptoms observed in human patients. Quantifying mouse social behaviours \([6]\) contributes to understanding how MPTP-induced neurodegeneration impacts specific behaviours.
To generate the locations and of each mouse keypoint, we also used the standard pose around 220,000 skeleton sequences. To obtain the location PDMB dataset [6], with added keypoint annotations, providing 640*480 resolution. All videos contain 9 behaviours of two (one top-view and two side-view) with frame rate of 30 fps and using three synchronised Sony Action cameras (HDR-AS15). This dataset consists of 12*3 annotated videos (6 videos for MPTP treated mice and 6 videos for control mice) recorded by University Belfast Animal Welfare and Ethical Review Body. Procedures) Act, 1986 (UK) and approved by the Queen’s with the Guidance on the Operation of the Animals (Scientific datasets. The CRIM13-Skeleton dataset contains 8 keypoints for each mouse (i.e., 0-left ear, 1-right ear, 2-snout, 3-centroid, 4-left lateral, 5-right lateral, 6-tail base and 7-tail end). The PDMB-Skeleton dataset contains 7 keypoints (i.e., 0-left ear, 1-right ear, 2-snout, 3-centroid, 4-left hip, 5-right hip and 6-tail base).

All experimental procedures were performed in accordance with the Guidance on the Operation of the Animals (Scientific Procedures) Act, 1986 (UK) and approved by the Queen’s University Belfast Animal Welfare and Ethical Review Body. This dataset consists of 12*3 annotated videos (6 videos for MPTP treated mice and 6 videos for control mice) recorded by using three synchronised Sony Action cameras (HDR-AS15) (one top-view and two side-view) with frame rate of 30 fps and 640*480 resolution. All videos contain 9 behaviours of two freely behaving mice and each video lasts around 10 minutes.

PDMB-Skeleton dataset is an extension of the original PDMB dataset [6], with added keypoint annotations, providing around 220,000 skeleton sequences. To obtain the location of each mouse keypoint, we also used the standard pose estimator, i.e., DeepLabCut [13], to generate the locations and confidence scores of 7 defined keypoints on every frame of 12 top-view videos. We adopted the same training and testing dataset split scheme as in the PDMB dataset, evenly dividing the entire dataset into training and testing sets, resulting in 110,000 training samples. Fig. 4(b) and S1(b) show the keypoint locations in the PDMB-Skeleton dataset. More details about data annotation and dataset construction can be found in Supplementary B.

3) Implementation Details: All the experiments are performed with PyTorch 1.4.0 on a server with an Intel Xeon CPU @ 2.40GHz and two 16GB Nvidia Tesla P100 GPUs. The parameters are optimised by the Adam algorithm. For the two datasets, we use the initial learning rate of 1e-4 and all the keypoint locations are normalised before training. No data augmentation is used for a fair performance comparison. β and batch size Bp are set to 0.5 and 128, respectively. As far as it concerns the model architecture, we use 3 cascaded CS-NLI+IAT modules, whose feature dimensions are 64, 128 and 256, respectively. The source code will be available at: https://github.com/FeixiangZhou/CS-IGANet.

B. Ablation Study

In this section, we launch comprehensive experiments to investigate the effectiveness of each model component, i.e., Cross-Skeleton Node-level Interaction, Interaction-Aware Transformer, Self-supervision for cross-skeleton node similarity learning in our proposed CS-IGANet. We conduct ablation experiments on the CRIM13-Skeleton and PDMB-Skeleton datasets and use a 3-layer single-stream (keypoint) GCN-TCN network [26] as our baseline model, where feature dimensions are 64, 128 and 256, respectively. Normally, classification accuracy is defined as the percentage of the samples that are correctly classified against the number of the overall samples. While it is a valid measure, this metric cannot disclose the characteristics of the datasets that have a severe imbalanced classification problem. Following [6], we here employ the averaging recognition rate per behaviour to better measure the system performance.

1) Effects of Cross-Skeleton Node-Level Interaction: Here, we investigate the influences of the proposed Cross-Skeleton Node-level Interaction module. We study the effects of dense geometric distance in Eq. (2), dense velocity in Eq. (3) as well as inter-skeleton interaction (denoted as inter-skeleton) block, presented in Table I. We observe that although the baseline method only using the dense skeleton possesses the fewest parameters and FLOPs, it exhibits the poorest performance, especially for behaviours such as ‘approach’, ‘chase’ and ‘walk away’. The low accuracy is due to the fact it only focuses on intra-skeleton interaction without considering the social interaction between mice. In addition, directly modelling cross-skeleton interaction of the same mouse based

| Methods             | Approach | Attack | Copulation | Chase | Circle | Drink | Eat | Clean | Sniff | Up | Walk away | Other | Average | Params | FLOPs |
|---------------------|----------|--------|------------|-------|--------|-------|-----|-------|-------|----|-----------|-------|---------|--------|-------|
| Baseline            |          | 46.70  | 79.41      | 69.32 | 22.09  | 57.92 | 77.02| 53.02 | 77.17 | 69.68 | 71.87     | 47.59 | 73.79   | 62.13  | 1.05M  |
| CS-NLI (w/o inter)  |          |        |            |       |        |       |     |       |       |     |           |       |         |        | 0.23G |
| CS-NLI (w inter)    |          |        |            |       |        |       |     |       |       |     |           |       |         |        | 0.33G |
| CS-NLI (w/o inter)  |          |        |            |       |        |       |     |       |       |     |           |       |         |        | 0.33G |
| CS-NLI (w inter)    |          |        |            |       |        |       |     |       |       |     |           |       |         |        | 0.33G |
| CS-NLI (w/o inter)  |          | 62.04  | 78.01      | 75.55 | 36.89  | 59.75 | 77.54| 65.87 | 80.55 | 66.51 | 80.71     | 45.40 | 56.67   | 65.46  | 2.89M  |
| CS-NLI (w inter)    |          | 69.24  | 81.81      | 78.91 | 44.73  | 66.23 | 79.12| 57.70 | 86.87 | 65.72 | 85.12     | 59.10 | 45.02   | 68.30  | 2.90M  |

Fig. 4. Keypoint labels of (a) CRIM13-Skeleton and (b) PDMB-Skeleton datasets. The CRIM13-Skeleton dataset contains 8 keypoints for each mouse (i.e., 0-left ear, 1-right ear, 2-snout, 3-centroid, 4-left lateral, 5-right lateral, 6-tail base and 7-tail end). The PDMB-Skeleton dataset contains 7 keypoints for each mouse (i.e., 0-left ear, 1-right ear, 2-snout, 3-centroid, 4-left hip, 5-right hip and 6-tail base).
on dense geometric distance or velocity information, without inter-skeleton interaction, can help slightly improve the average accuracy. The performance can be further improved to 65.46% by fusing the two types of information while the number of parameters only increases by 0.19M and the time complexity increases by 0.01G, confirming that the dense geometric distance and velocity information are beneficial to cross-skeleton interaction encoding. With respect to the inter-skeleton interaction modelling, we witness that the models with inter-skeleton interaction based on different types of information all achieve better performance than the models without such interaction. It is noticeable that the addition of inter-skeleton interaction almost negligibly increases the model’s complexity in terms of parameter count and FLOPs but significantly improves the average accuracy. In particular, for social behaviours such as ‘approach’, ‘chase’ and ‘walk away’, we can obtain 7.2%, 7.84% and 13.7% improvements, respectively by modelling inter-skeleton interaction (see the last two rows). Such significant improvements are due to the strong interaction modelling ability of our CS-NLI module. By jointly modelling intra-, inter- and cross-skeleton interactions based on multi-order dense information, our method achieves the best average accuracy of 68.3%, which is a 6.17% improvement against the baseline model.

We also visualise the relevant interaction patterns identified by our CS-NLI module, including the intra-, inter-, and cross-skeleton interactions. Fig. 5 shows the corresponding topologies of a behaviour sample ‘approach’ in the top branch of Fig. 2 (i.e., skeleton $s_1$ to skeleton $s_2$). The values close to 0 indicate weak relationships between keypoints and vice versa. From Fig. 5(a), we observe that the two topologies representing the intra-skeleton interactions of mice are very similar, where the correlations between some keypoint pairs are relatively strong, e.g., the correlation between the centroid and the tail base, and the correlation between the left lateral and the tail base. However, these independent relationships derived from each mouse are insufficient to be exploited to encode complex social interactions. The inter-skeleton interaction modelling is able to learn new interactions between mice that the independent skeleton graph cannot provide, as shown in Fig. 5(b). For instance, our CS-NLI module pays much attention to the interactions between the tail bases of mice, and between the left ear and the snout when considering the effect of mice $k_2$ on $k_1$. Moreover, in Fig. 5(c), our CS-NLI module further models the cross-skeleton interactions, where the tail bases of the same mice or different mice from different scales are highly related. To examine the difference of the topologies during training, we also show topologies learned...
Fig. 6. Learned topologies (i.e., inter-skeleton interaction $A_{k_1 \rightarrow k_2}$) by our CS-NLI module for three sample social behaviours (e.g., approach, chase and walk away) (a) and the motion trajectories for three behaviours in the CRIM13-Skeleton dataset (b). For each behaviour shown in (a), each topology refers to the social interaction between mice in the current frame. The corresponding motion trajectories of keypoints are shown in (b). In (b), blue and red points indicate the keypoints of the resident mouse and the intruder, respectively. Blue and red arrows represent the direction of motion. More examples of motion trajectories can be found in Fig. S4.

| Methods                  | Approach | Attack | Copulation | Chase | Circle | Drink | Eat | Clean | Sniff | Up | Walk away | Other | Average |
|--------------------------|----------|--------|------------|-------|--------|-------|-----|-------|-------|----|-----------|-------|---------|
| CS-NLI (w/o self, $\lambda = 0$) | 69.24    | 81.81  | 78.91      | 44.73 | 66.23  | 79.12 | 57.70 | 86.87 | 65.72 | 85.12 | 59.10 | 45.02 | 68.30  |
| CS-NLI (w self, $\lambda = 0.1$) | 59.65    | 79.62  | 74.83      | 57.16 | 64.91  | 80.87 | 61.49 | 81.26 | 63.45 | 80.07 | 75.32 | 49.79 | 69.65  |
| CS-NLI (w self, $\lambda = 0.5$) | 62.40    | 77.69  | 74.55      | 61.62 | 79.43  | 83.33 | 65.60 | 85.17 | 63.40 | 87.07 | 63.94 | 42.76 | 70.58  |
| CS-NLI (w self, $\lambda = 1$) | 67.50    | 81.43  | 65.73      | 48.66 | 77.04  | 82.81 | 64.09 | 84.96 | 66.72 | 89.04 | 59.44 | 45.25 | 69.39  |
| CS-NLI (w self, $\lambda = 1.5$) | 59.68    | 76.45  | 80.03      | 47.43 | 77.61  | 83.33 | 55.15 | 80.52 | 61.75 | 86.90 | 66.24 | 52.51 | 68.97  |

by our CS-NLI module that is not fully trained, as shown in Fig. S2. We observe that the model generates a relatively dense fully connected graph at the beginning of training, especially for the inter- and cross-interactions, where interactions may not be related to behaviours. On the contrary, our final model tends to better focus on behaviour related interactions. To show how our CS-NLI module works, we display the learned topologies representing the inter-skeleton interaction, i.e., $A_k^{\lambda=1} \rightarrow k_2$, in Fig. 6. From this figure, we observe that the CS-NLI module gives much attention to the interactions between mice, e.g., left lateral and left ear for ‘approach’, tail base and left ear for ‘chase’, and tail bases for ‘walk away’.

2) Effects of Interaction-Aware Transformer: In order to validate the effectiveness of our proposed IAT module, we design six structures using the baseline model. IAT (I & w/o DC) refers to the case where we only keep the encoder of the IAT and use the graph-level representation aggregated by $IAT(\cdot) (I)$ to perform behavioural classification, while IAT (I & w DC) refers to a structure with the encoder and the decoder. IAT (I+M & w DC) and IAT (I+A & w DC) mean that we enhance the graph-level representation by combining graph-level representation from the encoder and that generated by spatial max pooling and average pooling, respectively, where we simply use the sum operation to fuse different representations. The last IAT ($G(I,A,M) & w DC$)
refers to the structure that models the interactions between multi-level graph representations. From Table II, we observe that the IAT without any decoder achieves higher accuracy than the baseline model for all 8 behaviours, indicating that the intra-skeleton interaction of each mouse and inter-skeleton interactions between mice are important to graph-level representation learning. The performance is further improved by constructing an encoder-decoder framework, resulting in the highest accuracy of 83.37% and 61.49% for ‘attack’ and ‘walk away’, respectively. This is because the node-level representation can be adaptively updated by our proposed dual-path decoder, before it is fed into the next layer, which helps to identify higher-level features. In addition, three models combining different graph-level representations through a straightforward summation operation all outperform IAT (I & w DC), without incurring an increase in the number of parameters and FLOPs, suggesting that different graph-level representations carry complementary spatio-temporal information that helps improve the identification. Instead of fusing different graph-level representations by the sum operation, we explore the structural relations by our proposed interaction-aware self-attention unit, leading to a 1.15% improvement against IAT (I+A+M & w DC). Regarding our network involving two skeletons, we fuse different graph-level representations from two skeleton branches by the interaction-aware self-attention module. In contrast to the baseline, our proposed IAT demonstrates a noteworthy 6.87% improvement in average accuracy, although the computational complexity of this approach is more than twice that of the GAP-based method.

3) Effects of Self-Supervision for Cross-Skeleton Node Similarity Learning: In this subsection, we study the effect of the proposed auxiliary self-supervised loss function, controlled by the hyper-parameter $\lambda$. To analyse the impact of this parameter, we train several models (i.e., CS-NLI module) with different values of $\lambda$. As shown in Table III, all models with self-supervision ($\lambda = 0.1, 0.5, 1, 1.5$) lead to an improvement over the baseline without self-supervision. Increasing $\lambda$ from 0 to 0.5 significantly improves the accuracy. This is mainly because the important attributes (i.e., similarity) underlying cross-skeleton pairwise nodes are explicitly exploited in the node representation learning. When $\lambda = 0.5$, we achieve significant improvements on the accuracy of ‘chase’, ‘circle’, ‘drink’ and ‘eat’, and the highest average accuracy of 70.58%. However, there is significant degradation in the performance when we increase $\lambda$ to 1.5. This drop is due to the fact that the self-supervised loss severely penalises the inherent attributes of node pairs from different skeletons. Hence, our default value is $\lambda = 0.5$.

We also investigate the contribution of the proposed CS-NLI, IAT and Self-supervision modules to the whole network on both datasets. In addition to adding each proposed module to the baseline model separately, we further employ the proposed modules applied to the baseline model simultaneously. The results are reported in Table IV. On the two datasets, our method achieves the highest average accuracy, 73.75% and 62.33% respectively, with the three proposed modules applied simultaneously, which are of 11.62% and 9.66% increments compared to the baseline model. In terms of the model complexity (FLOPs and number of parameters), applying either CS-NLI or IAT to the baseline results in an approximately twofold increase. However, the incorporation of Self-supervision does not lead to an additional computational cost. Though the model complexity of our method with the three components is higher than the baseline, it receives a much higher accuracy.

### C. Comparison to State-of-the-Art Approaches

In this section, we compare the proposed model against several state-of-the-art graph-based methods and transformer-based methods on two datasets: CRIM13-Skeleton and PDMB-Skeleton. In our experiments, we follow the default settings of all existing methods to ensure a fair comparison. We use the same configuration parameters, modality information and hyperparameters as specified by the authors of those methods. Specifically, the selected methods include single-stream [30], [37] and multi-stream [21], [23], [25], [26], [31], [34], [38], [39], [56], [65] frameworks. Most methods (e.g., 2s-AGCN (2-stream, denoted as 2s), MS-G3D(2s), MV-IGNet(2s), CTR-GCN(4s), InfoGCN(4s), STEP...
CATFormer(4s)) employ a multi-stream fusion framework, where different modalities (e.g., joint/keypoint, bone, joint motion/velocity, and bone motion) are used as inputs to separately train the same network to obtain better results. Although these networks have proven effective, multiple separate networks will increase the number of parameters. Some methods (e.g., SGN(2s), EfficientGCV(3s), ST-TR(2s), PA-ResGCN(3s)) fuse several types of information in the early stage (in input) to reduce the computational cost caused by the multi-stream structure. We also compare three methods of interactive action recognition, i.e., 2s-DRAGCN [27], 2P-GCN [28] and ISTA-Net [30]. Similar to ST-GCN [37] and ISTA-Net [30], we only employ the keypoint information as input without explicitly fusing other information. Notably, our proposed method can be considered as the multi-stream structure based on early fusion but we extract dense geometric distance and velocity information to better describe nuanced social interactions between mice, which is different from all the above approaches. The results on two datasets are presented in Tables V and VI, respectively.

As shown in Table V, our proposed modules, along with their combined architecture, demonstrate superior performance compared to other state-of-the-art models in terms of average accuracy while having relatively few parameters and FLOPs. This is because our method jointly models the intra-, inter- and cross-skeleton interactions, and dynamically learns graph-level representation of mouse social behaviours, which is very effective in representation learning of mouse social behaviour. Although ST-GCN [37] achieves the highest classification accuracy on ‘other’, its average accuracy of all the behaviours is the lowest among these methods. This is because it only uses a fixed skeleton topology to model the relations between keypoints of each mouse, limiting its ability to encode/decode intra-skeleton interaction for some specific behaviours, such as ‘eat’ and ‘up’. Compared with CTR-GCN [39] using mul-stream fusion, our CS-IGANet exhibits.

### Table V

| Methods          | Approach | Chase | Circle | Eat | Clean | Sniff | Up | Walk away | Other | Average | Params | FLOPs |
|------------------|----------|-------|--------|-----|-------|-------|----|-----------|-------|---------|--------|-------|
| ST-GCN [37]      | 34.34    | 75.68 | 68.97  | 35.56 | 34.65 | 73.33 | 45.87 | 73.53     | 64.31 | 69.75   | 26.27  | 75.80 |
| 2s-AGCN [26]     | 51.03    | 83.40 | 75.84  | 36.46 | 53.40 | 75.61 | 63.65 | 76.71     | 75.80 | 76.25   | 40.04  | 51.15 |
| SGN [34]         | 49.57    | 80.43 | 74.90  | 34.90 | 66.42 | 74.34 | 63.26 | 73.83     | 66.19 | 67.59   | 41.78  | 75.41 |
| PA-ResGCN [21]   | 57.04    | 74.56 | 77.81  | 22.98 | 51.64 | 80.88 | 55.90 | 82.99     | 75.42 | 85.15   | 44.04  | 66.91 |
| MS-3G3 [31]      | 57.38    | 80.08 | 69.79  | 56.61 | 75.28 | 65.79 | 72.74 | 66.00     | 55.96 | 82.00   | 52.03  | 49.29 |
| MV-IGNet [25]    | 55.33    | 73.77 | 77.81  | 48.80 | 60.31 | 81.75 | 51.59 | 80.81     | 78.43 | 83.40   | 57.83  | 64.67 |
| ST-TR [38]       | 45.92    | 74.86 | 74.93  | 44.93 | 78.43 | 79.12 | 54.49 | 84.84     | 72.85 | 88.19   | 60.69  | 23.04 |
| EfficientGCN [65]| 52.99    | 78.47 | 75.12  | 33.12 | 44.97 | 81.23 | 55.36 | 80.20     | 77.96 | 83.23   | 55.29  | 66.39 |
| CTR-GCN [39]     | 52.33    | 79.94 | 75.89  | 41.59 | 53.08 | 66.67 | 55.50 | 68.12     | 73.70 | 87.71   | 48.83  | 60.58 |
| InfoGCN [23]     | 51.98    | 83.45 | 72.71  | 50.64 | 61.74 | 76.32 | 53.07 | 70.82     | 57.76 | 74.59   | 49.49  | 65.02 |
| STEP CATFormer [56]| 50.49  | 75.52 | 70.00  | 51.09 | 61.03 | 71.43 | 60.45 | 64.87     | 67.78 | 75.44   | 43.44  | 55.26 |
| 2s-DRAGCN [27]   | 53.03    | 75.52 | 75.90  | 54.41 | 55.72 | 74.21 | 51.47 | 75.44     | 61.76 | 70.02   | 53.20  | 60.56 |
| 2P-GCN [28]      | 57.35    | 78.99 | 70.04  | 53.79 | 65.47 | 72.11 | 56.86 | 69.49     | 65.05 | 77.53   | 54.72  | 58.63 |
| ISTA-Net [30]    | 58.34    | 78.57 | 71.08  | 57.59 | 64.78 | 70.22 | 56.50 | 72.56     | 68.54 | 75.51   | 57.65  | 56.75 |
| Ours(CS-NLI-self)| 62.40    | 77.69 | 74.55  | 61.62 | 79.43 | 83.33 | 65.60 | 85.17     | 63.40 | 87.07   | 63.94  | 42.76 |
| Ours(IAT)        | 63.20    | 75.90 | 73.83  | 55.0 | 75.91 | 76.95 | 67.29 | 83.95     | 67.21 | 84.46   | 60.60  | 41.03 |
| Ours(CS-IGANet)  | 67.30    | 84.30 | 75.30  | 69.32 | 82.08 | 81.75 | 64.44 | 88.20     | 73.70 | 86.79   | 66.79  | 43.85 |

**Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.**
a substantial performance advantage, where the average accuracy holds 10.92% improvement and the computational complexity, measured in FLOPs, is reduced from 1.22G to 0.61G. This suggests that dynamic refinement of channel-wise topology is not powerful enough to maintain the quality of mouse social behaviour representation. We also notice that, among the 12 existing methods, MV-IGNet [25] achieves the best performance with relatively fewer parameters and lower computational costs, but there is still a large gap (i.e., 5.89%) between MV-IGNet and our CS-IGANet. Compared to Transformer-based methods such as ST-TR [38] and STEP CATFormer [56], our method consistently exhibits superior performance and the parameter numbers and computational costs of them are several times that of our method. Another Transformer-based method, i.e., ISTA-Net [30], models interactive relations of diverse interacting subjects. Although it improves the accuracy of certain behaviours such as ‘chase’, the overall performance still lags behind ours by approximately 8% and the model complexity is higher than that of our method. In addition, for social interactions such as approach, chase and walk away, our proposed CS-IGANet improves the accuracy with large margins of 8.96%, 11.73% and 6.1%, respectively, compared with their close competitors. We also show the confusion matrix of our CS-IGANet on the CRIM13-Skeleton dataset, as shown in Fig. S3(a).

Notably, the accuracy for behaviour ‘other’ is significantly lower compared to almost all other methods. The ‘other’ category refers to instances where no behaviour of interest is occurring and it normally constitutes over 50% of the entire dataset [63]. However, in mouse behaviour research, accurately identifying other meaningful behaviours or interactions is more crucial [6], [63]. Our method improves the accuracy of most meaningful behaviours by effectively capturing the spatio-temporal dynamics of social interactions. On the other hand, methods, such as ST-GCN, lack the ability to model meaningful social interactions. Consequently, these methods tend to classify various behaviours as the ‘other’ category, leading to higher accuracy in this category but poor performance in meaningful behaviour classes. Despite the lower accuracy in the ‘other’ category, our proposed CS-IGANet significantly improves the average accuracy across all behaviour classes. We believe this trade-off highlights the effectiveness of our approach in achieving a more balanced and accurate overall classification, particularly in the more accurate recognition of meaningful behaviours.

Additionally, we present in Fig. 7 the t-SNE visualisation of the representations learned by our model and other 3 state-of-the-art methods (i.e., 2s-AGCN, MS-G3D and MV-IGNet). For our model, the representation is the concatenation of graph-level outputs of different layers, i.e., $\mathbf{X}^{\text{cross}}$. Our proposed CS-IGANet leads to better separation of the 11 behaviour classes. In particular, for some similar behaviours such as ‘approach’ and ‘walk away’, our model can better distinguish them.

As for the PDMB-Skeleton dataset, our approach also achieves the state-of-the-art performance with average accuracy of 62.33%, which is a 6% improvement compared with the closest competitor, i.e., MV-IGNet. In addition, our CS-IGANet significantly outperforms the other state-of-the-art methods on behaviours ‘approach’, ‘chase’, ‘circle’ and ‘walk away’, and achieves comparable performance on ‘clean’ and ‘sniff’, compared to [25] and [31]. The confusion matrix of our CS-IGANet on the PDMB-Skeleton dataset is shown in Fig. S3(b). To further evaluate the generalisation capability of our proposed method, we also conduct experiments on two human datasets (NTU-Interaction [66] and NTU120-Interaction [67]), as shown in Tab. S3. Our method continues to exhibit comparable or competitive performance.

V. Conclusion

In this work, we have presented a novel architecture called Cross-Skeleton Interaction Graph Aggregation Network (CS-IGANet) for representation Learning of mouse social behaviour. Cross-Skeleton Node-level Interaction module (CS-NLI) strengthens the node-level representation of each mouse by modelling intra-, inter- and cross-skeleton interactions in a unified way. We also designed a novel Interaction-Aware Transformer (IAT) to hierarchically aggregate node-level representation into graph-level representation of social behaviour, and adaptively update the node-level representation, which is guided by our interaction-aware self-attention unit. An auxiliary self-supervised learning task was also proposed to focus on the similarity between cross-skeleton pairwise nodes, enhancing the representation ability of our model. Experimental results on CRIM13-Skeleton and PDMB-Skeleton datasets demonstrated that the proposed approach outperformed most of the baseline methods. Our proposed solution is currently working on two mice cases but is extendable to three or more mice. The only difference is scaling up the complexity of computation. In addition, the proposed method for modelling social interactions of mice can be potentially extended to collaborative human behaviour prediction, especially for some scenarios involving nuanced behaviour patterns. For example, in collaborative work environments, understanding subtle cues and interactions among individuals is crucial for predicting actions and ensuring effective collaboration. Our future work in this direction could involve refining our approach to capture human-specific social dynamics and evaluating its performance in collaborative scenarios. We also plan to improve the efficiency and scalability of our approach to better meet the requirement for investigating more complex social interactions of more than two mice.

REFERENCES

[1] Y. K. Urbach et al., “Automated phenotyping and advanced data mining exemplified in rats transgenic for Huntington’s disease,” J. Neurosci. Methods, vol. 234, pp. 38–53, Aug. 2014.

[2] L. Lewejohann, A. M. Hoppmann, P. Kegel, M. Kritzler, A. Krüger, and N. Sachser, “Behavioral phenotyping of a murine model of Alzheimer’s disease in a seminaturistic environment using RFID tracking,” Behav. Res. Methods, vol. 41, no. 3, pp. 850–856, Aug. 2009.

[3] C. A. Wilson and J. I. Koenig, “Social interaction and social withdrawal in rodents as readouts for investigating the negative symptoms of schizophrenia,” Eur. Neuropsychopharmacol., vol. 24, no. 5, pp. 759–773, May 2014.

[4] S. R. Blume, D. K. Cass, and K. Y. Tseng, “Stepping test in mice: A reliable approach in determining forelimb akinesia in MPTP-induced parkinsonism,” Exp. Neurol., vol. 219, no. 1, pp. 208–211, Sep. 2009.
[5] Z. Jiang et al., “Context-aware mouse behavior recognition using hidden Markov models,” IEEE Trans. Image Process., vol. 28, no. 3, pp. 1133–1148, Mar. 2019.

[6] Z. Jiang et al., “Multi-view mouse social behaviour recognition with deep graphic model,” IEEE Trans. Image Process., vol. 30, pp. 5490–5504, 2021.

[7] A. Arac, P. Zhao, B. H. Dobkin, S. T. Carmichael, and P. Golumbsh, “DeepBehavior: A deep learning toolbox for automated analysis of animal and human behavior imaging data,” Frontiers Syst. Neurosci., vol. 13, p. 20, May 2019.

[8] N. G. Nguyen et al., “Applying deep learning models to mouse behavior recognition,” J. Biomed. Sci. Eng., vol. 12, no. 2, pp. 183–196, 2019.

[9] M. Marks et al., “Deep-learning-based identification, tracking, pose estimation and behaviour classification of interacting primates and mice in complex environments,” Nature Mach. Intell., vol. 4, no. 4, pp. 282–286, Apr. 2022.

[10] M. P. J. Camilleri, R. S. Bains, and C. K. I. Williams, “Of mice and mates: Automated classification and modelling of mouse behaviour in groups using a single model across cages,” 2023, arXiv:2306.03066.

[11] T. C. Moulin, L. E. Covill, P. M. Itskov, M. J. Williams, and M. P. J. Camilleri, R. S. Bains, and C. K. I. Williams, “Of mice and mates: Automated classification and modelling of mouse behaviour in groups using a single model across cages,” 2023, arXiv:2306.03066.

[12] M. Decourt, H. Jiménez-Urbieta, M. Benoit-Marand, and P.-O. Fernagut, “Neuropsychiatric and cognitive deficits in Parkinson’s disease and their modeling in rodents,” Biomedicines, vol. 9, no. 6, p. 684, Jun. 2021.

[13] A. Mathis et al., “DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning,” Nature Neurosci., vol. 21, no. 9, pp. 1281–1289, Sep. 2018.

[14] J. M. Graving et al., “DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning,” eLife, vol. 8, p. 1–42, Oct. 2019.

[15] F. Zhou et al., “Structured context enhancement network for mouse pose estimation,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 5, pp. 2807–2813, May 2022.

[16] Z. Cao, G. Hidalgo, T. Simo, S.-E. Wei, and Y. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 1, pp. 172–186, Jan. 2021.

[17] N. L. Goodwin et al., “Simple behavioral analysis (SimBA) as a platform for explainable machine learning in behavioral neuroscience,” Nature Neurosci., pp. 1–14, Feb. 2022.

[18] X. Liu et al., “OptiFlex: Video-based animal pose estimation using deep learning enhanced by optical flow,” BioRxiv, Apr. 2020, Art. no. 2020–04.

[19] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” 2016, arXiv:1609.02907.

[20] P. Gupta et al., “Quo vadis, skeleton action recognition?” Int. J. Comput. Vis., vol. 129, no. 7, pp. 2097–2112, Jul. 2021.

[21] Y.-F. Song, Z. Zhang, C. Shan, and E. Wang, “Stronger, faster and more explainable: A graph convolutional baseline for skeleton-based action recognition,” in Proc. 28th ACM Int. Conf. Multimedia, Oct. 2020, pp. 1625–1633.

[22] H. Xia and X. Gao, “Multi-scale mixed dense graph convolution network for skeleton-based action recognition,” IEEE Access, vol. 9, pp. 36475–36484, 2021.

[23] H.-G. Chi, M. H. Ha, S. Chi, S. W. Lee, Q. Huang, and K. Ramani, “InfoGCN: Representation learning for human skeleton-based action recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 20154–20164.

[24] L. Hedgesa, N. Heidari, and A. Iosifidis, “Temporal attention-augmented graph convolution for skeleton-based action recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 143–152.

[25] Y. Qi, J. Hu, L. Zhuang, and X. Pei, “Semantic-guided multi-scale human skeleton action recognition,” Appl. Intell., vol. 53, no. 9, pp. 9763–9778, May 2023.

[26] J. Lauer et al., “Multi-animal pose estimation, identification and tracking with DeepLabCut,” Nature Methods, vol. 19, no. 4, pp. 496–504, Apr. 2022.

[27] P. Zhang, C. Lan, W. Zeng, J. Xing, J. Xue, and N. Zheng, “Semantic-guided neural networks for efficient skeleton-based human action recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 1112–1121.

[28] L. Giancardo et al., “Automatic visual tracking and social behaviour analysis with multiple mice,” PLoS ONE, vol. 8, no. 9, Sep. 2013, Art. no. e74557.

[29] C. Segalin et al., “The mouse action recognition system (MARS) software pipeline for automated analysis of social behaviors in mice,” eLife, vol. 10, p. 63720, Nov. 2021.

[30] S. Yan, Y. Xiong, and D. Lin, “Spatial temporal graph convolutional networks for skeleton-based action recognition,” in Proc. 32nd AAAI Conf. Artif. Intell., 2018, pp. 7444–7452.

[31] C. Pizzari, M. Cannici, and M. Matteucci, “Skeletal-based action recognition via spatial and temporal transformer networks,” Comput. Vis. Image Understand., vol. 208–209, Jul. 2021, Art. no. 103219.

[32] X. Wu, Z. Zhang, C. Yuan, B. Li, Y. Deng, and W. Hu, “Channel-wise topology refinement graph convolution for skeleton-based action recognition,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., Oct. 2021, pp. 13359–13368.

[33] X. Hao, J. Li, Y. Guo, T. Jiang, and M. Yu, “Hypergraph neural network for skeleton-based action recognition,” IEEE Trans. Image Process., vol. 30, pp. 2263–2275, 2021.

[34] H. Yang, D. Yan, L. Zhang, Y. Sun, D. Li, and S. J. Maybank, “Feedback graph convolutional network for skeleton-based action recognition,” IEEE Trans. Image Process., vol. 31, pp. 164–175, 2022.

[35] N. Heidari and A. Iosifidis, “Temporal attention-augmented graph convolutional network for efficient skeleton-based human action recognition,” in Proc. 25th Int. Conf. Pattern Recognit. (ICPR), Jan. 2021, pp. 7907–7914.

[36] H. Sahbi, “Learning Laplacians in Chebyshev graph convolutional networks,” Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW), Oct. 2021, pp. 2064–2075.

[37] J. Brunh, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs,” 2013, arXiv:1312.6203.

[38] H. Sahbi, “Learning connectivity with graph convolutional networks,” in Proc. 25th Int. Conf. Pattern Recognit. (ICPR), Jan. 2021, pp. 5049–5054.

[39] H. Sahbi et al., “Context-dependent kernels for object classification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 4, pp. 699–708, Apr. 2011.

[40] H. Sahbi, “Phase-field models for lightweight graph convolutional networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2023, pp. 4644–4650.

[41] H. Sahbi, “TCMP: End-to-end topologically consistent magnitude pruning for miniaturized graph convolutional networks,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Apr. 2024, pp. 3065–3069.
[49] H. Sahbi, “Topologically-consistent magnitude pruning for very lightweight graph convolutional networks,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Oct. 2022, pp. 3495–3499.

[50] H. Sahbi, “DAMP: Distribution-aware magnitude pruning for budget-sensitive graph convolutional networks,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Apr. 2024, pp. 3070–3074.

[51] A. Vaswani et al., “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst. (NIPS), 2017, pp. 5998–6008.

[52] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, “Graph transformer networks,” in Proc. Adv. Neural Inf. Process. Syst., vol. 32, 2019, pp. 11983–11993.

[53] D. Q. Nguyen, T. D. Nguyen, and D. Phung, “Universal graph transformer self-attention networks,” 2019, arXiv:1909.11853.

[54] Y. Zhang, B. Wu, W. Li, L. Duan, and C. Gan, “STST: Spatial–temporal specialized transformer for skeleton-based action recognition,” in Proc. 29th ACM Int. Conf. Multimedia, Oct. 2021, pp. 3229–3237.

[55] K. Gedamu, Y. Ji, L. Gao, Y. Yang, and H. T. Shen, “Relation-mining self-attention network for skeleton-based human action recognition,” Pattern Recognit., vol. 139, Jul. 2023, Art. no. 109455.

[56] B. L. N. Huu and T. Matsui, “STEP CATFormer: Spatial–temporal effective body-part cross attention transformer for skeleton-based action recognition,” in Proc. Brit. Mach. Vis. Conf., 2023, pp. 1–12.

[57] T. D. Pereira et al., “Fast animal pose estimation using deep neural networks,” Nature Methods, vol. 16, no. 1, pp. 117–125, Jan. 2019.

[58] T. D. Pereira et al., “SLEAP: A deep learning system for multi-animal pose tracking,” Nature Methods, vol. 19, no. 4, pp. 486–495, Apr. 2022.

[59] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian, “Dynamic multiscale graph neural networks for 3D skeleton based human motion prediction,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 214–223.

[60] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and Ł. Kaiser, “Universal transformers,” 2018, arXiv:1807.03819.

[61] J. Lee, Y. Lee, J. Kim, A. R. Kosiorek, S. Choi, and Y. W. Teho, “Set Transformer: A framework for attention-based permutation-invariant neural networks,” in Proc. Int. Conf. Mach. Learn. (ICML), 2019, pp. 3744–3753.

[62] W. Jin et al., “Self-supervised learning on graphs: Deep insights and new direction,” 2020, arXiv:2006.10141.

[63] X. P. Burgos-Artizzu, P. Dollár, D. Lin, D. J. Anderson, and P. Perona, “Social behavior recognition in continuous video,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2012, pp. 1322–1329.

[64] V. Jackson-Lewis and S. Przedborski, “Protocol for the MPTP mouse model of Parkinson’s disease,” Nature Protocols, vol. 2, no. 1, pp. 141–151, Jan. 2007.

[65] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, “Constructing stronger and faster baselines for skeleton-based action recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 2, pp. 1474–1488, Feb. 2023.

[66] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, “NTU RGB+D: A large scale dataset for 3D human activity analysis,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 1010–1019.

[67] 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 Trans. Pattern Anal. Mach. Intell., vol. 42, no. 10, pp. 2684–2701, Oct. 2020.