Symbiotic Graph Neural Networks for 3D Skeleton-based Human Action Recognition and Motion Prediction

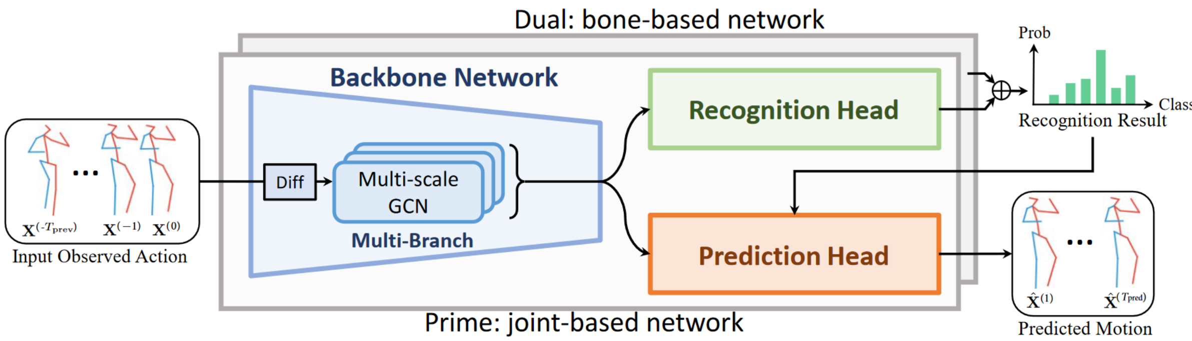
感觉这一篇文章可以作为主要参考文章，重点优化一下：
1. skeleton local information和结构的学习，2. 原始数据robustness (参考STST的信息优化），3. 在结构中使用transformer（目前觉得transformer 更适合做用于prediction module, recognition module应该采用一个类似GAN的模型；backbone network先调研有没有更好的transformer结构（目前没看到，之后再搜搜看），或者是可以看下有没有更快的GCN,（关于GCN的研究有侧重速度和robustness的文章，不知道可不可以嫁接到我们的recognition-prediction 任务上

1. Premilinaries

1.1. Definitions

1. model proposed:
    1. symbiotic graph neural network (Sym-GNN), simultaneously handle skeleton-based action recognition and motion prediction, using graph-based operations to capture spatial features
    2. operators to extract multi-scale spatial information:
        1. joint-scale graph convolution operators (JGC), based on
            1. actional graph inference module (AGIM), capture action-based relations
        2. part-scale graph convolution operators (PGC), part-scale graph
            1. nodes are integrated body-part features
            2. edges are based on body-part connections

2. proposed model:



1.2. Theories

2. Problem set

2.1. Target question

This article studies 3D Skeleton-based action recognition and motion prediction jointly.

2.2. notations

1. $X^{(t)} \in \mathbb{R}^{M \times D_x}$, where $t > 0$: action pose at time stamp t. M is the number of joints and $D_x = 3$ reflecting the 3D joint positions
2. $A \in \{0,1\}^{M \times M}$: adjacent matrix, where $(A)_{ij} = 1$ when ith and j th body-joints are connected with bones, otherwise 0
3. Action sequence: $\{X_{prev}, X_{pred}, y\}$, where $X_{prev} = [X^{-T_{prev}}, \dots, X^{(0)}] \in \mathbb{R}^{T_{prev} \times M \times D_x}$, denoting the previous motion tensor, $X_{prev} = [X^{(1), \dots, X^{(-T_{pred})}}] \in \mathbb{R}^{T_{pred} \times M \times D_x}$ denoting the future motion tensor, T_{prev} and T_{pred} are the frame numbers of previous and future motions respectively
4. $y \in \{0,1\}^C$ denotes class-label in C possible class category, one-hot vector. $\hat{y}, \hat{X}_{pred} = F(X_{prev}; \theta_{bk}, \theta_{recg}, \theta_{pred})$, where $\theta_b k, \theta_{recg}$ denote trainable parameters of the backbone, action-recognition head and the motion prediction head, respectively

2.3. model construction

2.3.1. joint-scale graph operators

1. route:

$$X_{prev} \xrightarrow{AGIM} A_{ACT}. \quad \left.\begin{matrix} A_{ACT} \\ x^{(t)} \end{matrix}\right\} \xrightarrow{AGC} Y_{AGC}.$$

2. Actional graph convolution: actional graph: $G_{act}(V, A_{act})$, where $V = \{v_1, \ldots, v_M\}$ is the joint set and $A_{act} \in \mathbb{R}^{M\times M}$ is the adjacency matrix, revealing pairwise joint-scale actional relations
3. AGIM: actional graphs inference module: learning A_{act} purly from observation without knowing action categories

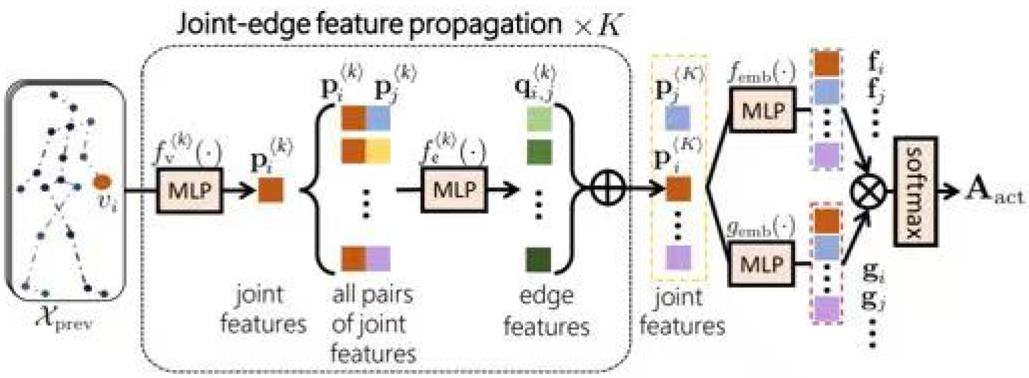


Fig. 3. Actional graphs inference module (AGIM) propagates features between joints and edges for K iterations and uses correlations between joint features to obtain actional graphs.

1. vector representation of the ith joint positions across all observed frames: $x_i = vec(X_{prev}[:, i, :]) \in \mathbb{R}^{D_x T_{prev}}$
2. $f_v^{<0>}(\cdot)$: multilayer perceptron(MLP) that maps the raw joint moving data x_i to joint features $p_i^{<0>}$
3. in the k^{th} iteration, the features propagated as:

$$q_{ij}^{<k>} = f_e^{<k>}([p_i^{<k-1>}, p_j^{<k-1>}]) \in \mathbb{R}^{D_e}$$

$$p_i^{<k>} = f_v^{<k>}\left(\frac{1}{M-1} \sum_{v_j \in V, j \neq i} q_{i,j}^{<k>}\right) \in \mathbb{R}^{D_v}$$

4. $p_i^{<k>}, q_{i,j}^{<k>}$ are the feature vectors of the ith joints and the edge connecting ith and jth joints at the kth iteration