Pose-Aided Video-Based Person Re-Identification via Recurrent Graph Convolutional Network

Honghu Pan, Qiao Liu†, Yongyong Chen*, Member, IEEE, Yunqi He, Yuan Zheng+, Feng Zheng, and Zhenyu He*, Senior Member, IEEE

Abstract—Existing methods for video-based person re-identification (ReID) mainly learn the appearance feature of a given pedestrian via a feature extractor and a feature aggregator. However, the appearance models would fail to learn a large inter-class variance when different pedestrians have similar appearances. Considering that different pedestrians have different walking postures and body proportions, we propose to learn the discriminative pose feature beyond the appearance feature for video retrieval. Specifically, we implement a two-branch architecture to separately learn the appearance feature and pose feature, and then concatenate them together for inference. To learn the pose feature, we first detect the pedestrian pose in each frame through an off-the-shelf pose detector, and construct a temporal graph using the pose sequence. We then exploit a recurrent graph convolutional network (RGCN) to learn the node embeddings of the temporal pose graph, which devises a global information propagation mechanism to simultaneously achieve the neighborhood aggregation of intra-frame nodes and message passing among inter-frame graphs. Finally, we propose a dual-attention method (DAM) consisting of node-attention and time-attention to obtain the temporal graph representation from the node embeddings, where the self-attention mechanism is employed to learn the importance of each node and each frame. We verify the proposed method on three video-based ReID datasets, i.e., Mars, DukeMTMC and iLIDS-VID, whose experimental results demonstrate that the learned pose feature can effectively improve the performance of existing appearance models.

Index Terms—Video surveillance, graph neural networks, recurrent neural networks.

I. INTRODUCTION

PERSON re-identification (ReID) [1], [2] is an important yet challenging task in public security and video surveillance. The visual data provided by the surveillance camera are usually video clips with multiple frames. Hence, the video-based person ReID is of great significance and thus has attracted increasing attention in recent years. Given a pedestrian video clip with a specific identity, the video-based person ReID aims to find out video clips with the same identity from the retrieval set. Existing methods mainly follow the pipeline of “feature extraction & feature aggregation”. Specifically, they first extract an appearance feature sequence from the input video clip via a deep convolutional neural network (CNN), and then aggregate the feature sequence into a video representation by an aggregator. Existing methods have explored multiple aggregators to capture the temporal and spatial clues of pedestrian appearance, such as RNN-based aggregator [3], [4], attention-based aggregator [5], [6], [7], 3D convolution-based aggregator [8], [9], and so on.

However, these methods only consider the appearance information of the given pedestrian, which would fail to learn a large inter-class variance for pedestrians with similar appearances as in shown Fig. 1; another disadvantage of the appearance feature is the misalignment of human body [10], [11]. While the pedestrian pose, which reflects the walking postures and body proportions of pedestrians, could be used as a supplement to alleviate these problems [10], [12]. Thereby, we propose to combine the discriminative pedestrian pose feature with the appearance feature for video-based ReID. To this end, we implement a two-branch architecture consisting of an appearance feature learning branch and a pose feature learning one. In the training stage, we train each branch separately; while in the inference stage, we concatenate the appearance feature and pose feature as the final video representation for pedestrian retrieval.

The crux is how to learn the pose feature of a given pedestrian. We observe that the keypoints of human pose...
Fig. 1. An illustration of pedestrians with similar appearances. In the above figure, the pedestrians of row (a) and row (b) are similar in appearance: they both wear white T-shirts and carry black schoolbags. However, they walk with different postures: (a) is playing a mobile phone and (b) is holding his glasses. Existing video-based ReID models mainly learn the pedestrian appearance feature, which are difficult to learn a large inter-class variance for pedestrians with similar appearances. While in this paper, we propose to learn the pose feature beyond the appearance feature for the pedestrian video retrieval.

Fig. 2. Qualitative analysis and quantitative analysis of the pedestrian video features learned by existing appearance model and our RGCN. In (a) and (b), we visualize the video features after dimensionality reduction by t-SNE [16], in which marks with the same color belong to the same identity. As can be seen, features learned by RGCN have a larger inter-class variance. In (c) and (d), we present the quantitative comparison of the appearance model and RGCN on the testing set of Mars [17], in which \(d_{\text{intra}}\) and \(d_{\text{inter}}\) denote the maximum intra-class distance and minimum inter-class distance of each identity, respectively. We can conclude that our RGCN outperforms the appearance-based model on most identities.

naturally form a graph structure. Concretely, nodes in graph refer to the pose keypoints, while edges are defined by the connections between keypoints. Therefore, we can construct a temporal pose graph from the given video clip with multiple frames, while each separate graph in the temporal graph corresponds to a frame. A similar task aiming at learning the node embeddings from temporal skeleton graph is the skeleton-based action recognition [13], [14], [15], in which the node-level connection between consecutive frames is usually constructed. However, this connection method would be not appropriate for pose feature learning, since the pose keypoints in the video frame may be partially occluded due to the camera view and pedestrian posture.

Therefore, we propose the **graph-level message passing** rather than the node-level temporal propagation to reduce the impact of occlusion. To this end, we develop a recurrent graph convolutional network (RGCN) to simultaneously perform the node neighborhood aggregation within graph and model the temporal correlations among graphs. In RGCN, the hidden state that contains the global information of pose graph is first fed to a graph convolutional (GCN) layer, then temporally propagated within a recurrent model, in which the recurrent model and GCN partially share weights. To achieve the person ReID, the final step is to obtain the temporal graph representation or pose feature from the features of multi-frame multi-node. Existing methods [13], [18] on skeleton feature learning mainly perform the average pooling on temporal graph, which fails to consider the frame importance and node importance. For instance, the occluded frames or keypoints may be less important than visible ones. To tackle this problem, we devise a dual-attention method (DAM), i.e., node-attention and time-attention, to learn the temporal graph representation, in which the self-attention mechanism is utilized in the node-attention and time-attention to evaluate the importance of each node and each frame, respectively. DAM is a novel module that simultaneously models the temporal and spatial importance by self-attention.

We note that our RGCN is an add-on method that can be plugged in existing appearance models to remedy the shortcomings of the appearance feature. We test our method on the basis of multiple appearance-based baselines, e.g., average pooling, RNN aggregation and attention aggregation. The experimental results on three widely-used datasets, i.e., Mars [17], DukeMTMC [19], [20] and iLIDS-VID [21], show that our method outperforms the baseline models by a large margin. For example, under the average pooling, the Rank1 on iLIDS-VID increases from 82.0% to 86.0% after introducing the pose feature. Moreover, the visualization in Fig. 2 demonstrates that the pose information can effectively enlarge the inter-class variance. The main contributions of this paper are three-fold:

- To the best of our knowledge, this is the first study that proposes to learn the pedestrian pose feature beyond the appearance feature for video-based ReID, which is able to enlarge the inter-class variance of pedestrians with similar appearance.
- To learn the node embeddings of the pedestrian pose graph, we integrate the recurrent model and graph
convolutional network into a recurrent graph convolutional network (RGCN) enabling the graph-level message passing, so that the hidden state of pose graph contains the global information.

- We exploit the node-attention and time-attention to obtain the temporal graph representation or pose feature from the node embeddings, where the self-attention mechanism is leveraged to learn the node and frame importance.

The remainder of this paper is organized as follows: Section II introduces some recent studies related with this paper; Section III elaborates the two-branch architecture, especially for the pose feature learning branch; Section IV presents the experiments and visualizations; Section V draws brief conclusions.

II. RELATED WORKS

A. Image-Based Person Re-Identification

Image-based ReID [22], [23], [24] aims to identify the identity of a given pedestrian image. In this area, learning an image representation [25], [26] via a deep convolutional neural network (CNN) is the most widely-used and most successful method. The global representation is first applied in image-based ReID. For example, TriNet [27] implemented a hard triplet mining strategy to learn the discriminative features of pedestrian images; OSM [28] combined the center loss and triplet loss to learn the compact global representations. The part-level features have also been explored by existing works: PCB [29] first proposed to learn the part-level features through a part-based convolutional module; MGH [30] combined the global feature and the part-level features to formulate the multi-granular feature as the pedestrian representation. Meanwhile, a great number of studies [10], [12], [31], [32] have attempted to learn the pose feature for pedestrian image retrieval: PDC [12] used the human body appearance to learn the pose-invariant features; PNIG [31] proposed a pose-normalization GAN [33] model to generate the image via the pedestrian pose; PIE [10] employed the pedestrian pose information to reduce the pedestrian misalignment; PVPM [32] proposed a pose-guided visible part matching method for the occluded person ReID. Even though the pose-based pedestrian ReID has been proven effective in image-based ReID, existing works of video-based ReID have not considered incorporating the pose information into the pedestrian feature.

B. Video-Based Person Re-Identification

Compared with image-based person ReID, the samples of video-based person ReID contain richer temporal and spatial information. Therefore, existing video-based person ReID methods [3], [4], [34], [35], [36] primarily focus on mining the temporal and spatial correlations via a feature aggregator. Early works [3], [4] employed a recurrent model as the aggregator. For example, RCN [3] combined the recurrent neural network and convolutional network to learn the representation of the input video clip; RFA-Net [4] proposed to learn a globally discriminate feature via LSTM [37]. However, the recurrent model only performs the temporal aggregation on video features, and it fails to capture the spatial clues. While the 3D convolution-based aggregator can model both temporal and spatial information: M3D [8] inserted the 3D convolutional layers into the 2D CNN model to enable the multi-scale feature learning; AP3D [9] implemented an appearance-preserving module by using the 3D convolution; SSN3D [11] adopted the 3D convolution across frames to alleviate the temporal appearance misalignment. The disadvantage of the 3D convolution is the high space complexity, while the attention-based aggregator [5], [6], [7], [38] can effectively reduce storage: RGSA [5] implemented a relation-guided spatial attention module and a relation-guided temporal refinement module to exploit both the spatial and temporal clues: SANet [39] devised a statistic attention block to learn high-order statistics of feature maps, which could jointly capture the long-range dependencies and subtle differences. Some works exploited information beyond appearance to learn pedestrian representations: MFA [40] proposed to learn pedestrian motion features via the motion feature aggregation method. The difference between our RGCN and MFA is that MFA represented the motion via optical flow, while we adopt the continuous walking postures.

C. Graph Convolutional Network

The GCN models [41], [42], [43] usually refer to the spectral-based graph neural networks, which define the node neighborhood aggregation via the graph Laplacian decomposition. SpecNet [42] proposed to learn the graph filter directly; to reduce the learning parameters and parameterize the graph filter, ChebNet [43] approximated the graph filter with the Chebyshev polynomial; GCN [41] proposed a layer-wise architecture to learn the graph embedding, where each layer aggregates 1-hop neighborhood for each node. Recently, a lot of studies [44], [45], [46] successfully applied the GCN model in person ReID. In image-based ReID, AAGCN [44] took advantage of the low-pass property of GCN to reduce the intra-class variance; HLGAT [47] proposed a local graph attention network to learn the intra-local and inter-local relations. In video-based ReID, CTL [48] employed a graph network to learn the corrections among human parts, where the nodes are extracted via a pose estimator; STGCN [45] implemented a temporal-spatial GCN to aggregate the temporal and spatial features; AGRL [49] constructed a part feature-based graph among video frames, then learned the relations or interactions between relevant regional features via GCN model. However, existing graph-based methods (e.g., AGRL and STGCN) constructed graphs by using feature maps of pedestrian video, which still only learned the appearance-level features. While our RGCN can model the pedestrian motion via the temporal pose graph.

D. Skeleton-Based Action Recognition

The skeleton-based action recognition methods using deep learning algorithms can be summarized as three categories: CNN-based [50], [51], RNN-based [52], [53], [54] and GCN-based [13], [55] models. The CNN-based models transform the 3D skeleton into the image data, then employ a CNN framework to learn the feature. For example, MTLN [50] generated
three video clips from each skeleton sequence, where each clip corresponds to one channel of the origin skeleton data; ESV [51] proposed an enhanced skeleton visualization method that converts the skeleton sequence into the 2D images. The CNN-based models could not capture the temporal correlations, and the RNN-based ones can model the continuous motion. EleAtt [52] combined the attention mechanism and a recurrent model to learn the importance of each element and model the temporal dynamics; SR-TSL [53] proposed a spatial reasoning module and a spatial reasoning module to achieve the spatial and temporal message passing; VA-LSTM [54] devised a LSTM-based adaptive framework to reduce view variations by automatically regulating observation viewpoints. Both the CNN-based [50], [51] and RNN-based [52], [53], [54] models fail to exploit the skeleton topology, i.e., the joint connections. In recent years, the GCN-based models have shown its advantage in skeleton-based action recognition since the graph models can explicitly learn the node corrections from the skeleton data. ST-GCN [13] presented the first strong baseline: it constructed a spatial-temporal graph for each skeleton sequence, then learned the node features by spatial and temporal aggregation. Many graph-based studies have been proposed on the basis of ST-GCN: DGGN [14] constructed a directed acyclic graph to model the relationship between joints and bones; DMGNN [15] constructed a multiscale graph to learn the part-level relations; Shift-GCN [56] integrated the shift operation and GCN to implement a lightweight architecture. The main difference between this task and our pose feature learning lies in the data source: the skeleton data [57] are collected by the depth sensor, while ours are detected by a pose detector. Hence, our pose data would be partially occluded. This paper devises a global information propagation mechanism to alleviate the influence from local node occlusion.

III. METHODOLOGIES

In this section, we first present the definition of video-based person ReID and the overall framework of the proposed model in Section III-A, then introduce the appearance feature learning baselines in Section III-B, next elaborate the pose feature learning paradigm by RGCN in Section III-C, last define the loss functions for model training in Section III-D.

A. Problem Definition and Overview

To enable the pedestrian video retrieval, the representation learning is widely employed in video-based ReID. Specifically, we learn a representation vector for each video clip by using deep neural networks, whose objective is to maximize the similarity between intra-class samples and minimize that between inter-class samples. We denote the given video clip as \( V = \{I_1, I_2, \cdots, I_T\} \), where \( I_t \) is the \( t \)-th frame and \( T \) is the length. We aim to learn a discriminative feature \( f \) from the image sequence \( V \).

Existing methods mainly focus on the appearance feature of the pedestrian, which follow the pipeline of “extraction & aggregation”. Generally, a backbone network (e.g., ResNet [58]) is first adopted to extract the appearance feature \( f^a \) of the image sequence \( I_t \), then an aggregator, e.g., average pooling, RNN-based aggregator or attention-based aggregator, performs the feature aggregation from the feature sequence \( f^a_t \) to the video appearance representation \( f^a \).

However, existing methods only consider appearances of pedestrians, which would hardly learn a large inter-class variance for pedestrians with similar appearances. To resolve this issue, we propose the pose-aided video-based person ReID, in which the pose information is involved in the learned video representation. To this end, we propose a two-branch architecture as in shown Fig. 3 to separately learn the appearance feature \( f^a \) and pose feature \( f^p \). The main contribution of this paper is the learning paradigm of \( f^p \). As can be seen in Fig. 3, we first construct a temporal pose graph with the detected pedestrian pose, then develop a recurrent graph convolutional network (RGCN) model to learn the node embeddings of the temporal graph, last exploit the dual-attention (DAM) mechanism to obtain the graph representation.

In this paper, we denote the matrices and vectors as uppercase characters and lowercase characters, respectively. The notations and their corresponding descriptions are presented in Table I.

| Notations | Descriptions |
|-----------|--------------|
| \( V \) | The input video |
| \( T \) | The length of \( V \) |
| \( I_t \) | The \( t \)-th frame of \( V \) |
| \( f^a \) | The appearance feature |
| \( f^p \) | The pose feature |
| \( G \) | The temporal pose graph |
| \( G_t \) | The \( t \)-th frame pose graph in \( G \) |
| \( X_{t}^{(0)} \) | The initialized feature of \( G_t \) |
| \( h_t \) | The hidden state of a sole node |
| \( H_t \) | The hidden state of \( G_t \) |
| \( a_t^{(n)} \) | The node weight learned by node attention |
| \( f_t^{(n)} \) | The frame weight learned by time attention |
| \( h_t^{(n)} \) | The graph feature learned by node attention |
| \( h_t^{(f)} \) | The graph feature learned by time attention |

B. Appearance Feature Learning

In this section, we introduce several appearance-based baselines to make this paper self-contained. Specifically, we present three widely-used aggregators for the appearance feature learning:

1) Average Pooling (AP): takes the average value of \( f^a_t \) as the final representation \( f^a \):

\[
f^a = \frac{1}{T} \sum_{t=1}^{T} f^a_t. \tag{1}
\]

2) Attention Aggregation (AA): performs the weighted average on \( f^a_t \), where the weight coefficients \( a_t \) are learned by the...
1) Graph Construction: Given a video clip containing $T$ frames, we employ OpenPose [59], a popular 2D pose estimator, to detect the pedestrian pose in each frame. As seen in Fig. 4(a), we retain 14 keypoints for each pedestrian, while each keypoint represents a specific part; for instance, keypoint 1 refers to the head, 3 and 6 refer to the shoulder. We then establish the connection between the keypoints according to the connections of human skeleton, for example, 1 and 2, 2 and 3, and etc.

With the pose keypoints and the connections, we construct a pose graph $G_t$ in the $t$-th frame, where $G_t$ is composed of the nodes $V_t$ and the edges $E_t$. Nodes $V_t$ consists of the detected keypoints, i.e., $V_t = \{v_{ti}|i = 1, 2, \cdots, 14\}$. The feature of $v_{ti}$ is initialized to the 2D position of keypoint $v_{ti}$ in the $t$-th frame. For keypoints that are not detected due to the occlusion, we use zero padding to define their initial features. Edges $E$ depict the connections between pairwise nodes. We employ an adjacency matrix $A = \{A_{ij}\}$ to represent $E$, where $A_{ij}$ is equal to 1 if $v_{ti}$ is connected to $v_{tj}$ and 0 if not. As can be seen in Fig. 4(b), the adjacency matrix is a symmetrical binary matrix, and it is constant over time. For a video clip, we could obtain $T$ pose graphs and then combine them together to form a temporal pose graph $G = \{G_t|t = 1, 2, \cdots, T\}$.

2) Recurrent Graph Convolutional Network: To learn the node embeddings of the continuous pose graph $G = \{G_t|t = 1, 2, \cdots, T\}$, we should consider both the node features within $G_t$ and the temporal information among $G_t$. A simple idea is to follow the GCN&RNN pipeline, which first learns the node embeddings of the continuous pose graph $G$-th frame node embeddings with a GCN model:

$$H_t = \{H_t^1, \cdots, H_t^T\},$$

where $H_t^i$ represents the node embedding of keypoint $v_{ti}$ in the $t$-th frame.

3) RNN Aggregation (RA): utilizes a recurrent model (i.e., LSTM [37]) to capture the temporal clues:

$$f^A = \frac{1}{T} \sum_{t=1}^{T} o_t^A,$$  \hspace{1cm} (3)

where $o_t^A$ denotes the output of the recurrent model in the $t$-th frame.

C. Pose Feature Learning

In this section, we first introduce the construction of the temporal pose graph, then elaborate the RGCN model for node embeddings learning, last present the dual-attention method for graph representation learning.

1) Graph Construction: Given a video clip containing $T$ frames, we employ OpenPose [59], a popular 2D pose estimator, to detect the pedestrian pose in each frame. As seen in Fig. 4(a), we retain 14 keypoints for each pedestrian, while each keypoint represents a specific part; for instance, keypoint 1 refers to the head, 3 and 6 refer to the shoulder. We then establish the connection between the keypoints according to the connections of human skeleton, for example, 1 and 2, 2 and 3, and etc.

With the pose keypoints and the connections, we construct a pose graph $G_t$ in the $t$-th frame, where $G_t$ is composed of the nodes $V_t$ and the edges $E_t$. Nodes $V_t$ consists of the detected keypoints, i.e., $V_t = \{v_{ti}|i = 1, 2, \cdots, 14\}$. The feature of $v_{ti}$ is initialized to the 2D position of keypoint $v_{ti}$ in the $t$-th frame. For keypoints that are not detected due to the occlusion, we use zero padding to define their initial features. Edges $E$ depict the connections between pairwise nodes. We employ an adjacency matrix $A = \{A_{ij}\}$ to represent $E$, where $A_{ij}$ is equal to 1 if $v_{ti}$ is connected to $v_{tj}$ and 0 if not. As can be seen in Fig. 4(b), the adjacency matrix is a symmetrical binary matrix, and it is constant over time. For a video clip, we could obtain $T$ pose graphs and then combine them together to form a temporal pose graph $G = \{G_t|t = 1, 2, \cdots, T\}$.

2) Recurrent Graph Convolutional Network: To learn the node embeddings of the continuous pose graph $G = \{G_t|t = 1, 2, \cdots, T\}$, we should consider both the node features within $G_t$ and the temporal information among $G_t$. A simple idea is to follow the GCN&RNN pipeline, which first learns the $t$-th frame node embeddings with a GCN model:

$$X_t = \sigma(\tilde{D}^{-1/2} \tilde{A} \tilde{D}^{-1/2} X_t^{(0)} W_g),$$  \hspace{1cm} (4)
RGCN with two GC layers can be denoted as:

$$H_t = \tanh(H_{t-1} W_h + \sigma(\hat{A} \hat{X}_t(0) W_s^{(1)} W_s^{(2)}) + b),$$  \hspace{1cm} (7)

where $W_s^{(1)}$ and $W_s^{(2)}$ denote the learning-based parameters in the first and second layer, respectively.

**Discussion:** Compared to the GCN&RNN pipeline and ST-GCN [13], our RGCN has the following advantages: 1) the hidden state $H_t$ of RGCN in Eq. (6) contains the global information of the pose graph; 2) RGCN runs faster, as it avoids to input the node one-by-one into model; 3) RGCN contains fewer parameters, which benefits from the weight-sharing. Specifically, the GCN&RNN pipeline contains three weight matrices in each layer, i.e., $W_h$ in Eq. (4), $W_h$ and $W_s$ in Eq. (5); while our RGCN model contains two learning-based weights ($W_h$ and $W_s$ in Eq. (6)).

RGCN of Eq. (6) combines GCN and the vanilla RNN model, while the vanilla RNN can also be replaced by other recurrent models, such as LSTM [37]. In this paper, we also implement an enhanced model using LSTM and GCN, whose equations are presented as follows:

$$F_i = \text{sigmoid}(\hat{A} \hat{X}_t(0) W_f + H_{t-1} U_f + b_f),$$

$$I_i = \text{sigmoid}(\hat{A} \hat{X}_t(0) W_i + H_{t-1} U_i + b_i),$$

$$O_i = \text{sigmoid}(\hat{A} \hat{X}_t(0) W_o + H_{t-1} U_o + b_o),$$

$$G_i = \tanh(\hat{A} \hat{X}_t(0) W_g + H_{t-1} U_g + b_g),$$

$$C_i = F_i \odot C_{i-1} + I_i \odot G_i,$$

$$H_t = O_i \odot \tanh(C_i),$$  \hspace{1cm} (8)

where sigmoid and tanh are activation functions, and $\odot$ denotes the Hadamard product. We dub the model of Eq. (8) as LGCN.

3) **Dual-Attention Method:** We take the hidden state $H_t$ as the embedding of $G_t$, then the node embeddings of the continuous pose graph $G$ can be denoted as $H = \{H_t | t = 1, 2, \cdots, T\}$. In $H$, there are two dimensions of information: time dimension and spatial dimension. We aim to obtain a feature vector $f^p$ from $H$ to represent the pedestrian pose. Directly performing average pooling on $H$ fails to distinguish important frames or nodes from global graph representation. To this end, we proposed a dual-attention method, i.e., node-attention and time-attention, to convert the node embeddings $H$ to the pose feature $f^p$. The node-attention method first performs the average pooling on the time-dimension of $H$:

$$\bar{H} = \frac{1}{T} \sum_{t=1}^{T} H_t,$$  \hspace{1cm} (9)

where $\bar{H} = \{\bar{h}_j | j = 1, 2, \cdots, 14\}$, and $\bar{h}_j \in R^{1 \times n}$ denotes the embedding of the $j$-th node. Then it learns the weights $a_j^{(n)}$ of each node with the self-attention mechanism:

$$a_j^{(n)} = \text{softmax}_j(\bar{h}_j w^{(n)}),$$  \hspace{1cm} (10)

last aggregates node features with $a_j^{(n)}$:

$$h^{(n)} = \sum_{j=1}^{14} a_j^{(n)} \bar{h}_j,$$  \hspace{1cm} (11)

where $w^{(n)} \in R^{n \times 1}$ is a learning-based parameter.
Similar with the node-attention method, the time-attention method can be divided three steps: it first performs the average pooling on $H_t = \{h_{tj}\} = 1, 2, \cdots, 14$:

$$\bar{h}_t = \frac{1}{14} \sum_{j=1}^{14} h_{tj}, \quad (12)$$

then learns the weights of each frame via the self-attention mechanism:

$$d_t^{(i)} = \text{softmax}(\bar{h}_t w^{(i)}), \quad (13)$$

finally aggregates the temporal features with the learned weights:

$$h^{(i)} = \sum_{t=1}^{T} d_t^{(i)} \bar{h}_t, \quad (14)$$

where $w^{(i)} \in \mathbb{R}^{n \times 1}$ is a learning-based parameter.

**a) Rationale behind DAM:** The rationale behind the node-attention is that different keypoints in each video clips correspond to different importances. For instance, occluded keypoints may be less important than visible ones. While the rationale behind the time-attention is that the occluded frames may contribute less to the retrieval results. In conclusion, we learn the feature $h^{(n)}$ and $h^{(t)}$ by the node-attention and time-attention, respectively. Finally, we concatenate $h^{(n)}$ and $h^{(t)}$ as the pose feature $f^p$ of the input video as shown in Fig. 3.

**b) Rationale behind the architecture design:** The RGCN architecture in Fig. 3 contains two branches: one for learning the appearance feature, and another for learning the pose feature. For the pose feature learning branch, DAM concatenates features learned by node-attention and time-attention. While RGCN concatenates the appearance feature learned by baseline model and pose feature learned by DAM to formulate the final pedestrian representation. Thus, DAM can be seen as a sub-module of RGCN. In Section IV-B, besides concatenation, we will explore other fusion methods for DAM and RGCN.

### D. Model Training

Given a video clip $V_t$ corresponding to a label $y_t$, we can learn an appearance feature $f^a_i$ and a pose feature $f^p_i$ by the appearance model and RGCN or LGCN model, respectively. Following the existing ReID methods [30], [60], we train our model with the triplet loss and identity loss. For the triplet loss, we train $f^a_i$ and $f^p_i$ separately:

$$L_{tri}^a = \sum_{i=1}^{N} \max(0, \delta + \max_{y_{i,j} = y_t} \{\|f^a_i - f^a_j\|_2\}) - \min_{y_{i,j} \neq y_t} \{\|f^a_i - f^a_j\|_2\}, \quad (15)$$

$$L_{tri}^p = \sum_{i=1}^{N} \max(0, \delta + \max_{y_{i,j} = y_t} \{\|f^p_i - f^p_j\|_2\}) - \min_{y_{i,j} \neq y_t} \{\|f^p_i - f^p_j\|_2\}, \quad (16)$$

where $\delta$ is a pre-defined margin and $N$ is the training batch size. We then concatenate $f^a_i$ and $f^p_i$ as the final feature $f_i$, and define the identity loss as follows:

$$L_{id} = -\frac{1}{N} \sum_{i=1}^{N} \log(\frac{\exp(z_i)}{\sum_{i=1}^{N} \exp(z_i)}). \quad (17)$$

In Eq. (17), $z_i = FC(f_i)$ is the classification logits, where $FC$ refers to a fully-connected layer.

In this section, we propose an adaptive training strategy to optimize $L_{tri}^a$ and $L_{tri}^p$, whose purpose is to impose a larger punishment to the larger loss. To this end, we define an adaptive parameter $\lambda$:

$$\lambda = \frac{L_{tri}^a}{L_{tri}^a + L_{tri}^p}. \quad (18)$$

Then the final loss can be denoted as the linear combination of above-mentioned losses:

$$L = L_{id} + \lambda L_{tri}^a + (1 - \lambda) L_{tri}^p. \quad (19)$$

By minimizing $L$, we could jointly learn the discriminative appearance feature $f^a_i$ and pose feature $f^p_i$.

## IV. Experiments

### A. Datasets and Experimental Implementations

1) **Datasets:** To validate the effectiveness of the learned pose feature in video-based ReID, we conduct our experiments on three widely-used datasets: Mars [17], DukeMTMC [19], [20] and iLIDS-VID [21]. Table II summarizes the statistics of three datasets. For the evaluation metrics, we report the mean Average Precision (mAP) and Cumulative Match Characteristic (CMC).

| Dataset   | Identities | Videos  | Camera | Splits |
|-----------|------------|---------|--------|--------|
| Mars      | 1,261      | 20,751  | 6      | 625/636|
| DukeMTMC  | 1,404      | 4,832   | 8      | 702/702|
| iLIDS-VID | 300        | 600     | 2      | 702/702|

### B. Ablation Studies

In this section, we conduct our ablation studies on Mars [17] and iLIDS-VID [21]; we report mAP and Rank1 of CMC for comparison.
TABLE III
COMPARISON WITH THE APPEARANCE MODELS

| Method      | Mars     | iLIDS-VID |
|-------------|----------|-----------|
|             | mAP   | Rank1 | mAP   | Rank1 |
| AP          | 81.1% | 85.2% | 87.6% | 82.0% |
| AP+RGCN    | 82.6% | 87.9% | 89.4% | 84.7% |
| AP+LGCN    | 82.3% | 88.0% | 90.4% | 86.0% |
| AA          | 73.0% | 80.8% | 85.2% | 78.7% |
| AA+RGCN    | 76.4% | 83.6% | 87.3% | 81.3% |
| AA+LGCN    | 76.4% | 83.5% | 87.8% | 80.7% |
| RA          | 80.7% | 85.5% | 87.9% | 80.7% |
| RA+RGCN    | 81.8% | 86.8% | 88.7% | 83.3% |
| RA+LGCN    | 81.9% | 87.5% | 89.3% | 84.7% |
| BiC [62]    | 86.0% | 90.2% | 91.1% | 83.8% |
| BiC+RGCN   | 86.0% | 90.7% | 91.5% | 89.6% |
| BiC+LGCN   | 86.5% | 91.1% | 91.6% | 90.2% |
| GRL [63]    | 84.6% | 90.7% | 90.9% | 89.6% |
| GRL+RGCN   | 85.0% | 90.9% | 91.2% | 90.3% |
| GRL+LGCN   | 85.2% | 91.0% | 91.4% | 90.2% |
| SINet [64]  | 86.1% | 90.6% | 93.8% | 90.0% |
| SINet+RGCN | 86.3% | 90.6% | 93.6% | 90.5% |
| SINet+LGCN | 86.9% | 90.8% | 94.5% | 90.6% |

1) Comparison With the Appearance Models: In this section, we compare our method with multiple appearance models, including the average pooling (AP), attention aggregation (AA), RNN aggregation (RA), BiCnet-TKS [62], GRL [63] and SINet [64]. The equations of the first three aggregators, i.e., AP, AA and RA, are presented in Section III-B, while BiCnet-TKS (BiC for short), GRL and SINet are the most recent state-of-the-arts (SOTA) appearance models for video-based ReID. We implement the RGCN and LGCN with one graph convolutional layer.

We report the experimental results in Table III. The experimental results show that the proposed pose feature learning works well on some appearance models with basic aggregators (AP, AA and RA) and can improve their performance by more than 1%. For example, for the AP baseline, the pose feature improve mAP and rank1 by 2.8% and 4.0% on iLIDS-VID. On the most recent SOTA models, the pose feature learning can achieve about 0.5% improvement. We can conclude that the pose feature can gain a significant improvement on mediocre appearance-based baselines, and gain a slight improvement on excellent baselines. This also demonstrates the limitation of pedestrian appearance and the effectiveness of the pose feature learning in video-based ReID.

2) Comparison With Other Temporal GCN Models: In this section, we compare our RGCN and LGCN model with other temporal GCN models, including GCN&RNN pipeline, GCN&LSTM pipeline, ST-GCN [13] and EvolveGCN [18]. The GCN&RNN pipeline is illustrated in Eq. (4) and Eq. (5), while the GCN&LSTM pipeline could be implemented by combining Eq. (4) and LSTM model. ST-GCN [13] is a popular framework for temporal skeleton feature learning. EvolveGCN [18] is recently proposed for dynamic graph representation learning.

We use two Tesla V100 GPU to train Mars, and only use one to train iLIDS-VID. The experimental results, training time and the number of parameters in the pose feature learning module on Mars and iLIDS-VID are reported in Table IV and Table V, respectively. As can be seen, even though the GCN&RNN (GCN&LSTM) could achieve similar performance than RGCN (LGCN), it costs more time for model training and contains more training parameters: the GCN&RNN pipeline has 0.19 more million parameters than...
RGCN model, while the GCN&LSTM pipeline has 0.27 more million parameters than LGCN model. We conclude that the proposed entire graph message passing mechanism in pose feature learning can effectively reduce the model size and complexity.

Compare to ST-GCN [13] and EvolveGCN [18], the proposed LGCN outperforms them, while RGCN achieves similar performance. ST-GCN [13] constructs a spatial-temporal graph, in which each node connects its temporal and spatial neighborhoods. The temporal information only is propagated within temporally adjacent nodes. EvolveGCN [18] employs a recurrent model to evolve and learn weights of GCN model, rather than node features. We notice that both ST-GCN and EvolveGCN did not model long-term temporal correlation of nodes. This is an important and critical step in sequence tasks.

For a deeper analysis, in Table VI, we report Davies-Bouldin Index (DBI) [65] indicator to evaluate inter-class variance of learned features. DBI is calculated as the average similarity of each cluster with its most similar cluster, and a smaller DBI indicates better performance. We conclude from Table VI that LGCN outperforms ST-GCN and EvolveGCN.

3) Validation of the Dual-Attention Method: In this section, we validate the effectiveness of the dual-attention method (DAM). Specifically, we compare our DAM with mean pooling, time-attention method (TAM) and node-attention method (NAM): mean pooling takes the mean value of the node embeddings as the pose feature; TAM and NAM obtain graph representation with time-level attention and node-level attention, respectively. As shown in Table VII, the attention method can improve the performance of the mean pooling, while our DAM outperforms TAM and NAM.

Furthermore, we explore the fusion modes of feature $h^{(t)}$ learned by time attention and $h^{(n)}$ learned by node attention. Specifically, we compare three fusion modes: concatenation, sum and adaptive sum. For the last one, we implement a fully-connected layer to adaptively learn the fusion weights $w_{adp} = FC(h^{(t)} || h^{(n)})$, then pose feature $f^p$ can be computed by $f^p = \text{sigmoid}(w_{adp}) \odot h^{(t)} + (1 - \text{sigmoid}(w_{adp})) \odot h^{(n)}$, where $\odot$ denotes the element-wise multiplication. We report the comparison in Table VIII. As can be seen, the adaptive sum-based fusion could achieve similar performance to concatenation-based, but the sum-based one obtains worse results.

Discussion on the effectiveness of DAM. DAM utilizes the attention mechanism to learn pedestrian pose feature, in which the node attention learns the importance of skeleton joints, and time-attention learns the importance of frames. By node attention, the important joints, such as joints in wrist and ankle, can have larger weights when aggregating node features. By time attention, the visible frames can have larger aggregating weights than the occluded ones. This distinguishes the unimportant information from important information and avoids the shortcomings of mean pooling.

4) Impact of Graph Convolutional Layers: In this section, we test the performance of RGCN and LGCN with different graph convolutional (GC) layers. Specifically, we choose the average pooling (AP) as the baseline model and implement RGCN and LGCN with one to four GC layers. The experimental results on Mars and iLIDS-VID are reported in Fig. 5 and Fig. 6, respectively. As can be seen, the accuracy decreases as the number of GC layers increases. In our method, each pose graph only contains 14 nodes, which would result the over-smoothing [66], [67] after multiple GC layers.

---

**Table VIII**

| Method             | Mars |             | iLIDS-VID |             |
|--------------------|------|-------------|-----------|-------------|
|                    | mAP  | Rank1       | mAP       | Rank1       |
| RGCN+concatenation | 82.6%| 87.9%       | 89.4%     | 84.7%       |
| RGCN+sum           | 81.9%| 87.0%       | 88.3%     | 83.5%       |
| RGCN+adaptive sum  | 82.4%| 88.0%       | 89.3%     | 84.0%       |
| LGCN+concatenation | 82.3%| 88.0%       | 90.4%     | 86.0%       |
| LGCN+sum           | 81.5%| 87.3%       | 90.0%     | 85.2%       |
| LGCN+adaptive sum  | 82.2%| 88.1%       | 90.5%     | 85.7%       |
5) Impact of the Sequence Length: Limited by the GPU memory, we input a video clip with a specific length $T$ for each iteration. In this section, we explore the retrieval accuracy under different $T$. Specifically, we verify four values for $T$: 4, 6, 8 and 10. We choose the average pooling (AP) as the baseline model and implement RGCN and LGCN with this setting. The retrieval results on Mars and iLIDS-VID are presented in Fig. 7 and Fig. 8, respectively. As can be seen, the overall tendency is that the larger $T$ contributes to the higher accuracy.

6) Validation of the Adaptive Loss: Eq. (19) defines an adaptive loss by a changeable $\lambda$. In this section, we aim to compare the performance of the adaptive $\lambda$ and fixed $\lambda$. We test two fixed values for $\lambda$, i.e., 0.5 and 1.0. Their respective performances are reported in Table IX. As can be seen, the adaptive $\lambda$ outperforms the fixed $\lambda$ on both Mars and iLIDS-VID. Moreover, we visualize the pose triplet loss $L_{tri}^p$ during training iLIDS-VID [21] in Fig. 9. The conclusion is that the loss defined by adaptive $\lambda$ could converge to a smaller value compared with fixed $\lambda$.

7) Discussion: In this section, we first explore the formulation of pedestrian features from the appearance feature and pose feature. Our method in Fig. 3 take the concatenation of appearance feature and pose feature as the final pedestrian feature. In this section, we explore two other fusion methods: sum and adaptive sum. Specifically, sum adds the appearance feature and pose feature to formulate the final pedestrian feature. Adaptive sum first computes a fusion weights $w_{adp}$ from the appearance feature $f^a$ and pose feature $f^p$ by $w_{adp} = FC(f^a||f^p)$, then adaptively fuses $f^a$ and $f^p$ by $f = \text{sigmoid}(w_{adp}) \odot f^a + (1 - \text{sigmoid}(w_{adp})) \odot f^p$, where $\odot$ denotes the element-wise multiplication. We report the results in Table X. As can be seen, the fusion method concatenation outperforms sum by a large margin, and slightly outperforms adaptive sum.

Next, we test the performance of single-modality feature (appearance feature or pose feature) and multi-modality feature.

---

**Table IX**

| Method | $\lambda$ | Mars | iLIDS-VID |
|--------|-----------|------|-----------|
|        |           | mAP  | Rank1     | mAP  | Rank1     |
| RGCN   | Adaptive  | 82.6%| 87.9%     | 89.4%| 84.7%     |
|        | 0.5       | 82.2%| 86.6%     | 88.7%| 84.0%     |
|        | 1.0       | 82.1%| 86.8%     | 87.5%| 82.0%     |
| LGCN   | Adaptive  | 82.3%| 88.0%     | 90.4%| 86.0%     |
|        | 0.5       | 82.0%| 87.1%     | 89.5%| 84.7%     |
|        | 1.0       | 81.7%| 87.3%     | 89.3%| 84.3%     |

**Table X**

| Method               | Mars | iLIDS-VID |
|----------------------|------|-----------|
|                      | mAP  | Rank1     |
|                      | mAP  | Rank1     |
| Concatenation        | 82.6%| 87.9%     |
| Sum                  | 81.3%| 85.9%     |
| Adaptive sum         | 82.3%| 86.5%     |

**Table XI**

| Method               | Mars | iLIDS-VID |
|----------------------|------|-----------|
|                      | mAP  | Rank1     |
|                      | mAP  | Rank1     |
| Appearance           | 81.1%| 85.2%     |
| Pose                 | 44.4%| 55.8%     |
| Appearance & Pose    | 82.6%| 87.9%     |

---

Fig. 8. Performance of RGCN and LGCN with different video lengths $T$ on iLIDS-VID [21]. Numbers shown in the figure are percentages.

Fig. 9. The pose triplet loss $L_{tri}^p$ under the adaptive $\lambda$ and fixed values (0.5 and 1.0) on iLIDS-VID [21]. The abscissa and ordinate indicate the training epoch and loss value, respectively.

Fig. 10. Visualization of the attention scores of each frame and each node learned by DAM.
feature (appearance feature and pose feature), and present the results in Table XI. The performance of only using the pose feature is much worse than that of appearance feature and multi-modality feature. And models fusing data from different modalities outperform their unimodal counterparts. The appearance feature involves the body shape, clothing and other appearance characteristic. While the pose feature depicts the walking patterns of pedestrian. The multi-modality features are heterogeneous and complementary, thus are superior than the single-modality features. This conclusion is also demonstrated in other multi-modal learning tasks [70], [71].

### C. Comparison and Visualization

1) **Comparison With the State-of-the-Arts:** In this section, we compare our method with the state-of-the-art (SOTA) methods, including STRF [69], GRL [63], CTL [48], SSN3D [11], TCLNet [68], CPA [72], and so on. We choose BiCnet-TKS [62] as our baseline appearance model. As shown in Table XII, our method achieves the competitive results on these three datasets.

2) **Visualization:** In this section, we visualize the retrieval results from Mars [17] in Fig. 11. As can be seen, for the AP, the video clips of the second and third columns have the similar appearances with the query video, which results the false positive results. While for RGCN, even though the video clips of the third and fourth columns have the different appearances and camera views with the query video, it still achieves the true positive results. The similar phenomenon is shown in LGCN. Thereby, we could draw the conclusion that the learned pose feature can effectively enlarge the inter-class variance for pedestrians with similar appearances.

Moreover, we visualize the learned attention scores by DAM in Fig. 10. As can be seen, the visible keypoints are more important than occluded ones; scores of visible frames are higher than those of occluded ones. Moreover, we found that the time-attention could learn more discriminative scores compared with the node-attention.

### V. Conclusion

In this paper, we proposed to learn the pose feature to remedy the shortcomings of the appearance feature for the
video-based ReID. To this end, we implemented a two-branch architecture to separately learn the pose feature and appearance feature, and we concatenated them together for video retrieval. To learn the pose feature, we constructed a temporal pose graph via an off-the-shelf pose detector, whose nodes and edges denote the pose keypoints and skeleton connections, respectively. We then implemented a RGCN model by combining the recurrent model and GCN to learn the node embeddings of the temporal graph, which could alleviate the influence from local node occlusion via the global information propagation mechanism. In RGCN, the GCN model and recurrent model partly shared weights to achieve entire graph message passing. Finally, we employed the self-attention mechanism to obtain the temporal graph representation, where the node-attention and time-attention are leveraged to evaluate the importance of each node and each frame, respectively. We tested our method on the basis of multiple appearance models, and the experimental results demonstrated the superiority of the learned pose feature: it can improve the performance of multiple appearance-based baselines.

REFERENCES

[1] Z. Wang et al., “Robust video-based person re-identification by hierarchical mining,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8179–8191, Dec. 2022.
[2] C. Chen, M. Ye, M. Qi, J. Wu, Y. Liu, and J. Jiang, “Saliency and granularity: Discovering temporal coherence for video-based person re-identification,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 6100–6112, Sep. 2022.
[3] N. McLaughlin, J. M. Del Rincón, and P. Miller, “Recurrent convolutional network for video-based person re-identification,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* (CVPR), Jun. 2016, pp. 1325–1334.
[4] Y. Yan, B. Ni, Z. Song, C. Ma, Y. Yan, and X. Yang, “Person re-identification via recurrent feature aggregation,” in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 701–716.
[5] X. Li, W. Zhou, Y. Zhou, and H. Li, “Relation-guided spatial-temporal attention network for video-based person re-identification,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 11434–11441.
[6] Y. Fu, X. Wang, Y. Wei, and T. Huang, “STA: Spatial–temporal attention for large-scale video-based person re-identification,” in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 33, 2019, pp. 8287–8294.
[7] X. Zhu, J. Liu, H. Wu, M. Wang, and Z.-J. Zha, “ASTA-Net: Adaptive spatio-temporal attention network for person re-identification in videos,” in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 1706–1715.
[8] J. Li, S. Zhang, and T. Huang, “Multi-scale 3D convolution network for video based person re-identification,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8618–8625.
[9] X. Gu, H. Chang, B. Ma, H. Zhang, and X. Chen, “Appearance-preserving 3D convolution for video-based person re-identification,” in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2020, pp. 228–243.
[10] L. Zheng, Y. Huang, H. Lu, and Y. Yang, “Pose-invariant embedding for deep person re-identification,” *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4500–4509, Sep. 2019.
[11] X. Jiang, Y. Qiao, and J. Yan, “SSN3D: Self-separated network to align parts for 3D convolutional video person re-identification,” in *Proc. IEEE Conf. Artif. Intell.*, vol. 35, no. 2, 2021, pp. 1691–1699.
[12] C. Su, J. Li, S. Zhang, J. Xing, W. Gao, and Q. Tian, “Pose-driven deep convolutional model for person re-identification,” in *Proc. IEEE Int. Conf. Comput. Vis.* (ICCV), Oct. 2017, pp. 3960–3969.
[13] M. Jiang, J. Dong, D. Ma, J. Sun, J. He, and L. Lang, “Inception spatial temporal graph convolutional networks for skeleton-based action recognition,” in *Proc. Int. Symp. Control Eng. Robot. (ISCER)*, Feb. 2022, pp. 208–213.
[14] L. Shi, Y. Zhang, J. Cheng, and H. Lu, “Skeleton-based action recognition with directed graph neural networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7912–7921.
[15] 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.
[16] M. Van and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, no. 11, 2008.
[17] L. Zheng et al., “MARS: A video benchmark for large-scale person re-identification,” in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 868–884.
[18] A. Pareja et al., “EvolveGCN: Evolving graph convolutional networks for dynamic graphs,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 5363–5370.
[19] X. Wang and R. Zhao, “Person re-identification: System design and evaluation overview,” in *Person Re-Identification*, 2014, pp. 351–370.
[20] S. Wu, Y.-C. Chen, X. Li, A.-C. Wu, J.-J. You, and W.-S. Zheng, “An enhanced deep feature representation for person re-identification,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–8.
[21] T. Wang, S. Gong, X. Zhu, and S. Wang, “Person re-identification by video ranking,” in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 698–703.
[22] H. Xian, X. Liu, Y. Bian, and B. Fan, “Person re-identification with elastic loss for person re-identification,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 1, pp. 160–171, Jan. 2022.
[23] D. Cheng, J. Zhou, N. Wang, and X. Gao, “Hybrid dynamic contrast and probability distillation for unsupervised person Re-Id,” *IEEE Trans. Image Process.*, vol. 31, pp. 3334–3346, 2022.
[24] D. Zhang, D. Meng, and J. Han, “Co-saliency detection via a self-paced multi-instance learning framework,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017.
[25] H. Pan, Y. Chen, Z. He, F. Meng, and N. Fan, “TCDesc: Learning topology consistent descriptors for image matching,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 5, pp. 2845–2855, May 2022.
[26] D. Cheng, Y. Gong, X. Chang, W. Shi, A. Hauptmann, and N. Zheng, “Deep feature learning via structured graph Laplacian embedding for person re-identification,” *Pattern Recognit.*, vol. 82, pp. 94–104, Oct. 2018.
[27] A. Hermans, L. Beyer, and B. Leibe, “In defense of the triplet loss for person re-identification,” 2017, arXiv:1703.07737.
[28] W. Wang, W. Pei, Q. Cao, S. Liu, G. Lu, and Y. Tai, “Push for center learning via orthogonalization and subspace masking for person re-identification,” *IEEE Trans. Image Process.*, vol. 30, pp. 907–920, 2021.
[29] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, “Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline),” in *Proc. Europ. Conf. Comput. Vis.*, 2018, pp. 480–496.
[30] Y. Yan et al., “Learning multi-granular hypergraphs for video-based person re-identification,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2899–2908.
[31] X. Qian et al., “Pose-normalized image generation for person re-identification,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 650–667.
[32] S. Gao, J. Wang, H. Lu, and Z. Liu, “Pose-guided visible part matching for occluded person ReID,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11434–11451.
[33] I. Goodfellow et al., “Generative adversarial nets,” in *Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–11.
[34] Z. Chen et al., “Frame-guided region-aligned representation for video person re-identification,” in *Proc. AAAI Conf. Artif. Intell.* (AAAI), vol. 34, no. 7, 2020, pp. 10591–10598.
[35] H. Pan, Y. Chen, and Z. He, “Multi-granularity graph pooling for video-based person re-identification,” *Neural Netw.*, vol. 160, pp. 22–33, 2023.
[36] C. Huang et al., “Self-supervised attentive generative adversarial networks for video anomaly detection,” *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 5, 2022, doi: 10.1109/TNNLS.2022.3159538.
[37] S. Bai, B. Ma, H. Chang, R. Huang, S. Shan, and X. Chen, “SANet: Statistical attention network for video-based person re-identification,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 6, pp. 3866–3879, Jun. 2022.
[41] T. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in Proc. Int. Conf. Learn. Represent., 2017, pp. 1–14.

[42] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs,” in Proc. Int. Conf. Learn. Represent., 2014, pp. 1–14.

[44] H. Pan, Y. Bai, Z. He, and C. Zhang, “AAGCN: Adjacency-aware graph convolutional network for person re-identification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 3289–3299.

[45] J. Yang, W. Zheng, Q. Yang, Y.C. Chen, and Q. Tian, “Spatial–temporal graph convolutional network for video-based person re-identification,” in Proc. IEEE Trans. Neural Inf. Process. Syst., 2016, pp. 3844–3852.

[46] G. Wang, Y. Yuan, X. Chen, J. Li, and X. Zhou, “Learning discriminative features with multiple granularities for person re-identification,” in Proc. 26th ACM Int. Conf. Multimedia, Oct. 2018, pp. 274–282.

[47] Z. Zhang, H. Zhang, and S. Liu, “Person re-identification using heterogeneous local graph attention networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 12136–12145.

[48] J. Liu, Z.-J. Zha, W. Wu, K. Zheng, and Q. Sun, “Spatial–temporal correlation and topology learning for person re-identification in videos,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 4370–4379.

[49] Y. Wu et al., “Adaptive graph representation learning for video person re-identification,” IEEE Trans. Image Process., vol. 29, pp. 8821–8830, 2020.

[50] L. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, “A new representation of skeleton sequences for 3D action recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3288–3297.

[51] M. Liu, H. Liu, and C. Chen, “Enhanced skeleton visualization for view invariant human action recognition,” Pattern Recognit., vol. 68, pp. 346–362, Aug. 2017.

[52] P. Zhang et al., “Adding attentiveness to the neurons in recurrent neural networks,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 135–151.

[53] C. Si, Y. Jing, W. Wang, L. Wang, and T. Tan, “Skeleton-based action recognition with spatial reasoning and temporal stack learning,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 103–118.

[54] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, “View adaptive recurrent neural networks for high performance human action recognition from skeleton data,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2117–2126.

[55] H. Pan, Y. Chen, T. Xu, Y. He, and Z. He, “Toward complete-view and high-level pose-based gait recognition,” IEEE Trans. Inf. Forensics Security, vol. 18, pp. 2104–2118, 2023.

[56] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, “Skeleton-based action recognition with shift graph convolutional network,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 183–192.

[57] W. Kay et al., “The kinetics human action video dataset,” 2017, arXiv:1705.06959.

[58] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[59] Z. Cao, G. Hidalgo, T. Simon, S. 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.

[60] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang, “Bag of tricks and a strong baseline for deep person re-identification,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2019, pp. 1487–1495.

[61] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[62] R. Hou, H. Chang, B. Ma, R. Huang, and S. Shan, “BiCnet-TKS: Learning efficient spatial–temporal representation for video person re-identification,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 2014–2023.
Yunqi He is currently pursuing the degree with the School of Computer Science, Northeast Forestry University, Harbin, China. His research interests include computer vision and machine learning.

Yuan Zheng received the bachelor’s and master’s degrees from the Harbin Institute of Technology (HIT) in 2006 and 2008, respectively and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2014. She was a Postdoctoral Researcher with the School of Computer Science, Shenzhen Graduate School, Harbin Institute of Technology, China, from 2014 to 2017. She is currently an Associate Professor with the College of Computer Science, Inner Mongolia University, China. Her main research interests include deep learning, computer vision, and pattern recognition.

Feng Zheng received the Ph.D. degree from the University of Sheffield, U.K. He is currently an Associate Professor with the Southern University of Science and Technology (SUSTech), China. He has published 100 academic papers in international top journals and conferences, including IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Intelligent Transportation Systems, IEEE Transactions on Image Processing, AAAI, NeurIPS, CVPR, and ECCV. His research interests include machine learning, computer vision, and cross-media intelligence. He is a member of the program committee for several top international AI conferences, including ICLR, ICML, AAAI, IJCAI, NeurIPS, KDD, and UAI. He is a reviewer of several leading journals, including IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Multimedia, and IS. He serves as the Local Co-Chair for IEEE ICME and IEEE IJCB and the Area Chair for ACM MM 2020 and 2021. He serves as the Associate Editor for IET Image Processing.

Zhenyu He (Senior Member, IEEE) received the Ph.D. degree from the Department of Computer Science, Hong Kong Baptist University, Hong Kong, in 2007. From 2007 to 2009, he was a Postdoctoral Researcher with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology. He is currently a Full Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His research interests include machine learning, computer vision, image processing, and pattern recognition.