results:

- Existing PAR algorithms adopt CNN as the backbone network to extract the feature representation of input images which learns the local features well. As is known to all, global relations in the pixel-level space are also very important for fine-grained attribute recognition. Some researchers resort to the Transformer network to capture such global information [7], [8], however, their models support the image-based attribute recognition only.
- Chen et al. [16] formulate the video-based pedestrian attribute recognition as a multi-task classification problem and try to learn a mapping from a given video to attributes. The attribute labels are transformed into binary vectors for network optimization. However, the high-level semantic information is greatly missing which is very important for pedestrian attribute recognition.

Therefore, it is natural to raise the following questions: *how to design a novel video-based pedestrian attribute recognition framework that simultaneously captures the global features of vision data, and aligns the vision and semantic attribute labels well?*

To answer this question, in this paper, we take the video frames and attribute set as the input and formulate the video-based PAR as a multi-modal fusion problem. As shown in Fig. 2, a novel CLIP-guided Visual-Text Fusion Transformer for Video-based PAR is proposed. To be specific, the video frames are transformed into video tokens using a pre-trained CLIP [17] which is a multimodal foundation model. The attribute set is transformed into corresponding language descriptions using split, expand, and prompt engineering. Then, the text encoder of CLIP is used for the language embedding. After that, we concatenate the video and text tokens and feed them into a fusion Transformer for multi-modal information interaction which mainly contains layer normalization, multi-head attention, and MLP (Multi-Layer Perceptron). The output will be fed into a classification head for pedestrian attribute recognition.

Different from the standard fully fine-tuning strategy, in this work, we propose a novel spatiotemporal side-tuning strategy to optimize the parameters of our framework. As the pre-trained foundation model contains large number of parameters, adjusting all parameters incurs high computational cost. In addition, the multimodal features are already well aligned and hasty fine-tuning may disrupt the original feature space. Specifically, we fix all the parameters of the multimodal foundation model and only optimize the light-weight integrated external side network. We consider both spatial and temporal views of the input pedestrian features and achieve parameter-efficient fine-tuning using our spatiotemporal side-tuning strategy. Extensive experiments on two large-scale video-based PAR datasets demonstrate that our proposed spatiotemporal side tuning strategy performs better on the GPU memory usage, time cost in the inference phase, and F1 score, compared with existing PEFT methods.

To sum up, the main contributions of this paper can be concluded as following three aspects:

1). We propose a novel CLIP-guided Visual-Text Fusion Transformer for Video-based Pedestrian Attribute Recognition, which is the first work to address the video-based PAR from the perspective of visual-text fusion.

2). We introduce the pre-trained big model CLIP as our backbone network, which makes our model robust to the aforementioned challenging factors. A novel spatiotemporal side-tuning strategy is specifically designed to optimize our PAR framework efficiently.

3). Extensive experiments on two large-scale video-based PAR datasets fully validated the effectiveness of our proposed video PAR framework and the parameter-efficient fine-tuning method.

This paper is an extension of our previous work which was published in CVPR workshop@NFVLR2023 [1]. The key extensions can be summarized as the following two aspects: **1). New Siding Tuning Strategy**: In our conference version, we extract the features of pedestrian videos and attribute text using pre-trained big model CLIP [17] and directly fuse them using a multi-modal Transformer. In this work, we propose a novel siding tuning strategy to extract better spatiotemporal features of pedestrian video. **2). More Experiments**: In this work, we conduct more ablation studies and parameter analyses to better illustrate the effectiveness of our proposed framework. Another benchmark dataset DukeMTMC-VID attribute dataset [18] is also used for comparison with other state-of-the-art video-based pedestrian attribute recognition algorithms.

Organization of this paper: In Section II, we will review the works most related to ours, including Pedestrian Attribute Recognition, Visual-Language Fusion, and Self-attention and Transformers. In Section III, we mainly introduce our framework with a focus on the overview, network architectures, and loss function. Then, we conduct experiments on two widely used video-based pedestrian attribute recognition datasets in Section IV. Finally, in Section V, we conclude this paper and propose possible research directions on the video-based PAR.

II. Related Works

In this section, we will give a brief introduction to Pedestrian Attribute Recognition, Pre-trained Vision-Language Foundation Models, and Parameter Efficient Fine Tuning Methods. More related works can be found in the following survey [1], [9] [2].

A. Pedestrian Attribute Recognition

Current pedestrian attribute recognition can be divided into two main streams, i.e., the RGB frame-based [19]–[24] and video-based PAR [16], [25]. For the RGB frame-based PAR, the early research works mainly analyzed pedestrian attributes from the perspective of multi-label classification [19], [20] using convolutional neural networks (CNN). Specifically, Abdulnabi et al. [19] propose a multi-task learning approach, utilizing multiple CNNs to learn attribute-specific features while sharing knowledge among them. Zhang et al. introduce the PANDA [20], a strategy that integrates a part-aware model with human attribute classification based on CNN. This

[1] https://nfvlr-workshop.github.io/

[2] github.com/wangxiao5791509/Pedestrian-Attribute-Recognition-Paper-List

framework accelerates the training of CNN, enabling it to learn robust normalized features even from smaller datasets. Due to the RNN (Recurrent Neural Networks) effectively modeling the sequential dependencies between human attributes, some researchers exploit the application of RNN on the PAR task. For example, Wang et al. [21] utilize Long Short-Term Memory (LSTM) to establish robust semantic dependencies among labels in pedestrian attribute recognition. By integrating previously predicted labels, the visual features can dynamically adapt to subsequent ones. Zhao et al. propose the GRL [22] to exploit the potential dependencies between pedestrian attributes by considering intra-group attribute mutual exclusion and inter-group attribute association. The graph neural network (GNN) is also introduced into the PAR task to model the semantic relations of different attributes. Specifically, VC-GCN [23] and A-AOG [24] represent attribute correlations through conditional random fields and graphical models. Li et al. [23] take pedestrian attribute recognition as an attribute sequence prediction problem, which utilizes GNN as a basic layer for the whole framework to model the spatial and semantic relations between pedestrian attributes.

Recently, the Transformer whose core operation is the self-attention mechanism, has drawn more and more attention in the artificial intelligence community. Many PAR works are also developed based on the Transformer network, for example, Fan et al. [26] introduce a PARformer to extract features instead of CNN, which combines global and local perspectives. VTB [2] proposes a novel baseline that treats pedestrian attribute recognition by introducing an additional text encoder, which interacts information respectively.

For the video-based PAR, it is a relatively new research topic compared with image-based methods. To be specific, Chen et al. [16] introduce a novel multi-task model that includes an attention module to pay attention to each frame for each attribute. Specker et al. [25] introduce different information from different frames through global features before temporal pooling. Lee et al. [27] achieve robust pedestrian attribute recognition by selecting unobstructed frames through sparsity-based temporal attention module. Thakare et al. [28] compute the pedestrian features through cross-correlation of attribute prediction of different view of pedestrian images. Liu et al. [29] capture the correlations among different attributes in both spatio and temporal domains through a novel spatio-temporal saliency module. Note that, to better mine the correlations among these patterns, a spatio-temporal attribute relationship learning module is proposed. Different from these models, in this work, we formulate the video-based PAR as a video-language fusion problem and propose to fuse the dual modalities using a pre-trained vision-language foundation model. More importantly, we propose a new spatiotemporal side tuning strategy to achieve parameter-efficient fine-tuning for the large model-based video pedestrian attribute recognition.

B. Pretrained Vision-Language Foundation Models

Inspired by the success of pre-trained large language models (for example, the BERT [30], GPT series [31], [32], LLaMA [33], LaMDA [34], Baichuan-2 [35]) and multi-modal models (e.g., CLIP [17], ALIGN [36], LXMERT [37], ViLBERT [38], SAM [39]), many researchers resort to these big models to improve their performance further. Specifically, the CLIP model [40] is pre-trained on 400 million image-text pairs and aligns the dual modalities in the feature space well. SAM [39] is proposed to address the segmentation problem which takes the image and prompt information (e.g., point, bounding box, text) from humans to achieve accurate and interactive segmentation. Due to the good generalization performance on the downstream tasks, Wang et al. propose the PromptPAR [41] which adopts the CLIP as the backbone and optimizes its parameters using prompt tuning. Jin et al. [42] formulate the attribute recognition as a phrase generation problem and take the pre-trained CLIP model as the visual and text encoder for the input encoding. Some researchers also exploit the human-centric pre-training and validate their model on the pedestrian attribute recognition task, such as Hulk [43], HAP [44], PLIP [45]. Although these works perform well on the frame-based PAR, however, their model can't process the video-based PAR task. In our conference version of this paper [46], we also exploit the pre-trained CLIP model to learn the dual modalities to better align the human vision features and semantic features. To further improve the training and inference efficiency, we fix the pre-trained CLIP model and further introduce a novel spatiotemporal side network to achieve efficient parameter tuning.

C. Parameter Efficient Fine Tuning

As the size of the model continues to grow rapidly, the question of how to effectively fine-tune the parameters in the model becomes common. Parameter Efficient Fine Tuning (PEFT), aims to achieve comparable performance to full parameter tuning by fine-tuning only a small number of parameters when using a large pre-trained model. Previously Prompt Tuning has been widely used in natural language processing, e.g., GPT3 [32], BERT [30], etc., for constructing templates to convert downstream classification or generative tasks into masked language modeling tasks during pre-training, in order to reduce the gap between pre-training and fine-tuning. Later Prompt Tuning took more forms, such as In-Context Learning [47], Instruction-tuning [48], and Chain-of-Thought [49], and was widely used in fine-tuning and reasoning for large-scale models. Jia et al. [50] first proposed Visual Prompt Tuning (VPT) to be introduced into the visual domain to fine-tune the frozen backbone by means of a set of successive learnable vectors. Zhou et al. [51], [52], for the first time, applied Prompt to vision, proposed to use continuous vectors as templates for categories and proposed to use a meta-net to extract instance information in conjunction with Prompt, which dramatically improves CLIP [17] to migrate to downstream tasks and generalize to invisible categories. Adapter [53] was first introduced into the Transformer [7] architecture by Houlsby [54], using the under-projection-activation-up-projection structure embedded in the Transformer layer to fine-tune the whole model. Gao et al. [55] proposed CLIP-Adapter, which mixes the original CLIP

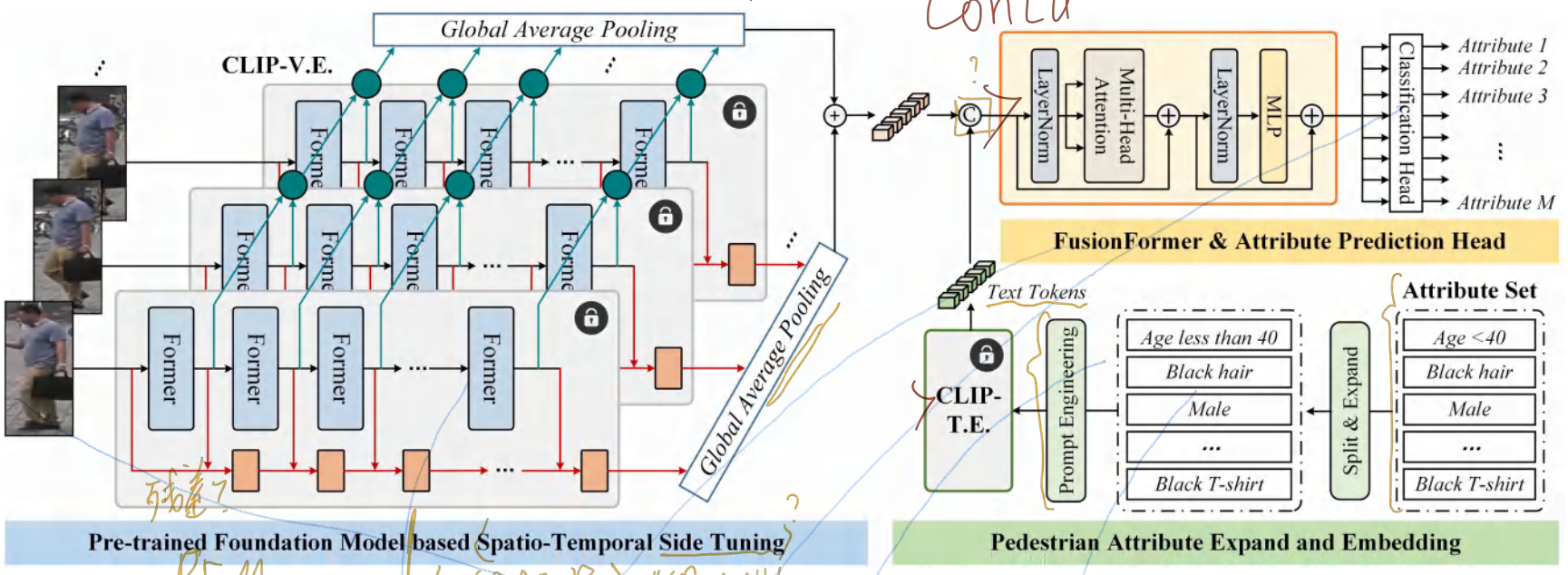

Fig. 2. An illustration of our proposed video-based pedestrian attribute recognition framework, termed VTFPAR++. It formulates video-based attribute recognition as a video-language fusion problem, which takes the pedestrian video and attribute set as the input. The pre-trained multi-modal foundation model CLIP is adopted as the basic feature extraction network. We further propose the lightweight spatiotemporal side network to aggregate the features from different Transformer layers and video frames. These features are fused into a unified representation via global average pooling operators. We process the given attributes into language descriptions via split, expand, and prompt engineering, and extract its features using CLIP text encoder. Then, we align the vision-language features using a fusion Transformer and classify the attributes via the attribute prediction head. Our framework requires lower GPU memory consumption, fewer parameter adjustments, and more efficient model training and deployment, yet still achieves leading attribute recognition accuracy on two public datasets.

knowledge and the knowledge of the few-shot through two linear layers to achieve the outstanding few-shot recognition performance. LoRA [56] divides the fine-tuned pre-trained model into the original weights and the updated weights, and the updated weights are not full-ranked, which can be changed into two low-dimensional matrices A and B by low-rank decomposition, and reduce the number of parameters during fine-tuning by connecting the partial weights to the adjoint matrices formed by A and B. Dou et al. [57] solved the problem of catastrophic forgetting by combining LoRA with MoE [58], which can dynamically generate LoRA based on inputs from different tasks. Zhang et al. [59] added a lightweight side net to the original backbone to fuse with the original output to achieve a fine-tuning effect. Sung et al. [60] argued that the backpropagation of previous PEFT methods all need to go through the backbone, which cannot achieve training efficiency, so they introduced an additional lightweight ViT [8] next to the backbone, which drastically improves the training efficiency. We consider that the video-based pedestrian attribute recognition task requires both fine-grained spatial features and modeling of temporal information over multiple frames, thus, we propose spatial and temporal side networks to augment CLIP spatial information and empower CLIP temporal modeling, respectively.

III. Our Proposed Approach

In this section, we will first give an overview of our proposed framework VTFPAR++, then, we will give more details about this network architecture, including input encoding, spatiotemporal side tuning, video-text fusion Transformer, and attribute prediction head. After that, we will introduce the loss function which is used in the training phase.

A. Overview

As shown in Fig. 2, we formulate the video-based pedestrian attribute recognition as a vision-language fusion problem and propose a novel video-based PAR framework, termed VTFPAR++. Given the input pedestrian video, we first adopt the pre-trained CLIP vision encoder with spatiotemporal side tuning networks to extract the visual features. The spatial and temporal side networks will efficiently extract and aggregate the features obtained from different frames and various layers from the CLIP vision encoder. The spatial and temporal features are obtained via the global average pooling operators and concatenated as the final visual representation of the input video. To help our video PAR model understand the pedestrian attributes, in this work, we first split the attributes into discrete word combinations and expand each attribute into a natural language description via prompt engineering. Then, we adopt the CLIP text encoder to obtain the language representations of the expanded attribute phases. The visual and text features are concatenated as unified tokens and fed into a multi-modal Transformer for fusion. The enhanced features will be further fed into the attribute prediction head for final recognition. More details will be introduced in the following sub-sections.

B. Network Architecture

Our proposed VTFPAR++ contains four main modules, including the CLIP text encoder, CLIP vision encoder with spatiotemporal side tuning network, multi-modal fusion Transformer, and attribute prediction head.

Input Encoding. Give a sequence of pedestrian frames $V \in \mathrm{R}^{T\times H\times W\times C}$, T, H, W, C denotes the number of video frames, height, width, and channel of video frame, respectively, and the attribute set $A = \{a_1, a_2, a_3, ..., a_M\}$, M is the number of human attributes that need to predict, we will

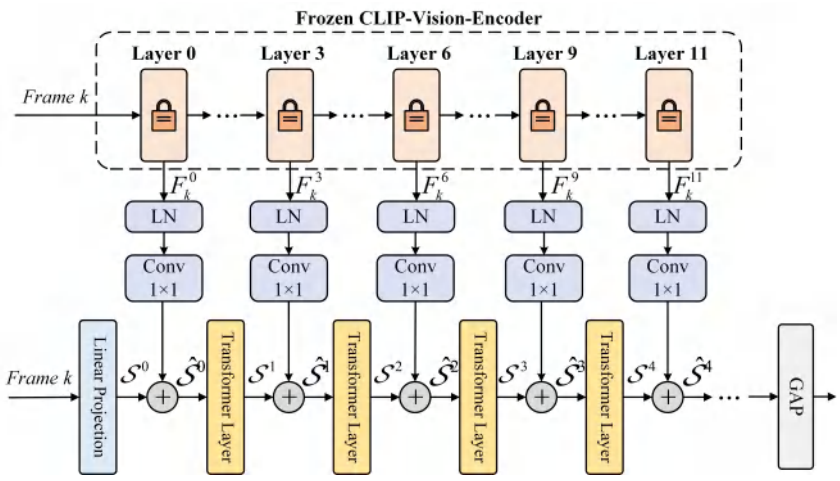

Fig. 3. An illustration of our proposed Spatial Side Network(SSN), SSN models the spatial relationship of multi-level CLIP visual features for each frame separately, and finally the modeling results of multiple frames are interacted with text features after GAP aggregation.

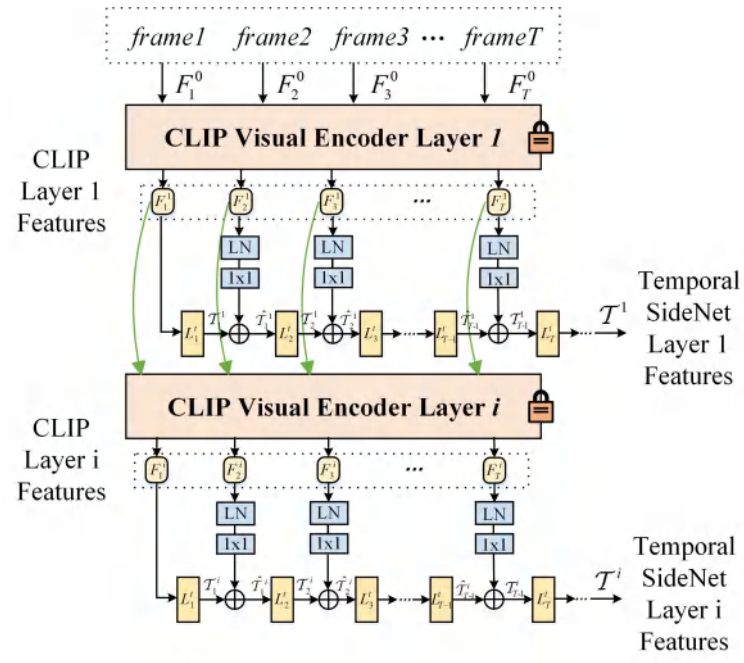

Fig. 4. An illustration of our proposed Temporal Side Network(TSN). TSN primarily models temporal relationships of the same layer of CLIP visual features over multiple frames to mitigate the effects of challenges such as occlusion and blurring.

adopt the pre-trained vision-language foundation model CLIP to extract the multi-modal features. For the video input, we first divide each frame into non-overlapping patches and feed them into a linear projection layer to get the visual tokens, for example, the tokens of k-th video frame can be denoted as $F^k \in \mathrm{R}^{N \times D}$. Then, we fed these video frames patch tokens into the Transformer layers of the CLIP foundation model after concatenate with the classification token. Thus, the visual tokens of the whole video at i-th layer of the CLIP can be represented as $F_v^i = \{F_1^i, F_2^i, F_3^i, ..., F_T^i\}$.

The key operation of the Transformer layer used in the pre-trained CLIP model is the *multi-head self-attention*. The calculation of self-attention in each head can be formulated as:

$$SelfAtten(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d}})V, \quad (1)$$

where the input feature vectors Q, K, V are query, key, and value features, respectively. These three features are transformed from a single input feature using a projection layer. d is the dimension of the input feature vectors. The framework uses similar operations to process both visual and text tokens. Further details will not be elaborated upon.

For the attribute set $A = \{a_1, a_2, a_3, ..., a_M\}$, we also input them into our framework to help the PAR model better understand the attributes it needs to recognize. Specifically, we first split the attribute into phrases, e.g., "Age < 40" is split and expanded into "Age less than 40". Then, this phrase is transformed into a sentence using prompt engineering. Usually, the prompt is a sentence like "The attribute ___ of this pedestrian is _____". By combining the attribute and prompt, we can transform the word/phrase into a language description, such as "The attribute age of this pedestrian is less than 40 ". Then, the pre-trained CLIP text encoder is utilized to extract the semantic attribute representation F_a.

Spatio-Temporal Side Tuning. After we embed the input video and text using the pre-trained CLIP model, we can directly train these networks in a fine-tuning way as we do in our conference paper [46]. However, the feature space of the pre-trained large CLIP model has been aligned with a large-scale image-text dataset. If fine-tuning is done hastily, it may disrupt the original feature space. Additionally, the model has a large number of parameters, so direct fine-tuning can cause a sudden increase in computational overhead. Wang et al. propose the PromptPAR [41] which resorts to prompt tuning for RGB frame-based PAR, however, their method still consumes a large amount of GPU memory due to the prompt tuning need back forward through their framework.

In this paper, we attempt to propose a novel spatiotemporal side tuning strategy to efficiently adjust embedded lightweight networks for video-based PAR. Specifically, the parameters of pre-trained CLIP vision and text encoders are all fixed. We introduce lightweight spatiotemporal side networks along different layers of CLIP vision encoder and different video frames to extract high-quality multi-scale and temporal feature representations, as illustrated in Fig. 2.

For the spatial side tuning network, as shown in Fig. 3, 8 Transformer layers are utilized. In this sub-network, we aggregate the features from the Transformer layers of CLIP vision encoder and the spatial side network via the fusion module (layer normalization layer, convolutional layer with kernel size 1×1). The two features are then added and fed into the subsequent Transformer layer of the spatial side network, the processing of the k-th frame can be formulated as:

$$\hat{S}^j = LN(F_k^i) * \omega_s + S^j \quad (2)$$

where $LN(\cdot)$ is layer normalization, and i is the layer we choose to fuse, j is the spatial side network layer corresponding to i-th CLIP layer. S^j is the feature propagated through the j-th layer of spatial side network. ω_s denotes the learnable weights of the 1×1 convolution. Similar operations are executed for feature fusion of all other different Transformer layers.

For the temporal side tuning network, as illustrated in Fig. 4, after embedding the T video frames into corresponding visual tokens, we first adopt the LN layer and 1×1 convolutional

layer to process the CLIP vision feature. Meanwhile, we feed the embedded visual tokens into temporal side networks and get the output features T.

$$\hat{T}_k^i = LN(F_k^i) * \omega_t + T_k^i \quad (3)$$

$$T_{k+1}^i = L_k^t(\hat{T}_k^i) \quad (4)$$

where k denotes the index of frames, and ω_t is the linear layer to transform the dimension of i-th CLIP visual feature to match the temporal feature. Then, we combine the processed feature from the CLIP vision encoder and our newly proposed temporal side networks in a frame-by-frame manner. Similar operations are also conducted for other CLIP Transformer layers. Through the spatiotemporal side networks, we get the features of temporal and spatial interaction $S = \{S^1, S^2, S^3, \ldots, S^T\}$ and $T = \{T^0, T^3, T^6, T^9, T^{11}\}$. Then the spatial and temporal features will be added after global average pooling and concatenated with the text features to be fed into the multi-modal Transformer.

Our proposed spatiotemporal side network has a similar architecture to the standard CLIP [17] visual backbone ViT-B/16 [8], but is more lightweight. We set the width (i.e., the dimension of feature vector) of the side network to 240, the number of attention heads to 6, the depth to 8, and the patch size to 16. Extensive experiments demonstrate that our proposed side networks can aggregate the spatiotemporal visual features in a more efficient and accurate way.

Video-Text Fusion Transformer. After we get the enhanced spatial and temporal features, we concatenate and feed them into a multi-modal Transformer layer for video-text fusion. A standard Transformer layer is adopted for this module, which contains the layer normalization, multi-head self-attention, and MLP (Multi-Layer Perceptron) layer. The output features will be later fed into the attribute recognition head.

Attribute Prediction Head. After we get the output F_f from the multi-modal Transformer, we utilize the vision tokens for final attribute prediction. A prediction head is proposed to achieve this target which contains k dense layers. The results are further processed by batch normalization BN and Sigmoid function, i.e.,

$$P = \sigma(BN(Dense(F_f))) \quad (5)$$

where σ is Sigmoid function, and $Dense$ denotes the dense layers (also termed fully connected layers).

C. Loss Function

To measure the distance between the predicted results $P = \{p_1, p_2, p_3, ..., p_Z\}$ from the attribute prediction head and the ground truth $Y = \{y_1, y_2, y_3, ..., y_H\}$, we use the weighted cross-entropy loss function which can be formulated as:

$$L = -\frac{1}{M}\sum_{i=1}^{N}\sum_{j=1}^{M} w_j(y_{ij}\log(p_{ij}) + (1-y_{ij})\log(1-p_{ij})) \quad (6)$$

where ω_j considers the imbalanced distributions of each attribute, p_{ij} is the attribute predicted by our model, N, M denotes the number of tracklets, attributes, respectively.

$$\omega_j = \begin{cases} e^{1-r_j}, & y_{ij} = 1 \\ e^{r_j}, & y_{ij} = 0 \end{cases} \quad (7)$$

where r_j is the ratio of positive samples of j-th attribute in the train set.

IV. Experiments

In this section, we will first introduce the datasets and evaluation metrics that we use in subsection IV-A. The implement details will be given in subsection IV-B. After that, we compare our method with other state-of-the-art algorithms in subsection IV-C. Then, we conduct some extended experiments on our newly proposed framework in subsection IV-D. We also provide a detailed analysis of the parameter in subsection IV-E. The visualization will also be provided in subsection IV-F to help the readers better understand our work. We discuss the failed cases and limitation analysis in the subsection IV-G.

A. Dataset and Evaluation Metric

In our experiments, the **MARS-Attribute dataset** [66] and the **DukeMTMC-VID-Attribute dataset** [66] are used [3] which are re-annotated based on MARS [67] and DukeMTMC-VID [68] dataset.

- **MARS-Attribute dataset** is an annotated pedestrian attributes dataset of the MARS [67] dataset by Chen et al. [66]. Five sets of multi-label attributes such as action, pedestrian orientation, color of upper/lower body, age, and nine sets of binary attributes such as gender, hair length, and length of upper/lower garment were annotated. We split the multi-label attribute into binary attributes as well, which means we use 43 binary attributes for training and testing. The training subset contains 8,298 sequences from 625 different ID pedestrians and the testing subset contains 8,062 sequences corresponding to 626 pedestrians. For each sequence, there are 60 frames on average, from which we randomly selected 6 frames for training and testing.
- **DukeMTMC-VID-Attribute dataset** is also annotated pedestrian attributes dataset of the DukeMTMC-VID [68] dataset by Chen et al. [66]. There are four sets of multi-label attributes, such as motion, pedestrian pose, and color of upper/lower body, and eight sets of binary attributes such as backpack, shoes, and boots were annotated. We use 37 binary attributes for training and testing by splitting the multi-label attribute into binary attributes. The training subset contains 702 different ID pedestrians and 16522 images and the testing subset contains 17661 images corresponding to 702 pedestrians. For each sequence, there are 169 frames on average, from which we randomly selected 6 frames for training and testing.

For the evaluation of our and the compared PAR models, we adopt the widely used Accuracy, Precision, Recall, and F1-score as the evaluation metric. Note that, the results reported

[3] https://irip.buaa.edu.cn/mars_duke_attributes/index.html