

# Multi-direction and Multi-scale Pyramid in Transformer for Video-based Pedestrian Retrieval

Xianghao Zang, Ge Li, and Wei Gao

**Abstract**—In video surveillance, pedestrian retrieval (also called person re-identification) is a critical task. This task aims to retrieve the pedestrian of interest from non-overlapping cameras. Recently, transformer-based models have achieved significant progress for this task. However, these models still suffer from ignoring fine-grained, part-informed information. This paper proposes a multi-direction and multi-scale Pyramid in Transformer (PiT) to solve this problem. In transformer-based architecture, each pedestrian image is split into many patches. Then, these patches are fed to transformer layers to obtain the feature representation of this image. To explore the fine-grained information, this paper proposes to apply vertical division and horizontal division on these patches to generate different-direction human parts. These parts provide more fine-grained information. To fuse multi-scale feature representation, this paper presents a pyramid structure containing global-level information and many pieces of local-level information from different scales. The feature pyramids of all the pedestrian images from the same video are fused to form the final multi-direction and multi-scale feature representation. Experimental results on two challenging video-based benchmarks, MARS and iLIDS-VID, show the proposed PiT achieves state-of-the-art performance. Extensive ablation studies demonstrate the superiority of the proposed pyramid structure. The code is available at <https://git.openi.org.cn/zangxh/PiT.git>.

**Index Terms**—video-based pedestrian retrieval, vision transformer, multi-direction and multi-scale pyramid.

## I. INTRODUCTION

**P**EDESTRIAN retrieval is a critical task in intelligent surveillance [1] [2] [3]. Given a pedestrian image as the query, pedestrian retrieval aims to find the right images in a large gallery. The query image and the matched gallery images must be from different cameras. Pedestrian retrieval has important practical applications in both society and industry, such as finding the criminal suspects and tracking pedestrian movement. Compared to the image-based pedestrian retrieval, the video-based one can provide much more gait and view information of pedestrians and alleviate the negative effects of occlusion situations. Therefore, video-based pedestrian retrieval is getting more and more attention from researchers [4].

For CNN-based pedestrian retrieval, dividing the feature map into multiple horizontal stripes and individually training each one are general operations. After the training is complete,

This work was supported by the National Key R&D Program of China (2020AAA0103501). (Corresponding author: Wei Gao.)

Xianghao Zang, Ge Li and Wei Gao are with School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China (e-mail: zangxh@pku.edu.cn; geli@ece.pku.edu.cn; gaowei262@pku.edu.cn).

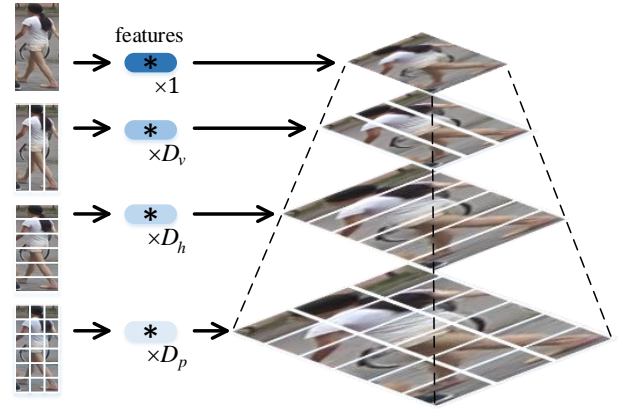


Fig. 1: Multi-direction and multi-scale pyramid in transformer for video-based pedestrian retrieval. Different layers employ different-direction division strategies. After the process of transformer layer, part-informed features are extracted. Each layer contains features with different scales. The four layers have 1,  $D_v$ ,  $D_h$ , and  $D_p$  features, respectively.

all the stripe features are assembled to generate the convolution descriptor for each image, which provides the model a rich feature representation [5]. Based on the horizontal division strategy, a pyramid of stripes is proposed to exploit the partial information of each pedestrian [6]. Although these methods above improve the model performance, the direction of division strategy is limited.

Recently, the transformer structure has achieved incredible progress in computer vision. The transformer structure is a popular model in natural language processing (NLP) and can handle the sequence data effectively. In computer vision, the input image is split into many patches. These patches are regarded as tokens similar to the words in the NLP task. A sequence of feature embedding of these patches is fed to the transformer layers. With the help of the multi-head self-attention module, the transformer can obtain global-level relationships among all the patches without losing information. Meanwhile, the CNN convolution kernel can only perceive limited scope, and the down-sampling operation inevitably loses much information. Moreover, the transformer structure has achieved competitive performance compared with CNN [7]. Although the transformer has a global perception, the fine-grained information may be neglected, which results in a limited performance.

This paper proposes a multi-direction and multi-scale Pyra-

mid in Transformer (PiT) for video pedestrian retrieval, as illustrated in Fig. 1. The pyramid contains four layers, which adopt “no division”, vertical, horizontal, and patch-based division strategies, respectively. The first layer includes a global-level feature of the pedestrian image. The second, third, and fourth layers contain  $D_v$ ,  $D_h$ , and  $D_p$  part-level features. In this way, the proposed PiT applies multi-direction division strategies in the pedestrian image and extracts a multi-scale feature representation for each pedestrian image.

Concretely, each pedestrian image is split into many patches. A class token and the feature embeddings of all patches are flattened and fed to multiple transformer layers. Then the processed patch tokens are rearranged into a two-dimension structure according to their original positions. Different division strategies are applied to this two-dimension structure, which generates different-direction parts. The class token and patch tokens within the same part are flattened to form a new token sequence. After the process of the last transformer layer, the class token learns the multi-direction part-informed information. A feature pyramid for each pedestrian image is obtained by combining all the features with different scales from different layers. The corresponding features of all the images within the same video are fused to generate the final multi-direction and multi-scale feature pyramid.

The traditional CNN-based methods usually apply horizontal division to feature map [5], which is reasonable because each horizontal stripe usually contains the head, torso, or legs. However, the vertical division can divide the human body into the right limb, head and torso, and the left limb, which introduces part-informed clues with more dimensions. Applying vertical and horizontal division simultaneously, which forms the patch-based division, can also provide more fine-grained information. Combining these multi-direction and multi-scale features can effectively improve the model performance. Experiments on two challenging video-based benchmarks, MARS [8] and iLIDS-VID [9], show the proposed PiT achieves state-of-the-art performance. Extensive ablation studies also demonstrate the superiority of the proposed pyramid structure.

The main contributions of this paper can be summarized as follows:

- **Multi-direction:** the proposed vertical and horizontal division strategies in transformer introduce fine-grained, part-informed information from different directions.
- **Multi-scale:** the global and local-level features with different scales form a feature pyramid. This multi-scale combination makes the feature representation rich and discriminative.
- **Performance:** the proposed PiT achieves state-of-the-art performance on two challenging video-based benchmarks, and extensive ablation studies demonstrate the superiority of the proposed multi-direction and multi-scale pyramid structure.

The rest of this paper is organized as follows: the related works are reviewed and analyzed in Section II, and then the proposed method is introduced in Section III. Experimental results and analysis are presented in Section IV, and Section V concludes this paper.

## II. RELATED WORK

### A. Video-based Pedestrian Retrieval

An early method for image-based pedestrian retrieval fused various features, such as RGB, HSV, HoG, LOMO, etc., to achieve the multi-feature fusion and overcome the challenge from pedestrian appearance changes [10]. Whereas video-based pedestrian retrieval has multiple images for each pedestrian. These consecutive pedestrian images can provide abundant temporal and spatial information, which can alleviate the negative effects of appearance change, occlusion, pose variation, etc [11]. Therefore, existing methods focus on exploiting both spatial and temporal clues from pedestrian video. GRL [12] employ video-level features to guide the generation of correlation map and disentangle the frame-level features into high-correlation and low-correlation features. BiCnet-TKS [13] introduced a bilateral complementary network to mine the divergent body parts of each pedestrian and proposed a temporal kernel selection module to explore temporal relations adaptively. CTL [14] employed a key-point estimator to extract multi-scale semantic features to form a topology graph. Then a 3D graph convolution is used to capture hierarchical spatial and temporal dependencies. Besides, AGW+ [15] employed a frame-level average pooling for video feature representation, which is simple but effective.

These methods above mainly utilized CNN to extract spatial and temporal clues. However, the CNN convolution kernel cannot capture long-range relationships, and the CNN down-sampling operation results in inevitable information loss.

### B. Vision Transformer

Recently, transformer architecture has become a de-facto standard for natural language processing. ViT [7] introduced this architecture to computer vision and achieved better performance than many state-of-the-art methods on the image classification task. Following this improvement, many works were proposed to improve the performance of the transformer-based framework. For the video-based classification task, Swin transformer [16] proposed a hierarchical structure and utilized shifted windows to solve the non-overlapping patch division problem. A 3D shifted window is also proposed to preprocess video data.

These methods above demonstrated the transformer structure could perform well for the video classification task. However, the video-based pedestrian retrieval is very different from the video classification task. The video-based pedestrian retrieval task depends highly on appearance information rather than motion information. Therefore, the transformer structure with the ability to perceive more fine-grained information is needed for the video-based pedestrian retrieval task.

### C. Division Strategies for Pedestrian Retrieval

For the pedestrian retrieval task, dividing the feature map into multiple stripes is a typical operation. PCB [5] proposed to horizontally divide the feature map into multiple stripes and achieved significant performance improvement compared with the original model. SPP [17] divided the feature map into

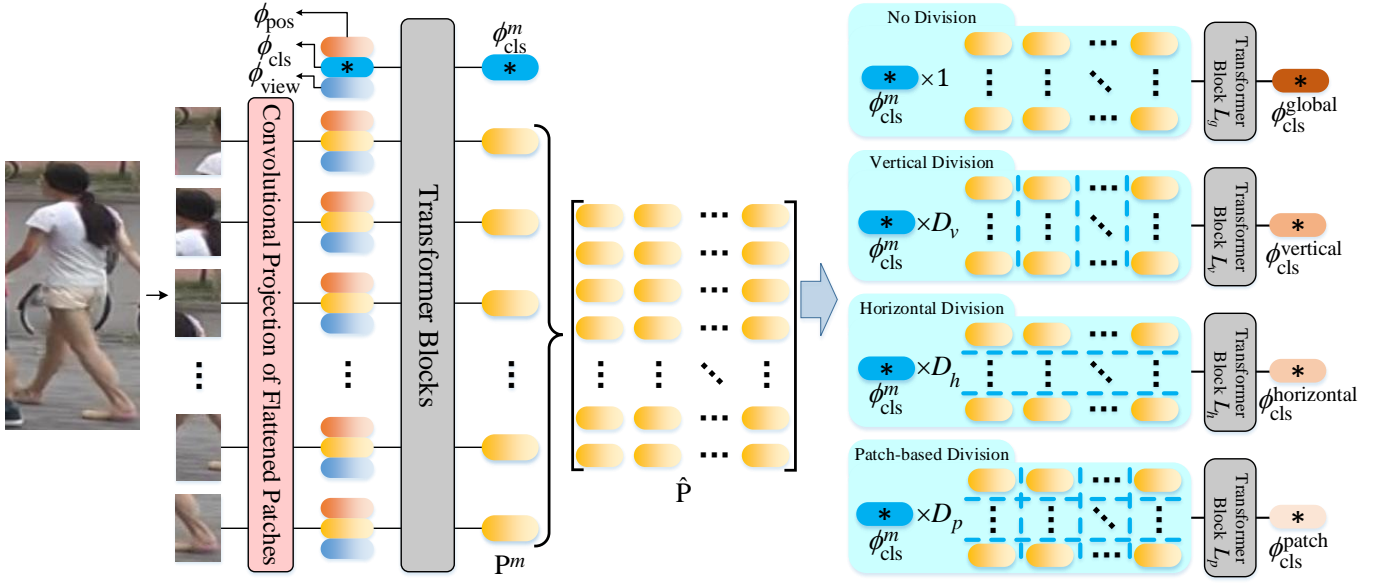


Fig. 2: The proposed feature Pyramid in Transformer (PiT) for each pedestrian image. The feature pyramid with four layers contains multi-direction and multi-scale feature representations.

equal patches and assembled the multi-scale feature patches into a pyramid structure. HPP [6] employed the horizontal division to form a multi-scale pyramid and improved the model performance for this task.

Although the models above utilized different division strategies, performance comparison among different division strategies in a unified framework has not been conducted, thus the differences between these strategies remain to be explored.

### III. MULTI-DIRECTION AND MULTI-SCALE PYRAMID IN TRANSFORMER

#### A. Transformer-based Framework

The Vision Transformer (ViT) [7] is employed to construct the framework, as illustrated in Fig. 2. Given some pedestrian videos  $\{V_1, V_2, \dots\}$  and pedestrian IDs  $\{y_1, y_2, \dots\}$ , each video  $V$  contains  $K$  pedestrian images  $I$ , as  $V = \{I_1, I_2, \dots, I_K\}$ . A convolution layer is used to embed the pedestrian image into multiple feature embeddings. Concretely, after applying convolution operation on the pedestrian image, a feature map  $f \in \mathbb{R}^{h \times w \times c}$  is obtained. Then the feature map is flattened to generate  $N$  features, where  $N = h \cdot w$  and the size of each feature is  $1 \times c$ . In this way, each feature can be treated as the feature embedding of each image patch, and the size of each image patch is the same as convolution kernel size  $k$ . The convolution stride  $s$  determines the interval of the adjacent image patches.

The feature embedding of each image patch from the image  $I$  is also called patch token  $p$ . A class token  $\phi_{cls}$  with the size of  $1 \times c$  is also introduced to represent the feature embedding of the whole image. After flattening class and patch tokens into a sequence  $\{\phi_{cls}; p_1; \dots; p_N\}$ , the patch tokens lose the location information in original pedestrian image. Therefore, a position embedding  $\phi_{pos} \in \mathbb{R}^{(N+1) \times c}$  is employed to retain this location information. A camera embedding  $\hat{\phi}_{view}$  is also

introduced to keep the camera information. Since the class token and all the patch tokens belong to the same camera, the size of  $\hat{\phi}_{view}$  is set to  $1 \times c$ . Then  $\hat{\phi}_{view}$  is copied  $N + 1$  times to form a new embedding  $\phi_{view} \in \mathbb{R}^{(N+1) \times c}$ . After these operations, the token sequence  $z^0$  from the pedestrian image  $I$  is calculated as follows,

$$\begin{aligned} z^0 &= [\phi_{cls}; p_1; \dots; p_N] + \lambda_1 \phi_{pos} + \lambda_2 \phi_{view} \\ &= [\phi_{cls}^0; p_1^0; \dots; p_N^0], \end{aligned} \quad (1)$$

where  $\lambda_1$  and  $\lambda_2$  are trade-off parameters. Then  $m$  transformer layers are employed to learn the relationship between the class token and all patch tokens. Each transformer layer is composed of a Multi-head Self-Attention (MSA) block and a Multi-Layer Perception (MLP) block. A LayerNorm (LN) layer is applied before MSA and MLP blocks, and a shortcut connection is also employed as follows,

$$\begin{aligned} z' &= z^{d-1} + \text{MSA}(\text{LN}(z^{d-1})), \\ z^d &= z' + \text{MLP}(\text{LN}(z')). \end{aligned} \quad (2)$$

After  $m$  transformer layers, the vector sequence  $z^m$  is obtained.

$$z^m = [\phi_{cls}^m; p_1^m; p_2^m; \dots; p_N^m] := [\phi_{cls}^m; P^m]. \quad (3)$$

In Eq. 3,  $N$  patch tokens  $\{p_i^m\}_{i=1}^N$  are denoted as  $P^m$ ,

$$P^m = [p_1^m; p_2^m; \dots; p_N^m]. \quad (4)$$

#### B. Multi-direction and Multi-scale Pyramid

To explore the fine-grained, part-informed information, the patch tokens  $P^m$  are rearranged into a new form  $\hat{P}$  according



to their original positions in pedestrian image  $I$ ,

$$\underbrace{[p_1^m; p_2^m; \dots; p_N^m]}_{P^m \in \mathbb{R}^{N \times c}} \xrightarrow{\text{rearrange}} \underbrace{\begin{bmatrix} p_1^m & p_2^m & \dots & p_w^m \\ p_{w+1}^m & p_{w+2}^m & \dots & p_{2w}^m \\ \vdots & \vdots & \ddots & \vdots \\ \dots & \dots & \dots & p_N^m \end{bmatrix}}_{\hat{P} \in \mathbb{R}^{h \times w \times c}}. \quad (5)$$

The rearranged patch tokens  $\hat{P}$  have the same size as the feature map  $f \in \mathbb{R}^{h \times w \times c}$ . We learn from the division strategies in Convolution Neural Network (CNN) and apply similar strategies to patch tokens  $\hat{P}$ .

1) **Multi-direction Division Strategies:** The rearranged tokens  $\hat{P}$  are copied four times. Multi-direction division strategies are applied to these four copies.

For the first copy, “no division” is applied to the tokens  $\hat{P}$ . Then the class token  $\phi_{\text{cls}}^m$  along with  $\hat{P}$  are flattened to a new token sequence  $z_{\text{global}} \in \mathbb{R}^{(N+1) \times c}$ , which equals  $z^m$ .

$$z_{\text{global}} = [\phi_{\text{cls}}^m; p_1^m; p_2^m; \dots; p_N^m] \Leftrightarrow z^m. \quad (6)$$

The transformer layer  $L_g$  receives this sequence and outputs a new sequence  $z'_{\text{global}} \in \mathbb{R}^{(N+1) \times c}$ . We follow the general operation in [7] [16], which discard all the patch tokens and only keep the class token for the following operations. Therefore, only the class token  $\phi_{\text{cls}}^{\text{global}} \in \mathbb{R}^{1 \times c}$  is kept as the feature representation of the whole pedestrian image  $I$ .

For the second copy, “vertical division” is applied to the patch tokens  $\hat{P}$  along the vertical direction to generate  $D_v$  parts. Each part has  $N/D_v$  patch tokens. The class token is copied  $D_v$  times, and each one is assigned to one part. Then the class token and all the patch tokens in the corresponding part are flattened along the vertical direction to form a new vector sequence  $z_{\text{vertical}} \in \mathbb{R}^{(N/D_v+1) \times c}$ . There are  $D_v$  sequences in total. For example, the first sequence  $z_{\text{vertical},1}$  is shown as follows,

$$z_{\text{vertical},1} = [\phi_{\text{cls}}^m; p_1^m; p_{w+1}^m; p_{2w+1}^m; \dots]. \quad (7)$$

Each sequence  $z_{\text{vertical},i}$  is followed by the parameter-sharing transformer layer  $L_v$ . In this way, the relationship between the class token and a specific vertical part is explored. After the process of the transformer layer  $L_v$ ,  $D_v$  class tokens  $\{\phi_{\text{cls},i}^{\text{vertical}}\}_{i=1}^{D_v}$  are kept and denoted as  $\phi_{\text{cls}}^{\text{vertical}}$ . Moreover, this paper first proposes the “vertical division” strategy, which extracts fine-grained feature representation and also introduces significant performance improvement.

For the third copy, “horizontal division” is applied by dividing the patch tokens  $\hat{P}$  into  $D_h$  parts. Horizontal division strategy is applied along the horizontal direction, and each part has  $N/D_h$  patch tokens. After the division operation, the class token is copied  $D_h$  times. Each one and its corresponding patch tokens are flattened along the horizontal direction to form the token sequence  $z_{\text{horizontal}} \in \mathbb{R}^{(N/D_h+1) \times c}$ . The first sequence  $z_{\text{horizontal},1}$  is shown below as an example,

$$z_{\text{horizontal},1} = [\phi_{\text{cls}}^m; p_1^m; p_2^m; p_3^m; \dots; p_{N/D_h}^m]. \quad (8)$$

All the sequences  $\{z_{\text{horizontal},i}\}_{i=1}^{D_h}$  are followed by the parameter-sharing transformer layers  $L_h$ , and  $D_h$  class tokens

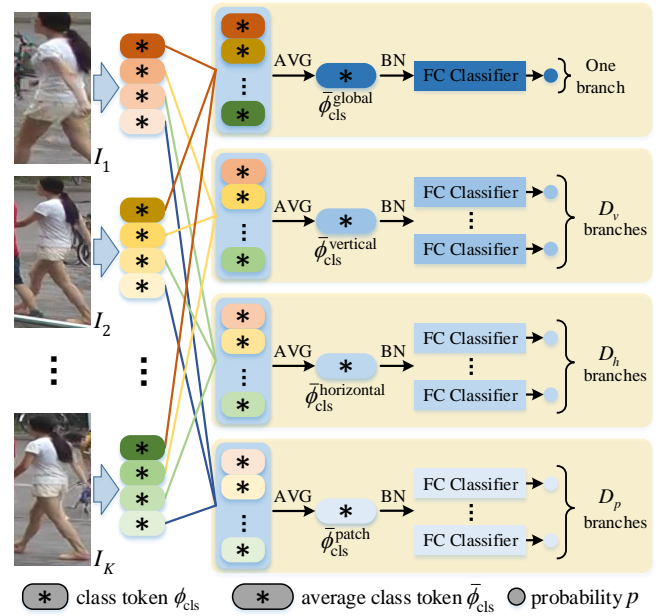


Fig. 3: The generation and training process of the feature pyramid for each video. The corresponding features of each pedestrian image are averaged to generate the final feature, and each feature is trained individually.

$\{\phi_{\text{cls},i}^{\text{horizontal}}\}_{i=1}^{D_h}$  are obtained and denoted as  $\phi_{\text{cls}}^{\text{horizontal}}$ . The “horizontal division” is proposed for CNN in previous work [6]. However, we propose a manner to apply this strategy to the new structure, *i.e.*, vision transformer, which is proven to be effective.

For the fourth copy, the vertical and horizontal division strategies are applied simultaneously to the rearranged patch tokens  $\hat{P}$  to form “patch-based division”. Each part has  $N/D_p$  patch tokens and has a more fine-grained receptive field. After the division operation, the class token and all patch tokens in the corresponding part are flattened along the horizontal direction to form  $D_p$  token sequences  $z_{\text{patch}} \in \mathbb{R}^{(N/D_p+1) \times c}$ , where  $D_p = D_v \times D_h$ . The first sequence  $z_{\text{patch},1}$  is shown below as an example,

$$z_{\text{patch},1} = [\phi_{\text{cls}}^m; p_1^m; p_2^m; \dots; p_{N/D_v}^m; p_{w+1}^m; p_{w+2}^m; \dots]. \quad (9)$$

Each token sequence  $z_{\text{patch},i}$  is followed by the transformer layers  $L_p$ , which generate  $D_p$  class tokens  $\{\phi_{\text{cls},i}^{\text{patch}}\}_{i=1}^{D_p}$ . These class tokens are denoted as  $\phi_{\text{cls}}^{\text{patch}}$ . A similar patch-based division strategy was proposed in CNN structure [17]. However, this paper proposes multi-direction division strategies, which are very different from the previous model.

After the transformer layer, each class token has the perception within its corresponding global/vertical/horizontal/patch-based part, making it obtain more fine-grained local information.

2) **Multi-scale Pyramid Structure:** The rearranged  $\hat{P}$  is divided using different scales, which generates multi-scale feature representations. These features are concatenated to form a pyramid structure  $z_{\phi} \in \mathbb{R}^{(1+D_v+D_h+D_p) \times c}$  for each

pedestrian image  $I$  as follows,

$$z_\phi = [\phi_{\text{cls}}^{\text{global}}; \phi_{\text{cls}}^{\text{vertical}}; \phi_{\text{cls}}^{\text{horizontal}}; \phi_{\text{cls}}^{\text{patch}}], \quad (10)$$

where  $\phi_{\text{cls}}^{\text{global}}$ ,  $\phi_{\text{cls}}^{\text{vertical}}$ ,  $\phi_{\text{cls}}^{\text{horizontal}}$ , and  $\phi_{\text{cls}}^{\text{patch}}$  are in the space of  $\mathbb{R}^{1 \times c}$ ,  $\mathbb{R}^{D_v \times c}$ ,  $\mathbb{R}^{D_h \times c}$ , and  $\mathbb{R}^{D_p \times c}$ , respectively.

There are  $K$  images in each pedestrian video  $V$ , and each image is represented by  $z_\phi$ . The feature pyramid  $\bar{z}_\phi$  of the video  $V$  is generated as illustrated in Fig. 3. Each feature of the video  $V$  is the corresponding average feature within it. For example, the class token  $\bar{\phi}_{\text{cls}}^{\text{global}}$  is calculated as follows,

$$\bar{\phi}_{\text{cls}}^{\text{global}} = \sum_{k=1}^K \phi_{\text{cls},k}^{\text{global}}. \quad (11)$$

The feature pyramid  $\bar{z}_\phi \in \mathbb{R}^{(1+D_v+D_h+D_p) \times c}$  of video  $V$  is expressed as follow,

$$\bar{z}_\phi = [\bar{\phi}_{\text{cls}}^{\text{global}}; \bar{\phi}_{\text{cls}}^{\text{vertical}}; \bar{\phi}_{\text{cls}}^{\text{horizontal}}; \bar{\phi}_{\text{cls}}^{\text{patch}}]. \quad (12)$$

The CNN-based methods usually design various fusion strategies to combine pedestrian image features within the same video. The feature map from CNN contains rich spatial and temporal information, and a sophisticated fusion strategy can effectively combine these features. However, in a transformer-based framework, the class token is not generated from the input image directly. In other words, these class tokens do not contain spatial and temporal information explicitly. Therefore, each feature of the video  $V$  is obtained in an average manner as Eq. 11.

The multi-scale pyramid structure  $\bar{z}_\phi \in \mathbb{R}^{(1+D_v+D_h+D_p) \times c}$  contains part-informed information of different scales, which is more rich and discriminative than the original global-level feature representation. The ablation study in the latter section demonstrates the superiority of this multi-scale pyramid structure.

### C. Model Training and Testing

This section describes the details of model training and testing. To simplify the expression, the class token  $\phi_{\text{cls}}^\theta \in \mathbb{R}^{1 \times c}$  is used to represent each element in  $\bar{z}_\phi$ . Each class token  $\phi_{\text{cls}}^\theta$  is followed by a BatchNorm layer and a classifier layer to generate the final probability  $p^\theta$ , as illustrated in Fig. 3. There are  $1+D_v+D_h+D_p$  classifier layers to train each token  $\phi_{\text{cls}}^\theta$  individually. The classification loss  $\mathcal{L}_{\text{cls}}$  and triplet loss  $\mathcal{L}_{\text{tri}}$  are used to supervise this model by averaging the losses of these independent branches.

$$\mathcal{L}_{\text{cls}} = -\frac{1}{TN_c} \sum_{i=1}^T \sum_{j=1}^{N_c} y_j \log p_{i,j}^\theta, \quad (13)$$

$$\mathcal{L}_{\text{tri}} = \frac{1}{BT} \sum_{i=1}^T \sum_{a \in b_i} \ln\{1 + \exp[\max d(\phi_{i,a}^\theta, \phi_{i,p}^\theta) - \min d(\phi_{i,a}^\theta, \phi_{i,n}^\theta)]\}, \quad (14)$$

where  $T=1+D_v+D_h+D_p$ ,  $N_c$  is the class number for a specific benchmark,  $b_i$  represents the  $i^{\text{th}}$  mini-batch,  $B$  is the number of pedestrian images in this mini-batch,  $a, p, n$  are anchor, positive, negative samples, respectively. Function  $d(\cdot)$

calculates the Euclidean distance between two features. The overall loss function  $\mathcal{L}$  is calculated as follows,

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{tri}}. \quad (15)$$

In the testing process, the multi-direction and multi-scale feature pyramid  $\bar{z}_\phi \in \mathbb{R}^{(1+D_v+D_h+D_p) \times c}$  is used to represent the feature representation of pedestrian video  $V$ .

## IV. EXPERIMENTS

### A. Experimental Setting

1) *Benchmarks*: The proposed Pyramid in Transformer (PiT) is evaluated in two challenging benchmarks: MARS [8] and iLIDS-VID [9]. There are also two other popular benchmarks: DukeMTMC-VideoReID and PRID2011. Existing methods [12] [13] [34] have achieved more than 0.95 in terms of Rank-1 metric on these two benchmarks. However, there is still much room for improvement on the challenging MARS and iLIDS-VID benchmarks.

- MARS is the largest video-based pedestrian retrieval benchmark and captured by six cameras on a university campus. It contains 20,478 videos from 1,261 identities. These videos are generated by employing the DPM detector and GMMCP tracker, which results in many videos with poor qualities.
- iLIDS-VID is captured by two cameras in an airport hall. It contains 600 videos from 300 identities. This benchmark is very challenging due to pervasive background clutter, mutual occlusions, and lighting variations.

2) *Evaluation Protocol and Metrics*: For MARS benchmark, the standard training and testing split provided by [8] is used for training the proposed PiT. The Cumulative Matching Characteristic (CMC) curve and mean Average Precision (mAP) are employed for evaluation. For iLIDS-VID benchmark, the whole set of videos is randomly divided into two halves. Then the trials are repeated ten times, and the CMC curve is used to evaluate the average results. For convenience, Rank-1, Rank-5, Rank-10, and Rank-20 are employed to represent the CMC curve.

3) *Implementation Details*: The proposed PiT is implemented using Pytorch. The transformer ViT-B16 [7], pre-trained on ImageNet, is employed as the backbone. We follow the operation in [35] to preprocess all the videos. Specifically, each video is divided equally into  $K$  snippets, where  $K$  equals 8. The first pedestrian image in each snippet is selected as the keyframe, and  $K$  images are used to represent this video. Each pedestrian image is resized to  $256 \times 128$ . The batch size and parameter  $m$  are set to 16 and 11. The kernel size  $k$  and stride  $s$  of the convolution layer are set to 16 and 12. The dimension  $h \times w \times c$  of feature embedding outputted by convolution layer is  $21 \times 10 \times 768$ . The trade-off parameters  $\lambda_1$  and  $\lambda_2$  are set to 1.0 and 1.5. The division parameters  $D_v$ ,  $D_h$ ,  $D_p$  are 2, 3, and 6. The standard Stochastic Gradient Descent (SGD) with momentum and an initial learning rate of 0.01 is used to train these models 120 epochs for each benchmark. Cosine annealing is employed to schedule the learning rate. The convolution layer and transformer layers are frozen in the first five epochs to train the classifier layers. After these five epochs, the whole network is trained.

TABLE I: Performance comparisons between proposed PiT and state-of-the-art methods on MARS and iLIDS-VID.

Methods	Venue	MARS				iLIDS-VID			
		Rank-1	Rank-5	Rank-10	mAP	Rank-1	Rank-5	Rank-10	Rank-20
ADFD [18]	CVPR2019	87.00	95.40		78.20	86.30	97.40		99.70
GLTR [19]	ICCV2019	87.02	95.76		78.47	86.00	98.00		
COSAM [20]	ICCV2019	84.90	95.50		79.90	79.60	95.30		
AGW+ [15]	TPAMI2020	87.60			83.00	83.20	98.30		
RTF [21]	AAAI2020	87.10			85.20	87.70			
FGRA [22]	AAAI2020	87.30	96.00		81.20	88.00	96.70	98.00	99.30
RGSATR [23]	AAAI2020	89.40	96.90		84.00	86.00	98.00		99.40
AMEM [24]	AAAI2020	86.70	94.00		79.30	87.20	97.70		99.50
CSTNet [25]	IJCAI2020	90.20	96.80		83.90	87.80	98.50		99.60
ASTA-Net [26]	ACM MM2020	90.40	97.00		84.10	88.10	98.60		
MG-RAFA [27]	CVPR2020	88.80	97.00		85.90	88.60	98.00		99.70
MGH [28]	CVPR2020	90.00	96.70		85.80	85.60	97.10		99.50
STGCN [29]	CVPR2020	89.95	96.41		83.70				
VRSTC [30]	CVPR2020	88.50	96.50	97.40	82.30	83.40	95.50	97.70	99.50
TCLNet-tri* [31]	ECCV2020	89.80			85.10	86.60			
AP3D [32]	ECCV2020	90.70			85.60	88.70			
AFA [33]	ECCV2020	90.20	96.60		82.90	88.50	96.80		99.70
SSN3D [34]	AAAI2021	90.10	96.60	98.00	86.20	88.90	97.30		98.80
GRL [12]	CVPR2021	91.00	96.70		84.80	90.40	98.30		99.80
BiCnet-TKS [13]	CVPR2021	90.20			86.00				
CTL [14]	CVPR2021	<b>91.40</b>	96.80		86.70	89.70	97.00		100.00
Proposed PiT		90.22	<b>97.23</b>	<b>98.04</b>	<b>86.80</b>	<b>92.07</b>	<b>98.93</b>	<b>99.80</b>	<b>100.00</b>

<sup>1</sup> The best results are in bold.

### B. Comparison with State-of-the-art Methods

Table I shows the comparison between the proposed PiT and twenty other state-of-the-art methods in terms of mAP score and CMC accuracy. These state-of-the-art methods are all within three years and employed ResNet50 as their backbone to explore the spatial and temporal information among pedestrian images. They used attribute information [18] [24], attention mechanism [20] [27], graph convolution [28] [29] [14], 3D convolution [32] [34], relation-guided models [22] [23] [12], Generative Adversarial Networks (GAN) [30], and new network architectures [15] [21] [25] [26] [31] [33] [19] [13], respectively, to generate the feature representation of each pedestrian video. Meanwhile, the proposed PiT employs a transformer-based framework and utilizes the simple average fusion to obtain the multi-direction and multi-scale feature pyramid.

1) *Performances on MARS*: Compared with other state-of-the-art methods, the proposed PiT achieves the best mAP score and competitive CMC accuracy. The best competitor, CTL [14], utilized a key-points estimator to extract human body local features as graph nodes and achieved topology learning for video-based pedestrian retrieval. In comparison, the proposed PiT does not explore the relationship among different pedestrian images within the same video and reaches a better mAP value. This demonstrates the proposed feature pyramid containing more fine-grained local information has a better generalization performance.

2) *Performances on iLIDS-VID*: Compared with other methods, the proposed PiT achieves state-of-the-art performance. The best competitor, GRL [12], used global correlation estimation to disentangle features into high-correlation and low-correlation features. Then GRL proposed temporal reciprocating learning to enhance the high-correlation semantic clues and accumulate the low-correlation sub-critical clues

TABLE II: Performance comparisons between different-direction division strategies.

One layer	Parameter	MARS		iLIDS-VID
		Rank-1	mAP	Rank-1
No Division (Baseline)	1×210	87.33	84.00	89.87
Vertical Division	105×2	88.26	85.07	89.73
	70×3	88.64	85.72	90.60
	42×5	88.80	85.56	<b>90.93</b>
	35×6	89.08	85.96	90.40
	30×7	<b>89.78</b>	<b>85.99</b>	89.93
Horizontal Division	2×105	88.26	85.33	90.07
	3×70	89.35	85.86	90.20
	5×42	<b>89.46</b>	<b>86.24</b>	<b>91.40</b>
	6×35	89.18	85.94	90.73
	7×30	89.35	86.00	90.67
Patch-based Division	6p	89.13	86.01	91.00
	14p	89.24	86.11	<b>91.67</b>
	15p	<b>89.73</b>	<b>86.17</b>	90.80

for the final feature representation. Meanwhile, the proposed PiT has a concise network structure and achieves 1.67% performance improvement on the metric of Rank-1 than GRL.

### C. Ablation Study

Different division strategies in a unified framework are compared in this section. In this paper, each image is split into 210 patches. They are then further divided into 2, 3, 5, 6, 7 parts. In other words, the optional values of parameters  $D_v$  and  $D_h$  are 2, 3, 5, 6, 7. To explicitly show the division details, 105×2, 70×3, 42×5, 35×6, and 30×7 are used to represent the division parameters using vertical division. 2×105, 3×70, 5×42, 6×35, and 7×30 are denoted as division parameters using horizontal division. Applying 105×2 vertical division and 3×70 horizontal division simultaneously forms a patch-based division and generates 6 parts. 105×2 and 7×30 generate 14 parts, and 42×5 and 3×70 generate 15 parts (*i.e.*, the

TABLE III: Performance comparisons between different combinations.

	Type	Parameter	MARS		iLIDS-VID
			Rank-1	mAP	Rank-1
Four Layers	Vertical Division	$1 \times 210_{105} \times 2_{42} \times 5_{30} \times 7$	89.34	86.30	91.13
	Horizontal Division	$1 \times 210_{2} \times 105_{5} \times 42_{7} \times 30$	89.56	86.32	91.40
	Patch-based Division	$1 \times 210_{6p_{14p_{15p}}$	89.51	85.96	90.47
	Proposed PiT	$1 \times 210_{105} \times 2_{3} \times 70_{6p}$	<b>90.22</b>	<b>86.80</b>	<b>92.07</b>

optional values of parameter  $D_p$  are 6, 14, 15). These patch-based division strategies are denoted as 6p, 14p, 15p. “No division” is denoted as  $1 \times 210$ .

1) *Effectiveness of Different-direction Division Strategies:*

This section compares different-direction division strategies in the transformer-based framework. The proposed pyramid with only one layer is used to compare the performances, as illustrated in Table II. Compared with the baseline method, this table shows that using different-direction division strategies improves the performance. Two conclusions can be made. First, although horizontal division is a commonly used strategy, the vertical and patch-based division strategies can also improve performance effectively. Second, the best division strategy and the number of parts are different for different benchmarks. These conclusions demonstrate the need to adopt a strategy based on the practical scene.

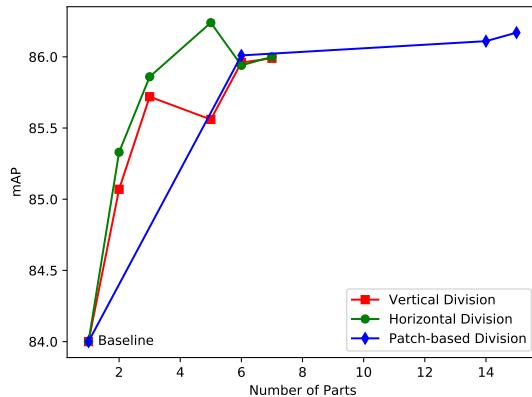


Fig. 4: The mAP score changes of different-direction division strategies on MARS benchmark. The values in this figure follow the data in Table II.

To allow for the performance analysis visually, the mAP scores of different division strategies on MARS benchmark are illustrated in Fig. 4. In this figure, increasing the number of parts can improve the performance gradually. However, more number of parts also introduces more computation complexity. For different division strategies, dividing the patch tokens into six parts achieves a better trade-off between the performance and computation complexity. Therefore, the parameter of proposed PiT is  $1 \times 210_{105} \times 2_{3} \times 70_{6p}$ , and its fourth layer splits the patch tokens into six parts.

2) *Effectiveness of Multi-scale Pyramid Structure:*

This section shows the performances of pyramid structures with different layers, as illustrated in Table IV. In this table, the proposed pyramid with one layer only employs “no division”. The proposed pyramid with two layers additionally uses the vertical division strategy. Then the horizontal division and

TABLE IV: Performance comparisons between the proposed pyramid with different numbers of layers.

Number of Layers	Parameter	MARS		iLIDS-VID
		Rank-1	mAP	Rank-1
1	$1 \times 210$	87.33	84.00	89.87
2	$1 \times 210_{105} \times 2$	88.59	85.65	90.20
3	$1 \times 210_{105} \times 2_{3} \times 70$	88.75	85.72	91.13
4	$1 \times 210_{105} \times 2_{3} \times 70_{6p}$	<b>90.22</b>	<b>86.80</b>	<b>92.07</b>

patch-based division strategies are added one by one. Each layer contains part-informed information with different scales. As the number of layers increases, the performance improves gradually. These improvements demonstrate that fusing multi-scale feature representations can improve performance effectively.

3) *Effectiveness of Multi-direction Pyramid Structure:*

This section compares performances between multi-direction and single-direction pyramid structures, as illustrated in Table III. All the pyramids in this table have four layers, and the differences between them are dependent upon which division strategy is employed. The type “Vertical Division” only employs vertical division strategies. The types “Horizontal Division” and “Patch-based Division” have the same meanings. For vertical and horizontal division strategies, the part numbers 2, 5, 7 are chosen to form the bottom three layers. For patch-based division, 6p, 14p, 15p are employed to form the bottom three layers.

Compared with other single-direction pyramids, the proposed PiT achieves the best performance. These comparisons demonstrate fusing multi-direction division strategies provides more improvement. On the other side, the type “Horizontal Division” performs better than the “Vertical Division”. This shows the horizontal division strategy is more suitable for the pedestrian retrieval task. In conclusion, the proposed PiT fusing multi-direction and multi-scale feature representations is the best combination.

TABLE V: Performance comparisons between proposed PiT with different parameters.

	Parameter	MARS		iLIDS-VID
		Rank-1	mAP	Rank-1
Four Layers	$1 \times 210_{105} \times 2_{3} \times 70_{6p}$	<b>90.22</b>	<b>86.80</b>	<b>92.07</b>
	$1 \times 210_{105} \times 2_{7} \times 30_{14p}$	89.24	86.13	91.20
	$1 \times 210_{42} \times 5_{3} \times 70_{15p}$	89.24	85.82	90.40

4) *Performances of Proposed Pyramid with Different Parameters:*

The proposed PiT contains four layers, and each layer adopts different division strategies. The performances of the proposed PiT with different parameters are presented in Table V. For example, the parameter  $1 \times 210_{105} \times 2_{3} \times 70_{6p}$



TABLE VI: Computation complexity and running time of the proposed PiT with different parameters.

Number of Layers	Parameter	MACs	Trainable Parameters	Using One 24G NVIDIA TITAN RTX GPU			
				MARS		iLIDS-VID (10 trials)	
				Running Time	Rank-1	Running Time	Rank-1
1	1×210	18.05G	85.76M	4.25h	87.33	8.00h	89.87
2	1×210_105×2	19.56G	93.09M	4.45h	88.59	8.75h	90.20
3	1×210_105×2_3×70	21.06G	100.53M	4.70h	88.75	9.50h	91.13
4	1×210_105×2_3×70_6p	22.59G	108.32M	5.00h	90.22	10.70h	92.07

represents “no division” (1×210), vertical division (105×2), horizontal division (3×70), and patch-based division (6p) are employed.

As illustrated in Table V, the proposed PiT with parameter 1×210\_105×2\_3×70\_6p achieves the best performance. The other two pyramids introduce more fine-grained parts, yet their performances get poor. On the other side, the pyramid with the parameter 1×210\_105×2\_7×30\_14p splits the patch tokens into more horizontal parts than using the parameter 1×210\_42×5\_3×70\_15p and achieves better performance. Although their fourth layers have a close number of parts, more horizontal parts introduce more performance improvement.

5) *Qualitative Analysis*: The retrieval examples are illustrated in Fig. 6. One pedestrian image is selected to represent the video for convenience, and the top eight retrieval results for each query are illustrated in this figure. We select the query pedestrian video from MARS benchmark according to the Average Precision (AP) value. Fig. 6(a)(b)(c)(d) show the successful cases where the proposed PiT has a better AP than the baseline method, and Fig. 6(e)(f) show the failed cases where the baseline method performs better.

For the successful case in Fig. 6(a)(b), the proposed PiT retrieves many correct videos in the gallery. In contrast, the baseline method retrieves many incorrect results, including a man in a blue shirt in Fig. 6(a) and a road sign in the second and third places in Fig. 6(b). For the failed case in Fig. 6(d), the baseline method puts the correct videos in the top two places. However, the proposed PiT also retrieves pedestrians with similar appearances. In Fig. 6, we employ the AP value to determine whether the proposed method is successful or not. For practical application, people usually look for the person of interest from the top-k results, not just the top-1 result. Therefore, the proposed PiT can be utilized effectively for Fig. 6(e)(f).

To explore the difference between the proposed PiT and the baseline method, the attention maps of the query pedestrian image are illustrated in Fig. 5. With the same input image, the proposed PiT and the baseline method have different attention maps. The baseline method cannot extract the fine-grained local features. Therefore, it cannot reduce the unfavorable impacts from the man in a blue shirt in Fig. 5(a) and the road sign in Fig. 5(b). Therefore, more fine-grained feature representation can help the model recognize the pedestrian of interest.

6) *Computation Complexity and Running Time*: Table VI adopts Multiply-ACcumulate operations (MACs) and trainable parameters to show the computation complexity. We use one 24G NVIDIA TITAN RTX GPU to conduct the experiments.

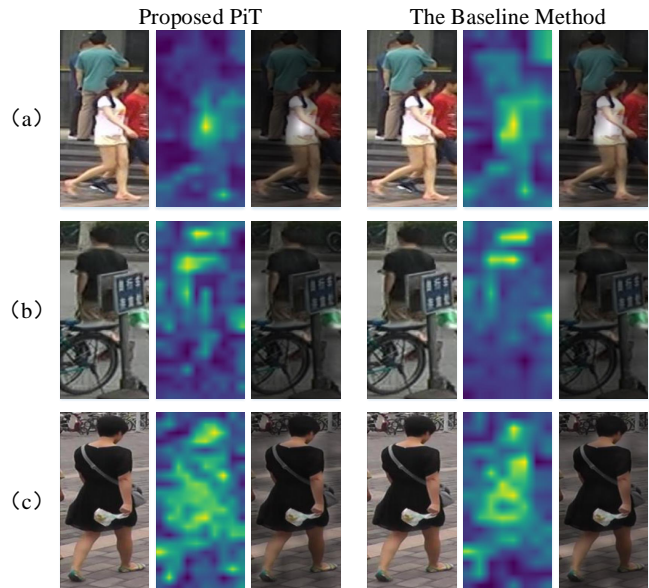


Fig. 5: Attention map examples. Three images in each group are the input image, the attention map of this image, and the product result between input image and its attention map.

For the MARS benchmark, the training and testing processes of the proposed PiT take 5.00 hours, including training 8,298 videos from 625 pedestrian IDs and testing another 11,310 videos for evaluation. For the iLIDS-VID benchmark, the experiments include ten trials to ensure statistical stability, and the total running time takes 10.70 hours. Each trial contains training 300 videos from 150 IDs and testing another 300 videos. In Table VI, the proposed PiT has acceptable MACs, trainable parameters, and running time, which have the same order of magnitude as the baseline method. The superiority of the proposed method can be seen from the performance improvement in terms of the Rank-1 metric.

## V. CONCLUSION

This paper proposes a multi-direction and multi-scale Pyramid in Transformer (PiT) for video-based pedestrian retrieval. The proposed PiT contains four layers, and each layer applies different division strategies on the patch tokens to generate different-direction parts. The class token and the patch tokens in each generated part are fed to the corresponding transformer layer. In this way, the class token perceives the fine-grained, part-informed features. Then multi-direction and multi-scale features are combined to form a feature pyramid for each pedestrian image. The feature pyramids of pedestrian images



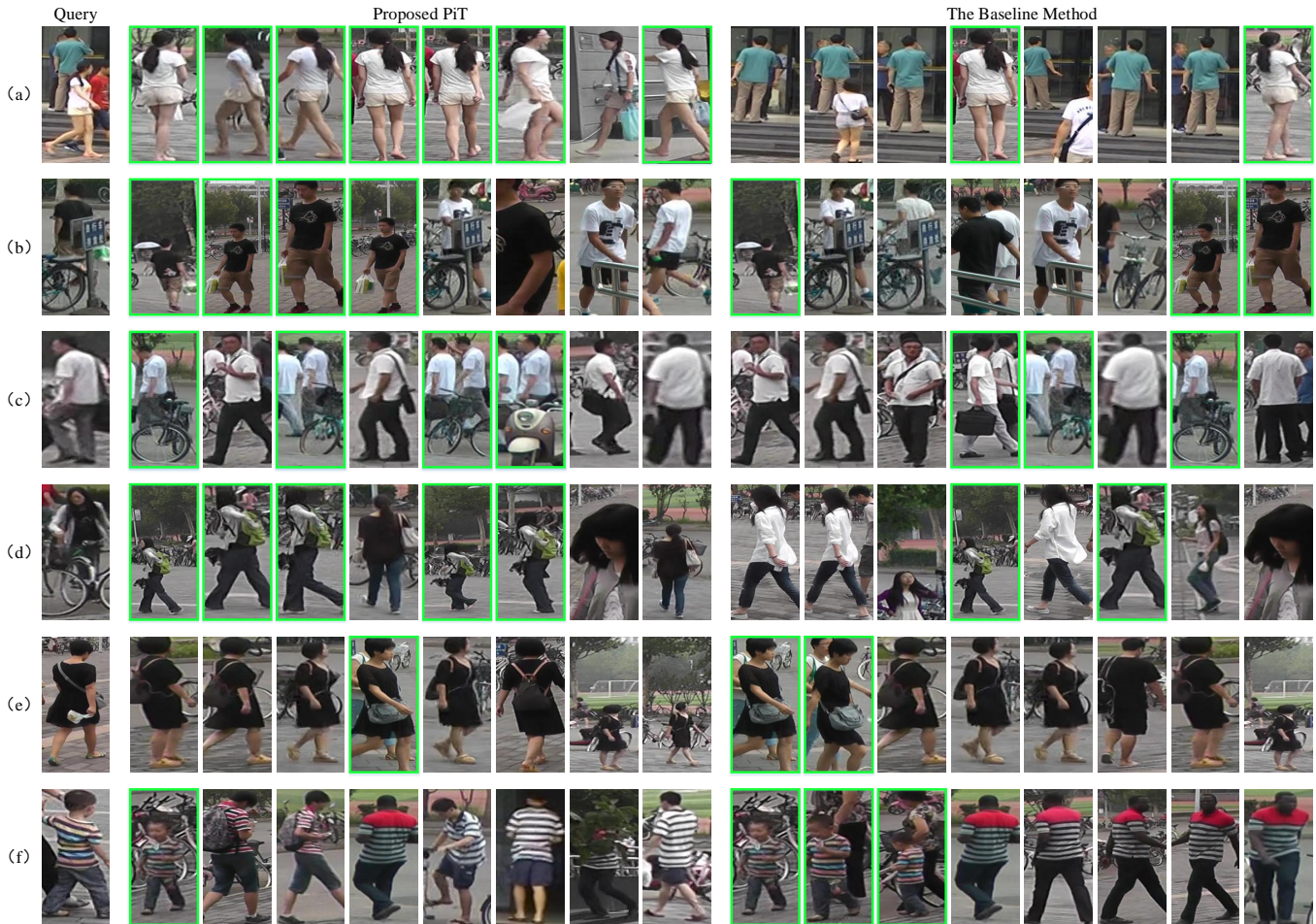


Fig. 6: Retrieval examples. Each video is represented by one pedestrian image within it. The query video is selected from MARS benchmark, and the top eight retrieval results for each query are illustrated in this figure. (a)(b)(c)(d) show the successful cases, and (e)(f) show the failed cases. The correct results are in green boxes.

belonging to the same video are fused to generate the final feature pyramid. Experimental results on two challenging benchmarks, MARS and iLIDS-VID, show the proposed PiT achieves state-of-the-art results. The comprehensive ablation studies demonstrate the superiority of the proposed multi-direction and multi-scale pyramid structure.

## REFERENCES

- [1] M. Ye, Y. Cheng, X. Lan, and H. Zhu, "Improving night-time pedestrian retrieval with distribution alignment and contextual distance," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 615–624, 2019.
- [2] J. García, A. Gardel, I. Bravo, and J. L. Lázaro, "Multiple view oriented matching algorithm for people reidentification," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1841–1851, 2014.
- [3] X. Zang, G. Li, W. Gao, and X. Shu, "Learning to disentangle scenes for person re-identification," *Image and Vision Computing*, vol. 116, p. 104330, 2021.
- [4] Z. Zeng, Z. Li, D. Cheng, H. Zhang, K. Zhan, and Y. Yang, "Two-stream multirate recurrent neural network for video-based pedestrian reidentification," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3179–3186, 2017.
- [5] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, Conference Proceedings, pp. 480–496.
- [6] Y. Fu, Y. Wei, Y. Zhou, H. Shi, G. Huang, X. Wang, Z. Yao, and T. Huang, "Horizontal pyramid matching for person re-identification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, Conference Proceedings, pp. 8295–8302.
- [7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, and S. Gelly, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020, Conference Proceedings.
- [8] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian, "Mars: A video benchmark for large-scale person re-identification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, 2016, Conference Proceedings, pp. 868–884.
- [9] T. Wang, S. Gong, X. Zhu, and S. Wang, "Person re-identification by video ranking," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2014, Conference Proceedings, pp. 688–703.
- [10] R. Zhou, X. Chang, L. Shi, Y.-D. Shen, Y. Yang, and F. Nie, "Person reidentification via multi-feature fusion with adaptive graph learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1592–1601, 2019.
- [11] X. Zang, G. Li, W. Gao, and X. Shu, "Exploiting robust unsupervised video person re-identification," *IET Image Processing*, 2021.
- [12] X. Liu, P. Zhang, C. Yu, H. Lu, and X. Yang, "Watching you: Global-guided reciprocal learning for video-based person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, Conference Proceedings, pp. 13 334–13 343.
- [13] R. Hou, H. Chang, B. Ma, R. Huang, and S. Shan, "Bicnet-tks: Learning efficient spatial-temporal representation for video person re-identification," in *Proceedings of the IEEE/CVF Conference on Com-*

- puter Vision and Pattern Recognition*, 2021, Conference Proceedings, pp. 2014–2023.
- [14] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, Conference Proceedings, pp. 4370–4379.
- [15] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, “Deep learning for person re-identification: A survey and outlook,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [16] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, “Video swin transformer,” *arXiv preprint arXiv:2106.13230*, 2021.
- [17] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1904–16, 2014.
- [18] Y. Zhao, X. Shen, Z. Jin, H. Lu, and X.-s. Hua, “Attribute-driven feature disentangling and temporal aggregation for video person re-identification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, Conference Proceedings, pp. 4913–4922.
- [19] J. Li, J. Wang, Q. Tian, W. Gao, and S. Zhang, “Global-local temporal representations for video person re-identification,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, Conference Proceedings, pp. 3958–3967.
- [20] A. Subramaniam, A. Nambiar, and A. Mittal, “Co-segmentation inspired attention networks for video-based person re-identification,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, Conference Proceedings, pp. 562–572.
- [21] X. Jiang, Y. Gong, X. Guo, Q. Yang, F. Huang, W.-S. Zheng, F. Zheng, and X. Sun, “Rethinking temporal fusion for video-based person re-identification on semantic and time aspect,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, Conference Proceedings, pp. 11 133–11 140.
- [22] Z. Chen, Z. Zhou, J. Huang, P. Zhang, and B. Li, “Frame-guided region-aligned representation for video person re-identification,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, Conference Proceedings, pp. 10 591–10 598.
- [23] X. Li, W. Zhou, Y. Zhou, and H. Li, “Relation-guided spatial attention and temporal refinement for video-based person re-identification,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, Conference Proceedings, pp. 11 434–11 441.
- [24] S. Li, H. Yu, and H. Hu, “Appearance and motion enhancement for video-based person re-identification,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, Conference Proceedings, pp. 11 394–11 401.
- [25] J. Liu, Z. J. Zha, X. Zhu, and N. Jiang, “Co-saliency spatio-temporal interaction network for person re-identification in videos,” in *Twenty-Ninth International Joint Conference on Artificial Intelligence and Seventeenth Pacific Rim International Conference on Artificial Intelligence IJCAI-PRICAI-20*, 2020, Conference Proceedings.
- [26] 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 *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, Conference Proceedings, pp. 1706–1715.
- [27] Z. Zhang, C. Lan, W. Zeng, and Z. Chen, “Multi-granularity reference-aided attentive feature aggregation for video-based person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, Conference Proceedings, pp. 10 407–10 416.
- [28] Y. Yan, J. Qin, J. Chen, L. Liu, F. Zhu, Y. Tai, and L. Shao, “Learning multi-granular hypergraphs for video-based person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, Conference Proceedings, pp. 2899–2908.
- [29] J. Yang, W.-S. Zheng, Q. Yang, Y.-C. Chen, and Q. Tian, “Spatial-temporal graph convolutional network for video-based person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, Conference Proceedings, pp. 3289–3299.
- [30] R. Hou, B. Ma, H. Chang, X. Gu, S. Shan, and X. Chen, “Vrsrc: Occlusion-free video person re-identification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, Conference Proceedings, pp. 7183–7192.
- [31] R. Hou, H. Chang, B. Ma, S. Shan, and X. Chen, “Temporal complementary learning for video person re-identification,” in *European Conference on Computer Vision*. Springer, 2020, Conference Proceedings, pp. 388–405.
- [32] X. Gu, H. Chang, B. Ma, H. Zhang, and X. Chen, “Appearance-preserving 3d convolution for video-based person re-identification,” in *European Conference on Computer Vision*. Springer, 2020, Conference Proceedings, pp. 228–243.
- [33] G. Chen, Y. Rao, J. Lu, and J. Zhou, “Temporal coherence or temporal motion: Which is more critical for video-based person re-identification?” in *European Conference on Computer Vision*. Springer, 2020, Conference Proceedings, pp. 660–676.
- [34] X. Jiang, Y. Qiao, J. Yan, Q. Li, W. Zheng, and D. Chen, “Ssn3d: Self-separated network to align parts for 3d convolution in video person re-identification,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, Conference Proceedings, pp. 1691–1699.
- [35] A. Porrello, L. Bergamini, and S. Calderara, “Robust re-identification by multiple views knowledge distillation,” in *European Conference on Computer Vision*. Springer, 2020, Conference Proceedings, pp. 93–110.