GaitStrip: Gait Recognition via Effective Strip-Based Feature Representations and Multi-level Framework

Ming Wang[1], Beibei Lin[2], Xianda Guo[3], Lincheng Li[4], Zheng Zhu[3], Jiande Sun[5], Shunli Zhang[1](✉), Yu Liu[1], and Xin Yu[6]

[1] Beijing Jiaotong University, Beijing, China
slzhang@bjtu.edu.cn
[2] National University of Singapore, Singapore, Singapore
[3] PhiGent Robotics, Beijing, China
[4] NetEase Fuxi AI Lab, Beijing, China
[5] Shandong Normal University, Jinan, China
[6] University of Technology Sydney, Ultimo, Australia

Abstract. Many gait recognition methods first partition the human gait into N-parts and then combine them to establish part-based feature representations. Their gait recognition performance is often affected by partitioning strategies, which are empirically chosen in different datasets. However, we observe that strips as the basic component of parts are agnostic against different partitioning strategies. Motivated by this observation, we present a strip-based multi-level gait recognition network, named GaitStrip, to extract comprehensive gait information at different levels. To be specific, our high-level branch explores the context of gait sequences and our low-level one focuses on detailed posture changes. We introduce a novel StriP-Based feature extractor (SPB) to learn the strip-based feature representations by directly taking each strip of the human body as the basic unit. Moreover, we propose a novel multi-branch structure, called Enhanced Convolution Module (ECM), to extract different representations of gaits. ECM consists of the Spatial-Temporal feature extractor (ST), the Frame-Level feature extractor (FL) and SPB, and has two obvious advantages: First, each branch focuses on a specific representation, which can be used to improve the robustness of the network. Specifically, ST aims to extract spatial-temporal features of gait sequences, while FL is used to generate the feature representation of each frame. Second, the parameters of the ECM can be reduced in test by introducing a structural re-parameterization technique. Extensive experimental results demonstrate that our GaitStrip achieves state-of-the-art performance in both normal walking and complex conditions. The source code is published at https://github.com/M-Candy77/GaitStrip.

1 Introduction

Gait recognition is one of the most popular biometric techniques. Since it can be used in a long-distance condition and cannot be disguised, gait recognition

L. Wang et al. (Eds.): ACCV 2022, LNCS 13844, pp. 711–727, 2023.
https://doi.org/10.1007/978-3-031-26316-3_42

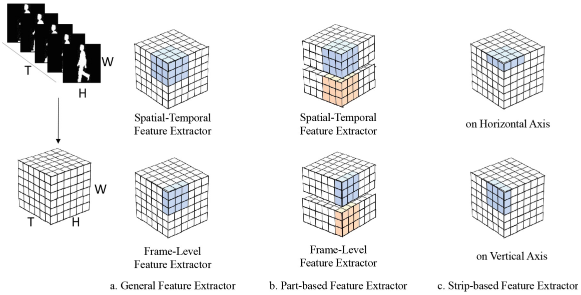
T
W
H
Spatial-Temporal Feature Extractor
Frame-Level Feature Extractor
a. General Feature Extractor
Spatial-Temporal Feature Extractor
Frame-Level Feature Extractor
b. Part-based Feature Extractor
on Horizontal Axis
on Vertical Axis
c. Strip-based Feature Extractor

Fig. 1. Visualization of feature extractors of different methods.

is widely applied in video surveillance and access control systems. However, this technology has experienced a huge challenge due to the complexity of the external environment, such as cross-view, speed changes, bad weathers and variations in appearances [4,13,14,35,37].

Recently, many Convolutional Neural Networks (CNNs) based gait recognition frameworks have been proposed to generate discriminative feature representations [1,2,7,15–19,21,22,26–28,33,36,40,44,45]. As shown in Fig. 1(a), some researchers extract gait features directly from the whole gait sequence, which captures global context information of gait sequences [2,27,40]. As those methods take the human gait as a unit to extract features, some local gait changes that are important for gait recognition might not be fully captured, which may affect the recognition performance. On the other hand, some other researchers [7,44] propose part-based feature representation to represent the human gait, which is shown in Fig. 1(b). They first partition the human gait into N-parts and then extract the detailed information of each part. Although carefully choosing the number of partitions in different convolutional layers can achieve appealing performance, it is unclear how to build an accurate part-based model on new datasets, which limits the generalization of the methods.

According to these findings, we argue that the part-based feature representation is not a general feature representation for gait recognition. Hence, we question *whether there is a gait descriptor that is insensitive to various partitions?* Through carefully analysis of recent part-based methods, we find that strips are the minimal effective representation elements for gaits instead of parts. Using strips, we will be able to circumvent the handcrafted partition in part-based methods. As shown in Fig. 1(c), the strip can be considered as an extreme form of the part-based representation, thus it is not necessary to manually determine the reasonable number of the parts. Motivated by this observation, we propose a new gait recognition network, called GaitStrip, to learn more discriminative feature representations based on strips. Specifically, GaitStrip is implemented under a multi-level framework to improve the representation capability. The multi-level framework includes two branches, i.e., the low-level branch and the high-level one. In particular, the high-level branch extracts the global context information from low-resolution gait images, while the low-level one captures more details from high-resolution images.

Furthermore, we introduce Enhanced Convolution Module (ECM), as a multi-branch block, to our GaitStrip. ECM includes three branches, i.e., the StriP-Based feature extractor (SPB), the Spatial-Temporal feature extractor (ST) and the Frame-Level feature extractor (FL), where each branch corresponds to a specific representation. Specifically, SPB is designed to generate strip-based feature representations by taking each strip of the human body as a basic unit, ST aims to extract spatial-temporal information of a gait sequence, and FL is used to extract each frame's spatial features. On the other hand, we introduce a structural re-parameterization technique to reduce the parameters of the ECM module in test [6]. Specifically, the parameters of SPB, ST and FL can be merged into a single $3 \times 3 \times 3$ convolution.

After feature extraction, we obtain an effective feature representation by using temporal aggregation and spatial mapping operations. The temporal aggregation ensembles temporal information of a variable-length gait sequence [20]. The spatial mapping first partitions the feature maps into multiple horizontal vectors and aggregates each vector by Generalized-Mean (GeM) pooling operations [25] for better representation. Extensive experiments on widely-used gait recognition benchmarks demonstrate that our GaitStrip outperforms the state-of-the-arts significantly.

The main contributions of the proposed method are three-fold, shown as follows:

- Based on the observation that the strip-based method can achieve more effective gait representations than part-based partitioning, we propose a multi-level gait recognition framework with strip to extract more comprehensive gait features, in which the high-level representation contains the context information while the low-level representation extracts local details of gait sequences.
- We develop an effective enhanced convolution module including three branches, which can not only take the advantage of both frame-level and spatial-temporal features but also use SPB to enhance the representation ability. Furthermore, we use the structural re-parameterization technique to reduce the parameters for high efficiency in test.
- We compare the proposed method with several state-of-the-art methods on two public datasets, CASIA-B and OUMVLP. The experimental results demonstrate that the performance of the proposed method achieves superior performance to these approaches.

2 Related Work

2.1 Gait Recognition

Existing gait recognition methods can be divided into two types, *i.e.* , global-based and local-based.

The global-based methods usually take the human gait as a sample to generate global feature representations [27,34,40]. For instance, Shiraga et al. [27] first calculate the Gait Energy Image (GEI) by using the mean function to compress the temporal information of gait sequences, and then utilize 2D CNNs to extract gait features. However, the generation of the GEI causes the loss of temporal information, which may degrade the representation ability. Thus, some other researchers [2,3,10,43] use 2D CNNs to extract each frame's feature before building the template. On the other hand, some researchers [20,30,32] extract spatial-temporal information from gait sequences for representation. Recently, 3D CNN has been used in gait recognition to learn the spatial-temporal representation of the entire gait sequence. For example, Lin et al. [20] use 3D CNNs to extract spatial-temporal information, and employ temporal aggregation to integrate temporal information, addressing the variable-length issue of video sequences.

The local-based methods usually take the part of the human gait as input to establish the part-based feature representations [7,44]. For example, Fan et al. [7] propose a focal convolution layer to extract part-based gait features. The focal convolutional layer first splits the feature maps into several local parts and then uses a shared convolution to extract each part's feature. Zhang et al. [44] first partition the human gait into four parts and then use 2D CNN to obtain feature representations of each part. However, these local-based methods need to predefine the number of partitions for specific datasets, which limit the generalization ability.

2.2 Strip-Based Modeling

Recently many strip-based modeling methods have been proposed in the visual field. For example, Ding et al. [5] propose a novel block, called Asymmetric Convolution Block (ACB), to generate discriminative feature representations. They use 1D asymmetric forms (e.g. 3×1 Conv and 1×3 Conv) to improve the feature representation ability of the standard square-kernel convolution (3×3 Conv). Note that the asymmetric convolutions can exploit the information of the horizontal and vertical strips. In particular, the asymmetric convolutions can be fused into the original square-kernel convolution. Huang et al. [12] propose the CCNet network to capture global contextual information. CCNet which is built with Criss-Cross Attention blocks models the relationships of horizontal and vertical strips.

However, the aforementioned methods only focus on the spatial strip-based information, which do not capture the temporal changes of each strip. Therefore, in this paper, we propose a novel strip-based feature extractor, which can be used to establish each strip's spatial-temporal information. In particular, as far as we know, GaitStrip is the first network which models strip-based feature representations in gait recognition.

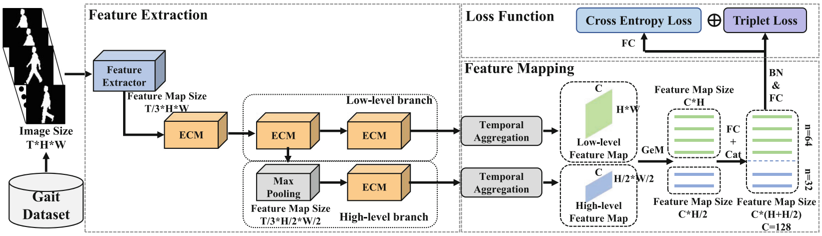

Fig. 2. Overview of the entire gait recognition framework.

3 Proposed Method

In this section, we first overview the whole gait recognition framework. Then, we describe the enhanced convolution module, the multiple-level structure and feature mapping in detail. Finally, we introduce the strategies of training and test.

3.1 Overview

The proposed gait recognition framework, GaitStrip, which includes the feature extraction stage and feature mapping stage is shown in Fig. 2. The GaitStrip is constructed based on 3D convolutions, which can effectively extract spatial-temporal information of gait sequences. During the feature extraction stage, a novel enhanced convolution module which uses both frame-level feature extractor and strip-based feature extractor to improve the representation ability of the traditional spatial-temporal feature extractor is proposed. Then, we design the multi-level framework which includes both the high-level and the low-level branches. During the feature mapping stage, the temporal aggregation operation is introduced to integrate the temporal information of feature maps [20]. Then, the feature maps are partitioned into multiple horizontal vectors and the information is aggregated by Generalized-Mean (GeM) pooling [25]. Finally, a combined loss function consisting of both cross-entropy loss and triplet loss is employed to train the proposed network.

3.2 Enhanced Convolution Module

Recently, many excellent feature extractors have been proposed to extract robust gait features, which can be divided into two types. One is the frame-level feature extractor which extracts gait features of each frame [2,3,7], and the other one is the spatial-temporal feature extractor which generates spatial-temporal feature representations of a gait sequence [20,30,32].

Assume that the feature map $X_{in} \in \mathbb{R}^{C_{in} \times T_{in} \times H_{in} \times W_{in}}$ is the input of a convolution operation, where C_{in} is the number of channels, T_{in} is the length of gait sequences and (H_{in}, W_{in}) is the image size of each frame. The frame-level and spatial-temporal feature extractors can be designed as

$$X_{FL} = c^{1\times3\times3}(X_{in}), \tag{1}$$

$$X_{ST} = c^{3\times3\times3}(X_{in}), \tag{2}$$

where $c^{a\times b\times c}(\cdot)$ represents the 3D convolution with kernel size (a, b, c). $X_{FL} \in \mathbb{R}^{C_{out}\times T_{in}\times H_{in}\times W_{in}}$ and $X_{ST} \in \mathbb{R}^{C_{out}\times T_{in}\times H_{in}\times W_{in}}$ are the output of the frame-level and spatial-temporal feature extractors, respectively.

The frame-level features ignore the temporal information of the gait sequence, while the spatial-temporal features focus on the spatial-temporal changes, which may not pay enough attention to the detailed information of each frame. Thus, we propose a combined framework which takes advantage of frame-level and spatial-temporal information as our backbone. The combined structure includes two branches, i.e. the spatial-temporal feature extractor and frame-level feature extractor, which can be designed as

$$X_{STFL} = X_{FL} + X_{ST}. \tag{3}$$

To further improve the global representation and address the inflexibility issue in the part-based representation, we present a StriP-Based feature extractor (SPB) which extracts strip-based features on horizontal axis and vertical axis, respectively. The strip-based feature extractor on horizontal axis first splits the human body into multiple horizontal strips and then applies convolution to extract spatial-temporal information of each horizontal strip. This extractor can be denoted as

$$X_{SPB-H} = c^{3\times1\times3}(X_{in}), \tag{4}$$

where $c^{3\times1\times3}(\cdot)$ denotes the 3D convolution with kernel size (3, 1, 3). $X_{SPB-H} \in \mathbb{R}^{C_{out}\times T_{in}\times H_{in}\times W_{in}}$ is the output of this extractor.

Similarly, the strip-based feature extractor is used for the vertical strip's spatial-temporal extraction, represented as

$$X_{SPB-V} = c^{3\times3\times1}(X_{in}), \tag{5}$$

where $c^{3\times3\times1}(\cdot)$ denotes the 3D convolution with kernel size (3, 3, 1). $X_{SPB-V} \in \mathbb{R}^{C_{out}\times T_{in}\times H_{in}\times W_{in}}$ is the output of this extractor. Finally, by combining the horizontal-based and vertical-based feature extractors, the strip-based feature extractor can obtain the following feature maps

$$X_{SPB} = X_{SPB-H} + X_{SPB-V}. \tag{6}$$

The proposed SPB can be used to enhance the feature representation ability of the traditional feature extractor. By combining SPB with aforementioned spatial-temporal feature extractor and frame-level feature extractor, as shown in Fig. 3, the ECM module can be obtained as follows

$$X_{ECM} = X_{ST} + X_{FL} + X_{SPB}. \tag{7}$$

Thus more comprehensive feature representations can be achieved.

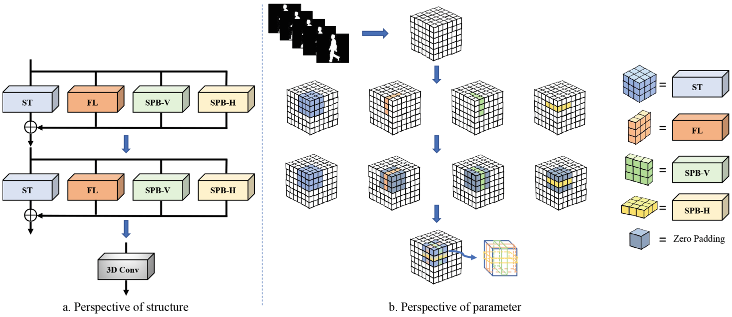

Fig. 3. Overview of the enhanced convolution module. ST represents the Spatial-Temporal feature extractor, FL represents the Frame-Level feature extractor, SPB-V represents StriP-Based feature extractor in vertical and SPB-H represents StriP-Based feature extractor in horizontal.

3.3 Structural Re-parameterization

To reduce the parameters of the proposed ECM, we introduce the structural re-parameterization [6] method to ensemble different convolutions during the test stage. As shown in Eq. 7, the ECM block includes four convolutions: $c^{3\times3\times3}(\cdot)$, $c^{1\times3\times3}(\cdot)$, $c^{3\times1\times3}(\cdot)$ and $c^{3\times3\times1}(\cdot)$. During the test stage, these convolutions can be integrated into a single 3D convolution $c_{emb}^{3\times3\times3}(\cdot)$, which can be designed as

$$c_{emb}^{3\times3\times3} = c^{3\times3\times3} + c_t^{3\times3\times3} + c_h^{3\times3\times3} + c_w^{3\times3\times3}, \tag{8}$$

where $c_t^{3\times3\times3}$, $c_h^{3\times3\times3}$ and $c_w^{3\times3\times3}$ are zero-padding expansions of $c^{1\times3\times3}$, $c^{3\times1\times3}$ and $c^{3\times3\times1}$, respectively, to make the kernels maintain the same dimensions. According to Eq. 8, the ECM in the test stage can be designed as

$$X_{ECM} = c_{emb}^{3\times3\times3}(X_{in}). \tag{9}$$

Note that although four convolutions are employed to improve the representation ability in the training stage, only a single convolution is required in the test stage, which does not increase the parameter number and not degrade the inference running efficiency.

3.4 Multi-level Framework

To further improve the representation ability, we design the multi-level framework based on the proposed ECM block for both high-level and low-level feature extraction. The low-level branch directly extracts features from the large-size feature maps, which focuses on details of the human body. By contrast, the high-level one which works on down-sampled feature maps based on max pooling can extract more abstract information.

3.5 Temporal Aggregation and Spatial Mapping

To adaptively aggregate the temporal information of variable-length gait sequences, we introduce the temporal aggregation [20]. Assuming that the feature map $X_{out} \in \mathbb{R}^{C_f \times T_f \times H_f \times W_f}$ is the output of the feature extraction module, the temporal aggregation operation can be represented as

$$Y_{ta} = F_{Max}^{1 \times T_f \times 1 \times 1}(X_{out}), \tag{10}$$

where $Y_{ta} \in \mathbb{R}^{C_f \times 1 \times H_f \times W_f}$ is the output of the temporal aggregation module.

For the spatial mapping, we generate multiple horizontal feature representations and then combine them to improve the spatial representation ability [2,7,20,24,39]. The spatial mapping can be represented as

$$Y_{out} = F_s(F_{GeM}^{1 \times 1 \times 1 \times W_f}(Y_{ta})), \tag{11}$$

where $Y_{out} \in \mathbb{R}^{C_f \times 1 \times H_f \times 1}$ is the output of the spatial mapping. $F_{GeM}(\cdot)$ means the Generalized-Mean (GeM) pooling operation [25]. $F_s(\cdot)$ denotes the multiple separate fully connected (FC) layers. After spatial mapping, we obtain the final feature representation Y by concatenating the high-level and low-level feature maps in horizontal axis.

3.6 Loss Function

To train the proposed network, we employ the combined loss function which consists of the triplet loss and cross entropy loss. Besides the traditional cross entropy loss used for classification, the triplet loss is also introduced to make the samples from the same ID as close as possible while those from different IDs have larger distance in the feature space. The combined loss function is calculated with the obtained the output of spatial mapping, which is represented as

$$L_{combined} = L_{tri} + L_{cse}, \tag{12}$$

where the L_{tri} and L_{cse} denote the triplet loss and cross entropy loss, respectively. L_{tri} is defined as

$$L_{tri} = \max(d(r,s) - d(r,t) + m, 0) \tag{13}$$

where r and s are samples of the same category, while r and t are samples from different categories. $d(\cdot)$ represents the Euclidean distance between the two samples and m is the margin of the triplet loss.

3.7 Training and Test Details

Training. In this paper, we introduce a combined loss function consisting of cross-entropy loss and triplet loss [2,7,31,38,41,42] to train the proposed GaitStrip. Specifically, the feature representation Y is fed into the triplet loss function for calculation [2], and input into the cross-entropy loss function through an FC layer. The Batch ALL (BA) [8] is used as the sampling strategy. The number of samples of each batch is $P \times K$, which contains P classes and each class corresponds to K samples. Considering the memory limit, the length of gait sequences is set to T in the training stage.

Test. During the test stage, we input the whole sequences into the GaitStrip to produce the feature representation Y. After that, Y is flattened into a vector to represent the corresponding sample. In general, the gait datasets are usually divided into two sets, *i.e.* , the gallery set and the probe set. The feature vectors from the gallery set are taken as the standard view to be retrieved, while those from the probe set are used to evaluate the performance. Specifically, we calculate the Euclidean distance between the feature vectors in the probe set and all feature vectors in the gallery set. The class label of the gallery sample with the smallest distance will be assigned to the probe sample.

4 Experiments

4.1 Datasets and Evaluation Protocol

CASIA-B. The CASIA-B dataset [37] is one of the largest gait datasets for evaluation. It includes 124 subjects, each of which contains 10 groups of gait sequences (six groups of normal walking (NM) #01-#06, two groups of walking with a bag (BG) #01-#02 and two groups of walking with a coat (CL) #01-#02). Each group contains 11 view angles ($0°$–$180°$) and the sampling interval is $18°$. Hence, the CASIA-B dataset contains 124 (subject) $\times$ 10 (groups) $\times$ 11 (view angle) = 13,640 gait sequences. The dataset is divided into two subsets, the training set and the test set. We use the protocol [2] for evaluation, which includes three different settings, i.e., Small-sample Training(ST), Medium-sample Training (MT) and Large-sample Training (LT). In the three settings, 24, 62 and 74 subjects are used to form the training set, respectively, and the rest are used for test. During the test stage, four groups of sequences (NM#01-#04) are used as the gallery set and the rest (NM#05-#06, BG#01-#02 and CL#01-#02) are taken as the probe set.

OUMVLP. The OUMVLP dataset [29] is one of the most popular gait datasets, which includes 10,307 subjects. Each subject was collected in two groups of video sequences (Seq#00 and Seq#01), each of which contains 14 view angles ($0°$, $15°, ..., 75°$, $90°$, $180°$, $195°, ..., 255°$, $270°$). During the test phase, the sequences in Seq#01 are used as the gallery set and the sequences in Seq#00 are taken as the probe set.

4.2 Implementation Details

Gait sequences are preprocessed and normalized into the same size 64×44 on both datasets [2]. In CASIA-B, GaitStrip has four blocks, where the last three blocks are built with the proposed ECM. The channel number of the four blocks

Table 1. Rank-1 accuracy (%) on CASIA-B under all view angles, different settings and conditions, excluding identical-view case.

Gallery NM#1-4			0°–180°											
Probe			0°	18°	36°	54°	72°	90°	108°	126°	144°	162°	180°	Mean
ST (24)	NM#5-6	GaitSet [2]	64.6	83.3	90.4	86.5	80.2	75.5	80.3	86.0	87.1	81.4	59.6	79.5
		MT3D [20]	71.9	83.9	90.9	90.1	81.1	75.6	82.1	89.0	91.1	86.3	69.2	82.8
		GaitGL [23]	77.0	87.8	93.9	92.7	83.9	78.7	84.7	91.5	92.5	89.3	74.4	86.0
		Ours	**79.6**	**89.5**	**95.6**	**94.3**	**86.4**	**82.0**	**86.6**	**93.0**	**93.6**	**90.1**	**75.1**	**87.8**
	BG#1-2	GaitSet [2]	55.8	70.5	76.9	75.5	69.7	63.4	68.0	75.8	76.2	70.7	52.5	68.6
		MT3D [20]	64.5	76.7	82.8	82.8	73.2	66.9	74.0	81.9	84.8	80.2	63.0	74.0
		GaitGL [23]	68.1	81.2	87.7	84.9	76.3	70.5	76.1	84.5	87.0	83.6	65.0	78.6
		Ours	**71.4**	**82.6**	**90.4**	**88.1**	**77.9**	**73.6**	**79.8**	**86.4**	**89.1**	**86.3**	**71.3**	**81.5**
	CL#1-2	GaitSet [2]	29.4	43.1	49.5	48.7	42.3	40.3	44.9	47.4	43.0	35.7	25.6	40.9
		MT3D [20]	46.6	61.6	66.5	63.3	57.4	52.1	58.1	58.9	58.5	57.4	41.9	56.6
		GaitGL [23]	46.9	58.7	66.6	65.4	58.3	54.1	59.5	62.7	61.3	57.1	40.6	57.4
		Ours	**54.3**	**67.8**	**75.0**	**71.6**	**66.2**	**59.7**	**65.5**	**70.5**	**69.6**	**63.6**	**46.6**	**64.6**
MT (62)	NM#5-6	GaitSet [2]	86.8	95.2	98.0	94.5	91.5	89.1	91.1	95.0	97.4	93.7	80.2	92.0
		MT3D [20]	91.9	96.4	98.5	95.7	93.8	90.8	93.9	97.3	97.9	95.0	86.8	94.4
		GaitGL [23]	93.9	97.6	**98.8**	97.3	95.2	92.7	95.6	98.1	98.5	96.5	91.2	95.9
		Ours	**94.0**	**98.0**	98.7	**97.8**	**95.6**	**93.0**	**96.1**	**98.2**	**98.6**	**97.0**	**92.6**	**96.3**
	BG#1-2	GaitSet [2]	79.9	89.8	91.2	86.7	81.6	76.7	81.0	88.2	90.3	88.5	73.0	84.3
		MT3D [20]	86.7	92.9	94.9	92.8	88.5	82.5	87.5	92.5	95.3	92.9	81.2	89.8
		GaitGL [23]	88.5	95.1	95.9	94.2	91.5	85.4	89.0	95.4	97.4	94.3	86.3	92.1
		Ours	**88.8**	**95.2**	**96.8**	**95.5**	**92.7**	**87.4**	**90.7**	**95.7**	**97.6**	**95.3**	**87.0**	**93.0**
	CL#1-2	GaitSet [2]	52.0	66.0	72.8	69.3	63.1	61.2	63.5	66.5	67.5	60.0	45.9	62.5
		MT3D [20]	67.5	81.0	85.0	80.6	75.9	69.8	76.8	81.0	80.8	73.8	59.0	75.6
		GaitGL [23]	70.7	83.2	87.1	84.7	78.2	71.3	78.0	83.7	83.6	77.1	63.1	78.3
		Ours	**69.2**	**86.7**	**90.0**	**88.3**	**83.6**	**75.8**	**82.3**	**88.1**	**88.1**	**81.7**	**65.7**	**81.8**
LT (74)	NM#5-6	GaitSet [2]	90.8	97.9	99.4	96.9	93.6	91.7	95.0	97.8	98.9	96.8	85.8	95.0
		GaitPart [7]	94.1	98.6	99.3	98.5	94.0	92.3	95.9	98.4	99.2	97.8	90.4	96.2
		MT3D [20]	95.7	98.2	99.0	97.5	95.1	93.9	96.1	98.6	99.2	98.2	92.0	96.7
		GaitGL [23]	96.0	98.3	**99.0**	97.9	**96.9**	**95.4**	97.0	98.9	**99.3**	98.8	94.0	97.4
		Ours	**96.0**	**98.4**	98.8	**97.9**	96.6	95.3	**97.5**	**98.9**	99.1	**99.0**	**96.3**	**97.6**
	BG#1-2	GaitSet [2]	83.8	91.2	91.8	88.8	83.3	81.0	84.1	90.0	92.2	94.4	79.0	87.2
		GaitPart [7]	89.1	94.8	96.7	95.1	88.3	84.9	89.0	93.5	96.1	93.8	85.8	91.5
		MT3D [20]	91.0	95.4	97.5	94.2	92.3	86.9	91.2	95.6	97.3	96.4	86.6	93.0
		GaitGL [23]	92.6	96.6	96.8	95.5	93.5	89.3	92.2	96.5	98.2	96.9	**91.5**	94.5
		Ours	**92.8**	**96.6**	**97.2**	**96.5**	**95.2**	**90.5**	**93.5**	**97.5**	**98.3**	**97.6**	91.4	**95.2**
	CL#1-2	GaitSet [2]	61.4	75.4	80.7	77.3	72.1	70.1	71.5	73.5	73.5	68.4	50.0	70.4
		GaitPart [7]	70.7	85.5	86.9	83.3	77.1	72.5	76.9	82.2	83.8	80.2	66.5	78.7
		MT3D [20]	76.0	87.6	89.8	85.0	81.2	75.7	81.0	84.5	85.4	82.2	68.1	81.5
		GaitGL [23]	76.6	90.0	90.3	87.1	84.5	79.0	84.1	87.0	87.3	84.4	69.5	83.6
		Ours	**79.9**	**92.3**	**93.4**	**89.2**	**86.0**	**80.0**	**86.0**	**88.5**	**91.7**	**87.5**	**73.5**	**86.2**

is set to 32, 64, 128 and 128, respectively. In OUMVLP, we use five blocks to construct the proposed GaitStrip and the last two blocks are implemented by the ECM module. The channel number of the five blocks is set to 64, 128, 196, 256 and 256, respectively. The margin of the triplet loss is set to 0.2 and Adam is selected as the optimizer. During the training stage, the parameters P and K are both set to 8. And the length of sequences T is set to 30. The learning rate is set to 1e-4 and reset to 1e-5 in the last 10K iterations. For the settings ST, MT and LT on CASIA-B dataset, the iteration number is set to 60K, 70K and 80K, respectively. On OUMVLP dataset, the parameter $P \times K$ is set to 32×8. The iteration number is set to 210K. The learning rate is initialized to 1e-4 and reset to 1e-5 after 150K iterations.

4.3 Comparison with the State-of-the-Art

Evaluation on CASIA-B. We compare the proposed method with several gait recognition approaches including GaitSet [2], GaitPart [7], MT3D [20] and GaitGL [23] on the CASIA-B dataset. The experimental results are shown in Table 1. It can be observed that the proposed method achieves the highest average accuracy under all settings (ST, MT and LT) and conditions (NM, BG and CL). Furthermore, we explore the performance of the proposed method under different settings and conditions in details.

Evaluation Under Various Settings (ST, MT and LT). We observe that our method achieves high performance under all three settings (ST, MT and LT) and exceeds the best result reported before. We display the complete experimental results under these three settings in Table 1. The recognition accuracy of GaitGL under ST MT and ST settings in NM condition is 86.0%, 95.9% and 97.4%, respectively. For the proposed method, the gait recognition accuracy is 87.8%, 96.3% and 97.6%, respectively. Furthermore, our method obtains significant improvement comparing with other methods in all three settings.

Evaluation Under Various Conditions (NM, BG and CL). It can be seen that when the external environment changes and more challenges exist, the accuracy decreases heavily. Under the LT setting, the accuracy of GaitGL in NM, BG and CL conditions is 97.4%, 94.5% and 83.6%, respectively. Comparing with GaitGL, our results are 0.2%, 0.7% and 2.6% higher, respectively. Under ST and MT settings, we can also observe that the proposed method owns the best performance. In the ST setting, our method outperforms GaitGL by 1.8%, 2.9% and 7.2% under NM, BG and CL, respectively. In the MT setting, the accuracy of the proposed method is 96.3%, 93.0% and 81.8%, which exceeds GaitGL by 0.4%, 0.9% and 3.5%, respectively.

Evaluation on Specific Angles (0°, 90°, 180°). The proposed method shows significant improvement in some extreme view angles (0°, 90° and 180°). For example, the average accuracy of MT3D in the setting LT and NM is 96.7%, but the accuracy corresponding to the three specific view angles are 95.7%, 93.9% and 92.0%, respectively. For the proposed method, the accuracy in the setting LT and NM is 97.6%, which outperforms MT3D by 0.9%. And the accuracy corresponding to the specific view angles (0°, 90° and 180°) are 96.0%, 95.3% and 96.3%, respectively, which outperforms MT3D by 0.3%, 1.4% and 4.3%, respectively. The main reason may be that the proposed SPB module extracts the feature of each strip, making the proposed ECM obtain more effective feature representation in the specific view angles.

Evaluation on OUMVLP. Compared with the CASIA-B, the OUMVLP dataset contains more subjects. Hereby, we compare GaitStrip with several famous gait recognition methods, including GEINet [27], GaitSet [2], GaitPart [7], GLN [9], SRN+CB [10], GaitGL [23] and 3D Local [11] on this dataset. The experimental results are shown in Table 2 which indicates that the proposed method achieves the optimal performance in all conditions. For example, the accuracy of GaitGL with invalid probe is 89.7%. For the proposed method,

Table 2. Rank-1 accuracy (%) on OUMVLP dataset under different view angles, excluding identical-view cases.

Method	Probe view														Mean
	0°	15°	30°	45°	60°	75°	90°	180°	195°	210°	225°	240°	255°	270°	
GEINet [27]	24.9	40.7	51.6	55.1	49.8	51.1	46.4	29.2	40.7	50.5	53.3	48.4	48.6	43.5	45.3
GaitSet [2]	84.5	93.3	96.7	96.6	93.5	95.3	94.2	87.0	92.5	96.0	96.0	93.0	94.3	92.7	93.3
GaitPart [7]	88.0	94.7	97.7	97.6	95.5	96.6	96.2	90.6	94.2	97.2	97.1	95.1	96.0	95.0	95.1
GLN [9]	89.3	95.8	97.9	97.8	96.0	96.7	96.1	90.7	95.3	97.7	97.5	95.7	96.2	95.3	95.6
SRN+CB [10]	91.2	96.5	98.3	98.4	96.3	97.3	96.8	92.3	96.3	98.1	98.1	96.0	97.0	96.2	96.4
GaitGL [23]	90.5	96.1	98.0	98.1	97.0	97.6	97.1	94.2	94.9	97.4	97.4	95.7	96.5	95.7	96.2
3D Local [11]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	96.5
Ours	**92.8**	**97.0**	**98.4**	**98.5**	**97.6**	**98.2**	**97.8**	**96.0**	**96.2**	**97.8**	**97.9**	**96.6**	**97.3**	**96.7**	**97.0**

Table 3. Rank-1 accuracy (%) of different ECM blocks.

ST	FL	SPB	NM	BG	CL
✓			97.4	94.9	85.3
✓		✓	97.4	95.2	85.5
	✓		96.2	92.9	78.5
	✓	✓	97.2	94.9	85.2
✓	✓		97.4	95.2	85.9
✓	✓	✓	**97.6**	**95.2**	**86.2**

the accuracy in the same conditions is 90.5%, which outperforms GaitGL by 0.8%. The accuracy of GaitGL excluding invalid probe sequences is 96.2%, while the accuracy of the proposed method is 97.0%.

4.4 Ablation Study

In this paper, to obtain effective feature representation, we propose the GaitStrip with ECM block, SPB feature extractor and multi-level framework. Therefore, we design several ablation studies to explore the contribution of the key components.

Analysis of the SPB module. We propose the novel SPB extractor to extract more discriminative gait features. To explore the contribution of the SPB, we first design three groups of comparative experiments, i.e., only using the ST to compare with the combination of ST and SPB, only using the FL to compare with the combination of FL and SPB, and comparing the combination of ST and FL to the combination of ST, FL and SPB. The experimental results are shown in Table 3. We can find that the performance of the modules with SPB is improved compared with that without SPB. The accuracy of methods with and without SPB in NM condition is very close, but the methods with SPB in CL condition perform better. Specifically, the accuracy in CL condition by using FL is 78.5%, while the accuracy in CL condition with the combination of FL and SPB is 85.2%, which increases by 6.7%. In the CL condition, the accuracy with the combination of FL, ST and SPB is 86.2%, which increases 0.3% compared with that with only the combination of

Table 4. Rank-1 accuracy (%) of different levels.

Multi-level structure		NM	BG	CL
High-level	Low-level			
✓		97.3	94.4	83.4
	✓	97.2	94.4	84.4
✓	✓	**97.6**	**95.2**	**86.2**

Table 5. The accuracy (%) of different strip-based modeling on the CASIA-B dataset.

Method	NM	BG	CL
baseline+ECM	**97.6**	**95.2**	**86.2**
baseline+ACB	96.1	92.8	79.4
baseline+CCA	80.4	75.1	67.6

FL and ST. Hence, the SPB can help to extract more comprehensive gait features, which plays an important role in recognition improvement.

Analysis of the ECM Block. In this paper, we propose the ECM to generate the discriminative feature representations by taking full advantage of the frame-level and strip-based information. The ECM consists of the ST, FL and SPB. To explore the advantage of the combination of the ST, FL and SPB in robust feature extraction, we design ablation experiments by using only one or two modules. The results of the ablation experiments are shown in Table 3. In NM condition, the accuracy of the combination of ST and SPB is 97.4%, the accuracy of the combination of FL and SPB is 97.2%, and the combination of the ST, FL and SPB is 97.6%, which increases by 0.2% and 0.4%, respectively, compared with the other two modules. The accuracy of the study shows that the combination of the ST, FL and SPB can obtain better accuracy in NM, BG and CL conditions than using only one or two of the modules.

Analysis of Multi-level Framework. The proposed GaitStrip works with multiple levels. To investigate the contribution of the low-level and high-level branches, we design the comparison methods with only one branch. The experimental results are shown in Table 4, from which we can observe that the accuracy of the methods with only high-level or low-level branch is 97.3% and 97.2%, respectively, while the accuracy with both levels is 97.6%, which achieves 0.3% and 0.4% improvement, respectively, demonstrating that the multi-level structure can effectively enhance the representation ability and then improve the recognition performance.

4.5 Comparison with Other Strip-Based Modeling

In Sect. 2.2, we introduce two different modules to model the strip-based information. To analyze their performance, we design some experiments by using the Asymmetric Convolution Block (ACB) or Criss-Cross Attention Block to replace the ECM module. All experiments are built with the LT setting on CASIA-B. The experimental results are shown in Table 5. It can be observed that the proposed ECM achieves better performance than other strip-based modelings. This may be because our ECM utilizes the spatial-temporal information of each strip, improving the feature representation ability. The accuracy of the ECM method in NM, BG and CL is 97.6%, 95.2% and 86.2% respectively, which exceeds the ACB method by 1.5%, 2.4% and 6.8%. The accuracy of the CCA method in NM, BG and CL is 80.4%, 75.1% and 67.7% respectively, which is inferior to our method as well. By comparing with other strip-based methods, we can note that the proposed method can better exploit the spatial-temporal representation, especially in some complex conditions, which achieves significant improvement.

4.6 Computational Analysis

In the inference stage, the proposed ECM can be embedded into a standard 3D convolution, which reduces parameters and inference time. The computational analysis is shown in Table 6. It can be observed that the average accuracy of using ECM is 93.0%, outperforming the accuracy of using ST by 0.5%. However, the parameters of both modules are equal.

Table 6. The accuracy (%), inference time (second/sequence) and parameters (M) of different methods on CASIA-B dataset

	Re-param	ST	ST+FL	ECM
Accuracy	–	92.5	92.8	**93.0**
Inference time	×	0.025	0.027	0.035
Parameters	×	3.87	4.33	5.25
Accuracy	–	92.5	92.8	**93.0**
Inference time	✓	0.025	0.025	0.025
Parameters	✓	3.87	3.87	3.87

5 Conclusion

In this paper, we propose a novel gait recognition network GaitStrip with ECM block and multi-level framework. On the one hand, the proposed ECM which aggregates spatial-temporal, frame-level and strip-based information can generate more comprehensive feature representations. Moreover, the spatial-temporal, frame-level and strip-based feature extractors can be embedded into a common 3D convolution in the inference stage, which does not introduce additional parameters. On the other hand, the multi-level structure containing both low-level and high-level branches can ensemble global semantic and local detailed information. The experiment results verify that the proposed GaitStrip achieves appealing performance in normal environment as well as complex conditions.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (61976017 and 61601021), the Beijing Natural Science Foundation (4202056), the Fundamental Research Funds for the Central Universities (2022JBMC013) and the Australian Research Council (DP220100800, DE230100477). The support and resources from the Center for High Performance Computing at Beijing Jiaotong University (http://hpc.bjtu.edu.cn) are gratefully acknowledged.

References

1. Chai, T., Mei, X., Li, A., Wang, Y.: Silhouette-based view-embeddings for gait recognition under multiple views. In: ICIP (2021)
2. Chao, H., He, Y., Zhang, J., Feng, J.: GaitSet: regarding gait as a set for cross-view gait recognition. In: AAAI (2019)
3. Chao, H., Wang, K., He, Y., Zhang, J., Feng, J.: GaitSet: cross-view gait recognition through utilizing gait as a deep set. TPAMI **44**(7), 3467–3478 (2021)
4. Connor, P., Ross, A.: Biometric recognition by gait: a survey of modalities and features. In: CVIU (2018)
5. Ding, X., Guo, Y., Ding, G., Han, J.: ACNet: strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks. In: ICCV (2019)
6. Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: RepVGG: making VGG-style convnets great again. In: CVPR (2021)
7. Fan, C., et al.: GaitPart: temporal part-based model for gait recognition. In: CVPR (2020)
8. Hermans, A., Beyer, L., Leibe, B.: In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737 (2017)
9. Hou, S., Cao, C., Liu, X., Huang, Y.: Gait lateral network: learning discriminative and compact representations for gait recognition. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12354, pp. 382–398. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58545-7_22
10. Hou, S., Liu, X., Cao, C., Huang, Y.: Set residual network for silhouette-based gait recognition. TBIOM **3**(3), 384–393 (2021)
11. Huang, Z., et al.: 3D local convolutional neural networks for gait recognition. In: ICCV (2021)
12. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: CCNet: criss-cross attention for semantic segmentation. In: ICCV (2019)
13. Jin, Y., Sharma, A., Tan, R.T.: DC-ShadowNet: single-image hard and soft shadow removal using unsupervised domain-classifier guided network. In: ICCV (2021)
14. Jin, Y., Yang, W., Tan, R.T.: Unsupervised night image enhancement: when layer decomposition meets light-effects suppression. arXiv preprint arXiv:2207.10564 (2022)
15. Li, S., Liu, W., Ma, H.: Attentive spatial-temporal summary networks for feature learning in irregular gait recognition. TMM **21**(9), 2361–2375 (2019)
16. Li, X., Makihara, Y., Xu, C., Yagi, Y., Ren, M.: Gait recognition invariant to carried objects using alpha blending generative adversarial networks. PR **105**, 107376 (2020)
17. Li, X., Makihara, Y., Xu, C., Yagi, Y., Ren, M.: Gait recognition via semi-supervised disentangled representation learning to identity and covariate features. In: CVPR (2020)
18. Li, X., Makihara, Y., Xu, C., Yagi, Y., Yu, S., Ren, M.: End-to-end model-based gait recognition. In: ACCV (2020)

19. Lin, B., Liu, Y., Zhang, S.: GaitMask: mask-based model for gait recognition. In: BMVC (2021)
20. Lin, B., Zhang, S., Bao, F.: Gait recognition with multiple-temporal-scale 3D convolutional neural network. In: ACM MM (2020)
21. Lin, B., Zhang, S., Liu, Y., Qin, S.: Multi-scale temporal information extractor for gait recognition. In: ICIP (2021)
22. Lin, B., Zhang, S., Wang, M., Li, L., Yu, X.: GaitGL: learning discriminative global-local feature representations for gait recognition. arXiv2208 (2022)
23. Lin, B., Zhang, S., Yu, X.: Gait recognition via effective global-local feature representation and local temporal aggregation. In: ICCV (2021)
24. Liu, J., et al.: Leaping from 2D detection to efficient 6DoF object pose estimation. In: Bartoli, A., Fusiello, A. (eds.) ECCV 2020. LNCS, vol. 12536, pp. 707–714. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-66096-3_47
25. Radenović, F., Tolias, G., Chum, O.: Fine-tuning CNN image retrieval with no human annotation. TPAMI **41**(7), 1655–1668 (2018)
26. Shen, C., Lin, B., Zhang, S., Huang, G.Q., Yu, S., Yu, X.: Gait recognition with mask-based regularization. arXiv preprint arXiv:2203.04038 (2022)
27. Shiraga, K., Makihara, Y., Muramatsu, D., Echigo, T., Yagi, Y.: GEINet: view-invariant gait recognition using a convolutional neural network. In: ICB (2016)
28. Song, C., Huang, Y., Huang, Y., Jia, N., Wang, L.: GaitNet: an end-to-end network for gait based human identification. PR **96**, 106988 (2019)
29. Takemura, N., Makihara, Y., Muramatsu, D., Echigo, T., Yagi, Y.: Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition. IPSJ Trans. Comput. Vis. Appl. **10**(1), 1–14 (2018). https://doi.org/10.1186/s41074-018-0039-6
30. Thapar, D., Jaswal, G., Nigam, A., Arora, C.: Gait metric learning Siamese network exploiting dual of spatio-temporal 3D-CNN intra and LSTM based inter gait-cycle-segment features. PRL **125**, 646–653 (2019)
31. Tian, Y., Yu, X., Fan, B., Wu, F., Heijnen, H., Balntas, V.: SOSNet: second order similarity regularization for local descriptor learning. In: CVPR (2019)
32. Wolf, T., Babaee, M., Rigoll, G.: Multi-view gait recognition using 3D convolutional neural networks. In: ICIP (2016)
33. Wu, H., Tian, J., Fu, Y., Li, B., Li, X.: Condition-aware comparison scheme for gait recognition. TIP **30**, 2734–2744 (2020)
34. Wu, Z., Huang, Y., Wang, L., Wang, X., Tan, T.: A comprehensive study on cross-view gait based human identification with deep CNNs. TPAMI **39**(2), 209–226 (2016)
35. Yeoh, T., Aguirre, H.E., Tanaka, K.: Clothing-invariant gait recognition using convolutional neural network. In: ISPACS (2016)
36. Yu, S., et al.: HID 2021: competition on human identification at a distance 2021. In: IJCB (2021)
37. Yu, S., Tan, D., Tan, T.: A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In: ICPR (2006)
38. Yu, X., et al.: Unsupervised extraction of local image descriptors via relative distance ranking loss. In: ICCV Workshops (2019)
39. Yu, X., Zhuang, Z., Koniusz, P., Li, H.: 6DoF object pose estimation via differentiable proxy voting loss. In: BMVC (2020)
40. Zhang, C., Liu, W., Ma, H., Fu, H.: Siamese neural network based gait recognition for human identification. In: ICASSP (2016)

41. Zhang, J., et al.: Gigapixel whole-slide images classification using locally supervised learning. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) Medical Image Computing and Computer Assisted Intervention–MICCAI 2022. Lecture Notes in Computer Science, vol. 13432, pp. 192–201. Springer, Cham (2022)
42. Zhang, X., et al.: Sleep stage classification based on multi-level feature learning and recurrent neural networks via wearable device. Comput. Biol. Med. **103**, 71–81 (2018)
43. Zhang, Y., Huang, Y., Wang, L., Yu, S.: A comprehensive study on gait biometrics using a joint CNN-based method. PR **93**, 228–236 (2019)
44. Zhang, Y., Huang, Y., Yu, S., Wang, L.: Cross-view gait recognition by discriminative feature learning. TIP **29**, 1001–1015 (2019)
45. Zhu, Z., et al.: Gait recognition in the wild: a benchmark. In: ICCV (2021)