GaitSet: Cross-View Gait Recognition Through Utilizing Gait As a Deep Set

Hanqing Chao, Kun Wang, Yiwei He, Junping Zhang, *Member, IEEE*, and Jianfeng Feng

Abstract—Gait is a unique biometric feature that can be recognized at a distance; thus, it has broad applications in crime prevention, forensic identification, and social security. To portray a gait, existing gait recognition methods utilize either a gait template which makes it difficult to preserve temporal information, or a gait sequence that maintains unnecessary sequential constraints and thus loses the flexibility of gait recognition. In this paper, we present a novel perspective that utilizes gait as a *deep set*, which means that a set of gait frames are integrated by a global-local fused deep network inspired by the way our left- and right-hemisphere processes information to learn information that can be used in identification. Based on this *deep set* perspective, our method is *immune to frame permutations*, and can naturally *integrate frames from different videos* that have been acquired under different scenarios, such as diverse viewing angles, different clothes, or different item-carrying conditions. Experiments show that under normal walking conditions, our single-model method achieves an average rank-1 accuracy of 96.1 percent on the CASIA-B gait dataset and an accuracy of 87.9 percent on the OU-MVLP gait dataset. Under various complex scenarios, our model also exhibits a high level of robustness. It achieves accuracies of 90.8 and 70.3 percent on CASIA-B under bag-carrying and coat-wearing walking conditions respectively, significantly outperforming the best existing methods. Moreover, the proposed method maintains a satisfactory accuracy even when only small numbers of frames are available in the test samples; for example, it achieves 85.0 percent on CASIA-B even when using only 7 frames. The source code has been released at https://github.com/AbnerHqC/GaitSet

Index Terms—Gait recognition, biometric authentication, GaitSet, deep learning

---

1 INTRODUCTION

UNLIKE other biometric identification sources such as a face, fingerprint, or iris, gait is a unique biometric feature that can be recognized from a distance without any intrusive interactions with subjects. This characteristic gives gait recognition high potential for use in applications such as crime prevention, forensic identification, and social security.

However, a person's variational poses in walking, which forms the basic information for gait recognition, is easily affected by exterior factors such as the subject's walking speed, clothing, and item-carrying condition as well as the camera's viewpoint and frame rate. These factors make gait recognition very challenging, especially cross-view gait recognition, which seeks to identify gait that might be captured from different angles. It thus is crucial to develop a practical gait recognition system.

The existing works have tried to tackle the problem from two aspects. They either regard gait as a single image or regard it as a video sequence. Methods in the first category

compress all gait silhouettes into one image, i.e., a gait template, for gait recognition [1], [2], [3], [4], [5], [6], [7]. Although various existing gait templates [5], [6], [7] encode information as abundantly as possible, the compression process omits significant features such as temporal information and fine-grained spatial information. To address this issue, the methods in the second category extract features directly from the original gait silhouette sequences [8], [9], [10]. These methods preserve more temporal information but would suffer a significant degradation when an input contains discontinuous frames or has a frame rate different from the training dataset.

To solve these problems, we present a novel perspective that regards gait as a set of gait silhouettes. Because gait is a periodic motion, it can be represented by a single period. Meanwhile, in a silhouette sequence containing one gait period, it can be observed that the silhouette in each position has a unique pose, as shown in Fig. 1. Given anyone's gait silhouettes, we can easily rearrange them into the correct order solely by observing their appearance. This suggests that the order of the poses in a gait period is not a key information to differentiate one person from others, since the pattern of the order is universal. Based on such an assumption, we can directly regard gait as a set of images and extract temporal information without ranking each frame like a video.

From this perspective, we propose an end-to-end deep learning model called Gaitset that extracts features from a gait frame set to identify gaits. Fig. 2 shows the overall scheme of Gaitset. The input to our model is a set of gait silhouettes. First, a CNN is used to extract frame-level features from each silhouette independently (local information).

- Hanqing Chao, Kun Wang, Yiwei He, and Junping Zhang are with the Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai 200438, China. E-mail: {hqchao16, KunWang17, heyw15, jpzhang}@fudan.edu.cn.
- Jianfeng Feng is with the Institute of Science and Technology for Brain-inspired Intelligence, Fudan University, Shanghai 200438, China. E-mail: jianfeng64@gmail.com.

Fig. 1. From top-left to bottom-right are silhouettes of a completed period of a subject in the *CASIA-B* gait dataset.

Second, an operation called Set Pooling is used to aggregate frame-level features into a single *set-level* feature (global information). Because this operation is conducted using high-level feature maps instead of the original silhouettes, it preserves spatial and temporal information better than a gait template; this aspect is experimentally validated in Section 4.5. The global-local fused deep network resembles the way our brain processes information [11]. Third, a structure called Horizontal pyramid mapping (HPM) is applied to project the set-level feature into a more discriminative space to obtain a final deep set representation. The superiority of the proposed method can be summarized into the following three aspects:

- *Flexible:* Our model is pretty flexible since it imposes no constraints on the input except the size of the silhouette. This means that the input set can consist of any number of nonconsecutive silhouettes filmed under different viewpoints with different walking conditions. Related experiments are presented in Section 4.7

- *Fast:* Our model directly learns the deep set gait representation of gait instead of measuring the similarity between a pair of gait templates or sequences. Thus, the representation of each sample only needs to be computed once and the recognition can be done by comparing euclidean distance between the representations of different samples.

- *Effective:* Our model substantially improves the state-of-the-art performance on the CASIA-B [12] and the OU-MVLP [13] datasets, exhibiting strong robustness to view and walking condition variations and high generalization ability to large datasets.

Compared with our previous AAAI-19 conference paper on this topic, we have extended our work in four ways: 1) we surveyed and compared more state-of-the-art gait recognition algorithms; 2) we conducted more comprehensive experiments to evaluate the performance of the proposed GaitSet model; 3) we achieved better performance by improving the loss function used in GaitSet; 4) a post feature dimension reduction module were included to enhance the practicality.

2 RELATED WORKS

In this section, we briefly survey developments in gait recognition and set-based deep learning methods.

2.1 Gait Recognition

Gait recognition can be broadly categorized into template-based and sequence-based approaches. In the former category, the previous works have generally divided this pipeline into two parts, i.e., template generation and matching. The goal of template generation is to compress gait information into a single image, e.g., a Gait Energy Image (GEI) [14] or a Chrono-Gait Image (CGI) [15]. To generate a template, these approaches first estimate the human silhouettes in each frame through background removal. Then, they generate a gait template by applying pixel level operators to the aligned silhouettes [15]. In the template matching procedure, they first extract the gait representation from a template image using machine learning approaches such as canonical correlation analysis (CCA) [16], linear discriminant analysis (LDA) [1], [17] and deep learning [18]. Then, they measure the similarity between pairs of representations using euclidean distance or other metric learning approaches [1], [3], [4], [7]. For example, the view transformation model (VTM) learns a projection between different views [19]; [2] proposed view-invariant discriminative projection (ViDP) to project the templates into a latent space to learn a view-invariant representation. Finally, they assign a label to the template based on the measured distance using a classifier, e.g., SVM or nearest neighbor classifier [4], [6], [7], [18], [20], [21].

In the second category, the video-based approaches directly take a sequence of silhouettes as an input. For
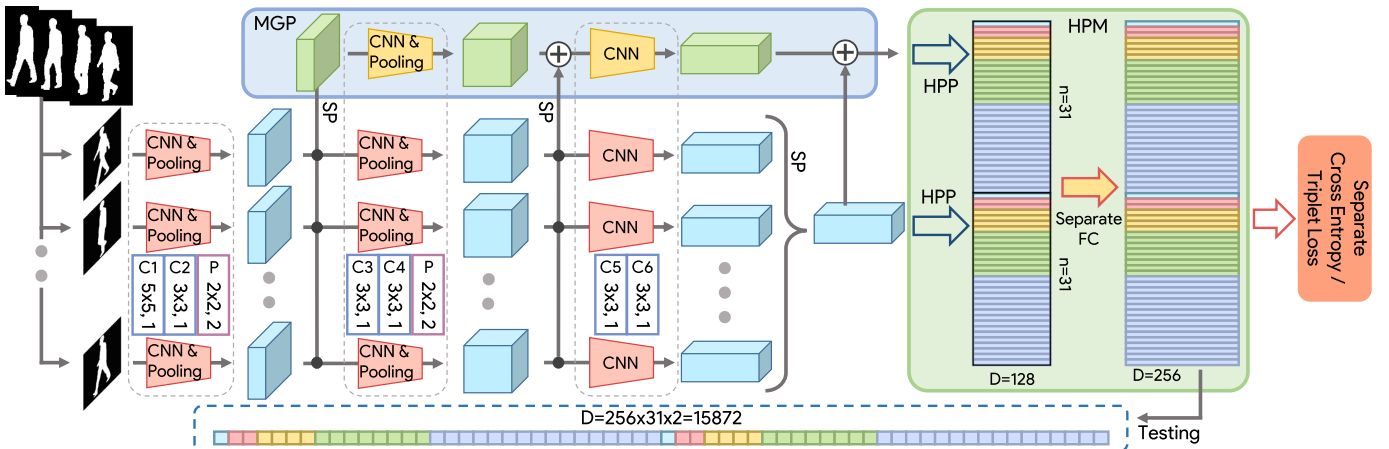

Fig. 2. The framework of GaitSet [26] . 'SP' represents set pooling. Trapezoids represent convolution and pooling blocks and those in the same column have the same configurations, as shown by the rectangles with capital letters. Note that although the blocks in MGP have the same configurations as those in the main pipeline, the parameters are shared only across blocks in the main pipeline – not with those in MGP. HPP represents horizontal pyramid pooling [27].

instance, the 3D CNN-based approaches [7], [9] extract temporal-spatial information using 3D convolutions; Liao [8] and An [c] utilize human skeletons to lean gait features which is robust to the change of clothing; [23] fused sequential frames by LSTM attention units; and [10] proposed the spatial-temporal graph attention network (STGAN) to uncover the graph relationships between gait frames, followed by obtaining attention for gait video. In recent, a part-based model [a] was proposed to capture spatial-temporal feature of each part and achieved promising results. To prevent redundancy feature in part-based model, two stage training strategy was used in [b] to learn compact features successfully.

In the second category, the video-based approaches directly take a sequence of silhouettes as an input. For instance, the 3D CNN-based approaches [7], [9] extract temporal-spatial information using 3D convolutions; Liao *et al.* [8] and An *et al.* [22] utilized human skeletons to learn gait features robust to the change of clothing; Zhang *et al.* [23] fused sequential frames by LSTM attention units; and Wu *et al.* [10] proposed the spatial-temporal graph attention network (STGAN) to uncover the graph relationships between gait frames, followed by obtaining attention for gait video. Recently, a part-based model [24] was proposed to capture spatial-temporal features of each part and achieved promising results. To prevent redundancy feature in part-based model, a two-stage training strategy was used in [25] to learn compact features effectively.

2.2 Deep Learning on an Unordered Set

Most deep learning works have been focusing on regular input representations such as video sequences or images. The initial goal for using unordered sets was to address point cloud tasks in the computer vision domain [28] based on PointNet. Using an unordered set, PointNet can avoid the noise introduced by quantization and the extension of data, leading to a high prediction performance. Since then, set-based methods have been widely used in the point cloud domain [29], [30], [31] and in content recommendation [32] and image captioning [33] by integrating features into the form of a set. [34] further formalized the deep learning tasks defined for sets and characterized of the permutations using invariant functions. To the best of our knowledge, this topic has not been studied in depth in the gait recognition domain except in our previous AAAI-19 conference version of this paper [26].

3 GAITSET

In this section, we introduce the details of our GaitSet method, which learns deep discriminative information from a set of gait silhouettes. To improve understanding, the overall pipeline is illustrated in Fig. 2.

3.1 Problem Formulation

The concept for regarding gait as a deep set will be formulated first. Given a dataset of N people with identities $y_i, i \in 1, 2, \ldots, N$, we assume that the gait silhouettes of a certain person are subject to a distribution $\mathcal{P}_i$ which is uniquely related to that individual. Therefore, all silhouettes in one or more sequences of a given person can be regarded as a

set of n silhouettes $\mathcal{X}_i = \{x_i^j | j = 1, 2, \ldots, n\}$, where $x_i^j \sim \mathcal{P}_i$. Under this assumption, we tackle the gait recognition task via 3 steps, formulated as follows:

$$f_i = H(G(F(x_i^1), F(x_i^2), \ldots, F(x_i^n))) \tag{1}$$

where F is a convolutional network that seeks to extract frame-level features from each gait silhouette. The function G is a permutation invariant function used to map a set of frame-level features to a set-level feature [34] based on set pooling (SP), which will be introduced in Section 3.2. The function H learns the deep set discriminative representation of $\mathcal{P}_i$ from the set-level feature through a structure called horizontal pyramid mapping (HMP) which will be discussed in Section 3.3. The input $\mathcal{X}_i$ is a tensor with four dimensions: set dimension, image channel dimension, image height dimension, and image width dimension.

3.2 Set Pooling

The goal of Set Pooling (SP) is to condense a set of gait information, formulated as $z = G(V)$, where z denotes the set-level feature and $V = \{v^j | j = 1, 2, \ldots, n\}$ denotes the frame-level features, where v^j means the j-th frame-level feature map and n denotes the number of gait frames in a set. Note that there are two constraints when performing an SP operation. First, if we expect to take a set as an input, the function should be a permutation invariant function satisfying:

$$G(\{v^j | j = 1, 2, \ldots, n\}) = G(\{v^{\pi(j)} | j = 1, 2, \ldots, n\}) \tag{2}$$

where π is any permutation [34]. Second, the function G should be able to take a *set* with arbitrary cardinality because the number of a person's gait silhouettes can be arbitrary in a real-world scenario. Next, we describe several instantiations of G. The experiments will show that although different instantiations of SP do influence the performances, they do not produce significant differences, and all SP instantiations achieve better performances than do the GEI-based methods by a large margin.

Basic Statistical Functions. To meet the invariant constraint requirement in Eq. (2), one rational strategy of SP is to use statistical functions on the set dimension. To balance the representativeness and the computational cost, we considered three statistical functions: $\max(\cdot)$, $\mathrm{mean}(\cdot)$ and $\mathrm{median}(\cdot)$. A comprehensive comparison will be given in Section 4.

Joint Functions. Further, two feasible ways to join the aforementioned 3 basic statistical functions are analyzed as follows:

$$G(\cdot) = \max(\cdot) + \mathrm{mean}(\cdot) + \mathrm{median}(\cdot) \tag{3}$$

$$G(\cdot) = 1\_1\mathrm{C}(\mathrm{cat}(\max(\cdot), \mathrm{mean}(\cdot), \mathrm{median}(\cdot))) \tag{4}$$

where cat means concatenating on the channel dimension, 1_1C denotes 1×1 convolutional layer, and $\max$, mean and median are applied to the set dimension. Eq. (4) is an extended version of Eq. (3), which allows the 1×1 convolutional layer to learn a proper weight for combining information extracted by different statistical functions.

Attention. Visual attention has been successfully applied in many computer vision tasks [35], [36], [37], and we also
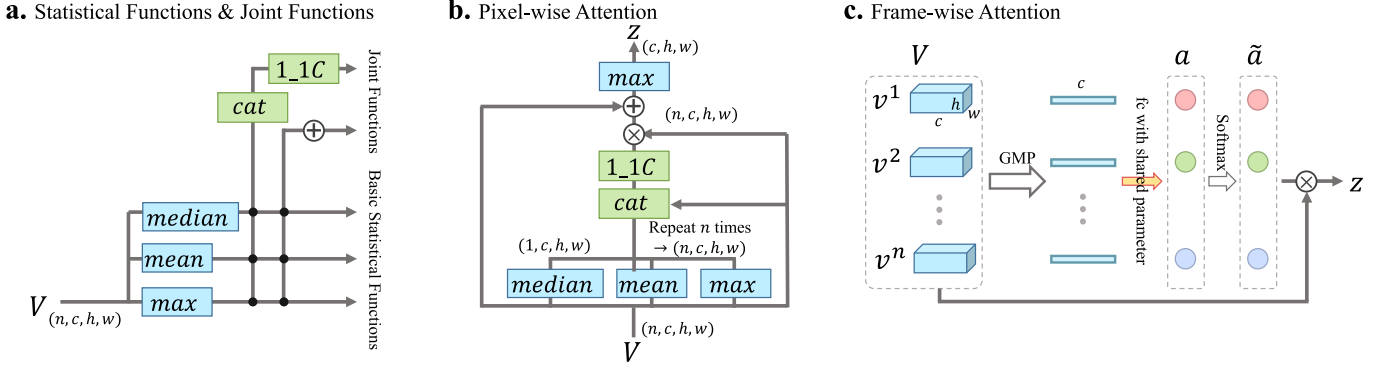
a. Statistical Functions & Joint Functions   **b.** Pixel-wise Attention   **c.** Frame-wise Attention

Fig. 3. Seven different instantiations of Set Pooling (SP). 1_1C and cat represent the 1×1 convolutional layer and the concatenate operation, respectively. Here, n represents the number of feature maps in a set, and c, h and w denote the number of channels, the height and width of a feature map, respectively. **a.** Three basic statistical SP and two joint SP. **b.** Pixel-wise attention SP. **c.** Frame-wise attention SP.

capitalize on attention to implement SP. We included two attention strategies in our work. The first one is a *pixel-wise attention*. Specifically, we refine the output of SP by utilizing the global information to learn an element-wise attention map for each frame-level feature map, as shown in Fig. 3b. First, global information is first collected by the statistical functions. Then, it is input into a 1×1 convolutional layer along with the original feature map to calculate an attention map for the refinement. The final set-level feature z is extracted by employing a MAX operation on the set of the refined frame-level feature maps. We use the residual structure to accelerate and stabilize the convergence. Another is a *frame-wise attention*, in which global max pooling is first applied on each v^j to get a compressed frame-wise feature. Then, based on the frame-wise feature, a fully connected layer is applied to calculate a frame-wise weight a^j. Finally, z is calculated by $\sum_{j=1}^{n} \tilde{a}^j v^j$, where $\tilde{a}^j$ is the softmax-normalized frame-wise weight. Fig. 3c illustrates the architecture of the frame-wise attention.

3.3 Horizontal Pyramid Mapping

In the literature, splitting feature maps into strips is a commonly used tactic in person re-identification tasks [27], [38]. For instance, [27] proposed horizontal pyramid pooling (HPP) through cropping and resizing the images into a uniform size based on pedestrian size while varying the discriminative parts from image to image. With 4 scales, HPP can thus help the deep network gather both local and global information by focusing on features with different sizes. Here, we improve HPP to adapt it to the gait recognition task; instead of applying a 1×1 convolutional layer after the pooling, we use independent fully connected layers (FC) for each pooled feature to map it into the discriminative space, as shown in Fig. 4. We call this approach horizontal pyramid mapping (HPM).

Concretely, HPM has S scales. On scale $s \in 1, 2, \ldots, S$, the feature map extracted by SP is split into 2^{s-1} strips on the height dimension, i.e., $\sum_{s=1}^{S} 2^{s-1}$ strips in total. Then, global pooling is applied to the 3-D strips to obtain 1-D features. For a strip $z_{s,t}$ where $t \in 1, 2, \ldots, 2^{s-1}$ represents the index of the strip in the scale, the global pooling is formulated as $f'_{s,t} = maxpool(z_{s,t}) + avgpool(z_{s,t})$, where $maxpool$ and $avgpool$ denote global max pooling (GMP) and global average pooling (GAP) respectively. Note that the functions $maxpool$ and $avgpool$ are used at the same time because the combined

results outperform applying either operation alone. The final step is to employ FCs to map the features f' into a deep discriminative space. Because the strips at different scales depict features of different receptive fields, and different strips at each scale depict features of different spatial positions, using independent FCs is a natural choice, as shown in Fig. 4.

3.4 Multilayer Global Pipeline

Generally, different layers of a convolutional network have different receptive fields. The deeper a layer is, the larger the receptive field will be. Thus, pixels in the feature maps of a shallow layer pay more attention to local and fine-grained information while those in deeper layers focus more on global and coarse-grained information. The set-level features extracted by applying SP to different layers are analogous. As shown in the main pipeline of Fig. 2, only one SP is applied to the last layer of the convolutional network. To collect different-level *set* information, we propose a multilayer global pipeline (MGP), which has a similar structure to the convolutional network in the main pipeline; however, we add the set-level features extracted by different layers to MGP. The final feature map generated by MGP is also mapped into $\sum_{s=1}^{S} 2^{s-1}$ features by HPM. Note that 1) the HPM that executes after MGP does not share parameters with the HPM that executes after the main pipeline. 2) The main pipeline is similar to that of human cognition, which focuses intuitively on a person's profile, whereas the MGH can preserve more details of a person's walking movements.

3.5 Loss Functions and Training Strategy

In the field of identification [39], [40], [41], two loss functions are widely used, i.e., cross entropy loss and triplet
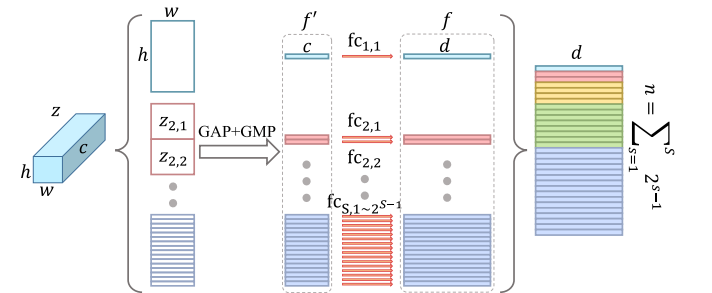
Fig. 4. The structure of horizontal pyramid mapping [26].

loss [40]. To obtain the best performance, we conducted comprehensive experiments on these two loss functions.

Cross-Entropy Loss is common in classification tasks. It measures the gap between a predictive distribution and the corresponding true distribution. In the recognition task, the output classes reflect all labels (identities) in the training set. As mentioned above, the output of the network are $2 \times \sum_{s=1}^{S} 2^{s-1}$ features with a dimension of d. During training, a cross entropy loss was calculated for each feature and then all losses were summed up as a total loss. During testing, the feature before the softmax layer is used for recognition.

Triplet loss was initially proposed for face recognition [40] but has become a popular loss function for metric embedding learning and has achieved high performances on various tasks [26], [40], [42], [43], [44]. It aims to pull semantically-similar points close to each other while pushing semantically-different points away from each other [42]. The specific version of triplet loss adopted in this paper is Batch All (BA_+) triplet loss [42].

Specifically, we denote a sample triplet as $r = (\alpha, \beta, \gamma)$ where α represents the anchor, β is a sample with the same label as the anchor α, and γ denotes a sample with a label different from that of the anchor α. Then, the BA_+ triplet loss of this triplet is defined as follows:

$$L(r) = ReLU(\xi + D_{\alpha,\beta} - D_{\alpha,\gamma}), \tag{5}$$

where ξ is the margin between intraclass distance $D_{\alpha,\beta}$ and interclass distance $D_{\alpha,\gamma}$. For a triplet r, each sample has $2 \times \sum_{s=1}^{S} 2^{s-1}$ features. We calculated the triplet loss for each corresponding feature triplet, i.e., $2 \times \sum_{s=1}^{S} 2^{s-1}$ triplet losses were calculated.

Using a Combination of Two Loss Functions. Our previous work [26] only used Batch All (BA_+) triplet loss and achieved state-of-the-art performance. In this study, to improve the learning ability, we combined the cross entropy loss with the triplet loss. First, cross entropy loss was used to train the network to converge. Then, a smaller learning rate with Batch All (BA_+) triplet loss was used to let the model find a more discriminant metric space. Experiments to compare these two losses are shown in Section 4.5.2.

3.6 Training and Test

Training. In the training phase, a batch with a size of $p \times k$ is sampled from the training set, where p denotes the number of persons and k is the number of training samples each person has in the batch. Note that although the experiment shows that our model performs well when its input is the set composed of silhouettes gathered from arbitrary sequences, during the training phase, a sample is composed only of silhouettes sampled in one sequence.

Testing. Given a query $\mathcal{Q}$, the goal is to retrieve all the *sets* with the same identity in gallery set $\mathbb{G}$. We denote a sample in $\mathbb{G}$ as $\mathcal{G}$. First, $\mathcal{Q}$ is input into GaitSet model to generate $2 \times \sum_{s=1}^{S} 2^{s-1}$ multiscale features, then all these features are concatenated into a final representation $\mathcal{F}_{\mathcal{Q}}$ as shown in Fig. 2. The same process is applied to each $\mathcal{G}$ to obtain $\mathcal{F}_{\mathcal{G}}$. Finally, $\mathcal{F}_{\mathcal{Q}}$ is compared with every $\mathcal{F}_{\mathcal{G}}$ to calculate the rank-1 recognition accuracy, which means the percentage of the correct subjects ranked first, based on nearest euclidean distance.

3.7 Post Feature Dimension Reduction

As introduced in Section 3.6, the identification is achieved by comparing $\mathcal{F}_{\mathcal{Q}}$ with every $\mathcal{F}_{\mathcal{G}}$ to find the nearest neighbor. Let d_f denotes the dimension of the final representation $\mathcal{F}$. The computational complexity of this identification process is $O(d_f|\mathbb{G}|)$ where $|\cdot|$ calculates the cardinality of a set. In practical applications, $|\mathbb{G}|$ could be extremely large. A small d_f will be the key to keep the process efficient. Thus, we proposed a post feature dimension reduction module which is a post trained linear projection to reduce the dimension of the output feature while maintaining a competitive recognition accuracy. Specifically, based on a trained GaitSet model with fixed parameters, we feed the learned d_f dimensional feature into a fully connected layer with an output of dimension d_f'. The new d_f' dimensional features are used to calculate a triplet loss to train the fully connected layer. In Section 4.6 we show that compared with directly shrinking the output dimension of the HPM, our post feature dimension reduction model can effectively reduce the feature dimension into a significantly smaller size while preserving a high recognition performance.

4 EXPERIMENTS

In this section, we report the results of comprehensive experiments conducted to evaluate the performance of the proposed GaitSet. First, we compared GaitSet with other state-of-the-art methods on two public gait datasets: CASIA-B [12] and OU-MVLP [13]. Then, we conducted a set of ablation studies on CASIA-B. Third, we studied the effectiveness of feature dimension reduction. Finally, we analyzed the practicality of GaitSet from three aspects: limited silhouettes, multiple views, and multiple walking conditions.

4.1 Datasets

The *CASIA-B* dataset [12] is a popular gait dataset that contains 124 subjects labeled from 001 to 124. Each subject has 3 walking conditions, i.e., normal (NM) (6 video sequences per subject), walking with a bag (BG) (2 video sequences per subject) and wearing a coat or jacket (CL) (2 video sequences per subject). Each sequence is simultaneously framed under 11 views ($0°, 18°, \ldots, 180°$). Thus, this dataset contains $124 \times (6 + 2 + 2) \times 11 = 113,640$ videos in total. Since this dataset does not include official training and test subset partitions, we conducted our experiments using three kinds of division popular in the current literature. Based on the sizes of the training sets, we name these three kinds of division small-sample training (ST), medium-sample training (MT) and large-sample training (LT). In ST, the first 24 subjects (001-024) were used for training and the remaining 100 subjects were used for testing with no overlap. In MT, the first 62 subjects (001-062) were used for training, and the remaining 62 subjects were used for testing. In LT, the first 74 subjects (001-074) were used for training and the remaining 50 subjects were used for testing. For the test sets in all three settings, the first 4 sequences of the NM condition (i.e., NM #1-4) were kept in the gallery, and the remaining 6 sequences were divided into 3 probe subsets, i.e., the NM subset condition #5-6, the BG subset containing BG #1-2 and the CL subset containing CL #1-2.

TABLE 1
Batch Size (B_S), Learning Rate (L_R), and Training Iterations (Iter) on OU-MVLP and
the Three Settings of CASIA-B (CASIA-ST, CASIA-MT, and CASIA-LT)

		CASIA-ST	CASIA-MT	CASIA-LT	OU-MVLP
CE	B_S	128	128	128	512
	L_R	1e-4	1e-4	1e-4	Iter $<$ 150k: 1e-4; Iter $>$ =150k: 1e-5
	Iter	70k	80k	90k	800k
Triplet Only	B_S	$p = 8, k = 16$	$p = 8, k = 16$	$p = 8, k = 16$	$p = 32, k = 16$
	L_R	1e-4	1e-4	1e-4	Iter $<$ 150k: 1e-4; Iter $>$ =150k: 1e-5
	Iter	80k	80k	100k	800k
Triplet Tune	B_S	$p = 8, k = 16$	$p = 8, k = 16$	$p = 8, k = 16$	$p = 32, k = 16$
	L_R	1e-5	1e-5	1e-5	Iter $<$ 600k: 1e-4; Iter $>$ =600k: 1e-5
	Iter	30k	40k	60k	700k

When being trained with cross entropy loss, a mini-batch is randomly selected from the training set. When being trained with triplet loss, a mini-batch is composed as described in Section 3.6.

The *OU-MVLP* dataset [13] is currently the largest public gait dataset. It contains 10,307 subjects with 14 views ($0°$, $15°, \ldots, 90°; 180°, 195°, \ldots, 270°$) per subject and 2 sequences (#00-01) per view. According to its protocol, the sequences of 5,154 subjects are used for training, and the sequences of the remaining 5,154 subjects are used for testing. In the test set, sequences with index #01 are kept in the gallery and those with index #00 are used as probes.

4.2 Parameter Setting

In all the experiments, the input was a set of aligned silhouettes of size 64×44. The silhouettes were directly provided by the datasets and were aligned based on the methods described in [13]. We adopted the Adam optimizer [45] for training our GaitSet network. The code for all the experiments was written in Python with Pytorch 0.4.0. The models were trained on a computer equipped with 4 NVIDIA 1080TI GPUs. Unless otherwise stated, the set cardinality during the training phase was set to 30. The margin ξ in the triplet loss function (Eq. (5)) was set to 0.2. The number of HPM scales S was set to 5. *1)*, For the *CASIA-B* dataset, we set the number of channels in $C1$ and $C2$ as 32, in $C3$ and $C4$ as 64, and in $C5$ and $C6$ as 128. Under these settings, the average computational complexity of our model is 8.6GFLOPs. *2)* On the *OU-MVLP* dataset, which contains 20 times more sequences than CASIA-B, we used convolutional layers with more channels, i.e., $C1 = C2 = 64, C3 = C4 = 128, C5 = C6 = 256$. For convenience, the details of batch size, learning rate, and training iterations on different experimental settings are listed in Table 1. Furthermore, rank-1 accuracy is adopted as a criterion in the subsequent evaluations.

4.3 Brief Introduction of Compared Methods

Among our compared methods, View-invariant Discriminative Projection (ViDP) [2] uses a unitary linear projection to project the templates intoa latent space to learn a view-invariant represent. Correlated Motion Co-Clustering (CMCC) [46] first uses motion co-clustering to partition the most related parts of gaits from different views into the same group, and then applies canonical correlation analysis (CCA) on each group to maximize the correlation between gait information across views. Wu *et al.* proposed several

CNN based models in [7]. CNN-LB feeds GEIs of two gait sequences into a 3-layer CNN as two channels and judges whether the two GEIs belong to the same person; CNN-3D runs 3-layers 3D-CNN on 9 adjacent frames and averages predictions of 16 9-frame samples to get the final output; CNN-Ensemble aggregates outputs of 8 different networks and achieves the best performance in this work. Yu *et al.* [21] applied AutoEncoder (AE) to extract view-invariant features. He *et al.* [21] proposed a multi-task GAN (MGAN) to project gait features from one angle to another angle for multi-view gait recognition. Angle Center Loss (ACL) [23] which is robust to different local parts and temporal window sizes is proposed to learn discriminative gait features. GEINet [18] classifies GEIs of different persons with a 2-layer CNN followed by 2 fully connected layers. Takemura *et al.* [4] improved the structures proposed in [7] to leverage triplet loss.

4.4 Main Results

4.4.1 CASIA-B

Table 2 shows a comparison between the state-of-the-art methods[1] and the proposed GaitSet. Except for GaitSet, the other results were directly taken from their original papers. All the results were averaged on the 11 gallery views and the identical views were excluded. For example, the accuracy of probe view $36°$ was averaged on 10 gallery views, excluding gallery view $36°$. From Table 2, an interesting relationship between views and accuracies can be observed. In addition to $0°$ and $180°$, where low accuracies are expected, the accuracy for $90°$ is a local minimum value that is always worse than the accuracy for $72°$ or $108°$. The possible reason is that gait contains feature information not only those parallel to the walking direction, such as stride, which can be observed most clearly at $90°$ but also feature information vertical to the walking direction, such as the left-right swinging motions of the body or arms, which can be observed most clearly at $0°$ or $180°$. Therefore, both parallel and vertical perspectives lose some portion of the gait information while views such as $36°$ and $144°$ achieve a better balance between these two extremes.

1. Since Wu *et al* proposed more than one model [7], we cited the most competitive results under different experimental settings.

TABLE 2
Averaged Rank-1 Accuracies on *CASIA-B* Under Three Different Experimental Settings, Excluding Identical-View Cases

Gallery NM#1-4			0°–180°											
	Probe		0°	18°	36°	54°	72°	90°	108°	126°	144°	162°	180°	mean
ST (24)	NM#5-6	ViDP [2]	–	–	–	59.1	–	50.2	–	57.5	–	–	–	–
		CMCC [46]	46.3	–	–	52.4	–	48.3	–	56.9	–	–	–	–
		CNN-LB [7]	54.8	–	–	77.8	–	64.9	–	76.1	–	–	–	–
		GaitSet(ours)	**71.6**	**87.7**	**92.6**	**89.1**	**82.4**	**80.3**	**84.4**	**89.0**	**89.8**	**82.9**	**66.6**	**83.3**
	BG#1-2	GaitSet(ours)	64.1	76.4	81.4	82.4	77.2	71.8	75.4	80.8	81.2	75.7	59.4	75.1
	CL#1-2	GaitSet(ours)	36.4	49.7	54.6	49.7	48.7	45.2	45.5	48.2	47.2	41.4	30.6	45.2
MT (62)	NM#5-6	AE [21]	49.3	61.5	64.4	63.6	63.7	58.1	59.9	66.5	64.8	56.9	44.0	59.3
		MGAN [6]	54.9	65.9	72.1	74.8	71.1	65.7	70.0	75.6	76.2	68.6	53.8	68.1
		GaitSet(ours)	**89.7**	**97.9**	**98.3**	**97.4**	**92.5**	**90.4**	**93.4**	**97.0**	**98.9**	**95.9**	**86.6**	**94.3**
	BG#1-2	AE [21]	29.8	37.7	39.2	40.5	43.8	37.5	43.0	42.7	36.3	30.6	28.5	37.2
		MGAN [6]	48.5	58.5	59.7	58.0	53.7	49.8	54.0	61.3	59.5	55.9	43.1	54.7
		GaitSet(ours)	**79.9**	**89.8**	**91.2**	**86.7**	**81.6**	**76.7**	**81.0**	**88.2**	**90.3**	**88.5**	**73.0**	**84.3**
	CL#1-2	AE [21]	18.7	21.0	25.0	25.1	25.0	26.3	28.7	30.0	23.6	23.4	19.0	24.2
		MGAN [6]	23.1	34.5	36.3	33.3	32.9	32.7	34.2	37.6	33.7	26.7	21.0	31.5
		GaitSet(ours)	**52.0**	**66.0**	**72.8**	**69.3**	**63.1**	**61.2**	**63.5**	**66.5**	**67.5**	**60.0**	**45.9**	**62.5**
LT (74)	NM#5-6	CNN-3D [7]	87.1	93.2	97.0	94.6	90.2	88.3	91.1	93.8	96.5	96.0	85.7	92.1
		CNN-Ensemble [7]	88.7	95.1	98.2	96.4	94.1	91.5	93.9	97.5	98.4	95.8	85.6	94.1
		ACL [23]	**92.0**	98.5	**100.0**	**98.9**	**95.7**	91.5	94.5	97.7	98.4	96.7	**91.9**	96.0
		GaitSet(ours)	91.1	**99.0**	99.9	97.8	95.1	**94.5**	**96.1**	**98.3**	**99.2**	**98.1**	88.0	**96.1**
	BG#1-2	CNN-LB [7]	64.2	80.6	82.7	76.9	64.8	63.1	68.0	76.9	82.2	75.4	61.3	72.4
		GaitSet(ours)	**86.7**	**94.2**	**95.7**	**93.4**	**88.9**	**85.5**	**89.0**	**91.7**	**94.5**	**95.9**	**83.3**	**90.8**
	CL#1-2	CNN-LB [7]	37.7	57.2	66.6	61.1	55.2	54.6	55.2	59.1	58.9	48.8	39.4	54.0
		GaitSet(ours)	**59.5**	**75.0**	**78.3**	**74.6**	**71.4**	**71.3**	**70.8**	**74.1**	**74.6**	**69.4**	**54.1**	**70.3**

Small-Sample Training (ST). Our method achieved a high performance even with only 24 subjects in the training set and exceeded the best performance reported [7] by well over 10 percent on the reported values. There are two main reasons for these results. *1)* Because our model regards the input as a set, the number of samples (frames) available for training the convolutional network in the main pipeline is dozens of times higher than the number of samples used to train the template- or video-based models. Taking a mini-batch as an example, our model input consists of $30 \times 128 = 3,840$ silhouettes, while under the same batch size other template-based models only obtain 128 templates. *2)* Because the sample *sets* used in the training phase are composed of frames selected randomly from the sequence in the training set, each of which can generate multiple different sample *sets*; thus, any units related to set feature learning (such as MGP and HPM) can also be trained well.

In addition, it is noteworthy that in ST, all the other compared models were trained and tested only on the NM subset, whereas our model was trained and tested on all the NM, BG, and CL subsets. If we instead focus our model on only one subset, it performs even better, because then, the training and testing environments are the same and exhibit more consistency.

Medium-Sample Training (MT) & Large-Sample Training (LT). Generally, the performance of deep learning models depends heavily on the scale of the training sets. Thus, we evaluated our GaitSet using two different divisions for the training set and test set, i.e., MT and LT, as recommended in the prior literature. Table 2 shows that our model attains fairly good results on the NM subset, especially on LT, where the results of all views except $180°$ are over 90 percent. This result shows that the accuracy gains obvious

improvement when more training data is available for this subset.

Our model achieves satisfactory performance on the BG subset. On the CL dataset, the recognition performances are somewhat less satisfactory, although our model still exceeds the best performance reported so far [7] by over 15 percent. This reduced performance may be explained by the following three reasons: *1)* a coat can entirely change a person's appearance, e.g., a subject looks larger in a coat than in a T-shirt. *2)* A coat can hide the motions of both limbs and body. *3)* In the training set, the ratio of the CL subset is substantially lower than that of the NM subset, demanding stronger discriminative ability on the part of the model.

4.4.2 OU-MVLP

Table 3 compares GaitSet with the two other methods on the OU-MVLP dataset. As some of the previous works did not conduct experiments on all 14 views, we list our results based on two types of gallery sets, i.e., all 14 views and 4 typical views $(0°, 30°, 60°, 90°)$. All the results are averaged on the gallery views and identical views are excluded. The results show that our method generalizes well to a dataset with a large scale and wide view variation. Moreover, because the representation for each sample only needs to be calculated once, our model can complete the test involving all 133,780 sequences in only 14 minutes with 4 NVIDIA 1080TI GPUs.

It is noteworthy that because some subjects missing several gait sequences have not been removed from the probe set, the maximum rank-1 accuracy cannot reach 100 percent. If we do not count the cases that have no corresponding samples in the gallery, the average rank-1 accuracy of all

TABLE 3
Averaged Rank-1 Accuracies on *OU-MVLP*, Excluding
Identical-View Cases. GEINet: [18]. 3in+2diff: [4]

Probe	Gallery All 14 Views		Gallery $0°, 30°, 60°, 90°$		
	GEINet	Ours	GEINet	3in+2diff	Ours
0°	11.4	**81.3**	8.2	25.5	79.6
15°	29.1	**88.6**	-	-	87.1
30°	41.5	**90.2**	32.3	50.0	87.4
45°	45.5	**90.7**	-	-	89.8
60°	39.5	**88.6**	33.6	45.3	86.2
75°	41.8	**89.1**	-	-	88.0
90°	38.9	**88.3**	28.5	40.6	84.3
180°	14.9	**83.1**	-	-	81.8
195°	33.1	**87.7**	-	-	84.2
210°	43.2	**89.4**	-	-	87.7
225°	45.6	**89.7**	-	-	87.6
240°	39.4	**87.8**	-	-	86.3
255°	40.5	**88.3**	-	-	86.4
270°	36.3	**86.9**	-	-	85.8
mean	35.8	**87.9**	-	-	85.9

Fig. 5. The accuracy change process on OU-MVLP.

GEI experiment, the gait silhouettes are averaged into a single GEI before being fed into the network. There might be two main reasons for this improvement: *1)* our SP extracts the set-level feature from a high-level feature map where the temporal information is well preserved and the spatial information has been sufficiently processed; and *2)* as mentioned in Section 4.4, regarding gait as a set enlarges the volume of training data.

The Impact of SP. In Table 4, the results from the second row to the eighth row show the impact of different SP strategies. SP with pixel-wised attention achieves the highest accuracy on the NM and BG subsets and when $\max(\cdot)$ is used, it obtains the highest accuracy on the CL subsets. Considering the fact that SP with $\max(\cdot)$ also achieves the second best performance on the NM and BG subsets and has the most concise structure; thus, we choose it as the SP strategy in the final version of GaitSet.

The Impact of MGP. The second and the last rows of Table 4 show that MGP improves all three test subsets. This result is consistent with our experience mentioned in Section 3.4, i.e., that set-level features extracted from different layers of the main pipeline contain different discriminative information.

The Impact of HPM Scales and HPM Weight Independence. As shown in Table 5, HPM obtains better performance with more scales. Furthermore, the last two lines of Table 5 compare the impact of the weight independence of the fully connected layer in HPM. It can be seen that using independent weights increases the accuracy by more than 7 percent on each subset. During the experiments, we also noticed that

probe views will rise to 94.1 percent rather than 87.9 percent.

Fig. 5 shows the relationship between training iterations and test accuracy. We can see that after the cross entropy loss reaches its best performance, further tuning with triplet loss can still engender improvement.

4.5 Ablation Experiments and Model Studies

In this section, we report ablation experiments and model studies on CASIA-B, to examine the effectiveness of regarding gait as a set with set pooling, MGP, HPM, and different training strategies with different loss combinations. All the experiments were based on the settings under CASIA-B LT, as shown in Table 2.

4.5.1 Ablation Experiments

Set versus GEI. The first two rows of Table 4 show the effectiveness of regarding gait as a set. With totally identical networks, the result of using the *set* exceeds that of using GEI by more than 6 percent on the NM subset and more than 19 percent on the CL subset. The only difference is that in the
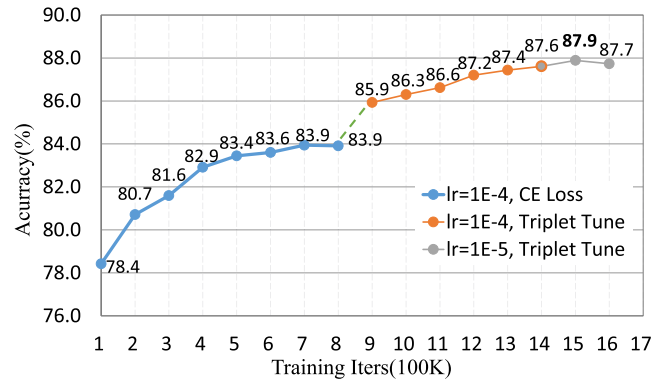
TABLE 4
Ablation Experiments Conducted on *CASIA-B* Using Setting LT

GEI	Set	Set Pooling							MGP	NM	BG	CL
		Max	Mean	Median	Joint sum 3	Joint 1_1C 4	Pix-att	Frame att				
✓										89.0	76.3	50.7
	✓	✓								95.4	88.7	69.9
	✓		✓							95.0	86.3	66.3
	✓			✓						94.8	84.9	63.7
	✓				✓					94.1	84.1	64.3
	✓					✓				94.9	86.9	66.8
	✓						✓			95.6	88.9	69.6
	✓							✓		95.0	85.1	65.3
	✓	✓							✓	**96.1**	**90.8**	**70.3**

The results are rank-1 accuracies averaged on all 11 views, excluding identical-view cases. The numbers in brackets indicate the second highest results in each column. Here 'att' is the abbreviation of attention.

TABLE 5
The Impact of Different HPM Scales and HPM Weight Independence Experiments Conducted on *CASIA-B* Using the Setting on LT

| HPM scales | HPM weights | | NM | BG | CL |
	Shared	Independent			
1(no HPM)		✓	91.8	82.4	59.4
2		✓	91.8	82.9	60.1
3		✓	91.7	83.0	64.2
4		✓	93.9	86.9	64.5
5	✓		91.1	82.0	60.9
		✓	96.1	90.8	70.3

The results are rank-1 accuracies averaged on all 11 views, excluding identical-view cases.

TABLE 6
Different Loss Functions Conducted on *CASIA-B* Using Setting LT

Loss function	BN	Dropout	NM	BG	CL
CEloss	✓		90.9	84.8	46.7
		✓	94.4	89.4	60.2
	✓	✓	95.6	90.1	64.8
Triplet loss	✓		95.5	87.3	69.3
		✓	94.5	87.1	66.9
	✓	✓	95.3	90.3	67.3
CEloss + Triplet loss	✓		95.8	90.3	69.5
		✓	94.7	88.7	66.9
	✓	✓	96.1	90.8	70.3

The results are rank-1 accuracies averaged on all 11 views, excluding identical-view cases.

introducing independent weights makes the network converge faster.

4.5.2 Training Strategies

Our previous AAAI-19 paper utilized triplet loss to achieve good performance. To further improve the gait recognition accuracy, we combined triplet loss and cross-entropy loss to train the GaitSet model.

Table 6 shows the results of the three training strategies, and the impact of batch normalization and dropout. All three training strategies exceed 95 percent rank-1 accuracy on the NM subset when using batch normalization and dropout. However, only the pretraining model that combines the two losses reaches the highest 96.1 percent rank-1 accuracy. The first and third lines of Table 6 reveal that the dropout layers is essential for a robust training performance of cross-entropy loss in this case. We also see that batch normalization improves all the training strategies.

4.6 Feature Dimension Reduction

As mentioned in Section 3.6, the testing feature after concatenating all the HPM ouputs has $256 \times 31 \times 2 = 15,872$ dimensions in a standard framework, which impairs the testing efficiency. Therefore, we conducted dimensional reduction using two methods. One is to set the feature dimension to a lower level by shrinking the output dimension of the HPM fully connected layers. The other method is to perform the testing task after introducing a new fully connected layer, which achieves a large compression of the original 15,872 dimensions.

HPM Output Dimensions. The HPM output dimensions were set to 32,64,128,256,512, and 1,024. Using these different dimension, we studied the recognition accuracy with approximately $50,000 - 60,000$ training iterations. As Fig. 6 shows, even when the output dimensions are as low as 32, the performance still reaches 93 percent on the NM subset with all three loss function strategies. However, there is still a negative impact on the performance if the HPM output dimensions are too low (down to 32) or too high (up to 1024). The reasons for this performance degeneration are that 1) the fully connected layers whose output dimensions are too high can easily be overfitting because they contain too many parameters, and 2) an output dimension that is too small would significantly constrain the fully connected layers' learning capacity. In particular, the model trained with CE loss is less robust on the CL subset with high dimensional HMP outputs, while the model with the pretraining strategy has a stable performance on the CL subset. By decreasing the HMP output dimensions, we can compress the final feature dimension from 15,872 to one quarter of that. While this compression is associated with a subtle performance impairment, the degeneration of recognition performance on the BG and CL subsets cannot be ignored.

Dimension Reduction With the New FC. Undoubtedly, we can directly reduce the final feature dimension. After the model has been well trained, a new fully connected layer is applied to the 15,872 dimension feature to reduce it into a lower dimension. This new layer is tuned for 10,000 iterations with triplet loss and a learning rate of $1e - 4$. We investigate the output dimensions of $128, 256, 512, 1024, 2048$, and 4,096.
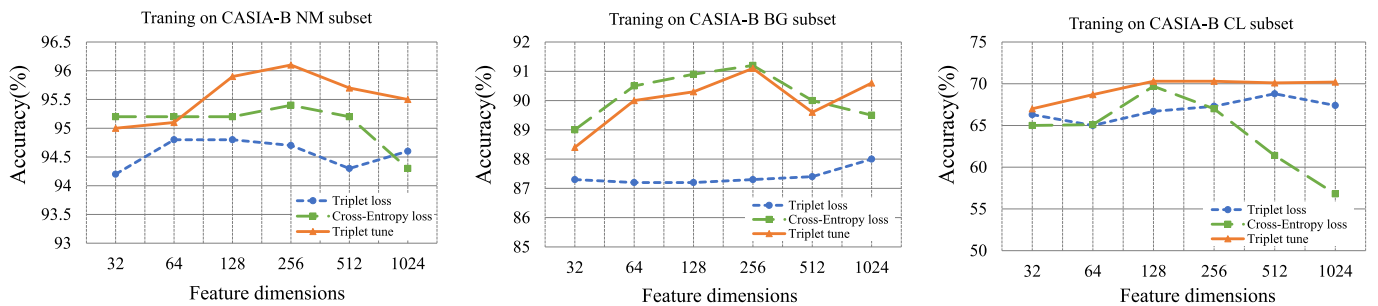

Fig. 6. Relationships between recognition accuracy and the HPM output dimension. From left to right are the individual results on the CASIA-B NM, BG, and CL subsets. The relationships vary with different training strategies, as shown by the different lines in each graph.

TABLE 7
The Recognition Accuracy After Dimensional
Reduction With the New FC

Feature dimension	NM	BG	CL
128	91.7	83.5	62.5
256	94.1	87.3	66.6
512	94.4	88.4	68.9
1024	95.0	89.3	69.1
2048	95.0	90.2	70.0
4096	94.9	90.2	70.3

TABLE 8
Multiview Experiments Conducted on *CASIA-B*
Using the LT Setting

View difference	18° / 162°	36° / 144°	54° / 126°	72° / 108°	90°	Single view
All silhouettes	98.9	99.6	97.6	96.2	99.3	96.1
10 silhouettes	94.8	97.2	95.9	91.7	97.25	89.5

Cases with the probe contains the views in the gallery are excluded.

As the Table 7 shows, the final feature dimension can be compressed to 1024 while maintaining the recognition accuracy at 95 percent on the NM subset; this is only 6.5 percent of the original 15,872 dimensions. Similar to changing the output dimension of HPM, a too-small feature dimension leads to a performance decrease. Although it runs counter to the idea of an end-to-end design, introducing this post-processing effectively compresses the learned feature representation, making the method more practical for real applications.

4.7 Practicality

Because of the flexibility of the gait set approach, GaitSet may be useful in more complicated practical conditions. In this section, we investigate the practicality of GaitSet through three novel scenarios: *1)* How does GaitSet perform when a input set contains only a few silhouettes? *2)* Can silhouettes with different views enhance the identification accuracy? *3)* Can the model effectively extract deep discriminative representation from a set containing silhouettes shot under different walking conditions? It is worth noting that we did not retrain or tune our model in these experiments; the exact model used in the Section 4.4 with the LT setting is used here. Note that here the reported accuracies are averaged on 10 times experiments with different random seeds.

Limited Silhouettes. In real forensic identification scenarios, cases occur in which no continuous sequence of a subject's gait is available, only some fitful and sporadic silhouettes. We simulate such a circumstance by randomly selecting a certain number of frames from sequences to compose each sample in both the gallery and probe. Fig. 7 shows
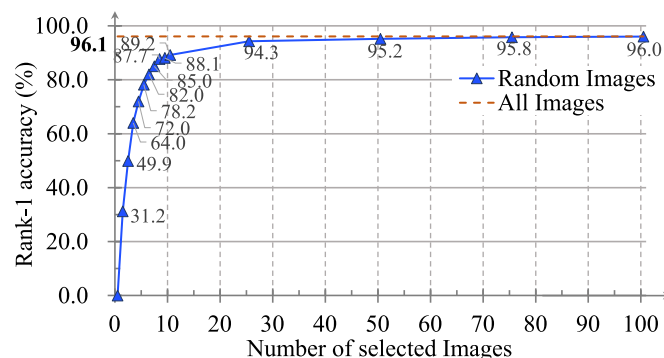

Fig. 7. Average rank-1 accuracies with constraints of silhouette volume on the *CASIA-B* dataset using the LT setting. The accuracy values are averaged on all 11 views excluding identical-view cases, and the final reported results are averaged across ten experimental repetitions.

the relationship between the number of silhouettes in each input set and the rank-1 accuracy averaged on all 11 probe views. Our method attains an 82 percent accuracy using only 7 silhouettes. This result also indicates that our model makes full use of the temporal information contained in a gait set. It can also be observed that *1)* the accuracy rises monotonically as the number of silhouettes increases, and *2)* the accuracy is close to the best performance when the samples contain more than 25 silhouettes. This number is consistent with the number of frames that one gait period contains.

Multiple Views. Here we study a scenario where one person's gait is collected from different views. We simulate these scenarios by constructing each silhouette sample selected from two sequences that have the same walking condition but different views. To alleviate the effects of the number of silhouettes, an experiment is performed under the case where the maximum silhouette number is 10. Unlike the previous contrast experiments of a single view where an input set consists of 10 silhouettes from one sequence, more concretely, in this experiment, an input set is made up of 5 silhouettes from each of two sequences in the two-view experiment. Note that in this experiment, only probe samples are composed by the aforementioned method, whereas the sample in the gallery is composed of all silhouettes from one sequence.

Because there are too many view pairs to display them all, we summarize the results by averaging the accuracies of each possible view difference, as indicated in Table 8. For example, the result of a 90° difference was averaged by the accuracies of 6 view pairs (0°&90°, 18°&108°, ..., 90°&180°). Furthermore, the 9 view differences were folded at 90° and those larger than 90° were averaged with the corresponding view differences of less than 90°. For example, the results of the 18° view difference were averaged with those of the 162° view difference.Our model effectively aggregates information from different views and boosts the final performance. This result can be explained by the pattern between views and accuracies discussed in Section 4.4. Including multiple views in the input set allows the model to gather both parallel and vertical information, resulting in performance improvements.

Multiple Walking Conditions. In real life, it is highly likely that gait sequences of the same person occur under different walking conditions. We simulate such different conditions by forming an input set using silhouettes from two sequences with the same view but different walking conditions and conduct experiments under the constraint of different numbers of silhouettes. Note that in this experiment, only

TABLE 9
Multiple Walking Condition Experiments Conducted on
CASIA-B Using the LT Setting

NM(10)	91.1	NM(10)+BG(10)	93.0	NM(20)	94.2
BG(10)	85.8	NM(10)+CL(10)	91.6	BG(20)	89.4
CL(10)	85.1	BG(10)+CL(10)	89.7	CL(20)	88.9

The results are rank-1 accuracies averaged on all 11 views, excluding identical-view cases. The numbers in brackets indicate the constraints of the silhouette number in each input set.

the probe samples are composed by the method discussed above. All the samples in the gallery are constructed using all the silhouettes from one sequence. Moreover, the probe-gallery division of this experiment is different. For each subject, the sequences NM #02, BG #02, and CL #02 are kept in the gallery, while the sequences NM #01, BG #01, and CL #01 are used as probes.

From Table 9 we can see that 1) the accuracies are still boosted as the number of silhouettes increases, and 2) when the number of the silhouettes is fixed, the results reveal the relationships between different walking conditions. Containing large yet complementary noises and information, the combination of silhouettes from BG and CL helps the model improve the accuracy. In contrast, silhouettes of NM contain little noise. Consequently, substituting silhouettes of the other two conditions for some of them does not provide extra useful information but only introduces noise, leading to a degraded performance.

5 CONCLUSION

In this paper, we presented a novel perspective that regards gait as a deep set, called a GaitSet. The proposed GaitSet approach extracts both spatial and temporal information more effectively and efficiently than do the existing methods, which regard gait as either a template or a sequence. Unlike other existing gait recognition approaches, the GaitSet approach also provides an innovative way to aggregate valuable spatiotemporal information from different sequences to enhance the accuracy of cross-view gait recognition. Experiments on two benchmark gait datasets indicate that GaitSet achieves the highest recognition accuracy compared with other state-of-the-art algorithms, and the results reveal that GaitSet exhibits a wide range of flexibility and robustness when applied to various complex environments, showing a great potential for practical applications. In addition, since the set assumption could fit various other biometric identification tasks including person re-identification and video-based face recognition, the structure of GaitSet can be applied to these tasks with few minor changes in the future.

ACKNOWLEDGEMENTS

REFERENCES

[1] Z. Liu and S. Sarkar, "Improved gait recognition by gait dynamics normalization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 6, pp. 863–876, Jun. 2006.
[2] M. Hu, Y. Wang, Z. Zhang, J. J. Little, and D. Huang, "View-invariant discriminative projection for multi-view gait-based human identification," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 12, pp. 2034–2045, Dec. 2013.
[3] Y. Guan, C.-T. Li, and F. Roli, "On reducing the effect of covariate factors in gait recognition: A classifier ensemble method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1521–1528, Jul. 2015.
[4] N. Takemura, Y. Makihara, D. Muramatsu, T. Echigo, and Y. Yagi, "On input/output architectures for convolutional neural network-based cross-view gait recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 9, pp. 2708–2719, Sep. 2019.
[5] X. Chen, J. Weng, W. Lu, and J. Xu, "Multi-gait recognition based on attribute discovery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 7, pp. 1697–1710, Jul. 2018.
[6] Y. He, J. Zhang, H. Shan, and L. Wang, "Multi-task GANs for view-specific feature learning in gait recognition," *IEEE Trans. Inform. Forensics Secur.*, vol. 14, no. 1, pp. 102–113, Jan. 2019.
[7] Z. Wu, Y. Huang, L. Wang, X. Wang, and T. Tan, "A comprehensive study on cross-view gait based human identification with deep CNNs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 2, pp. 209–226, Feb. 2017.
[8] R. Liao, C. Cao, E. B. Garcia, S. Yu, and Y. Huang, "Pose-based temporal-spatial network (PTSN) for gait recognition with carrying and clothing variations," in *Proc. Chin. Conf. Biometric Recognit.*, 2017, pp. 474–483.
[9] T. Wolf, M. Babaee, and G. Rigoll, "Multi-view gait recognition using 3D convolutional neural networks," in *Proc. IEEE Int. Conf. Image Process.*, 2016, pp. 4165–4169.
[10] X. Wu, W. An, S. Yu, W. Guo, and E. B. García, "Spatial-temporal graph attention network for video-based gait recognition," in *Proc. Asian Conf. Pattern Recognit.*, 2019, pp. 274–286.
[11] C. Peyrin, M. Baciu, C. Segebarth, and C. Marendaz, "Cerebral regions and hemispheric specialization for processing spatial frequencies during natural scene recognition. An event-related fMRI study," *Neuroimage*, vol. 23, no. 2, pp. 698–707, 2004.
[12] S. Yu, D. Tan, and T. Tan, "A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition," in *Proc. Int. Conf. Pattern Recognit.*, 2006, pp. 441–444.
[13] N. Takemura, Y. Makihara, D. Muramatsu, T. Echigo, and Y. Yagi, "Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition," *IPSJ Trans. Comput. Vis. Appl.*, vol. 10, no. 4, pp. 1–14, 2018.
[14] J. Han and B. Bhanu, "Individual recognition using gait energy image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 2, pp. 316–322, Feb. 2006.
[15] C. Wang, J. Zhang, L. Wang, J. Pu, and X. Yuan, "Human identification using temporal information preserving gait template," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2164–2176, Nov. 2012.
[16] X. Xing, K. Wang, T. Yan, and Z. Lv, "Complete canonical correlation analysis with application to multi-view gait recognition," *Pattern Recognit.*, vol. 50, pp. 107–117, 2016.
[17] K. Bashir, T. Xiang, and S. Gong, "Gait recognition without subject cooperation," *Pattern Recognit. Lett.*, vol. 31, no. 13, pp. 2052–2060, 2010.
[18] K. Shiraga, Y. Makihara, D. Muramatsu, T. Echigo, and Y. Yagi, "GEINet: View-invariant gait recognition using a convolutional neural network," in *Proc. IAPR Int. Conf. Biometrics*, 2016, pp. 1–8.
[19] Y. Makihara, R. Sagawa, Y. Mukaigawa, T. Echigo, and Y. Yagi, "Gait recognition using a view transformation model in the frequency domain," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 151–163.
[20] S. Yu, H. Chen, E. B. G. Reyes, and N. Poh, "GaitGAN: Invariant gait feature extraction using generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 532–539.
[21] S. Yu, H. Chen, Q. Wang, L. Shen, and Y. Huang, "Invariant feature extraction for gait recognition using only one uniform model," *Neurocomputing*, vol. 239, pp. 81–93, 2017.
[22] W. An *et al.*, "Performance evaluation of model-based gait on multi-view very large population database with pose sequences," *IEEE Trans. Biometrics Behav. Identity Sci.*, vol. 2, no. 4, pp. 421–430, Oct. 2020.

[23] Y. Zhang, Y. Huang, S. Yu, and L. Wang, "Cross-view gait recognition by discriminative feature learning," *IEEE Trans. Image Process.*, vol. 29, pp. 1001–1015, Jul. 2019.

[24] C. Fan *et al.* "Gaitpart: Temporal part-based model for gait recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14 225–14 233.

[25] S. Hou, C. Cao, X. Liu, and Y. Huang, "Gait lateral network: Learning discriminative and compact representations for gait recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 382–398.

[26] H. Chao, Y. He, J. Zhang, and J. Feng, "Gaitset: Regarding gait as a set for cross-view gait recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 8126–8133.

[27] Y. Fu *et al.*, "Horizontal pyramid matching for person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8295–8302.

[28] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 77–85.

[29] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, 2019.

[30] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4490–4499.

[31] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[32] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.

[33] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei, "A hierarchical approach for generating descriptive image paragraphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3337–3345.

[34] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola, "Deep sets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 3391–3401.

[35] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803.

[36] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.

[37] W. Li, X. Zhu, and S. Gong, "Harmonious attention network for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2285–2294.

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

[39] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1701–1708.

[40] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[41] L. Zheng, Y. Yang, and A. G. Hauptmann, "Person re-identification: Past, present and future," 2016, *arXiv:1610.02984*.

[42] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv: 1703.07737*.

[43] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech.*, 2018, pp. 1086–1090.

[44] X. Dong and J. Shen, "Triplet loss in Siamese network for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 459–474.

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representation*, 2015, pp. 1–15.

[46] W. Kusakunniran, Q. Wu, J. Zhang, H. Li, and L. Wang, "Recognizing gaits across views through correlated motion co-clustering," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 696–709, Feb. 2014.

Hanqing Chao received the BS degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2016, and the master's degree in computer science from Fudan University, Shanghai, China, in 2019. His research interests include machine learning, computer vision, and gait recognition.
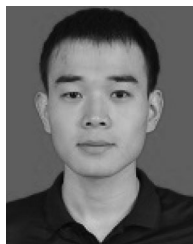
Kun Wang received the BS degree from the Xi'an Jiaotong University, Xi'an, China, in 2017, and the master's degree in computer science from Fudan University, Shanghai, China, in 2020. His research interests include machine learning, computer vision, and gait recognition.

Yiwei He received the BS degree from the Nanjing Audit University, Nanjing, China, in 2015, and the master's degree in computer science at Fudan University, Shanghai, China, in 2018. His research interests include machine learning, computer vision, and gait recognition.

Junping Zhang (Member, IEEE) received the BS degree in automation from Xiangtan University, Xiangtan, China, in 1992, the MS degree in control theory and control engineering from Hunan University, Changsha, China, in 2000, and the PhD degree in intelligent system and pattern recognition from the Institute of Automation, Chinese Academy of Sciences, in 2003. Currently, he is a professor at the School of Computer Science, Fudan University, since 2011. His research interests include machine learning, image processing, biometric authentication, and intelligent transportation systems. He has widely published in highly ranked international journals, such as the *IEEE Transactions on Pattern Analysis and Machine Intelligence* and *IEEE Transactions on Neural Networks and Learning Systems*, and leading international conferences, such as ICML and ECCV. He has been an associate editor of the IEEE Intelligent Systems Magazine, since 2009.
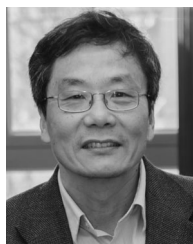
Jianfeng Feng received the BSc, MSc, and PhD degrees in mathematics, from Peking University, Peking, China. He is currently the chair professor of the Shanghai National Centre for Mathematic Sciences, and the dean of Brain-inspired AI Institute, and the head of Data Science School in Fudan University, since 2008. He has been developing new mathematical, statistical and computational theories and methods to meet the challenges raised in neuroscience, mental health and brain-inspired AI researches. He was awarded the Royal Society Wolfson Research Merit Award, in 2011, as a scientist being of great achievements or potentials. He was invited to deliver 2019 Paykel Lecture at the Cambridge University.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.