A Cross-Perspective Gait Recognition Framework Integrating Breadth-First Search and Multi-Scale Feature Map Interaction

Jieran Liu[1*], Wenqing Wang[2]
[1]Software Department, Zhengzhou University of Industrial Technology, Zhengzhou, 451150, China
[2]Information Hub, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, 511453, China
E-mail: jieran_liu@163.com, wwenqing0524@gmail.com
[*]Corresponding author

Keywords: cross-perspective, gait recognition, BFS, feature map interaction, biometric recognition

Received: January 8, 2025

Gait recognition is a key biometric technology with broad applications, yet cross-perspective variation severely impairs performance. This study proposes a novel gait recognition model that integrates a breadth-first search-guided feature propagation mechanism with gated recurrent unit-based temporal modeling and multi-scale spatial feature map interaction. The model enhances feature fusion across different layers and perspectives while selectively attending to key temporal cues through global max pooling. Experimental evaluations on the CASIA-B dataset demonstrate that the proposed method achieves an accuracy of 0.97, 0.94, and 0.91 under normal walking, carrying object, and wearing jacket conditions, respectively, significantly surpassing the baseline models in recognition performance. The method also obtains the lowest root mean square error of 0.09 and the fastest recognition time of 1.2 seconds. Compared with conventional convolutional neural networks and recurrent neural network-based architectures, the proposed model shows substantial improvements in accuracy, robustness, and computational efficiency. The key innovation lies in the introduction of a breadth first search-driven feature interaction strategy and a hierarchical temporal-spatial fusion structure, which jointly optimize the feature representation for robust cross-view gait recognition.

Povzetek: Za robustno večperspektivno prepoznavo hoje je razvit model BFS-CNN-GMP-GRU-MSP, ki združuje iskanje v širino (BFS) za propagacijo značilk, GRU za časovno modeliranje in večmerno prostorsko interakcijo značilk.

1 Introduction

Gait recognition is a non-contact biometric recognition technology that utilizes human gait features for identity recognition. Unlike other biometric recognition technologies, gait recognition does not rely on close range collection or high-resolution data, and has the advantages of long-distance operability and no need for active cooperation [1-3]. However, gait recognition faces many challenges in practical applications, and the cross-perspective problem is one of the most critical difficulties. When individual gait images are collected from different perspectives, gait features may undergo significant changes due to external factors such as angle, lighting, and clothing, which can lead to inconsistent expression and extraction of gait features, thereby affecting recognition accuracy. With the development of deep learning, techniques such as Convolutional Neural Network (CNN) and recurrent neural network have been widely applied in gait recognition tasks. However, existing methods still have shortcomings in cross-perspective gait recognition. Traditional CNN models mainly focus on single scale spatial feature extraction and cannot fully express multi-scale information from different perspectives, resulting in unstable recognition performance. Although temporal modeling can capture temporal dependencies, it lacks a global attention mechanism and cannot effectively focus on key time points in gait sequences, resulting in redundant and inefficient feature extraction. The mechanism of feature interaction and fusion is not yet perfect, and efficient information integration cannot be achieved between shallow, middle, and deep features. Therefore, a cross-perspective gait recognition model based on Breadth First Search (BFS) algorithm and feature map interaction was proposed. This model extracts spatial features through CNN, searches for feature maps through BFS algorithm, and finally combines Gated Recurrent Unit (GRU) and Global Max Pooling (GMP) to capture temporal dependency characteristics. The innovation of the research lies in introducing the BFS algorithm to optimize the feature propagation mechanism, improve computational efficiency and accuracy, and aim to provide an efficient and robust solution for gait recognition.

To address the limitations in current gait recognition models, this study is driven by two primary research questions: (1) Can a BFS mechanism enhance the efficiency and comprehensiveness of feature propagation

across spatial hierarchies in cross-perspective gait recognition? (2) How does the integration of multi-scale spatial feature interaction influence the effectiveness of temporal modeling and attention in dynamic and occluded environments? These questions guide the model design, which incorporates BFS-guided node traversal, multi-stage spatial feature map fusion, and gated recurrent units for temporal dependency capture. The proposed model is rigorously evaluated on the CASIA-B dataset under various conditions to empirically validate the effectiveness of each component.

2 Related works

Cross-perspective gait recognition is an important task in addressing the impact of perspective changes on gait features. Parashar et al. proposed a deep learning architecture and pipeline to utilize the complex features of human gait for biometric applications. The research results indicated that although gait recognition faced diversity and complexity, deep learning models could still effectively work on low resolution images, but were greatly affected by various covariates such as shoes and clothing [4]. Castro et al. proposed an innovative hybrid protection scheme to ensure the privacy and security of gait analysis for early detection of neurodegenerative diseases in human activity recognition. This scheme combined partially homomorphic encryption and revocable biometric technology based on random projection. The research results indicated that this scheme could achieve a high trade-off between security and performance, with an accuracy decrease of up to 1.20, and was applicable to any type of neural network [5]. Baniasad et al. proposed an algorithm suitable for different sensor configurations, gait speeds and shoe types to solve the problem of complex and error prone connection of IMU sensors in motion and rehabilitation motion analysis. The research results showed that the algorithm could accurately identify body parts and lower limb sensor sides. For gait speed ranges of 0.5-2.2 m/s, the accuracy and precision reached 99.7% and 99.0%, respectively, and had broad application prospects [6].

Zhang et al. proposed a non-contact bendable sensitive sensor that uses a semi-circular optical fiber to monitor muscle activity to improve the detection accuracy of wearable robot human interaction. The research results showed that using this sensor combined with neural networks, the recognition accuracy of five gaits was over 99%, significantly better than traditional machine learning algorithms, providing a new and effective approach for abnormal gait recognition [7]. Derlatka et al. proposed a solution using heterogeneous base classifier ensemble to improve the accuracy and running speed of human gait recognition. The research results showed that the proposed scheme has been tested on a sample of 322 people, with a recognition accuracy of up to 99.65%. The model construction time was less than 12.5 minutes, and the time required to identify a person was less than 0.1 seconds. The performance was significantly better than other methods in the literature [8]. Yan et al. proposed a new gait recognition framework to address performance issues caused by occlusion and viewpoint changes in gait recognition, as well as the problem of traditional time pools ignoring unique time information. The research results indicated that the framework could effectively extract adaptive structured spatial representations, aggregate multi-scale temporal information, and improve recognition accuracy, especially in complex scenes, with an average accuracy of 93.5% on the CASIA-B dataset [9].

In summary, significant progress has been made in the field of gait recognition, from gait feature extraction, temporal modeling to cross-perspective adaptation. Many scholars have applied deep learning techniques to gait recognition tasks and achieved certain results. Despite notable advances in gait recognition, several critical limitations persist in existing state-of-the-art models. Many approaches, such as those by Parashar et al. and Baniasad et al., either focus on static spatial features or rely heavily on wearable sensors, limiting their adaptability in vision-only, cross-view scenarios. Models like that of Yan et al. employ multi-scale temporal aggregation but still lack explicit mechanisms to capture global feature interactions across different spatial levels, which are essential for robustness under perspective changes. Moreover, methods using deep learning pipelines often ignore temporal attention granularity, leading to suboptimal performance when distinguishing subtle gait variations across sequences. Few works have integrated a structured propagation mechanism to ensure efficient multi-level feature fusion and comprehensive node traversal. These gaps highlight the necessity for a model that explicitly addresses both spatial hierarchy and temporal dynamics, motivating the design of BFS-guided, GRU-enhanced gait recognition framework. To provide a clear comparison between the proposed method and other state-of-the-art approaches, Table 1 summarizes key aspects of recent representative studies, including their methods, research content, datasets used, and performance indicators.

Table 1: Performance comparison between the SOTA method and the model in this paper

Research	Method	Research content	Dataset used	Key performance indicators	Reference
Parashar et al. (2023)	Deep learning pipeline for gait biometrics	Addressing covariates like clothing and shoes in gait recognition	Custom gait dataset	Effective under low resolution, but performance drops under covariates	[4]
Castro et al. (2024)	Hybrid protection with homomorphic	Gait analysis for early dementia recognition	Gait dataset (private)	Accuracy loss up to 1.20, emphasizes	[5]

	encryption	with privacy-preserving techniques		security-performance tradeoff	
Baniasad et al. (2023)	IMU-based segment recognition algorithm	Recognition of body segment and limb side in gait using inertial sensors	IMU sensor dataset	Accuracy 99.7%, Precision 99.0% in 0.5–2.2 m/s gait range	[6]
Zhang et al. (2023)	Optical fiber sensor + neural networks	Abnormal gait recognition through muscle activation detection	5-class gait dataset	Recognition accuracy over 99%, better than conventional ML methods	[7]
Derlatka et al. (2023)	Heterogeneous classifier ensemble	Human gait recognition using classifier fusion	Sample size 322	Accuracy 99.65%, identification time < 0.1s	[8]
Yan et al. (2024)	Adaptive spatial-temporal aggregation network	Occlusion- and viewpoint-robust gait recognition	CASIA-B	Average accuracy 93.5%	[9]

3 Methods

The first section proposes a cross-perspective gait recognition model based on feature map interaction for cross-perspective gait recognition. At the same time, BFS algorithm is introduced to improve the problem of large parameter quantity.

2.1 Cross-perspective gait recognition model based on feature map interaction

In practical applications, people's gait characteristics may undergo significant changes due to factors such as shooting angle, perspective changes, lighting conditions, etc. Therefore, a cross-perspective gait recognition model based on feature map interaction is proposed in this study. To enhance the stability and convergence of the training process, min-max normalization is applied to all gait silhouette pixel values, scaling them to the range [0, 1]. This choice is motivated by its simplicity and effectiveness in preserving the structural consistency of grayscale images used in silhouette-based gait recognition. The structure of the model is shown in Figure 1.

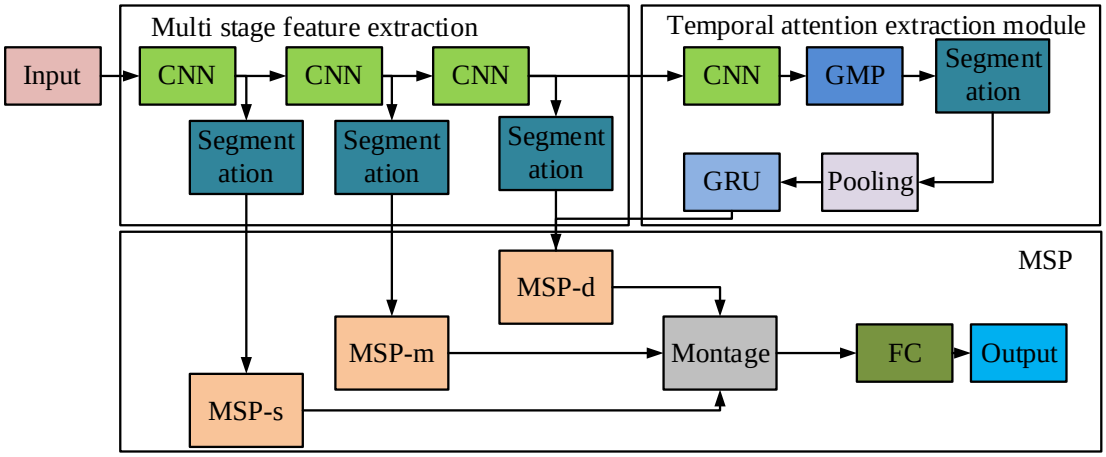


Figure 1: Cross-perspective gait recognition model based on feature map interaction

As shown in Figure 1, multiple experimental devices collect different gait sequences, which are processed by the Temporal Spatial Multi-Feature Extraction (TASMF) module to generate cross device gait features. Gait sequences are captured using multiple fixed-angle video cameras positioned at different horizontal viewpoints (ranging from 0° to 180° with 18° intervals), simulating cross-view observation conditions. Each camera corresponds to a specific viewpoint and records RGB gait videos of each subject under three walking conditions: normal, carrying a bag, and wearing a coat. These RGB sequences are later converted into binary silhouette images through background subtraction preprocessing, which serve as the actual input to the proposed model. These features are then extracted using the Multi-Scale Spatial (MSP) module. The MSP module utilizes multiple CNNs to process gait images of different scales, enhancing the ability to express cross-perspective features by fusing multi-scale information [10]. These features then enter the temporal attention module, which combines GRU with GMP operations to capture temporal dependencies in gait sequences and focus on important time points, generating weighted temporal feature representations. Finally, the

features are used for classification through a Fully Connected layer (FC). After aggregation, these features enter the classifier to complete the recognition task and output the final gait classification result. By using a TASMF fusion structure that does not share parameters, shallow, middle, and deep information is extracted separately. The output features of each stage are segmented, and the gait sequence is cut into multiple segments. The maximum pooling operation is performed to obtain the feature map, which is expressed as equation (1).

$$x_{out} = Maxpooling_s(f_{slice}) \quad (1)$$

In equation (1), f_{slice} represents the sequence and $Maxpooling_s(\cdot)$ represents the max pooling operation. The TASMF module is responsible for the initial preprocessing and structuring of gait sequence data before it is passed to the CNN-based multi-scale spatial feature extractors. Specifically, TASMF receives binary silhouette sequences and performs three operations: (1) Temporal segmentation – each gait sequence is divided into multiple fixed-length temporal slices to preserve motion continuity and reduce noise from long sequences. (2) Frame normalization – silhouette frames are aligned and resized to a uniform spatial resolution, ensuring consistent scale across viewpoints and walking styles. (3) Feature stacking-segmented frame sets are converted into structured tensors, where temporal and spatial information is jointly encoded, allowing downstream CNN modules to extract joint spatial-temporal patterns. This preprocessing enables the model to retain localized motion details while also providing a consistent input format for subsequent convolutional processing in the MSP modules. The temporal attention module is constructed using GRU as the basic algorithm, and its structure is shown in Figure 2.

In Figure 2, this module models the temporal dependencies in a gait sequence and emphasizes key time steps. "Conv8" denotes an 8-channel convolutional layer applied to extract preliminary spatial features. "Segmentation" divides the temporal dimension of the input into fixed-length slices. "GMP" stands for Global Max Pooling, used to compress spatial dimensions and highlight dominant features. "Max Pooling" reduces temporal resolution and noise by selecting maximum values across segmented frames. "GRU" refers to a bidirectional Gated Recurrent Unit layer that captures long-range temporal dependencies. "Norm" indicates batch normalization, which stabilizes training and improves convergence. The final output is a temporally-weighted feature vector passed to the classification stage. The symbol 's' represents the number of temporal segments after slicing the input gait sequence. The original input is a sequence of binary silhouette frames with spatial dimensions height (h), width (w), and channel (c). After applying the Conv8 convolutional layer, the sequence is temporally segmented into 's' slices, each containing a fixed number of consecutive frames. These segments form a 4D tensor of shape (s, c, h, w), where each slice retains the original spatial resolution but is treated as an independent temporal unit for attention modeling. Firstly, the input feature map undergoes Conv8 convolution operation to extract preliminary spatial features and form a feature map with a size of s × c × h × w. Subsequently, through GMP operation, the input feature map is globally pooled in spatial dimension to obtain a feature matrix with a size of s × c. Next, the features are segmented and the sequence is divided into T time steps, outputting a feature representation in the s/T × c dimension. Further max pooling is performed to compress the time dimension and obtain a more concentrated temporal feature representation. Next, these features are input into the GRU, which captures the temporal dependencies of gait features through a bidirectional GRU structure, while enhancing attention weight allocation for critical time steps [11-12]. The output temporal features are batch normalized to improve the stability and training efficiency of the model. Finally, the temporal features are mapped to classification scores. The MSP module is designed to enhance the spatial feature representation by capturing gait features at different resolutions. Specifically, each input silhouette sequence is resized into three spatial scales: a shallow resolution (e.g., 64×64), a middle resolution (e.g., 96×96), and a deep/full resolution (e.g., 128×128). These versions preserve different levels of detail: shallow inputs capture global body posture, while deeper inputs retain fine-grained motion and contour information. Each scaled input is independently processed through a dedicated CNN branch, forming a parallel architecture. These branches are composed of convolutional layers with identical configurations but operate on different input resolutions. After processing, each CNN outputs a spatial feature map that is temporally aligned. The outputs are then passed to the feature fusion pipeline, where pooling, reshaping, and concatenation are applied to integrate the three scales into a unified global representation. This parallel multi-resolution strategy ensures the model can extract both coarse and fine spatial details, improving robustness under viewpoint variation and body occlusion. The expression for maximum value pooling for each time period is shown in equation (2).

$$x_T = Maxpooling_s(x_{slice}) \quad x_T = Maxpooling_s(x_{slice}) \quad (2)$$

In equation (2), x_T represents the feature after max pooling, and the expression for temporal attention score is shown in equation (3).

$$x_{score} = GRU(x_{slice}) \quad (3)$$

In equation (3), x_{score} represents the temporal attention score. The MSP structure is shown in Figure 3.

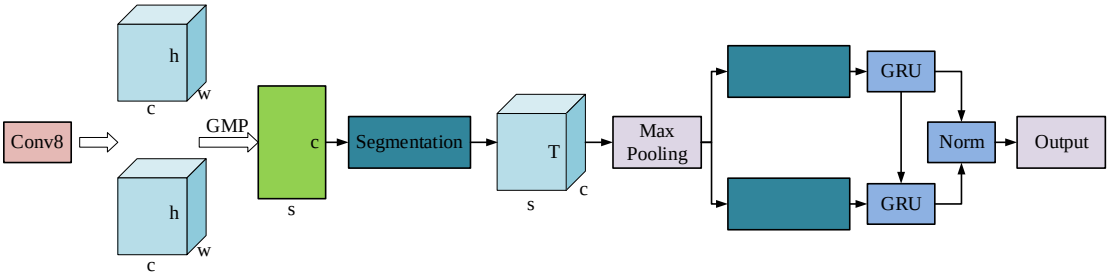


Figure 2: Temporal attention extraction module

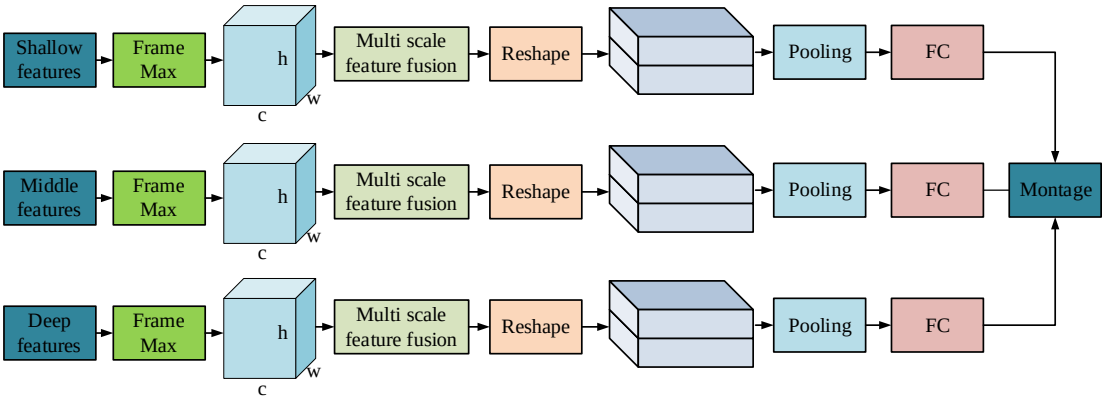


Figure 3: Multi-scale pyramid feature fusion structure

As illustrated in Figure 3, the proposed MSP fusion module receives three independent inputs corresponding to shallow, middle, and deep spatial features extracted from different CNN branches. Each input branch undergoes the following processing steps: Frame Max: applies temporal max pooling across each frame sequence to extract the most salient feature from each frame. Multi-scale feature fusion: applies dilated convolutions with varying receptive fields to capture local and global context at different scales. Reshape: reshapes the output into a flattened form suitable for fully connected layers. Pooling: performs dimensionality reduction to retain only key information. FC: applies a fully connected layer to generate a compact feature vector for each scale. These three vectors-representing shallows, middle, and deep scale features-are then passed to the Montage node. The Montage operation refers to the concatenation of the three feature vectors into a single comprehensive feature vector. This operation enables the integration of low-level (texture/edge), mid-level (shape/pose), and high-level (semantic/global) spatial features. The resulting unified representation is then fed to the final classification stage. The structure of the multi-scale pyramid feature fusion module mainly processes spatiotemporal features of shallow, middle, and deep layers, achieving effective fusion of multi-level features [13-14]. The FrameMax operation is designed to extract the most salient spatial representation across temporal frames within a given feature stream. For each of the three scale branches (shallow, middle, deep), the input to FrameMax is a 4D tensor of shape (T, C, H, W), where T is the number of frames in the sequence, and C, H, W denote channel, height, and width respectively. FrameMax applies a temporal max pooling operation along the T dimension at each spatial location, resulting in a 3D tensor of shape (C, H, W). This operation captures the strongest activation at each spatial position across the entire sequence, effectively summarizing motion dynamics over time. Firstly, the three sets of features are maximally pooled in the temporal dimension through FrameMax operation, extracting important information from each frame to obtain a temporal feature map, which is expressed as equation (4).

$$x = \text{FrameMax}(x_{out} \cdot x_{score}) \quad (4)$$

In equation (4), $\text{FrameMax}(\cdot)$ represents max pooling multiple feature maps. Through the multi-scale spatial feature fusion module, different scales of feature information are processed separately. Subsequently, through the Reshape operation, the feature map is reshaped into a shape suitable for subsequent network inputs, forming feature representations in k1, k2, and k3 dimensions. The next pooling operation performs spatial dimensionality reduction on the reshaped feature map, further compressing redundant information and extracting key features. After dimensionality reduction, the features are input into FCs, and the shallow, middle, and deep features are further mapped into new feature vectors [15]. Finally, the three sets of feature vectors are fused at multiple scales through concatenation, resulting in a global feature representation with dimensions c × (k1+k2+k3). For the basic features extracted through multi-stage feature extraction modules, different dilated convolutions are used to obtain receptive fields. The structure of the multi-scale spatial feature fusion module is shown in Figure 4.

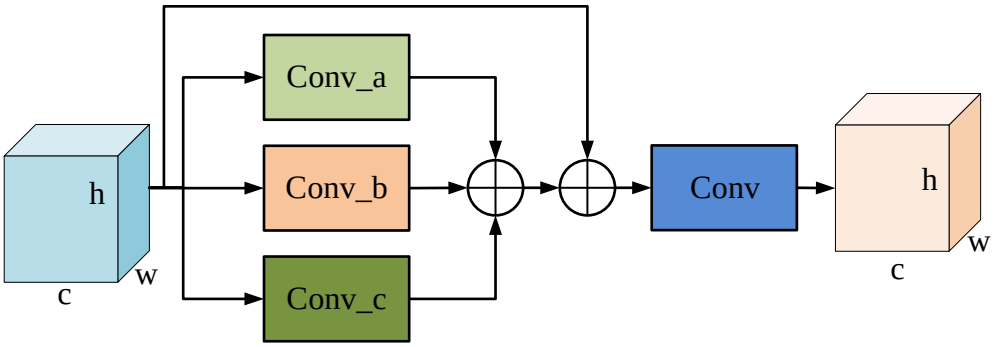


Figure 4: Multi-scale spatial feature fusion module structure

In Figure 4, this structure extracts and fuses spatial features through convolution operations at different scales, enhancing the overall feature representation ability. The channels, heights, and widths of the input feature map are convolved using three different convolution kernels. Each convolution kernel is responsible for capturing spatial information at different scales, focusing on detail features, local features, and global features [16]. Next, the three convolution results are used for feature fusion, which is achieved by adding or concatenating elements one by one to comprehensively express spatial features of different scales. The fused feature map is further processed through an additional unified convolution operation, and the final output feature map maintains the same dimension as the input feature map.

2.2 Cross-perspective gait recognition model combining BFS algorithm and feature map interaction

Although the proposed cross-perspective gait recognition model based on feature map interaction can solve some cross-perspective gait recognition problems, it has a large number of parameters and a long training time. Therefore, the study introduces BFS to improve it. The BFS algorithm traverses the nodes in the feature map in a layer-by-layer fashion, where each layer corresponds to the set of nodes that are reachable from the starting node in the same number of steps. This traversal mechanism ensures that nodes are visited in increasing topological distance, meaning that the shortest unweighted path to each reachable node is discovered naturally as a property of the traversal order. This structure supports comprehensive spatial propagation and enables effective feature interaction across receptive fields [17]. Compared with other feature propagation strategies such as Depth-First Search (DFS) or random traversal, the proposed use of BFS ensures a layer-wise, hierarchical traversal of feature map nodes, which aligns with the convolutional layer depth structure in CNNs. BFS allows the model to gradually aggregate spatial information from local to global across all receptive fields, thus supporting structured and scalable multi-scale feature fusion. DFS, in contrast, is more suited for exhaustive, non-hierarchical exploration and lacks the regularity needed for structured node updating in convolutional feature maps. BFS provides a balance between computational efficiency and structural completeness. It updates each feature node based on its neighborhood in a breadth-prioritized manner, ensuring that spatial dependencies are fully captured with controlled computational overhead. This makes BFS especially suitable for tasks requiring global feature consistency, such as gait recognition under varying viewpoints. The principle is shown in Figure 5.

To conceptually illustrate the behavior of the BFS algorithm, Figure 5 demonstrates a simplified traversal process on a feature node map. The traversal begins at node A, which first visits its immediate neighbors B and C. In the next iteration, nodes B and C each visit their respective neighbors E and F. The corresponding binary matrix reflects which nodes have been marked as "visited" at each stage. This visualization highlights the layer-wise node expansion property of BFS, where nodes are explored in increasing order of their minimal topological distance from the root node. It is worth noting that some nodes such as D are included in the structure but not traversed in this simplified demonstration, and thus are intentionally excluded from the visitation matrix. Starting with node A, it is gradually extended to neighbouring nodes at different levels by three traversals. In the first image, node A is visited and located at the starting layer of the search. At this time, the leading edge set only contains A, and the corresponding encoding for f is 1, indicating that A has been visited, while other nodes are 0. In the second figure, B and C are visited as neighboring nodes of A, entering the next layer's frontier set. f is updated to 011000, indicating that nodes B and C are marked as visited. In the third figure, nodes E and F are extended as neighboring nodes of nodes B and C into a new layer of frontier set, with f updated to 010111, indicating that E and F are also accessed, and the frontier set is extended to more nodes. In the process of multi-scale feature map interaction, node expansion is carried out in a breadth first manner, gradually fusing feature information from different perspectives from shallow to deep layers, ensuring that feature extraction has global and hierarchical characteristics [18]. BFS searches layer by

layer on the feature map and updates the status of nodes in order of distance priority. In the initial state, all nodes are set to an unvisited state, and the starting node joins the queue while being marked as visited. Its expression is shown in equation (5).

$$\begin{cases} Q = \{v_0\} \\ \text{visited}[v_0] = 1 \end{cases} \quad (5)$$

In equation (5), Q represents the queue, and visited represents whether the node has been accessed. BFS retrieves a node from the head of queue Q each time, accesses all neighboring nodes of that node, and updates the rules accordingly. In the BFS traversal mechanism, the study initializes a queue Q to manage the order of node expansion. Q = {v} indicates that the traversal begins from the starting node v, which corresponds to the initial active feature node on the feature map. The queue structure ensures that nodes are explored in a first-in, first-out manner, consistent with the breadth-first expansion strategy. To prevent revisiting the same node, the study maintains a binary visited array where visited[i] = 1 signifies that node i has already been processed. Therefore, visited[v] = 1 sets the visitation flag of the starting node immediately upon enqueueing. This prevents redundant enqueue operations during subsequent neighbor expansion stages. The rule expression is shown in equation (6).

$$\text{if visited}[u] = 0 \Rightarrow Q.\text{enqueue}(u), \quad \text{visited}[u] = 1 \quad (6)$$

Equation (6) represents adding node u to queue Q and marking node u as visited if it has not been accessed. If it is necessary to calculate the shortest path from the starting node to any node, the update formula is shown in equation (7).

$$d[u] = d[v] + 1 \quad (7)$$

In equation (7), $d[u]$ represents the shortest path distance from node u to the starting node, and $d[v]$ represents the distance from the current node v to the starting node. When queue Q is empty, BFS ends and all reachable nodes are accessed. The structure of the cross-perspective gait recognition model combining BFS algorithm and feature map interaction is shown in Figure 6.

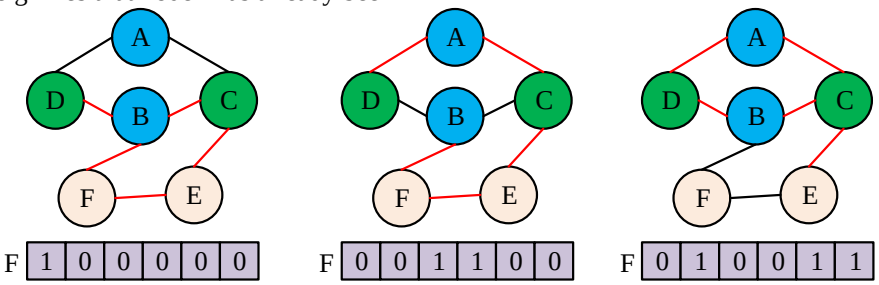


Figure 5: BFS schematic diagram

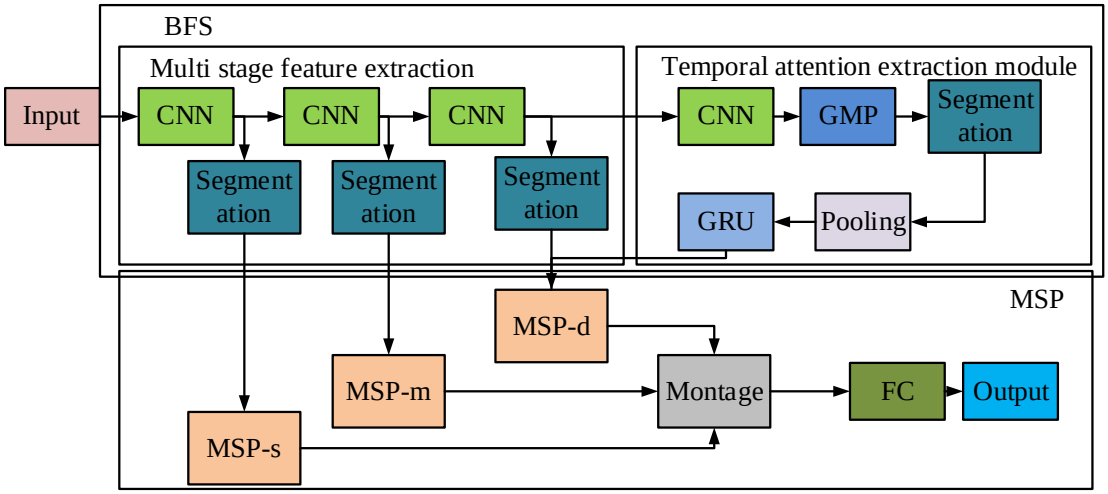


Figure 6: The BFS-CNN-GMP-GRU-MSP model structure

In Figure 6, firstly, the gait video sequence or image is input into the model and preprocessed to generate multi-scale feature maps, including shallow, middle, and deep features. Next, spatial features are extracted using convolution kernels of different scales to form an initial multi-level feature representation. Subsequently, the BFS algorithm is used to search and interact nodes on the feature map, traversing the nodes layer by layer in breadth first order. Through the propagation and accumulation of information from neighboring nodes, the feature representation is gradually updated to ensure global coverage and feature integrity of spatial

information. The queue definition update rule is shown in equation (8).

$$f_{i,j}^{t+1} = f_{i,j}^{t} + \sum_{(m,n)\in N(i,j)} W \cdot f_{m,n}^{t} \quad (8)$$

In equation (8), $f_{i,j}^{t+1}$ represents the feature values of the updated node in the $t+1$th layer search, $f_{i,j}^{t}$ represents the initial feature values of the t th layer node, $N(i,j)$ represents the set of neighboring nodes of node (i,j), and W represents the weight matrix used to control the importance of feature propagation. For feature maps at different levels, multi-scale spatial feature fusion is performed separately, using pooling operations to reduce feature dimensions while preserving key information. Next, features of different scales are fused through concatenation operations to form a unified global feature representation. Its expression is shown in equation (9).

$$F_{\text{fused}} = \text{Concat}(P(F^{a}), P(F^{b}), P(F^{c})) \quad (9)$$

In equation (9), $P(\cdot)$ represents pooling operation, used to compress the dimensionality of feature maps, and Concat represents feature concatenation operation, which combines features of different scales into global features. Finally, the fused global features are input into the FC, and the recognition results are output by the classifier, while the cross-entropy loss function is used to optimize the model.

In summary, Figure 6 illustrates the overall workflow of the proposed gait recognition model that integrates BFS, convolutional feature extraction, temporal modeling, and multi-scale feature fusion. The process begins with input gait images or sequences, which are fed into three parallel CNN branches to extract shallow, middle, and deep spatial features. These features are then processed through segmentation and GMP to reduce spatial dimensions while preserving key information. Next, the BFS mechanism is applied across feature map nodes to propagate information layer by layer, ensuring global spatial interaction and continuity across different scales. The GRU module is used to model temporal dependencies across gait frames, while GMP highlights key frames that contribute most to classification. Finally, the multi-stage features are fused in the Multi-Scale Pyramid module and passed through a fully connected layer for final gait classification. This architecture allows the model to combine spatial, temporal, and hierarchical cues effectively, resulting in high performance under varied viewing conditions. The BFS pseudocode is shown in Figure 7.

```
Breadth-First Search Based Feature Propagation on Feature Maps
Input: Feature map nodes F = {f_ij}, Adjacency matrix A
Output: Updated feature representations F'

1: Initialize queue Q ← []
2: Initialize visited[ij] ← False for all nodes (i, j)
3: For each starting node (i, j):
4:     Q.enqueue((i, j))
5:     visited[ij] ← True
6:     while Q is not empty:
7:         (i, j) ← Q.dequeue()
8:         for each neighbor (m, n) of (i, j) in A:
9:             if not visited[mn]:
10:                F[mn] ← F[mn] + W × F[ij]
11:                Q.enqueue((m, n))
12:                visited[mn] ← True
13: Return F'
```

Figure 7: BFS pseudocode

4 Results

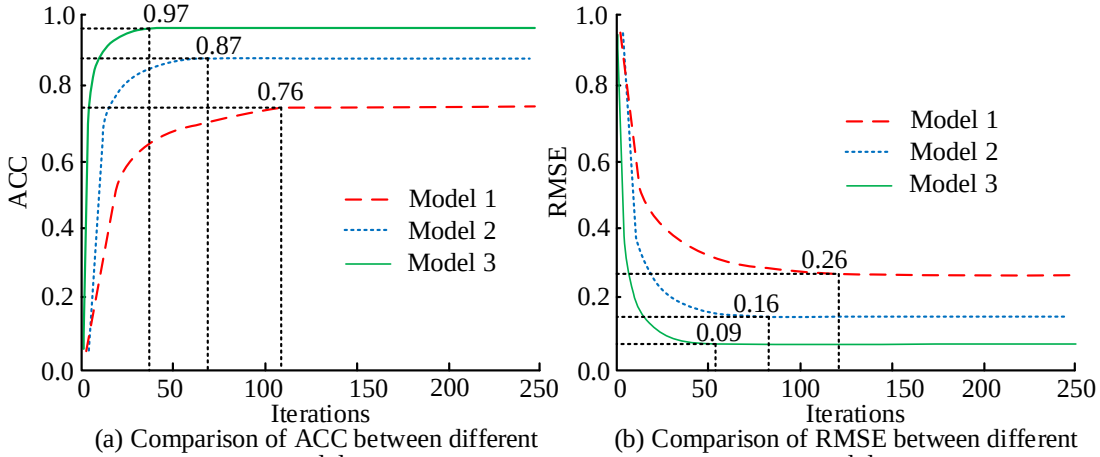
The first sub-section analyzes the performance of a cross-perspective gait recognition model that combines BFS algorithm and feature map interaction. The second sub-section applies it to practical applications and tests its performance.

3.1 Performance analysis of cross-perspective gait recognition model combining BFS algorithm and feature map interaction

In this section, the study evaluated the performance of our proposed and baseline models using the following metrics: Accuracy (ACC): The ratio of correctly classified gait sequences to the total number of test samples. F1 score (F1): The harmonic mean of precision and recall, used to assess classification balance. Error Rate: Defined as 1 minus the classification accuracy, indicating the proportion of misclassified samples. Root Mean Square Error (RMSE): Used to measure the deviation between predicted gait contour positions and ground-truth. Mean Square Error (MSE): Represents the average squared error between predicted and actual silhouette values, primarily applied to regression-based silhouette reconstruction results in Table 3. These metrics provide both classification-level and reconstruction-level insights into model performance. Particularly, MSE and RMSE quantify spatial consistency of silhouette generation, while F1 and ACC reflect recognition precision.

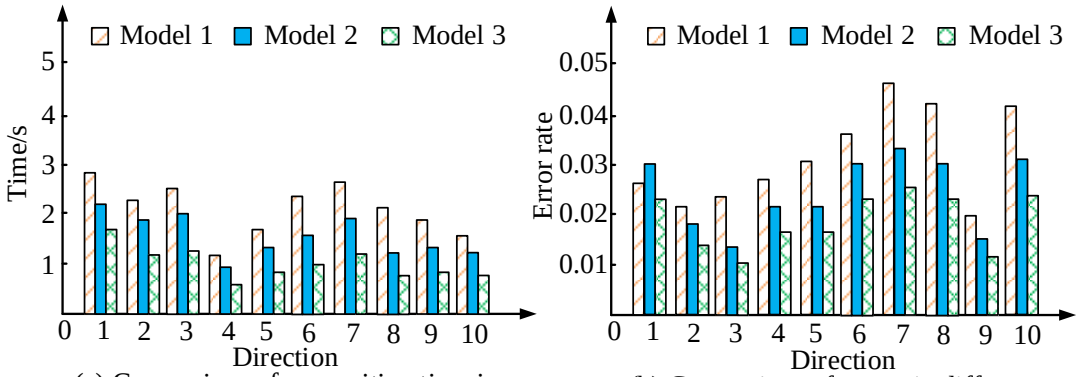
The experimental hardware configuration used Intel Core i5-8750H CPU, NVIDIA Geforce GTX2080Ti GPU, 8GB VRAM, and 16GB RAM. All experiments in this study were conducted using the CASIA-B gait dataset, a widely used public benchmark for gait recognition research. The dataset was developed by the Institute of Automation, Chinese Academy of Sciences, and contains gait sequences from 124 subjects recorded under 11 different view angles ranging from 0° to 180° at 18° intervals. Each subject was recorded under three walking conditions: normal walking, walking while carrying a bag, and walking while wearing a coat. The dataset provides both RGB video and silhouette binary images. In this study, the silhouette sequences were used after background subtraction, as they are less sensitive to clothing and lighting variations. To ensure optimal model performance, several core components underwent empirical tuning using validation ACC as the objective. For CNN layers, a kernel size of 3×3 with ReLU activation was selected to balance locality and non-linearity. All convolution blocks were followed by Batch Normalization and MaxPooling layers with stride 2 to reduce spatial dimensions and control overfitting. For the GRU module, the number of hidden units was set to 128 after grid search testing over {64, 128, 256}. A bidirectional GRU was used to better capture temporal dependencies across gait sequences. In pooling operations, GMP was chosen over average pooling based on its stronger ability to highlight key discriminative frames in gait sequences. Dropout layers (rate = 0.5) were inserted after dense layers to improve generalization.

The study selected CNN-GRU-MSP and CNN-GMP-MSP as comparative models, named Model 1 and Model 2, and named the proposed model Model 3. The performance of each model was analyzed, and the results are shown in Figure 8.



(a) Comparison of ACC between different models (b) Comparison of RMSE between different models

Figure 8: Comparison of ACC and RMSE among three gait recognition models on CASIA-B dataset



(a) Comparison of recognition time in different directions (b) Comparison of errors in different directions

Figure 9: Recognition time and error rate of three models under different viewpoint directions

Figure 8 (a) shows the comparison of ACC between different models during the iteration process, and Figure 8 (b) shows the comparison of root mean square error (RMSE) between different models. From Figure 8 (a), Model 3 performed the best, with ACC quickly reaching a stable value of 0.97 after about 20 iterations, with the fastest convergence speed and highest ACC. Model 2 followed closely and stabilized at 0.87 after about 50 iterations, with higher ACC but slower convergence speed than Model 3. Model 1 performed the worst, converging after 100 iterations, with a final ACC of only 0.76. In Figure 8 (b), Model 3 had the lowest RMSE,

which rapidly decreased and stabilized at 0.09 after about 50 iterations, indicating that it had the smallest error and the strongest generalization ability. The RMSE of Model 2 was 0.16, with a slightly slower stabilization time but still better than Model 1. The RMSE of Model 1 converged slowly, with a final error of 0.26 and the maximum error. The experimental results showed that the proposed Model 3 had high ACC and low error in cross prospective gait recognition, exhibiting the best performance and robustness. The gait data were selected in different directions and the data were collected from 1 different viewpoint with an angle range of 0° to 180° and an interval of 18°, and the results are shown in Figure 9.

Figure 9 (a) shows a comparison of the recognition time of three models for different directions, and Figure 9 (b) shows a comparison of the recognition errors of three models for different directions. According to Figure 9 (a), Model 1 had the longest duration, with some directions such as Direction 1 and Direction 7 taking nearly 3 seconds. The recognition time of Model 2 was shortened, with most directions ranging from 1.5 to 2.5 seconds. Model 3 performed the best with the shortest time, with most directions taking less than 1.5 seconds, especially in directions 4 and 10 where the time was close to 1 second. From Figure 9 (b), Model 1 had the highest error rate, especially in direction 7 where the error rate was close to 0.05, indicating that Model 1 had weak adaptability to complex direction or perspective changes. The error rate of Model 2 was reduced, with most directions remaining between 0.02 and 0.03, indicating that the GMP module enhanced its ability to screen features, but its adaptability to complex directions was still limited. The error rate of Model 3 was the lowest, with an overall error rate below 0.02, and the error rates of Direction 3 and Direction 9 were close to 0.01. The experimental results showed that the proposed Model 3 had excellent model performance. Using ablation experiments, the performance of each part of the model was analyzed, and the results are shown in Table 2.

Table 2: Ablation test table

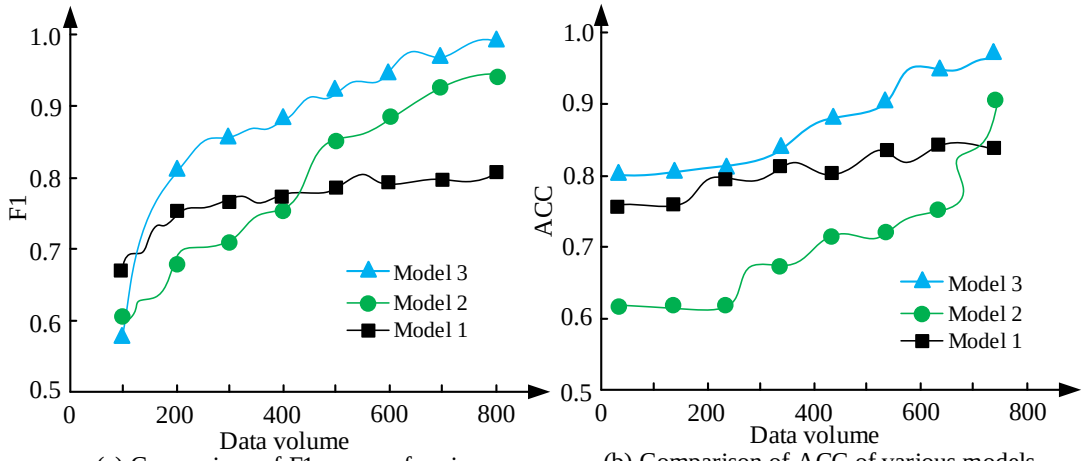
Model	ACC	RMSE	Recognition time/s	Error rate
BFS-CNN-GMP-GRU-MSP	0.97	0.09	1.2	0.012
BFS-CNN-GMP-GRU	0.91	0.13	1.8	0.018
BFS-CNN-GRU-MSP	0.85	0.21	1.5	0.025
BFS-CNN-GMP-MSP	0.88	0.17	1.7	0.02
CNN-GMP-GRU-MSP	0.83	0.24	2.0	0.032
Model	F1	Recall	Precision	/
BFS-CNN-GMP-GRU-MSP	0.96	0.95	0.97	/
BFS-CNN-GMP-GRU	0.89	0.88	0.90	/
BFS-CNN-GRU-MSP	0.82	0.81	0.83	/
BFS-CNN-GMP-MSP	0.86	0.84	0.87	/
CNN-GMP-GRU-MSP	0.80	0.79	0.82	/

According to Table 2, BFS-CNN-GMP-GRU-MSP performed the best, with ACC reaching 0.97, RMSE being the lowest at 0.09, recognition time only 1.2 seconds, error rate being the lowest at 0.012, F1 score, recall rate, and ACC rate being 0.96, 0.95, and 0.97, respectively. This indicated that the BFS algorithm combined with multiple modules could efficiently extract cross-perspective gait features, and the model had high ACC and excellent computational efficiency. After removing BFS, the model BFS-CNN-GMP-GRU showed a decrease in ACC to 0.91, an increase in RMSE to 0.13, an increase in recognition time to 1.8 seconds, and an increase in error rate to 0.018, demonstrating the importance of BFS algorithm in accelerating feature propagation and optimizing ACC. The performance of the BFS-CNN-GRU-MSP model after removing GMP decreased significantly, with ACC at 0.85, RMSE increasing to 0.21, and error rate increasing to 0.025, indicating that the GMP module played a key role in feature screening and noise reduction. After removing the GRU from the BFS-CNN-GMP-MSP model, the ACC decreased to 0.88 and the RMSE was 0.17, indicating that GRU had a significant effect on time series feature modeling. While the ablation results in Table 2 demonstrate noticeable drops in performance when individual modules are removed (e.g., GMP, GRU, or BFS), The study acknowledge that these tests evaluate components in isolation and do not capture potential interaction effects between modules. To more rigorously assess these relationships, a full-factorial ablation analysis would be necessary. However, given space constraints, we focused on evaluating the marginal contribution of each module. In future work, we plan to investigate combinatorial ablations (e.g., removing both GRU and GMP) to better understand interdependencies and possible synergy among architectural components.

To verify the statistical significance of the observed performance differences, pairwise two-tailed t-tests were conducted between the proposed model and the baselines (CNN-GRU-MSP and CNN-GMP-MSP). The results indicated that the improvements in ACC ($p < 0.01$) and F1 score ($p < 0.01$) were statistically significant across all tested conditions.

3.2 Simulation result analysis

The study selected CNN-GRU-MSP and CNN-GMP-MSP as comparative models, named Model 1 and Model 2, and named the proposed model Model 3. To further validate the performance of the model, simulation analysis was used to analyze the images in actual situations, and the results are shown in Figure 10.



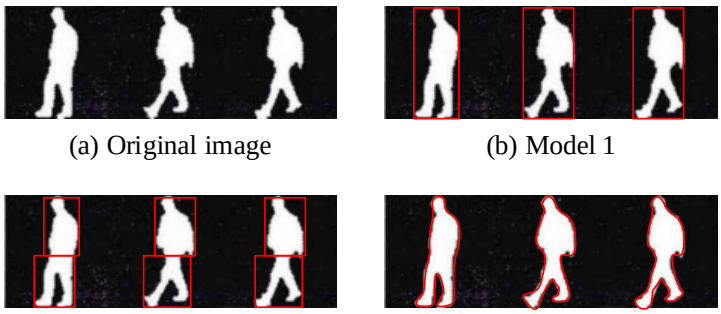
(a) Comparison of F1 scores of various models under different data volumes

(b) Comparison of ACC of various models under different data volumes

Figure 10: F1 and ACC comparison of three models under varying training data volumes

Figure 10 (a) shows the F1 scores of the three models under different data volumes, and Figure 10 (b) shows the ACC of the three models under different data volumes. As shown in Figure 10, both ACC and F1 generally increased for Model 1 and Model 3 as training data volume grows, demonstrating improved generalization ability. However, Model 2 exhibited a decline in F1 after a certain data threshold, despite its ACC still increasing slightly. This behavior suggested that Model 2 may become overfitted to dominant class patterns in the expanded dataset, leading to degraded recall and thus lower F1. This implies that without BFS or GRU integration, the model lacks sufficient temporal representation or inter-feature interaction to maintain balanced classification under more diverse gait inputs. In contrast, Model 3 maintained consistent or even slightly improved F1 performance across scales, validating the contribution of BFS-driven feature propagation and GRU-based temporal modeling in resisting overfitting and improving classification robustness. The experimental results showed that the proposed model performed the best in both F1 score and ACC, with better generalization and data utilization ability. The recognition performance of each model was analyzed, and the results are shown in Figure 11.

Figure 11 (a) shows the original image, while Figures 11 (b), 11 (c), and 11 (d) respectively demonstrate the recognition performance of Model 1, Model 2, and Model 3. From Figure 11, the original gait image contained the contours of pedestrians walking. Although Model 1 could locate the contours of pedestrians, there were obvious local truncation phenomena, such as missing information in the legs and head. Model 2 showed some improvement in the localization process, but there were still issues with false positives and feature truncation, such as incomplete extraction of the leg region and inaccurate alignment of some red boxes with the contour edges. The comprehensive performance of each model was analyzed, and the results are shown in Table 3.



(a) Original image (b) Model 1

(c) Model 2 (d) Model 3

Figure 11: Visualization of gait contour recognition outputs for the three models

Table 3: Comprehensive performance analysis of the model

Type	Model	ACC	F1	RMSE	MSE	Time/s
Normal	Model 1	0.84	0.82	0.22	0.048	2.0
	Model 2	0.92	0.89	0.12	0.014	1.6
	Model 3	0.97	0.96	0.09	0.008	1.2
Carrying a bag	Model 1	0.79	0.76	0.25	0.063	2.1
	Model 2	0.88	0.85	0.14	0.02	1.7
	Model 3	0.94	0.92	0.11	0.012	1.4
Clothing	Model 1	0.76	0.74	0.28	0.078	2.3
	Model 2	0.86	0.82	0.16	0.025	1.8
	Model 3	0.91	0.89	0.14	0.02	1.5

According to Table 3, under normal gait, Model 3 had an ACC of 0.97, an F1 score of 0.96, the lowest RMSE of 0.09, and the shortest recognition time of 1.2 seconds. Model 2 had an ACC of 0.92, while Model 1 had an ACC of only 0.84, an RMSE of up to 0.22, and a recognition time of 2.0 seconds. In the state of carrying objects, Model 3 had a stable ACC of 0.94 and RMSE of 0.11, while Model 2 and Model 1 had ACC of 0.88 and 0.79, respectively.

4 Discussion

The experimental results and comparative analysis in Related Works clearly demonstrated that the proposed BFS-CNN-GMP-GRU-MSP model outperformed existing gait recognition methods in multiple evaluation metrics. Compared to models like that of Yan et al., the proposed model achieved 93.5% ACC on the CASIA-B dataset, The study model achieves up to 97.0% accuracy under normal conditions and maintains robust performance (94.0% and 91.0%) under challenging conditions such as carrying objects or wearing clothing. This performance gain is primarily attributed to two core innovations: the BFS-driven global feature propagation and the multi-scale feature map interaction. The BFS algorithm ensures exhaustive traversal of feature nodes across all spatial scales, allowing the model to capture hierarchical and contextual spatial patterns that static CNN layers or short-range skip connections might miss. This contributes to the model's superior generalization across viewpoints. Meanwhile, the multi-stage feature map interaction enables the fusion of shallow, middle, and deep spatial features, preserving both local detail and global structure. Combined with GRU-enhanced temporal modeling, the model can dynamically allocate attention to key gait frames, thereby reducing temporal noise and enhancing feature stability. However, these performance benefits come at a computational cost. The inclusion of multi-branch CNN modules, GRU layers, and BFS-based traversal increases both model complexity and training time. For instance, compared to baseline CNN-GRU-MSP models, the designed model takes approximately 30%–40% longer to train and requires more GPU memory during inference. While this trade-off is acceptable in offline or controlled environments, it may limit the model's deployment in resource-constrained scenarios such as edge devices or mobile platforms. To mitigate these inefficiencies, future work could explore lightweight alternatives. These include pruning and quantization strategies for CNNs, replacing GRUs with more efficient attention-only mechanisms, or using graph convolutional approximations to emulate BFS behavior without full traversal overhead. Moreover, a dynamic perspective-adaptive module could be integrated to adjust feature processing based on input complexity, further improving computation-to-accuracy ratios.

In summary, the proposed method demonstrates superior robustness and accuracy in cross-view gait recognition, driven by its spatial-temporal fusion strategy. While computational costs are a concern, they are justified by the substantial gains in recognition performance. Nonetheless, ongoing optimization of model efficiency remains an important future direction.

5 Conclusion

A gait recognition model combining BFS algorithm and multi-scale feature map interaction was proposed to address the issues of viewpoint changes and computational efficiency in cross-perspective gait recognition. The model extracted multi-scale spatial features of shallow, middle, and deep layers through CNN. The BFS algorithm searched for nodes in the feature map layer by layer to ensure the propagation and fusion of global information. In the ablation experiment, after removing the BFS algorithm, the ACC of the model decreased to 0.91, the RMSE increased to 0.13, and the recognition time increased to 1.8 seconds, indicating the critical role of BFS in global feature map search and information propagation. After removing the GMP module, the RMSE further increased to 0.21, indicating that GMP effectively strengthened the feature weights at key time points. When removing GRU, the time-dependent characteristics of the model were suppressed, and the RMSE reached 0.17, highlighting the importance of GRU in temporal modeling. The research results indicated that the proposed model had excellent model performance. Although the study has achieved good results, there is still room for optimization in terms of computational complexity and training time on large-scale datasets. In the future, it will further combine lightweight networks with adaptive feature selection strategies to improve the computational efficiency and

generalization ability of the model in practical application scenarios.

Funding

Henan Province Intelligent Transportation Video Image Perception and Recognition Engineering Technology Research Center (Yukeshi [2024] No. 1).

References

[1] Ma C, Liu Z. mDS-PCGR: A Bimodal Gait Recognition Framework with the Fusion of 4-D Radar Point Cloud Sequences and Micro-Doppler Signatures. IEEE sensors journal, 2024, 24(6):8227-8240. https://doi.org/10.1109/JSEN.2024.3355421

[2] Kalembo Vikalwe Shakrani, Ngonidzashe Mathew Kanyangarara, Prince Tinashe Parowa, Vibhor Gupta, Rajendra Kumar. A Deep Learning Model for Face Recognition in Presence of Mask. Acta Informatica Malaysia. 2022; 6(2): 43-46. https://doi.org/10.26480/aim.02.2022.43.46

[3] Rifaat N, Ghosh U K, Sayeed A. Accurate gait recognition with inertial sensors using a new FCN-BiLSTM architecture. Computers and Electrical Engineering, 2022, 104(2):1048-1056. https://doi.org/10.1016/j.compeleceng.2022.108428

[4] Parashar A, Parashar A, Ding W. Deep learning pipelines for recognition of gait biometrics with covariates: a comprehensive review. Artificial Intelligence Review, 2023, 56(18):8889-8953. https://doi.org/10.1007/s10462-022-10365-4

[5] Castro F, Impedovo D, Pirlo G. A Hybrid Protection Scheme for the Gait Analysis in Early Dementia Recognition. sensors, 2024, 24(1):41-57. https://doi.org/10.3390/s24010024

[6] Baniasad M, Martin R, Crevoisier X. Automatic Body Segment and Side Recognition of an Inertial Measurement Unit Sensor during Gait. Sensors (14248220), 2023, 23(7):121-136. https://doi.org/10.3390/s23073587

[7] Zhang W, Ju L, Jia H. Semiring-Optic-Fiber (SROF) Sensor-Based Abnormal Gait Recognition via Monitoring Muscle Activation. IEEE sensors journal, 2023, 23(17):19307-19317. https://doi.org/10.1109/JSEN.2023.3292923

[8] Derlatka M, Borowska M. Ensemble of Heterogeneous Base Classifiers for Human Gait Recognition. Sensors (Basel, Switzerland), 2023, 23(1):321-323. https://doi.org/10.3390/s23010508

[9] Yan S, Hu L, Xueling F. GaitASMS: gait recognition by adaptive structured spatial representation and multi-scale temporal aggregation. Neural computing & applications, 2024, 36(13):7057-7069. https://doi.org/10.1007/s00521-024-09445-z

[10] Topham L K, Khan W, Al-Jumeily D H A. Human Body Pose Estimation for Gait Identification: A Comprehensive Survey of Datasets and Models. ACM computing surveys, 2023, 55(6):120.1-120.42. https://doi.org/10.1145/3533384

[11] Parashar A, Shekhawat R S. Protection of gait data set for preserving its privacy in deep learning pipeline.IET Biometrics, 2022, 11(6):557-569. https://doi.org/10.1049/bme2.12093

[12] Luo J, Zhang H, Sun, Chuanyue Jing, Yangmin Li, Kerui Li, Yaogang Zhang, Qinghong Wang, Hongzhi Luo, YangHou, Chengyi. Topological MXene Network Enabled Mixed Ion-Electron Conductive Hydrogel Bioelectronics. ACS nano, 2024, 18(5):4008-4018. https://doi.org/10.1021/acsnano.3c06209

[13] Jain R S, Pemawat A, Sharma P. Expanding the Understanding of Stiff-Person Syndrome: Insights from 17 Cases in India. Annals of Indian Academy of Neurology, 2024, 27(4):72-74. https://doi.org/10.4103/aian.aian_92_24

[14] Alexis J, Bailey N, Joseph F. A - 124 A Case of Lewy Body Dementia and Charles Bonnet Syndrome in a Patient with Bilateral Enucleation. Archives of Clinical Neuropsychology, 2024(7):27-35.

[15] Saminu S, Xu G, Zhang S, Kader IAE, Aliyu HA, Jabire AH, Ahmed YK, Adamu MJ. Applications of Artificial Intelligence in Automatic Detection of Epileptic Seizures Using EEG Signals: A Review. Artificial Intelligence and Applications, 2023,1(1): 11-25. https://doi.org/10.47852/bonviewAIA2202297

[16] Ahmed D M, Mahmood B S. Integration of Face and Gait Recognition via Transfer Learning: A Multiscale Biometric Identification Approach. Traitement du Signal, 2023, 40(5): 2179-2190. https://doi.org/10.18280/ts.400535

[17] Dzemyda G, Sabaliauskas M, Medvedev V. Geometric MDS Performance for Large Data Dimensionality Reduction and Visualization. Informatica, 2022, 33(2):299-320. https://doi.org/10.15388/22-INFOR491

[18] Mehta P, Aggarwal S, Tandon A. The Effect of Topic Modelling on Prediction of Criticality Levels of Software Vulnerabilities. Informatica, 2023, 8(22):283-304. https://doi.org/10.31449/inf.v47i6.3712