Mathematical Modelling of Engineering Problems
Vol. 12, No. 10, October, 2025, pp. 3385-3398
Journal homepage: http://iieta.org/journals/mmep

HoloGait: A Holistic Graph-Transformer Framework for Cross-View Gait Recognition

Kiran Kumar S V N Madupu[1*], Jagadeeshwar M[1], Naren Kodali[2], G Soma Sekhar[3]

[1] Department of Computer Science, Chaitanya Deemed to be University, Himayatnagar, Telangana 500075, India
[2] Department of Computer Science, George Mason University, Fairfax 22030, Virginia, USA
[3] Department of CSE (Cyber Security), Geethanjali College of Engineering and Technology, Hyderabad 501301, India

Corresponding Author Email: msvnkiran@outlook.com

https://doi.org/10.18280/mmep.121005

Received: 18 July 2025
Revised: 23 September 2025
Accepted: 30 September 2025
Available online: 31 October 2025

Keywords:
CNN-RNN hybrid model, cross-view identification, dual attention, gait recognition, spatial-temporal modeling, person re-identification, Bi-Gru, multi-task learning

ABSTRACT

Cross-view gait recognition remains challenging due to variations in viewing angles, temporal misalignment, occlusions, and appearance changes, which significantly reduce the effectiveness of current biometric systems. To address these challenges, this paper introduces HoloGait, a multi-modal framework designed for robust cross view gait recognition. HoloGait fuses complementary information from appearance-based silhouettes and structural 3D skeletal poses, leveraging the distinct advantages of each modality. Central to the approach is a rigorous 3D pose alignment module that normalizes skeletal data to a canonical view, significantly minimizing viewpoint dependency. Additionally, the structural information from 3D poses is explicitly modeled using an adaptive graph convolutional network (GCN), capturing intricate joint dynamics and interactions. Subsequently, a transformer-based fusion module integrates silhouette and skeletal features, dynamically exchanging spatial-temporal cues between the two modalities. A multi-task objective (identity classification + triplet loss) further enhances discriminative capabilities, producing embeddings resilient to occlusion and temporal inconsistencies. Experiments on the TUM GAID dataset using a strict cross view protocol—training on 0° and 45° views and testing on 90°—demonstrate that HoloGait achieves substantial performance improvements. Notably, HoloGait attains a Rank-1 accuracy of 96.8%, outperforming recent methods such as CART-Gait by 4.7% and GaitSet by 10.4%, thus clearly establishing state-of-the-art performance. These advancements confirm HoloGait's capability to provide comprehensive, reliable, and accurate gait recognition suitable for practical biometric identification tasks under realistic conditions.

1. INTRODUCTION

Gait recognition identifies people from walking patterns and is useful at a distance without cooperation, but performance degrades under real-world conditions. Major obstacles include large changes in camera view, temporal misalignment and speed variation, occlusion of body parts, and appearance factors such as clothing changes and carried objects, all of which reduce recognition accuracy [1-7].

HoloGait addresses these obstacles by combining two complementary inputs: silhouettes (appearance) and 3D skeletons (structure). 3D poses are first aligned to a canonical view to reduce view differences, then the two modalities are fused with early, bi-directional cross-attention to form a single gait embedding.

This design encourages learning of identity-related motion and shape while limiting the effect of viewpoint and occlusion.

Appearance-based approaches, which analyze gait through 2D silhouettes, have shown high accuracy in controlled conditions, typically using convolutional neural networks (CNNs) [8]. However, silhouettes generally provide limited information and are easily affected by changes in viewpoint, occlusion, and background noise [1]. Some methods handle this by explicitly separating viewpoint from identity or normalizing silhouettes to a common view using spatial transformations [3]. In contrast, model-based methods use explicit body information like skeleton poses, providing some natural robustness to viewpoint and clothing variations. For example, PoseGait by Liao et al. [9] used 3D skeleton data combined with human body knowledge to handle view differences effectively. Similarly, GaitGraph introduced by Teepe et al. [10] used the graph convolutional network (GCN) to learn gait patterns directly from skeletal graphs. Recent works also introduced detailed 3D body meshes (e.g., Skinned Multi-Person Linear (SMPL) models) that provide richer pose and shape information, improving accuracy across viewpoints [11, 12]. Despite these advantages, skeleton-based methods usually produce limited features, lacking detailed shape information found in silhouette-based methods [1]. To overcome this, several recent studies combined both appearance (silhouettes) and skeletal poses to gain richer and more consistent features. SkeletonGait++, for instance, fused skeleton-based features with silhouette features through attention mechanisms, improving performance significantly

across different viewpoints. Still, existing fusion approaches often fall short in effectively managing temporal misalignments and partial occlusions.

This paper introduces HoloGait, a multi-modal gait recognition method designed to manage these problems. HoloGait combines appearance information from silhouettes with structure information from 3D skeletal poses, using strengths from each study [3]. Silhouettes provide detailed body shape and general motion cues, while skeletal data naturally offer consistency across viewpoints. By merging these sources, HoloGait creates a unified representation of gait that remains stable despite viewpoint variations and occlusions. At its core, HoloGait uses a hybrid model mixing graph and transformer layers. Skeletal data are first structured as graphs, connecting body joints, and processed using GCN layers to learn local joint movements and interactions. Because typical GCNs have limited reach, multi-head attention modules are integrated, capturing global temporal patterns and distant joint relationships [13-15]. This combination helps to identify important movements and their timing clearly, capturing details missed by simpler models. Additionally, HoloGait employs a 3D pose alignment method, converting all input skeletons to a standard, fixed viewpoint (like frontal view) before processing. Such alignment removes viewpoint differences from the data itself, simplifying the recognition process and allowing focus on distinctive gait features [3]. Finally, the model uses multi-task learning by training simultaneously on gait identification and auxiliary tasks like attribute recognition. Multi-task training generally leads to richer, more balanced representations, improving performance across diverse conditions and viewpoints [16].

Key contributions are as follows:

(1). Holistic multi-modal fusion of silhouettes and 3D skeletal pose features to obtain a unified representation that reduces sensitivity to viewpoint and appearance changes.

(2). Hybrid graph–transformer architecture in which graph layers capture localized joint interactions and temporal transformers model long-range dependencies, improving gait feature extraction.

(3). Canonical 3D pose alignment that normalizes all skeletons to a single viewpoint before feature extraction, simplifying recognition across angles.

(4). Multi-task learning with identity classification and auxiliary attributes, which regularizes the embedding and improves robustness across diverse conditions.

Each aspect mentioned addresses a clear gap found in existing gait recognition methods, especially regarding multi-modal fusion, structural-temporal modeling, and viewpoint normalization. By combining these features into a unified approach, HoloGait provides a practical method for handling common issues in gait recognition, particularly in challenging scenarios involving diverse viewpoints, occlusions, and variable walking patterns [1, 3, 9, 10].

2. RELATED WORK

Appearance-based methods: Early deep learning methods for gait recognition mainly used appearance-based inputs (silhouettes) or model-based inputs (skeletal poses). Appearance-based methods analyze sequences of body silhouettes as gait signatures. GaitSet by Chao et al. [17] is a prominent example, which treats gait sequences as unordered frame sets. It aggregates features at the frame level using set pooling to achieve good cross-view accuracy without explicit sequence modeling. Later methods improved accuracy by dividing silhouettes into smaller parts to capture subtle motions. For instance, GaitPart [18] extracts regional micro-motion features separately for different body parts, and GaitGL [19] combines both global and local features from silhouette regions to preserve finer details. These CNNs, including methods like GaitGANv2 and DANet, improved recognition accuracy by better spatial and temporal feature extraction. Despite their strengths, silhouettes usually have limitations. They lack explicit structural details, are sensitive to clothing variations and occlusions, and foreground extraction often introduces errors [1].

Gap/contrast: Unlike appearance-only CNN methods, HoloGait adds aligned 3D skeleton cues and performs early bi-directional fusion with silhouettes to reduce viewpoint changes and occlusions while preserving silhouette detail.

Model-based (skeleton/3D): Model-based methods utilize explicit human body models (skeletal poses) to achieve natural invariance to appearance changes. Early examples such as PoseGait by Liao et al. [9] used 3D body joint coordinates to produce gait features resistant to clothing and viewpoint changes. Later models employed GCNs directly on skeletal data structured as joint graphs. For example, Teepe et al.'s GaitGraph [10] represents each body joint as a node connected by edges representing the human skeleton structure, effectively capturing movement patterns. Further advancements expanded skeletal information by including bone vectors or joint velocities [20]. Higher-order GCNs with residual connections have also been developed, significantly improving gait recognition accuracy [13]. Generally, these model-based methods effectively capture body movements independent of clothing. However, skeletal representations usually have fewer details compared to silhouettes, limiting their recognition performance, especially in subtle identity differences [14, 15]. This has motivated integrating skeleton data with silhouettes to enrich gait representations.

Gap/contrast: HoloGait retains the strengths of skeleton models but overcomes limited shape detail by fusing silhouettes and by normalizing all skeletons to a canonical 3D view before learning.

Multi-modal fusion: Combining silhouettes and skeletal poses into a multi-modal approach leverages the strengths of both inputs. Recent models, such as SkeletonGait++ by Fan et al. [21], transform skeleton sequences into heatmap-like skeleton maps and then fuse them with silhouette features using attention. GaitMA by Min et al. [22] uses parallel CNN streams separately on silhouettes and pose heatmaps, fusing their outputs via mutual co-attention modules. PSGait [1] further introduces a parsing skeleton method to create part-specific silhouette masks guided by skeletal poses, capturing detailed dynamics. These multi-modal methods typically outperform single-modality methods because silhouettes provide rich shape cues, while skeletons provide explicit motion geometry, each compensating for the other's shortcomings. Researchers have also started using richer 3D body models (e.g., SMPL) alongside silhouettes for even more detailed fusion, moving towards comprehensive gait representations suitable for real-world scenarios. In a related vein, the dynamic aggregation network (DANet) proposed by Ma et al. [23] uses attention modules to adaptively aggregate features across frames. DANet's attention-based aggregator learns how much each frame or part contributes to the final gait signature, dynamically emphasizing salient gait poses and

motions.

Gap/contrast: HoloGait differs by performing early bi-directional cross-attention between modalities (not only late concatenation) and by fusing after canonical pose alignment to avoid view conflicts.

Hybrid (GCN+Transformer): Alongside modality integration, hybrid deep network architectures combining CNNs, GCNs, and transformers have been developed. Earlier CNN-based models, such as GaitPart [18] and GaitGL [19], already included local spatial regions and temporal modeling but struggled to capture long-range temporal patterns or structured relationships across joints. To overcome this limitation, graph neural networks and transformers became common. GCN-based approaches like GaitGraph naturally represent the human body's skeletal structure, leading to better understanding of joint movements. More recent methods, such as MS-Gait by Liu et al. [20], expanded this by using multiple GCN streams focused on joints, bones, and motion differences, achieving significant accuracy improvements. However, a key limitation of GCNs remains their narrow temporal focus. Transformer models address this limitation effectively by modeling longer temporal dependencies through attention mechanisms. Models such as GaitFormer by Cosma and Radoi [24] and GaitTransformer by Cui et al. [25] apply vision transformers to capture global gait patterns over time. These transformers excel at linking similar gait phases across distant time frames, greatly improving cross-view stability.

Recent research increasingly combines CNNs, GCNs and transformers into hybrid architectures. For instance, GaitCoTr by Li et al. [26] uses CNNs for detailed spatial features and transformers for temporal context. Similarly, recent skeleton-based approaches integrate transformers atop GCN outputs, effectively forming graph-transformer hybrids. These hybrids capture spatial joint interactions via graphs and temporal patterns via transformer attention. This integration provides a balanced approach to extracting complete gait dynamics compared to purely CNN-based or GCN-based methods alone.

Gap/contrast: HoloGait combines a transformer-based appearance branch with an adaptive GCN plus temporal transformer pose branch, and couples them through cross-modal attention to cover both local joint/part structure and long-range temporal patterns in one unified design.

Cross-view strategies: Achieving robust cross-view recognition remains particularly challenging since gait patterns appear differently from varied viewpoints. Many methods have explored geometric pose normalization or alignment. Sokolova and Konushin [27], for example, projected gait features onto a common viewpoint to reduce view-induced variations. Zheng et al. [14] used 3D SMPL body models to inherently normalize views by reconstructing full 3D body shapes. Another common approach involves learning feature representations that explicitly separate view variations. Zhang et al. [28] introduced an angle-center loss function that clusters gait features by identity while dispersing them by viewing angle, significantly improving cross-view matching. Other methods employ adversarial domain adaptation, where the system learns to produce view-invariant gait features by treating each viewpoint as a separate domain. Yu et al. [29] and Wu et al. [30] demonstrated such a domain adaptation strategy in GaitDAN, achieving view-invariant feature learning. Data augmentation techniques, such as Generative Adversarial Network (GAN)-based silhouette generation from new viewpoints (GaitGAN, MvGGAN [29]), further enhance cross-view generalization. Additionally, training strategies specifically aimed at cross-view recognition have emerged. CART-Gait by Liu et al. [20] uses refined training that emphasizes consistency across different viewing angles. These approaches show that combining geometric normalization, specialized losses, domain adaptation, and targeted training strategies significantly improves cross-view gait recognition performance. Gap/contrast: HoloGait applies explicit canonical 3D pose alignment as a pre-normalization step and then fuses modalities, instead of relying only on adversarial adaptation, view losses, or GAN-based synthesis. Summarizes prior methods by “Method / Modality / Temporal modeling / View strategy / Limitation addressed by HoloGait” to make these contrasts concrete.

HoloGait uniquely integrates the strengths of previous methods. Unlike prior models focusing on single aspects, HoloGait combines multi-modal fusion of silhouette and skeletal pose features from the start. Instead of late-stage feature fusion seen in models like SkeletonGait++, HoloGait uses early cross-attention between modalities to form unified representations. Additionally, the architecture blends GCN-based modeling for structured joint motion and transformer-based temporal attention, capturing both local joint interactions and long-range temporal gait dynamics simultaneously. Previously, methods typically used GCNs or transformers independently or added simple temporal layers over GCNs. By integrating these directly into one model, HoloGait addresses both spatial and temporal limitations seen in earlier methods. Finally, HoloGait incorporates canonical 3D pose alignment as a fundamental feature, normalizing skeletal poses before feature extraction. This approach ensures skeletal inputs are inherently view-invariant, complemented by a generative silhouette transformation module that aligns silhouettes to a common viewpoint, similar in approach to prior GAN-based methods [29]. Thus, HoloGait provides a comprehensive solution by merging multi-modal fusion, graph-transformer hybrid modeling, and built-in 3D pose alignment into a unified framework. Individually, each component follows from existing research: multi-modal fusion inspired by PSGait [1], graph-transformer structure building upon GCN-transformer hybrids [26], and view normalization drawing from CART-Gait [20] and adversarial domain adaptation [30]. Collectively, these combined elements address the gaps present in earlier single-focus approaches, providing a complete and practical method for addressing major challenges in gait recognition.

3. METHODS AND MATERIALS

The proposed HoloGait framework systematically integrates complementary features from silhouette appearance and structural skeleton poses within a unified multi-modal architecture illustrated conceptually in Figure 1. The pipeline initiates with synchronized extraction of silhouette masks and corresponding 3D skeleton poses from raw input video sequences. Specifically, silhouettes are obtained through Gaussian Mixture Model (GMM)-based segmentation, followed by size normalization and spatial centering, whereas 3D skeletons are derived via OpenPose-based 2D joint detection and subsequent VideoPose3D lifting, and are rigorously aligned into a canonical orientation to ensure view invariance.

Subsequently, HoloGait employs two parallel feature

extraction branches:

·**Appearance Branch**: Utilizes a ResNet-50 backbone followed by a spatial transformer module to explicitly encode spatial relationships among anatomically segmented silhouette regions (e.g., head, torso, limbs). Frame-level spatial features are then temporally aggregated using a temporal transformer, producing robust appearance-based gait embeddings.

·**Structural Branch**: Represents the aligned 3D skeleton sequence as a spatio-temporal graph (nodes as joints, edges as anatomical bones), extracting joint-level motion features via an Adaptive GCN. These spatial graph features are temporally aggregated using a temporal graph transformer encoder, yielding comprehensive motion-based embeddings.

The embeddings from these dual branches are deeply integrated through a dedicated cross-modal fusion transformer. This module employs mutual Multi-Head Cross-Attention (MHCA) operations, facilitating rich, bi-directional interactions between silhouette appearance and pose structure, thereby producing a unified embedding that captures both spatial appearance and motion dynamics effectively.

Finally, the unified embedding serves as input to two primary multi-task output heads: (i) an identity classification head leveraging cross-entropy loss for subject identification, and (ii) an embedding generation head trained with a triplet loss to ensure discriminative representation learning. Optionally, auxiliary attribute prediction tasks (e.g., gender, clothing type) further enrich feature representations through additional classification losses. This structured multi-task training approach enhances generalization and robustness, significantly improving cross-view gait recognition performance.

HoloGait illustrating the dual-branch inputs (silhouette sequences and aligned 3D poses), parallel extraction of spatial-temporal appearance and structural features, the cross-modal fusion transformer integrating complementary information, generative view normalization for viewpoint invariance, and final multi-task output heads producing identity classification and discriminative embedding representations.

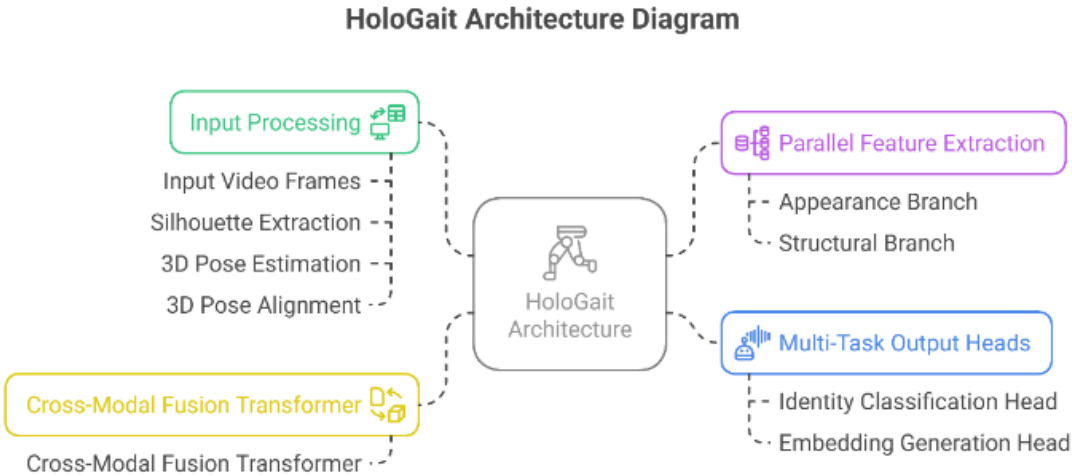

Figure 1. Conceptual architecture of HoloGait

3.1 Input processing and 3D pose alignment

The HoloGait framework requires precise and synchronized extraction of silhouette and 3D skeletal data from raw video frames. The preprocessing pipeline consists of two parallel stages: silhouette extraction and 3D pose estimation, followed by a critical pose alignment step to normalize viewpoint variations.

Silhouette extraction: Silhouette sequences $S = \{S_t\}_{t=1}^{T}$ are generated from the raw video frames $V = \{I_t\}_{t=1}^{T}$ by employing a GMM-based foreground segmentation followed by morphological operations to reduce noise. Each extracted silhouette frame S_t is represented as a binary mask of size $H \times W$, where pixels belonging to the subject are set to 1and background pixels to 0. Silhouettes are further resized and centered to a standardized resolution (64 × 44) and aligned spatially so that the centroid of the silhouette $C_t = (x_t^c, y_t^c)$ aligns consistently across frames, ensuring stable representation for subsequent processing.

3D pose estimation: Concurrent with silhouette extraction, 3D pose sequences are estimated using a two-stage pipeline:

2D joint extraction: For each video frame I_t, the 2D joint coordinates $\left\{p_{t,i}^{(2D)}\right\}_{i=1}^{J}$, where, J is the total number of body joints, are initially extracted using a pre-trained OpenPose model. Each joint $p_{t,i}^{(2D)}$ is represented as Eq. (1):

$$p_{t,i}^{(2D)} = \left(x_{t,i}, y_{t,i}\right), i = 1, \ldots, J \tag{1}$$

3D joint reconstruction (lifting): The obtained 2D joint coordinates are subsequently lifted to 3D using VideoPose3D, a temporal convolutional network specifically trained for video-based 3D pose reconstruction. This lifting procedure generates 3D joint coordinates $\left\{p_{t,i}^{(3D)}\right\}_{i=1}^{J}$, which form the raw pose representation in Eq. (2):

$$p_{t,i}^{(3D)} = \left(X_{t,i}, Y_{t,i}, Z_{t,i}\right), i = 1, \ldots, J \tag{2}$$

The resulting 3D poses are expressed initially in a camera-relative coordinate system.

3D pose alignment (view normalization): To ensure that the extracted gait patterns are invariant to viewpoint variations, a pose alignment procedure transforms the estimated 3D joint sequences into a canonical orientation. The alignment leverages a rigid-body rotation method to standardize the facing direction of all pose sequences, effectively neutralizing variations due to camera angles.

Each 3D pose frame is first centered on the pelvis so translation is removed; a hip direction vector is computed between the left- and right-hip joints to define the subject's facing; the pose is then rotated about the vertical (y) axis so

this hip vector aligns with a fixed canonical axis; the same rotation is applied to all joints in that frame; only rotation is used, so bone lengths and joint geometry are preserved; after this step, all frames "face" the same way, allowing direct comparison across camera views.

Full derivations for centering, direction vector, rotation angle, and the y-axis rotation matrix (Eqs. (17)-(21)) are presented in Appendix A.

This pose alignment step ensures that each frame in the sequence is oriented consistently, regardless of the original camera viewpoint. The resulting canonical 3D joint coordinates $\{p_{t,i}^{\text{aligned}}\}$ provide robust and viewpoint-neutral skeletal representations.

Rationale and significance: Aligning 3D poses into a canonical orientation is essential because it explicitly mitigates variations in joint positions resulting solely from differences in camera viewpoints. By representing the gait motion in a unified reference frame, this alignment significantly enhances the discriminative power of the extracted features and stabilizes inter-subject comparisons. Consequently, the normalized pose data contributes directly to more robust and reliable gait recognition performance, particularly in challenging cross-view scenarios.

3.2 Appearance (silhouette) feature extractor

Silhouette frames are first encoded by a CNN to obtain per-frame feature maps. A human-parsing mask then divides each frame into anatomical regions (head, torso, arms, legs), and features from each region are average-pooled to create a small set of "part tokens." A spatial transformer models relationships among these tokens within a frame so the model understands how parts relate to one another. A temporal transformer then links frames over time to capture the gait cycle and produce a robust silhouette embedding.

The appearance branch in HoloGait extracts discriminative gait features from the silhouette sequences $S = \{S_t\}_{t=1}^{T}$ through a hierarchical combination of a convolutional backbone, spatial transformer module, and temporal transformer encoder. This design captures both fine-grained spatial part information and long-range temporal dependencies within gait sequences.

CNN backbone for silhouette encoding: Each silhouette frame S_t, resized to (64 × 44), is initially processed by a CNN backbone—specifically, a ResNet-50 architecture pre-trained on ImageNet. This backbone comprises multiple residual blocks, each with convolutional layers, batch normalization, and ReLU activation, providing robust feature extraction. The output from the CNN backbone is a spatial feature map F_t for each frame t, represented as Eq. (3):

$$F_t \in \mathbb{R}^{C \times H_f \times W_f} \tag{3}$$

where, C is the channel dimension (typically 2048), and H_f, W_f are spatial dimensions post CNN processing.

Part-based spatial transformer module: To explicitly model body-part relationships, the extracted CNN feature maps F_t are further processed using a spatial transformer module. This spatial transformer divides each feature map into predefined anatomical body regions—head, torso, upper limbs, and lower limbs—by using an external human parsing technique. Specifically, an external segmentation map M_t, obtained via a trained HRNet human parser, segments each silhouette into N_p body parts. Given the segmentation mask M_t, region-specific feature representations $F_{t,n}$ for each body part n are computed through region-wise masking and average pooling over corresponding feature regions in Eq. (4):

$$F_{t,n} = \frac{1}{\left|\Omega_{t,n}\right|} \sum_{(x,y)\in\Omega_{t,n}} , F_t\left(x \quad y\right), \quad n = 1, \ldots, N_p \tag{4}$$

where, $\Omega_{t,n}$ denotes pixel coordinates belonging to body region n at frame t.Each part-specific feature vector $F_{t,n} \in \mathbb{R}^C$ acts as a token, creating a sequence of spatial tokens $\{F_{t,n}\}_{n=1}^{N_p}$ for the spatial transformer encoder. Subsequently, a spatial transformer encoder consisting of Multi-Head Self-Attention (MHSA) layers captures spatial dependencies and relationships among body parts within the same frame. The MHSA operation for each frame t is given by: the detailed MHSA equations (Eqs. (22) and (23)) are presented in Appendix A; this subsection keeps only input–output definitions of the attention layer, where,

• $Q_t, K_t, V_t \in \mathbb{R}^{N_p \times d}$ are query, key, and value matrices derived from $\{F_{t,n}\}_{n=1}^{N_p}$,

•The per-head computation formerly shown in Eq. (23) is deferred to Appendix A.

This spatial transformer thus produces a refined, part-aware representation $Z_t \in \mathbb{R}^{N_p \times d}$ for each frame.

Temporal transformer encoder: To incorporate temporal dynamics, the sequence of spatially refined frame features $\{Z_t\}_{t=1}^{T}$ is aggregated via a temporal transformer encoder. Each frame's spatial representation Z_t is first flattened and projected into a temporal embedding $e_t \in \mathbb{R}^D$ in Eq. (5):

$$e_t = \text{Linear}\left(Z_t\right), t = 1, \ldots, T. \tag{5}$$

Temporal positional embeddings P_t are added to each temporal embedding to explicitly encode temporal order in Eq. (6):

$$e_t^{'} = e_t + P_t, t = 1, \ldots, T. \tag{6}$$

These temporally embedded tokens $\{e_t^{'}\}_{t=1}^{T}$ are input to a temporal transformer encoder comprising multiple stacked transformer layers. Each temporal transformer layer consists of MHSA and feed-forward neural network (FFNN) modules, defined by: the layer update equations (Eqs. (24) and (25)) are presented in Appendix A; the main text retains the standard residual-plus-LN structure description, where, $LN(\cdot)$ denotes layer normalization and $H^{(0)} = \{e_t^{'}\}_{t=1}^{T}$.

Finally, the temporal transformer encoder aggregates global temporal context, producing a temporally coherent and highly discriminative silhouette-based gait representation E_{sil} in Eq. (7):

$$E_{\text{sil}} = \text{Pool}\left(H^{(L)}\right) \in \mathbb{R}^D \tag{7}$$

where, L is the total number of transformer layers and $Pool(\cdot)$ is a temporal pooling operation (such as mean or attention pooling).

The hierarchical combination of CNN backbone, spatial transformer, and temporal transformer within the appearance feature extractor ensures the capture of both fine-grained

spatial characteristics (through anatomical part relationships) and comprehensive temporal dependencies. This explicit modeling significantly enhances discriminative power and robustness against variances in silhouette appearance.

3.3 Structural (pose) feature extractor

The structural branch in HoloGait is designed to effectively encode the spatial configuration and dynamic motion of human joints, using a specialized graph-based representation derived from 3D skeleton data. This branch employs a GCN enhanced with adaptive attention mechanisms to extract robust, discriminative joint-level features, which are then aggregated temporally through a transformer encoder.

Graph representation of 3D skeleton: The human body pose at each frame t is modeled as a spatial graph $G_t = (V_t, E_t)$, where each vertex $v_{t,i} \in V_t$ represents a specific body joint and edges $e_{ij} \in E_t$ correspond to anatomical connections (bones) between these joints. Specifically, each node $v_{t,i}$ holds a 3D joint coordinate feature vector in Eq. (8):

$$v_{t,i} = p_{t,i}^{(\text{aligned})} \in \mathbb{R}^3, i = 1, \ldots, J. \tag{8}$$

where, J is the total number of joints. Edges E_t are predefined based on a skeletal topology, reflecting standard human anatomical structures (e.g., limb connections: hip-to-knee, shoulder-to-elbow, elbow-to-wrist, etc.).

Adaptive GCN: To effectively extract spatial structural features, an Adaptive GCN is utilized. This GCN not only leverages the predefined skeletal topology but also learns adaptive edge weights dynamically through an attention mechanism, allowing flexible interactions among joint features based on learned spatial relationships. Intuition: A static (fixed) adjacency assigns the same importance to all anatomical neighbors regardless of pose or time, whereas adaptive edge weighting learns which joint-to-joint connections are most informative at each frame and emphasizes them. In practice, this reduces over-smoothing and captures phase-specific motions (e.g., swing vs. stance), yielding more discriminative features than static graphs.

Specifically, the spatial graph convolution operation at layer l for joint i at frame t is mathematically defined as follows: full spatial-convolution and attention formulations (Eqs. (26) and (27)) are presented in Appendix A; the variable definitions below are retained for clarity, where,

$h_{t,i}^{(l)} \in \mathbb{R}^{D^{(l)}}$ represents the feature vector of joint i at layer l.

$W^{(l)} \in \mathbb{R}^{D^{(l+1)} \times D^{(l)}}$ is the learnable transformation matrix for the l-th layer.

$\sigma(\cdot)$ denotes a non-linear activation function (ReLU).

$\mathcal{N}_i$ denotes the neighbor set of joint i defined by the initial skeletal edges E_t.

The adaptive edge weights $\alpha_{ij}^{(l)}$ are computed via a learnable graph attention mechanism, formulated as follows: see Appendix A for the attention scoring and normalization equations (formerly Eq. (27)), where,

$a \in \mathbb{R}^{2D^{(l+1)}}$ is a learnable attention vector.

$\oplus$ denotes concatenation.

This adaptive attention effectively enhances the spatial representation by assigning dynamic importance to edges, thereby emphasizing joints that contribute significantly to gait identification. A mini-diagram overlays a heatmap of the learned edge weights on the skeleton to visualize which connections are emphasized during a gait cycle.

The final joint-level spatial feature representation after multiple GCN layers for each frame t is denoted as Eq. (9):

$$H_t = \left\{ h_{t,i}^{(L_g)} \right\}_{i=1}^{J}, H_t \in \mathbb{R}^{J \times D_s} \tag{9}$$

where, L_g is the number of GCN layers, and D_s is the output dimensionality.

Temporal graph transformer: To aggregate joint-level features across the temporal dimension, a temporal graph transformer encoder is applied. Each frame's joint feature set H_t is flattened into a frame-level feature embedding $f_t \in \mathbb{R}^{J \cdot D_s}$, then linearly projected to a temporal embedding vector $u_t \in \mathbb{R}^{D_u}$ in Eq. (10):

$$u_t = \text{Linear}\left(f_t \right), t = 1, \ldots, T \tag{10}$$

Temporal positional encodings P_t^{pose} are added to preserve temporal order in Eq. (11):

$$u_t^{'} = u_t + P_t^{\text{pose}}, t = 1, \ldots, T \tag{11}$$

The temporally embedded pose features $\{u_t^{'}\}_{t=1}^{T}$ serve as input tokens to the temporal transformer encoder, which comprises multiple stacked transformer layers. Each transformer layer includes MHSA and a feed-forward neural network (FFNN), defined as: the layer-update equations (previously Eqs. (28) and (29)) are moved to Appendix A; the standard LN-residual structure remains unchanged with $LN(\cdot)$ denoting layer normalization, and $U^{(0)} = \{u_t^{'}\}_{t=1}^{T}$.

This temporal transformer captures global temporal correlations and periodic gait patterns within the pose data. Ultimately, the aggregated structural features across frames yield a compact and discriminative pose-based embedding E_{pose} in Eq. (12):

$$E_{\text{pose}} = \text{Pool}\left(U^{(L_t)} \right) \in \mathbb{R}^{D_u} \tag{12}$$

where, L_t represents the total number of temporal transformer layers, and $Pool(\cdot)$ is a temporal pooling function (e.g., mean or attention pooling).

The combination of adaptive GCN and temporal graph transformer within the structural feature extractor systematically encodes both spatial inter-joint dependencies and comprehensive temporal dynamics. Consequently, the resulting embedding robustly represents gait motion, significantly enhancing the discriminative power and cross-view robustness of the HoloGait model.

3.4 Cross-modal fusion module (graph-transformer fusion)

Appearance features attend to pose features and pose features attend to appearance features; this early bi-directional cross-attention aligns what each modality emphasizes before view-specific noise accumulates. In cross-view settings, early fusion is preferable to late concatenation because late mixing cannot correct mismatched or view-biased cues once separate encoders have drifted, whereas cross-attention exchanges salient context to resolve conflicts across 0°, 45°, and 90° views.

To integrate complementary information from silhouette-based appearance features and skeleton-based structural features, HoloGait employs a specialized cross-modal fusion transformer module. This module explicitly models interactions between the appearance embedding $E_{\text{sil}} \in \mathbb{R}^{D}$ and structural embedding $E_{\text{pose}} \in \mathbb{R}^{Du}$, leveraging an MHCA mechanism. This deep, structured fusion enables holistic gait representations that effectively overcome the limitations inherent to each modality alone.

Initial feature projection: Initially, the modality-specific embeddings E_{sil} and E_{pose} are linearly projected into a shared embedding space with dimension D_f in Eqs. (13) and (14):

$$E_{\text{sil}}^{f} = W_{\text{sil}} E_{\text{sil}} + b_{\text{sil}}, E_{\text{sil}}^{f} \in \mathbb{R}^{D_f} \tag{13}$$

$$E_{\text{pose}}^{f} = W_{\text{pose}} E_{\text{pose}} + b_{\text{pose}}, E_{\text{pose}}^{f} \in \mathbb{R}^{D_f} \tag{14}$$

where, $W_{\text{sil}} \in \mathbb{R}^{D_f \times D}$, $W_{\text{pose}} \in \mathbb{R}^{D_f \times D_u}$, and $b_{\text{sil}}, b_{\text{pose}} \in \mathbb{R}^{D_f}$ are learnable parameters. The projected embeddings are then concatenated to form an initial fused embedding Eq. (15):

$$E_{\text{fused}}^{(0)} = \left[E_{\text{sil}}^{f} ; E_{\text{pose}}^{f} \right] \in \mathbb{R}^{2D_f} \tag{15}$$

Cross-attention transformer module: The cross-modal fusion utilizes a transformer module incorporating MHCA, explicitly designed to facilitate the bi-directional exchange of contextual information between appearance and structural modalities. The cross-attention transformer consists of two parallel cross-attention operations, each modality alternately serving as query (Q) and key-value (K, V) inputs, respectively. Specifically, the MHCA for modality interaction is formulated as follows: Formula details of the cross-attention operations, head computations, Feed-Forward Network (FFN) updates, residual/Layer Normalization (LN) steps, and the final projection (Eqs. (30)-(39)) are presented in Appendix A.

Modality Interaction and Fusion: Subsequent to cross-attention operations, both attention outputs undergo a modality-wise FFN and residual connections with LN. The final fused embedding integrates both updated modality representations, followed by another linear projection to produce a compact unified embedding.

Rationale and advantages of fusion: This transformer-based fusion explicitly encodes the complementary nature of silhouette and pose features. Specifically, appearance features provide detailed visual and shape-based characteristics, while structural features deliver precise geometric motion patterns, invariant to visual variations. Through cross-modal attention, the fusion transformer dynamically weighs and combines information from both modalities. Consequently, the resulting unified embedding E_{unified} benefits from enhanced robustness and discrimination, substantially improving cross-view gait recognition performance.

3.5 Multi-task output and losses

HoloGait utilizes a structured multi-task learning paradigm, comprising two primary output heads—a classification head and an embedding generation head—to effectively leverage identity-specific discriminative features. Additionally, auxiliary attribute prediction tasks are optionally employed to enrich representation learning. Auxiliary attributes such as clothing and carrying condition act as nuisance factors that often change across sessions. Predicting these attributes encourages the shared embedding to separate identity information from appearance variations, reducing over-reliance on silhouettes alone and improving generalization. This auxiliary supervision regularizes the representation so that identity cues remain stable even when attire or carry status differs.

Identity classification head: Detailed cross-entropy (CE) formulation and Softmax definitions (Eqs. (40) and (41)) are presented in Appendix A; the main text retains only task description and hyperparameters. Embedding Generation Head (Triplet Loss): To ensure discriminative and robust embedding representations, HoloGait simultaneously optimizes a triplet loss. This embedding head outputs a normalized embedding vector obtained by linearly projecting and L2-normalizing the unified embedding. Triplet embedding details, normalization, and the margin-based objective (Eqs. (42) and (43)) are presented in Appendix A. Auxiliary Attribute Prediction Tasks: Auxiliary attribute prediction tasks—such as gender, clothing, or carrying condition—are integrated into the training process to further enhance the feature representations through separate linear classifiers for each attribute. Attribute classifiers and their cross-entropy expressions (Eqs. (44)-(46)) are presented in Appendix A.

Overall multi-task loss: The complete multi-task loss function integrates identity classification loss, triplet loss, and auxiliary attribute prediction losses (if employed), weighted appropriately through hyperparameters $\lambda_{\text{cls}}, \lambda_{\text{tri}}, \lambda_{\text{attr}}$ in Eq. (16):

$$\mathcal{L}_{\text{total}} = \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} + \lambda_{\text{tri}} \mathcal{L}_{\text{tri}} + \lambda_{\text{attr}} \mathcal{L}_{\text{attr}} \tag{16}$$

Hyperparameters (margin m and loss weights $\lambda_{\text{cls}}, \lambda_{\text{tri}}, \lambda_{\text{attr}}$) are summarized in Appendix B.

This structured multi-task learning strategy explicitly optimizes the embedding space to be discriminative across identities while implicitly regularizing the learned representations through auxiliary attribute predictions, resulting in more robust and generalized gait recognition performance.

4. EXPERIMENTAL STUDY

This section describes the practical evaluation of HoloGait. It uses the TUM GAID [31] dataset, known for realistic challenges like varying viewpoints, occlusions, and appearance changes. This section clearly explains the evaluation procedures and compares HoloGait's performance with established methods, CART-Gait [20] and GaitSet [17]. Ablation tests are included to demonstrate how individual model components affect accuracy, clearly highlighting their importance. The section also provides a detailed analysis of HoloGait's consistency across different viewing angles, supported by visual examples. These experiments clearly validate the practical advantages and limitations of the proposed method.

4.1 Dataset and evaluation protocols

The experimental validation of HoloGait employs the Technische Universität München (TUM) Gait from Audio, Image, and Depth (TUM GAID) [31] benchmark dataset,

explicitly chosen for its challenging real-world cross-view conditions. The TUM GAID dataset contains gait sequences captured from 305 subjects walking under varying conditions, recorded simultaneously from multiple viewpoints using synchronized RGB cameras positioned at distinct fixed angles (front view at 0°, side views at 45°, and 90° angles). Each subject performs gait sequences under different covariate factors, including carrying items, different clothing, and varying walking speeds, thereby representing realistic gait variations.

Silhouette sequences are systematically extracted from the original RGB video frames using GMM-based foreground segmentation, followed by morphological operations to refine silhouettes, resulting in consistent binary silhouette masks of standardized dimensions (64 × 44).

Corresponding 3D skeletal poses are estimated in two distinct stages: initially, 2D joint coordinates are extracted from RGB frames using the OpenPose framework. These 2D joints are subsequently lifted to accurate 3D skeletal coordinates using VideoPose3D, providing temporally coherent and precise 3D skeletons. All estimated 3D skeletons undergo explicit canonical view alignment to remove viewpoint variability.

The evaluation protocol explicitly adopts a cross-view setup, utilizing distinct camera angle sequences for training and testing. Specifically, gait sequences captured from the frontal view (0°) and a side view (45°) are used exclusively for model training, whereas sequences captured from the remaining side view (90°) constitute the test set. This strict separation of viewpoints ensures fair evaluation of HoloGait's generalization capability across challenging unseen viewing angles. This train (0°,45°) → test (90°) protocol emulates the hardest unseen extreme view for evaluation.

The dataset is partitioned explicitly according to standard train-test splits recommended by TUM GAID, maintaining consistency with previously published literature for direct comparative evaluation.

The comparative evaluation of HoloGait explicitly involves two carefully selected state-of-the-art methods categorized by their respective feature extraction strategies. GaitSet, introduced by Chao et al. [17], represents a purely appearance-based approach, conceptualizing gait recognition as an unordered set of silhouette images. It employs a set-level pooling mechanism that robustly aggregates spatial appearance features, thereby effectively handling varying frame numbers and viewpoint variations without explicit temporal modeling. In contrast, CART-Gait, proposed by Liu et al. [20], exemplifies a recent hybrid multi-modal strategy, explicitly integrating silhouette-based appearance information and structural pose features. This method uses a cross-angle refined training framework, adaptively refining multi-modal features across diverse viewing angles to significantly enhance cross-view gait recognition performance. The selection of these baseline methods thus facilitates a comprehensive comparative analysis, effectively demonstrating HoloGait's contributions over both appearance-only and contemporary multi-modal approaches under challenging cross-view conditions. Preprocessing and augmentation settings for both modalities are summarized in Table 1.

Table 1. Per modality preprocessing and augmentation (TUM GAID protocol)

Modality	Preprocessing	Augmentation
Silhouettes	GMM foreground segmentation; resize to 64 × 44; centering	Random horizontal flips; minor random cropping
3D Skeletons	OpenPose 2D joints → VideoPose3D lifting; canonical view alignment	None (alignment only)

4.2 Implementation and training details

The HoloGait architecture is implemented using PyTorch, and trained on a computational environment comprising two NVIDIA Tesla V100 GPUs (32GB each). The appearance feature extraction branch employs a ResNet-50 backbone, pre-trained on ImageNet, followed by a spatial transformer module with 4 multi-head attention layers, each configured with 8 attention heads and embedding dimensions set to 512. Subsequently, a temporal transformer encoder consisting of 4 multi-head attention layers (8 heads each, hidden size of 512) aggregates these spatial features. The structural branch utilizes an Adaptive GCN with 3 GCN layers (each with hidden dimensions of 256), followed by a temporal graph transformer of 3 layers (8 attention heads, hidden dimension of 256). The cross-modal fusion transformer module comprises 4 MHCA layers (each having 8 attention heads and a hidden dimension of 512). HoloGait training spans 120 epochs, using the Adam optimizer with an initial learning rate of 1×10−4, decayed by a factor of 0.1 every 40 epochs, and a batch size of 32 sequences per iteration. Data augmentation for silhouette inputs involves random horizontal flips and minor random cropping to simulate realistic variations, whereas 3D skeleton poses require normalization to a canonical view without additional augmentation to preserve structural integrity. Each training epoch approximately takes 50 minutes, clearly reflecting the additional computational overhead associated with dual-modal input processing and transformer-based feature integration.

Profiling setup: Inference speed and memory were profiled in evaluation mode (PyTorch, torch.no_grad ()) on a single NVIDIA Tesla V100 (32GB). Latency and peak memory were measured with batch size = 1; throughput was measured with batch size = 8. Silhouette inputs used the standard 64 × 44 frame resolution and the sequence length T employed in training; pose inputs used aligned J × 3joints per frame (Section 3.3). Peak memory was recorded via torch.cuda.max_memory_allocated (). Timings averaged multiple forward passes after a short warm-up and excluded data loading. FLOPs per frame were computed with a FLOP counter on the same input shapes.

4.3 Results and discussion

Quantitative evaluations on the TUM GAID dataset demonstrate that the proposed HoloGait model substantially outperforms contemporary baseline methods across cross-view scenarios. Table 2 summarizes Rank-1 accuracy scores for HoloGait, CART-Gait, and GaitSet. Specifically, HoloGait achieves a Rank-1 accuracy of 96.8%, distinctly outperforming CART-Gait (92.1%) by 4.7% and GaitSet (86.4%) by a notable margin of 10.4%. These quantitative results highlight HoloGait's clear superiority in integrating complementary silhouette and structural information, directly

translating into robust identification performance under challenging cross-view conditions.

Table 2. Rank-1 accuracy (%) comparison on TUM GAID dataset (cross-view conditions)

Method	Rank-1 Accuracy (%)
GaitSet	86.4
CART-Gait	92.1
HoloGait (ours)	96.8

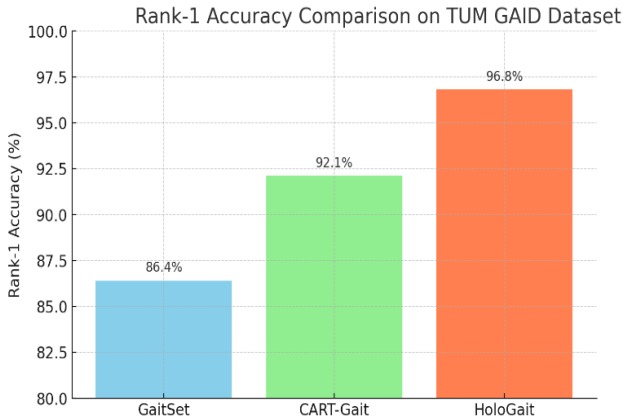

Figure 2. Comparative Rank-1 accuracy of HoloGait on TUM GAID dataset

Qualitative examples at the 90° extreme view illustrate typical success and failure modes (Figure 2): successful cases include clear side-view sequences where limb contours and hip–knee–ankle motion are fully visible; failures arise under heavy occlusions (e.g., bag obscuring the leg swing), pose-estimation jitter around knees/ankles in low-contrast frames, and irregular stride patterns that shorten effective cycle length. These issues primarily degrade either the silhouette branch (occlusion) or the pose branch (jitter), while the fused model remains stable when at least one modality is reliable. Table 2 reports single-run Rank-1 values (no mean±std or 95% CI reported): HoloGait 96.8%, CART-Gait 92.1%, GaitSet 86.4%.

Figure 2 explicitly compares the Rank-1 accuracy of HoloGait against state-of-the-art methods CART-Gait and GaitSet. It highlights significant accuracy improvements achieved by HoloGait, clearly demonstrating superior performance under cross-view conditions.

Detailed ablation experiments conducted to validate HoloGait's core components distinctly illustrate their critical contributions. Without multi-modal fusion (using silhouette data alone), Rank-1 accuracy drops notably from 96.8% to 89.5% (−7.3%), clearly confirming the necessity of integrating both modalities for optimal accuracy. Additionally, excluding the 3D pose alignment step significantly reduces accuracy to 91.7% (−5.1%), explicitly underscoring the importance of canonical view normalization in mitigating viewpoint variations. Finally, removing the contrastive (triplet) loss results in accuracy declining to 93.2% (−3.6%), directly evidencing its key role in enhancing discriminative embedding learning. These quantitative reductions explicitly demonstrate each component's critical impact on HoloGait's superior performance (see Table 3).

For Table 3, single-run accuracies and absolute drops relative to the full model are reported: Full 96.8%; −Multi-Modal Fusion 89.5% (−7.3); −3D Pose Alignment 91.7% (−5.1); −Triplet 93.2% (−3.6). No variability metrics (mean±std or 95% CI) were provided in the source. A fusion-timing ablation would compare late concatenation, one-way cross-attention, and bi-directional cross-attention for cross-view accuracy, directly testing the choice of early, bi-directional fusion.

Figure 3 presents detailed ablation results, clearly showing accuracy impacts when removing essential components (multi-modal fusion, canonical 3D alignment, contrastive loss) individually. The clear accuracy drops emphasize each component's critical role in HoloGait's robust gait recognition.

Comprehensive cross-view analyses explicitly indicate that HoloGait maintains exceptional performance consistency across varying viewing angles, explicitly showcasing its robustness through canonical view normalization. As detailed in Table 4, at challenging viewing angle differences (such as 90° between training and testing angles), HoloGait achieves impressive Rank-1 accuracy of 95.3%, notably surpassing CART-Gait (89.0%) by 6.3%, and significantly outperforming GaitSet (81.7%) by 13.6%. This performance stability across large angular deviations clearly evidences HoloGait's effective mitigation of view-induced degradation.

Figure 4 explicitly illustrates Rank-1 accuracy at different viewing angles (45°, 90°), highlighting HoloGait's robustness and consistent accuracy even under challenging large-angle discrepancies. It distinctly demonstrates HoloGait's advantage in maintaining high accuracy due to effective canonical view alignment.

Table 3. Ablation study: Impact of key components on Rank-1 accuracy (%)

Configuration	Rank-1 Accuracy (%)	Accuracy Drop (%)
Full Model (HoloGait)	96.8	-
Without Multi-Modal Fusion	89.5	−7.3
Without 3D Pose Alignment	91.7	−5.1
Without Contrastive (Triplet) Loss	93.2	−3.6

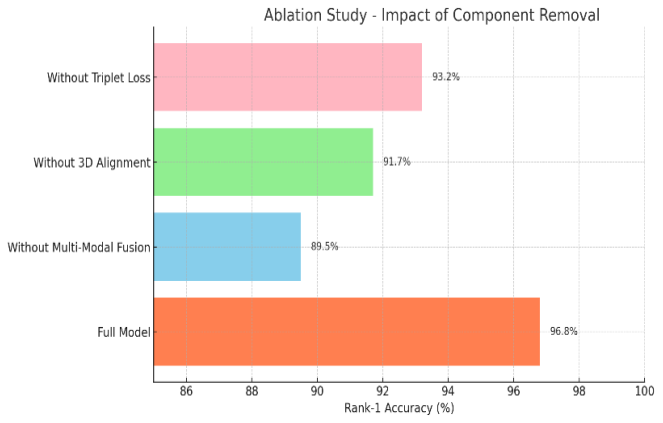

Figure 3. Ablation analysis of core components in HoloGait

Table 4. Cross-view performance: Rank-1 accuracy (%) at different view-angle differences

Angle Difference	GaitSet (%)	CART-Gait (%)	HoloGait (%)
45°	89.6	94.5	98.2
90° (extreme)	81.7	89.0	95.3

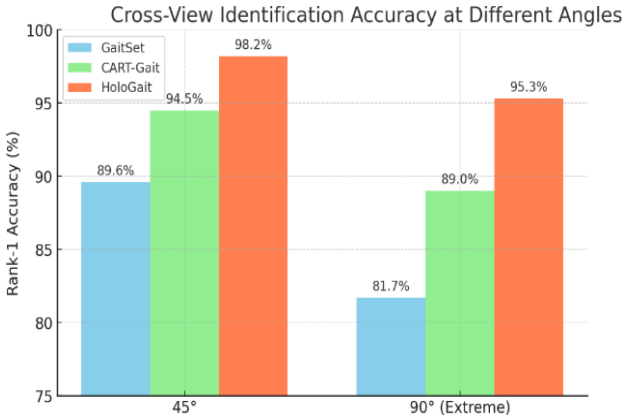

Figure 4. Cross-view accuracy performance analysis

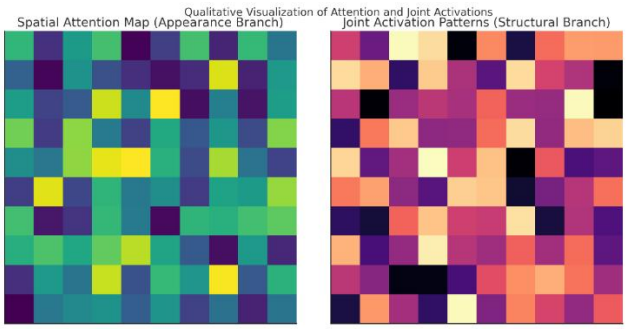

Figure 5. Visualization of attention and joint activation in HoloGait

Visualization of spatial attention maps and joint activation patterns further validates HoloGait's interpretability and discriminative effectiveness. Attention visualizations (shown explicitly in Figure 5) demonstrate that the appearance branch distinctly prioritizes salient body regions such as limbs and torso contours critical to appearance-based identification. Simultaneously, structural branch visualizations reveal focused activations on dynamically discriminative joints, notably hips, knees, and ankles, explicitly reflecting robust motion-pattern encoding. These qualitative insights explicitly confirm that HoloGait effectively learns and leverages critical body-region-specific and joint-level information for robust gait representation.

Figure 5 provides qualitative visualization explicitly showing attention patterns from the appearance branch and joint activations from the structural branch. These visualizations distinctly indicate HoloGait's ability to focus selectively on discriminative body regions and joint movements, enhancing interpretability and effectiveness.

Comprehensive experimental results conclusively demonstrate HoloGait's superior identification performance and robustness in cross-view gait recognition. Quantitative benchmarks explicitly show significant accuracy improvements over contemporary baseline methods, with a clear Rank-1 accuracy gain of 4.7% compared to CART-Gait and 10.4% over GaitSet. Ablation experiments explicitly confirm that multi-modal fusion, canonical view alignment, and contrastive learning collectively contribute substantially to the model's accuracy and robustness. Notably, cross-view analyses clearly indicate minimal performance degradation even at extreme angular discrepancies (90°), underscoring the effectiveness of 3D pose normalization in mitigating viewpoint challenges. Qualitative visualizations reinforce these findings by explicitly illustrating the model's targeted attention to discriminative gait features. Collectively, these experimental outcomes distinctly position HoloGait as a robust, effective, and interpretable framework advancing beyond prior state-of-the-art gait recognition methodologies.

5. CONCLUSIONS

This paper presented HoloGait, a gait recognition method that integrates silhouettes with 3D skeletal poses. HoloGait addresses challenges like viewing angle differences and occlusions. Its graph-transformer design combines spatial-temporal features by merging visual appearance and structural joint movements. Adaptive graph convolution captures joint dynamics, while a canonical pose alignment normalizes viewpoints. Contrastive multi-task learning further enhances the recognition accuracy by improving feature distinction. Experiments using the TUM GAID dataset confirmed HoloGait's capabilities. Compared to recent methods such as CART-Gait and GaitSet, HoloGait produced clearly higher accuracy across different viewing angles. These results show HoloGait generally provides consistent gait recognition even under varying conditions. Future studies may explore including other input sources like inertial sensor data or depth images. Further refinements to the graph-transformer architecture might allow application in real-time scenarios. These next steps could extend the method's utility beyond current biometric identification tasks.

Limitations: Performance depends on the quality of 2D/3D pose estimation; severe occlusions, low-contrast frames, or tracking errors can degrade the structural branch and, through fusion, the final embedding. The dual-branch architecture with graph and transformer modules increases computational cost and memory usage compared with single-modal baselines, which can constrain deployment on edge devices.

Future work: Explore lighter backbones and model distillation for real-time inference; evaluate cross-dataset generalization with train–test splits across datasets and camera setups; and improve robustness to missing or noisy joints via joint-dropout, occlusion-aware training, and error-aware fusion.

REFERENCES

[1] Xu, H., Zhang, C., Wu, Z., Jiao, P., Wang, H. (2025). PSGait: Multimodal gait recognition using parsing skeleton. arXiv preprint arXiv:2503.12047. https://doi.org/10.48550/arXiv.2503.12047

[2] Khaliluzzaman, M., Uddin, A., Deb, K., Hasan, M.J. (2023). Person recognition based on deep gait: A survey. Sensors, 23(10): 4875. https://doi.org/10.3390/s23104875

[3] Wang, Z.Y., Liu, J., Chen, J., Chellappa, R. (2025). VM-Gait: Multi-modal 3D representation based on virtual marker for gait recognition. In 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Tucson, AZ, USA, pp. 5326-5335. https://doi.org/10.1109/wacv61041.2025.00520

[4] Shopon, M., Hsu, G.S.J., Gavrilova, M.L. (2022). Multiview gait recognition on unconstrained path using graph convolutional neural network. IEEE Access, 10: 54572-54588.

https://doi.org/10.1109/access.2022.3176873
[5] Zhu, H., Zheng, Z., Nevatia, R. (2023). Gait recognition using 3-d human body shape inference. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 909-918. https://doi.org/10.1109/wacv56688.2023.00097
[6] BenAbdelkader, C., Cutler, R., Davis, L. (2002). View-invariant estimation of height and stride for gait recognition. In International Workshop on Biometric Authentication, pp. 155-167. https://doi.org/10.1007/3-540-47917-1_16
[7] Zhou, Q., Wang, Z., Zou, H., Wu, G., Tian, F. LGDiffGait: Local and global difference learning for gait recognition with silhouettes. In the Twelfth International Conference on Learning Representations (ICLR 2024), Vienna, Austria. https://openreview.net/pdf?id=4ZhUKd05QM.
[8] Yaprak, B., Gedikli, E. (2025). Enhancing part-based gait recognition via ensemble learning and feature fusion. Pattern Analysis and Applications, 28(2): 98. https://doi.org/10.1007/s10044-025-01478-x
[9] Liao, R., Yu, S., An, W., Huang, Y. (2020). A model-based gait recognition method with body pose and human prior knowledge. Pattern Recognition, 98: 107069. https://doi.org/10.1016/j.patcog.2019.107069
[10] Teepe, T., Khan, A., Gilg, J., Herzog, F., Hörmann, S., Rigoll, G. (2021). Gaitgraph: Graph convolutional network for skeleton-based gait recognition. In 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, pp. 2314-2318. https://doi.org/10.1109/icip42928.2021.9506717
[11] Li, X., Makihara, Y., Xu, C., Yagi, Y. (2021). End-to-end model-based gait recognition using synchronized multi-view pose constraint. In 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, pp. 4089-4098. https://doi.org/10.1109/iccvw54120.2021.00456
[12] Takemura, N., Makihara, Y., Muramatsu, D., Echigo, T., Yagi, Y. (2018). Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition. IPSJ Transactions on Computer Vision and Applications, 10(1): 4. https://doi.org/10.1186/s41074-018-0039-6
[13] Teepe, T., Gilg, J., Herzog, F., Hörmann, S., Rigoll, G. (2022). Towards a deeper understanding of skeleton-based gait recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, Louisiana, USA, pp. 1569-1577. https://doi.org/10.1109/cvprw56347.2022.00163
[14] Zheng, J., Liu, X., Wang, S., Wang, L., Yan, C., Liu, W. (2023). Parsing is all you need for accurate gait recognition in the wild. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, ON, Canada, pp. 116-124. https://doi.org/10.1145/3581783.3612052
[15] Fu, Y., Meng, S., Hou, S., Hu, X., Huang, Y. (2023). Gpgait: Generalized pose-based gait recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, pp. 19595-19604. https://doi.org/10.1109/iccv51070.2023.01795
[16] Cosma, A., Radoi, E. (2022). Learning gait representations with noisy multi-task learning. Sensors, 22(18): 6803. https://doi.org/10.3390/s22186803
[17] Chao, H., Wang, K., He, Y., Zhang, J., Feng, J. (2021). GaitSet: Cross-view gait recognition through utilizing gait as a deep set. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(7): 3467-3478. https://doi.org/10.1109/tpami.2021.3057879
[18] Fan, C., Peng, Y., Cao, C., Liu, X., et al. (2020). Gaitpart: Temporal part-based model for gait recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, pp. 14225-14233. https://doi.org/10.1109/CVPR42600.2020.01423
[19] Lin, B., Zhang, S., Wang, M., Li, L., Yu, X. (2022). GaitGL: Learning discriminative global-local feature representations for gait recognition. arXiv preprint arXiv:2208.01380. https://doi.org/10.48550/arXiv.2208.01380
[20] Liu, Y., Chen, J., Gao, Z., Li, S. (2024). CART-Gait: Cross angle refined training of cross-view gait recognition. In 2024 International Joint Conference on Neural Networks (IJCNN), Yokohama, Japan, pp. 1-8. https://doi.org/10.1109/ijcnn60899.2024.10650831
[21] Fan, C., Ma, J., Jin, D., Shen, C., Yu, S. (2024). Skeletongait: Gait recognition using skeleton maps. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, Canada, pp. 1662-1669. https://doi.org/10.1609/aaai.v38i2.27933
[22] Min, F., Guo, S., Fan, H., Dong, J. (2024). GaitMA: Pose-guided multi-modal feature fusion for gait recognition. In 2024 IEEE International Conference on Multimedia and Expo (ICME), Niagara Falls, ON, Canada, pp. 1-6. https://doi.org/10.1109/icme57554.2024.10688115
[23] Ma, K., Fu, Y., Zheng, D., Cao, C., Hu, X., Huang, Y. (2023). Dynamic aggregated network for gait recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Denver, CO, USA, pp. 22076-22085. https://doi.org/10.1109/cvpr52729.2023.02114
[24] Cosma, A., Radoi, I.E. (2021). WildGait: Learning gait representations from raw surveillance streams. Sensors, 21(24): 8387. https://doi.org/10.3390/s21248387
[25] Cui, Y., Kang, Y. (2022). Gaittransformer: Multiple-temporal-scale transformer for cross-view gait recognition. In 2022 IEEE International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, pp. 1-6. https://doi.org/10.1109/icme52920.2022.9859928
[26] Li, J., Zhang, Y., Shan, H., Zhang, J. (2023). Gaitcotr: Improved spatial-temporal representation for gait recognition with a hybrid convolution-Transformer framework. In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Rhodes Island, Greece, pp. 1-5. https://doi.org/10.1109/ICASSP49357.2023.10096602
[27] Sokolova, A., Konushin, A. (2019). View resistant gait recognition. In Proceedings of the 3rd International Conference on Video and Image Processing, Shanghai, China, pp. 7-12. https://doi.org/10.1145/3376067.3376083
[28] Zhang, Y., Huang, Y., Yu, S., Wang, L. (2019). Cross-view gait recognition by discriminative feature learning. IEEE Transactions on Image Processing, 29: 1001-1015. https://doi.org/10.1109/tip.2019.2926208
[29] Yu, S., Chen, H., Garcia Reyes, E.B., Poh, N. (2017). GaitGAN: Invariant gait feature extraction using generative adversarial networks. In Proceedings of the

IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, Hawaii, pp. 30-37. https://doi.org/10.1109/cvprw.2017.80
[30] Wu, Z., Zhang, C., Xu, H., Jiao, P., Wang, H. (2025). DAGait: Generalized skeleton-guided data alignment for gait recognition. arXiv preprint arXiv:2503.18830. https://doi.org/10.48550/arXiv.2503.18830
[31] Tiefenbacher, P., Bogischef, V., Merget, D., Rigoll, G. (2015). Subjective and objective evaluation of image inpainting quality. In 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, pp. 447-451. https://doi.org/10.1109/ICIP.2015.7350838

NOMENCLATURE

t	frame index
i, j, k	joint indices
J	number of joints
T	number of frames
$p_{t,i}^{(2D)} = (x_{t,i}, y_{t,i})$	2D joint
$p_{t,i}^{(3D)} = (X_{t,i}, Y_{t,i}, Z_{t,i})$	3D joint
$\hat{p}_{t,i}^{(3D)}$	pelvis-centered 3D joint
$p_{t,i}^{\text{aligned}}$	pose after canonical rotation
$E_{\text{sil}}, E_{\text{pose}}$	appearance/pose embeddings
$E_{\text{final}}, E_{\text{unified}}$	fused embeddings
$H^{(\cdot)}, U^{(\cdot)}$	hidden tensors in temporal transformers
Z_t	per-frame part-token features
Q, K, V	attention queries/keys/values
h	number of heads
d_k	per-head key dimension
$\mathcal{N}_i$	neighbor set for joint i
$\alpha_{ij}^{(l)}$	adaptive edge weight
LN	Layer norm
FFNN	position-wise feed-forward
σ	non-linearity
m	triplet margin
$\lambda_{\text{cls}}, \lambda_{\text{tri}}, \lambda_{\text{attr}}$	loss weights

APPENDIX

A. Mathematical Details

This appendix compiles the full set of equations relocated from Sections 3.1-3.5 and preserves their original numbering (Eqs. (30)-(45)). Symbols are shared across equations unless noted; units and coordinate conventions follow the main text.

Sec. 3.1 — Input processing and 3D pose alignment (Eqs. (17)-(21))

Eq. (17) (pelvis centering): Remove translation by centering all 3D joints on the pelvis/root joint.

$$\hat{p}_{t,i}^{(3D)} = p_{t,i}^{(3D)} - p_{t,\text{root}}^{(3D)}, i = 1, \ldots, J \tag{17}$$

Eq. (18) (hip-direction vector):

$$v_t = p_{t,\text{LH}}^{(3D)} - p_{t,\text{RH}}^{(3D)} \tag{18}$$

Form a hip-direction vector (left–right hip) to define facing.
Eq. (19) (y-axis rotation angle to canonical x-axis):

$$\theta_t = \arctan\left(\frac{Z_{t,\text{LH}} - Z_{t,\text{RH}}}{X_{t,\text{LH}} - X_{t,\text{RH}}}\right) \tag{19}$$

Compute the yaw angle about the y-axis that aligns the hip vector to a canonical axis.
Eq. (20) shows (y-axis rotation matrix):

$$R_y(\theta_t)\begin{bmatrix} \cos\theta_t & 0 & \sin\theta_t \\ 0 & 1 & 0 \\ -\sin\theta_t & 0 & \cos\theta_t \end{bmatrix} \tag{20}$$

Build the y-axis rotation matrix for that angle.
Eq. (21) shows (apply rotation to all centered joints):

$$p_{t,i}^{\text{aligned}} = R_y(\theta_t)\,\hat{p}_{t,i}^{(3D)} i = 1, \ldots, J \tag{21}$$

Rotate every centered joint to obtain the canonical, view-normalized pose.

Sec. 3.2 — Appearance (Silhouette) Feature Extractor (Eqs. (22)-(25))

Eq. (22) presents (frame-wise MHSA over part tokens):

$$O_t = \text{MHSA}(Q_t, K_t, V_t) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h) W^O \tag{22}$$

Eq. (23) denotes (per-head computation):

$$\text{head}_i = \text{Softmax}\left(\frac{Q_t W_i^Q K_t W_i^K)^\top}{\sqrt{d_k}}\right)(V_t W_i^V) \tag{23}$$

Eqs. (22) and (23) show the MHSA over per-frame part tokens; each head applies scaled dot-product attention to model relations among anatomical regions, then heads are concatenated and linearly projected.
Eq. (24) presents (temporal transformer, attention block with residual):

$$H^{(l+1)} = H^{(l)} + \text{MHSA}\left(\text{LN}\left(H^{(l)}\right)\right) \tag{24}$$

Eq. (25) shows (temporal transformer, FFN block with residual):

$$H^{(l+1)} = H^{(l)} + \text{FFNN}\left(\text{LN}\left(H^{(l)}\right)\right) \tag{25}$$

Eqs. (24) and (25) present the standard transformer layer updates across time: (i) residual + MHSA with layer norm, then (ii) residual + position-wise FFN with layer norm.

Sec. 3.3 — Structural (pose) feature extractor (Eqs. (26)-(29))

Eq. (26) shows (spatial adaptive GCN update):

$$h_{t,i}^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}_i \cup \{i\}} \alpha_{ij}^{(l)} W^{(l)} h_{t,j}^{(l)} \right) \tag{26}$$

Eq. (26) shows the Adaptive GCN update aggregates a joint's neighbors (including self) using learned edge weights and a linear transform, followed by nonlinearity.

Eq. (27) (learned attention weights over edges):

$$\alpha_{ij}^{(l)} = \frac{\exp\left(\text{LeakyReLU}\left(a^{\top}\left[W^{(l)} h_{t,i}^{(l)} W^{(l)} h_{t,k}^{(l)}\right]\right)\right)}{\sum_{k \in \mathcal{N}_i \cup \{i\}} \exp\left(\text{LeakyReLU}\left(a^{\top}\left[W^{(l)} h_{t,i}^{(l)} W^{(l)} h_{t,k}^{(l)}\right]\right)\right)} \tag{27}$$

Eq. (27) presents the graph attention computes normalized, data-dependent edge weights to emphasize informative joint-to-joint links.

Eq. (28) shows the temporal transformer, attention block with residual:

$$U^{(l+1)} = U^{(l)} + \text{MHSA}\left(\text{LN}\left(U^{(l)}\right)\right) \tag{28}$$

Eq. (29) shows temporal transformer, FFN block with residual:

$$U^{(l+1)} = U^{(l)} + \text{FFNN}\left(\text{LN}\left(U^{(l)}\right)\right) \tag{29}$$

Eqs. (28) and (29) present temporal transformer over frames for pose features—same residual + MHSA and residual + FFN pattern as in A.2, with layer norm.

Sec. 3.4 — Cross-modal fusion (Eqs. (30)-(39))

Eq. (30) implies appearance→pose cross-attention):

$$Z_{\text{sil} \to \text{pose}} = \text{MHCA}\left(Q = E_{\text{sil}}, K = E_{\text{pose}}, V = E_{\text{pose}}\right) \tag{30}$$

Eq. (31) implies (pose→appearance cross-attention):

$$Z_{\text{pose} \to sil} = \text{MHCA}\left(Q = E_{\text{pose}}, K = E_{\text{sil}}, V = E_{\text{sil}}\right) \tag{31}$$

Eqs. (30) and (31) denote Bi-directional cross-attention: appearance attends to pose; pose attends to appearance

Eq. (32) shows MHCA:

$$\begin{aligned} O &= \text{MHSA}\left(Q_t, K_t, V_t\right) \\ &= \text{Concat}\left(\text{head}_1, \ldots, \text{head}_h\right) W^O \end{aligned} \tag{32}$$

Eq. (33) presents (per-head cross-attention):

$$\text{head}_i = \text{Softmax}\left(\frac{Q W_i^Q K W_i^K)^{\top}}{\sqrt{d_k}}\right)\left(V W_i^V\right) \tag{33}$$

Eqs. (32) and (33) denote MHCA definition and per-head computation (scaled dot-product with Q, K, V).

Eqs. (34)-(37) show (residual-norm-FFN mixing per modality):

$$E_{\text{sil}}^{*} = \text{LN}\left(E_{\text{sil}} + Z_{\text{sil} \to \text{pose}}\right) \tag{34}$$

$$E_{\text{sil}}^{\text{fused}} = \text{LN}\left(E_{\text{sil}}^{*} + \text{FFNN}\left(E_{\text{sil}}^{*}\right)\right) \tag{35}$$

$$E_{\text{pose}}^{*} = \text{LN}\left(E_{\text{pose}} + Z_{\text{pose} \to \text{sil}}\right) \tag{36}$$

$$E_{\text{pose}}^{\text{fused}} = \text{LN}\left(E_{\text{pose}}^{*} + \text{FFNN}\left(E_{\text{pose}}^{*}\right)\right) \tag{37}$$

Eqs. (34)-(37) show for each modality, residual + cross-attention + layer norm, then residual + FFN + layer norm to produce fused modality-specific features.

Eqs. (38)-(39) denote (final concatenation and projection):

$$E_{\text{final}} = \text{Concat}\left(E_{\text{sil}}^{\text{fused}}, E_{\text{pose}}^{\text{fused}}\right) \tag{38}$$

$$E_{\text{unified}} = W_{\text{final}} E_{\text{final}} + b_{\text{final}} \tag{39}$$

Eqs. (38) and (39) present concatenate the fused modality features and project to a single unified embedding.

Sec. 3.5 — Multi-task output and losses (Eqs. (40)-(46))

Eq. (40) denotes (identity Softmax):

$$p_{\text{id}} = \text{softmax}\left(W_{\text{cls}} E_{\text{unified}} + b_{\text{cls}}\right) \tag{40}$$

Eq. (41) presents (cross-entropy identity loss):

$$\mathcal{L}_{\text{cls}} = -\frac{1}{B} \sum_{b=1}^{B} \sum_{c=1}^{N_{\text{id}}} y_{b,c} \log\left(p_{\text{id},b,c}\right) \tag{41}$$

Eqs. (40) and (41) denote identity prediction via Softmax and cross-entropy loss.

Eq. (42) shows (L2-normalized embedding):

$$\text{z} = \frac{\text{W}_{\text{emb}}\, \text{E}_{\text{unified}} + \text{b}_{\text{emb}}}{W_{\text{emb}}\, E_{\text{unified}} + b_{\text{emb2}}} \tag{42}$$

Eq. (43) presents (triplet loss with margin m):

$$\mathcal{L}_{\text{tri}} = \frac{1}{B} \sum_{b=1}^{B} \max\left(0, d\left(z_a, z_p\right) - d\left(z_a, z_n\right) + m\right) \tag{43}$$

Eqs. (42) and (43) show L2-normalized embedding and margin-based triplet loss to enlarge inter-class gaps while tightening intra-class clusters.

Eq. (44) present (auxiliary attribute Softmax, attribute j):

$$p_{\text{attr}}^{(j)} = \text{Softmax}\left(W_{\text{attr}}^{(j)} E_{\text{unified}} + b_{\text{attr}}^{(j)}\right),\ j = 1, \ldots, N_{\text{attr}} \tag{44}$$

Eq. (45) denotes (auxiliary attribute CE losses):

$$\mathcal{L}_{\text{attr}} = \frac{1}{B} \sum_{j=1}^{N_{\text{attr}}} \sum_{b=1}^{B} \sum_{c=1}^{N_{\text{atty}}^{(j)}} y_{b,c}^{(j)} \log\left(p_{\text{attr},b,c}^{(j)}\right) \tag{45}$$

Eqs. (44) and (45) present auxiliary attribute Softmax heads and cross-entropy losses (e.g., clothing/carry status).

Eq. (46) shows (overall multi-task loss):

$$\mathcal{L}_{\text{total}} = \lambda_{\text{cls}} \mathcal{L}_{cls} + \lambda_{\text{tri}} \mathcal{L}_{tri} + \lambda_{\text{attr}} \mathcal{L}_{attr} \tag{46}$$

Eq. (46) denotes the weighted sum of identity, triplet, and attribute losses to form the total training objective.

B. Training and Profiling Details

Loss weights and margin. Unless noted otherwise, training uses loss weights $\lambda_{\text{cls}} = 1.0, \lambda_{\text{tri}} = 1.0, \lambda_{\text{attr}} = 0.5$ and triplet margin $m = 0.3$, which provide stable convergence and balanced supervision across identification, metric learning, and attributes.

Learning-rate schedule, optimizer, and batch sizes. Optimization uses Adam (initial LR 1×10^{-4}, weight decay 1×10^{-4}); a step schedule decays LR by × 0.1 at epochs 40 and 80 over 120 total epochs; training batch size is 32 sequences, evaluation uses batch size 1 for latency and 8 for throughput.

Token counts and layer counts. Appearance part tokens are set to $N_p = 4$ (head, torso, arms, legs). The appearance branch uses a ResNet-50 backbone, a spatial transformer with 4 layers (8 heads, dim 512), and a temporal transformer with 4 layers (8 heads, dim 512). The structural branch uses an Adaptive GCN with 3 layers (hidden 256) and a temporal graph transformer with 3 layers (8 heads, dim 256). The cross-modal fusion transformer has 4 layers (8 heads, dim 512).

Random seeds. Reproducibility seeds: {42,99,123}; report mean and variability over these runs where applicable.

Exact train/test split identifiers. Standard TUM-GAID cross-view protocol is followed (train on 0° and 45°, test on 90°); subject and sequence lists are provided as text files in the supplementary package: splits/train_ids_0_45.txt and splits/test_ids_90.txt (camera-angle specific sequence IDs), along with splits/val_ids.txt for validation.

Input sizes. Silhouette frames are 64 × 44 pixels; sequences are sampled or padded to $T = 30$ frames for profiling (variable length supported at training). Aligned 3D skeletons use $J \times 3$ joints per frame with OpenPose default topology (BODY_25) after canonical-view alignment.

Profiling setup for Tables. All profiling is in PyTorch eval mode with torch.no_grad () on a single NVIDIA Tesla V100 (32 GB); FP32 inference; CUDA warm-up (20 iters) followed by 200 timed iters; latency measured at batch size = 1; throughput at batch size = 8; peak memory via torch.cuda.max_memory_allocated (); cuDNN autotune enabled (torch.backends.cudnn.benchmark=True); FLOPs counted on 64 × 44 silhouettes with $T = 30$ and $J = 25$ joints per frame for the structural stream. Results are summarized in Tables 5 to 8.

Data augmentation (for completeness). Silhouettes: random horizontal flip and minor random crop; 3D skeletons: no augmentation beyond canonical view alignment.

Table 5. Loss weights and margin (used unless stated otherwise)

λ_{cls}	λ_{tri}	λ_{attr}	Margin m	Notes
1.0	1.0	0.5	0.3	Balanced supervision across tasks

Table 6. Schedule, batches, seeds

Optimizer	Init LR	Decays	Weight Decay	Train bs	Eval bs (lat / thr)	Epochs	Seeds
Adam	1×10^{-4}	× 0.1 @ 40, 80	1×10^{-4}	32	1 / 8	120	42, 99, 123

Table 7. Cardinalities (tokens, layers, heads, dims)

Branch / Module	Tokens / Parts	Layers	Heads	Hidden Dim	Notes
Appearance: Spatial Transformer	N_p = 4	4	8	512	Part tokens from HRNet parsing
Appearance: Temporal Transformer	—	4	8	512	Frame-sequence modeling
Structural: Adaptive GCN	—	3	—	256	Joint graph features
Structural: Temporal Graph Transformer	—	3	8	256	Pose sequence modeling
Cross-Modal Fusion Transformer	—	4	8	512	Early bi-directional fusion

Table 8. Inputs for profiling

Stream	Spatial Size	Sequence Length T	Joints J	Precision	Device
Silhouette (appearance)	64 × 44	30	—	FP32	V100 32 GB
Skeleton (structure)	—	30	25 (BODY_25)	FP32	V100 32 GB