PuppetGait: Generalizing Gait Recognition via 3D Body-Aligned LVM Features

Chengwei Ye[1], **Dingqiang Ye**[2], **Chuanfu Shen**[3], **Zirui Zhou**[1], **Chao Fan**[4], **Shuaijun Jiang**[1], **Vishal M. Patel**[2], **Shiqi Yu**[1,†]

[1] Department of Computer Science and Engineering, Southern University of Science and Technology
[2] Department of Electrical and Computer Engineering, Johns Hopkins University
[3] Department of Data and Systems Engineering, The University of Hong Kong
[4] School of Artificial Intelligence, Shenzhen University

12531329@mail.sustech.edu.cn, dye6@jhu.edu, noahshen@connect.hku.hk, 12331257@mail.sustech.edu.cn
chaofan996@szu.edu.cn, 12432671@mail.sustech.edu.cn, vpatel36@jhu.edu, yusq@sustech.edu.cn

Abstract

Large vision model (LVM)-based gait recognition has achieved remarkable performance. However, current LVM-based methods largely rely on 2D image-plane representations and often neglect the underlying 3D structure of the human body. Without 3D structural guidance, 2D representations are sensitive to viewpoint variations. This sensitivity leads to unreliable gait features and limits generalizability in real-world 3D scenarios. To bridge this gap, we propose **PuppetGait**, a framework that aligns LVM features into a unified 3D canonical body space. The method conceptually manipulates the features like a puppet, adjusting them to a predefined pose and viewpoint. Comprehensive evaluations on CCPG, CCGR-MINI, CASIA-B*, and SUSTech1K show that PuppetGait maintains strong within-domain performance (96.54% mean Rank-1 on CCPG) while substantially improving cross-domain generalization, reaching 23.22% Rank-1 on CCPG-to-CCGR-MINI and 63.28% mean Rank-1 on CCGR-MINI-to-CCPG. All the models and code are available at https://github.com/ShiqiYu/OpenGait/.

1 Introduction

Gait recognition identifies individuals by their unique walking patterns, offering a non-intrusive biometric modality for long-distance and non-cooperative identification [14, 27, 34]. These properties make gait recognition valuable for security applications such as surveillance-based retrieval, suspect tracking, and identity verification, where subjects may be observed by ordinary cameras without active cooperation. Despite substantial progress in RGB-based gait analysis, robust cross-view recognition remains a fundamental challenge. This critical issue has been explicitly studied since early multi-view benchmarks such as CASIA-B [54] and OU-MVLP [40], and becomes even more entangled with clothing, carrying conditions, and camera-domain shifts in recent practical benchmarks like CCGR [59].

Prior works have explored cross-view challenges through two main routes as illustrated in Figure 1. Most existing methods learn view-invariant descriptors from 2D observations [26, 2, 6, 22]. These methods are practical and compatible with camera inputs, but their representations remain tied to the observed image plane, so cross-view body-part correspondence is mostly learned implicitly from training data. In contrast, structure-aware methods introduce pose [21], parametric body meshes [55], LiDAR point clouds [35], or explicit 3D galleries [60]. While these methods provide a stronger basis for cross-view correspondence, they often impose additional constraints, such as sparse pose

[†]Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

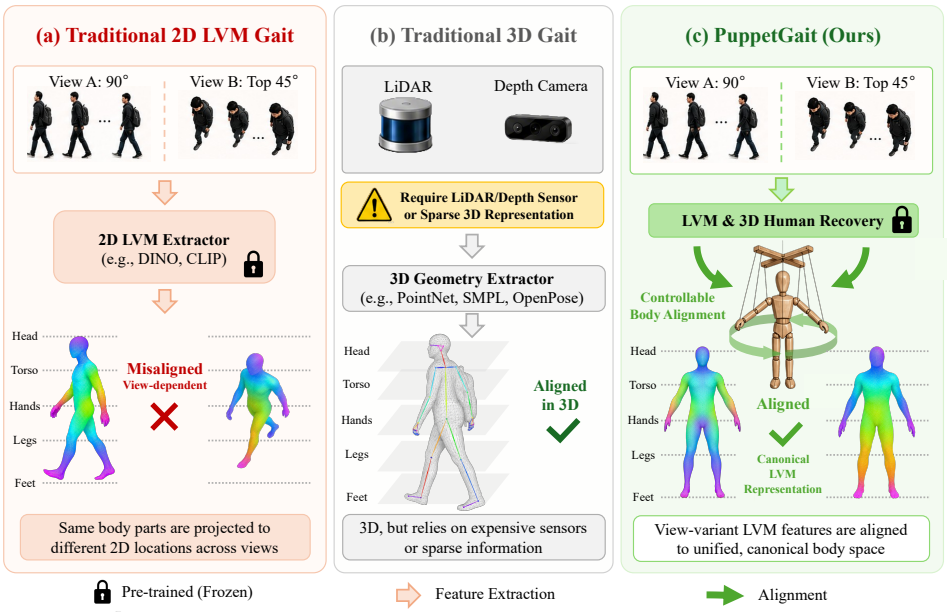


Figure 1: Overview and motivation of PuppetGait. Traditional 2D LVM methods preserve rich RGB semantics but remain view-dependent in the image plane. Traditional 3D methods are inherently aligned but rely on depth/LiDAR sensors or sparse 3D representations. PuppetGait uses LVM inferred 3D body to reorganize LVM features into a canonical feature layout.

representations, template-confined meshes, dedicated depth sensors, or subject-specific 3D enrollment. This reveals a key trade-off: image-plane methods preserve practical RGB pipelines but lack explicit spatial alignment, whereas explicit 3D methods improve structural correspondence but are less flexible as general RGB recognition solutions.

Recent advances in pretrained vision foundation models provide powerful tools for gait recognition. On the representation side, LVM-based gait methods have introduced strong pretrained visual features into gait modeling. BigGait[50] adapts task-agnostic LVM features for gait recognition, while BiggerGait[51] further shows that intermediate LVM layers contain discriminative and complementary identity cues. However, existing LVM-based methods primarily exploit feature discriminability within the standard 2D recognition pipeline, without introducing an explicit alignment mechanism. As a result, their robustness to viewpoint changes and domain shifts still largely depends on learned invariance from large-scale data, rather than explicit body-part correspondence. Meanwhile, LVM-based human 3D mesh recovery model[49] can infer dense, camera-aware body geometry from ordinary RGB frames, including body shape, articulated pose, and camera configuration. This geometry is represented by a parametric human body model, enabling the recovered body to be controlled by changing its pose and rendering viewpoint while preserving the inferred body shape.

This motivates us to revisit a long-standing intuition in gait recognition: reducing spatial variation by mapping human body toward a common view or canonical layout. Unlike view-transformation methods that operate on 2D gait templates, we apply this alignment to **dense LVM feature maps with the help of 3D body geometry**. Specifically, the parametric body recovered by a 3D human mesh recovery model serves as a controllable puppet: we first anchor image-plane LVM features to the inferred body and then manipulate the body into a preset target configuration, such as an A-pose rendered from a fixed view.

Following this insight, we introduce **PuppetGait**, a controllable body-alignment framework for LVM-based gait recognition. PuppetGait leverages a frozen LVM-based 3D human mesh recovery model to extract LVM features and estimate body geometry. The inferred body geometry is reconfigured into a preset target configuration to establish consistent body-part correspondence across viewpoints, as illustrated in Fig. 1(c). Guided by Geometry-aware Optimal Transport (GAOT), the original view-dependent 2D features are mapped into a canonical feature layout. This design preserves rich LVM semantics while improving cross-view comparability and cross-domain generalization.

To the best of our knowledge, this is the first LVM-based RGB gait framework that uses RGB-inferred 3D human geometry as a structural prior for feature-level body alignment. Our main contributions are summarized as follows:

- **A 3D-Guided Canonical LVM Framework:** We identify the spatial misalignment inherent in view-dependent LVM features and propose to use LVM-inferred body geometry as a structural carrier rather than an independent modality. Through Puppet Alignment and GAOT, PuppetGait anchors 2D image-plane LVM features to the inferred body surface and transports them into a canonical body-aligned feature layout.
- **State-of-the-art performance and strong generalization.** Extensive experiments on CCPG, CCGR-MINI, CASIA-B*, and SUSTech1K show that PuppetGait establishes state-of-the-art results in both within-domain and cross-domain settings. The gains are especially pronounced under cross-domain evaluation, demonstrating the effectiveness of 3D-guided canonical alignment for generalizable gait recognition.

2 Related Work

2.1 Gait representation learning

Early gait recognition relies on handcrafted body templates, including gait energy images and view transformation models [11, 26]. Modern deep methods typically learn sequence-level representations from silhouettes, with representative models improving set modeling, part-level temporal modeling, and global-local representation [2, 6, 22]. OpenGait and GaitBase further provide practical benchmarks and strong baselines for evaluating architectural progress [7, 9]. Beyond silhouettes, pose, skeleton, and parsing-based representations have been explored to inject human structure into gait recognition [21, 43, 8]. Mesh-based gait methods provide a denser body representation for structural reasoning [55, 47]. These methods demonstrate that body structure matters, but they usually commit to a fixed upstream representation. PuppetGait instead keeps rich LVM features and uses inferred body geometry only to establish controllable body-part correspondence, rather than another input modality.

2.2 Large vision models and human mesh recovery

Foundation models have changed visual representation learning by pretraining transferable backbones on large-scale data. CLIP learns visual features from language supervision [32], DINO-style models learn self-supervised visual features with strong transfer [1, 28, 37], and SAM provides promptable segmentation priors [18]. BigGait introduces LVM features to gait recognition by transforming all-purpose features into implicit gait representations, while BiggerGait shows that intermediate LVM layers contain complementary gait information [50, 51]. Other recent works explore denoising, large-language-model sequence modeling, and multi-modal gait feature construction [16, 48, 15]. In parallel, human mesh recovery and 3D body models have advanced from SMPL/SMPL-X based parametric bodies [24, 29] to transformer and promptable full-body recovery models [36, 5, 49]. PuppetGait connects these two trends by using downstream inferred body geometry to control the alignment of upstream dense LVM gait features.

2.3 Cross-view, cross-covariate, and multi-modal gait

Practical gait recognition must handle cross-view and cross-covariate variations, including clothing, carrying, occlusion, speed, and dataset domain shifts [54, 40, 19, 59]. In-the-wild benchmarks such as GREW and Gait3D reveal further domain complexity, while CCGR emphasizes mixed covariates beyond isolated view or clothing changes [58, 55, 59]. Recent methods address these issues through dynamic temporal representations, quality-aware modeling, causal intervention, and cross-granularity alignment [42, 44, 56, 41]. Multi-modal and 3D gait methods provide another path by using LiDAR, pointclouds, SMPL meshes, or 3D galleries [35, 15, 60]. These works suggest that geometry can reduce view ambiguity, but many require additional sensors, fixed upstream modalities, or explicit 3D enrollment. PuppetGait targets the same geometric need with standard RGB walking videos and an off-the-shelf 3D human prior.

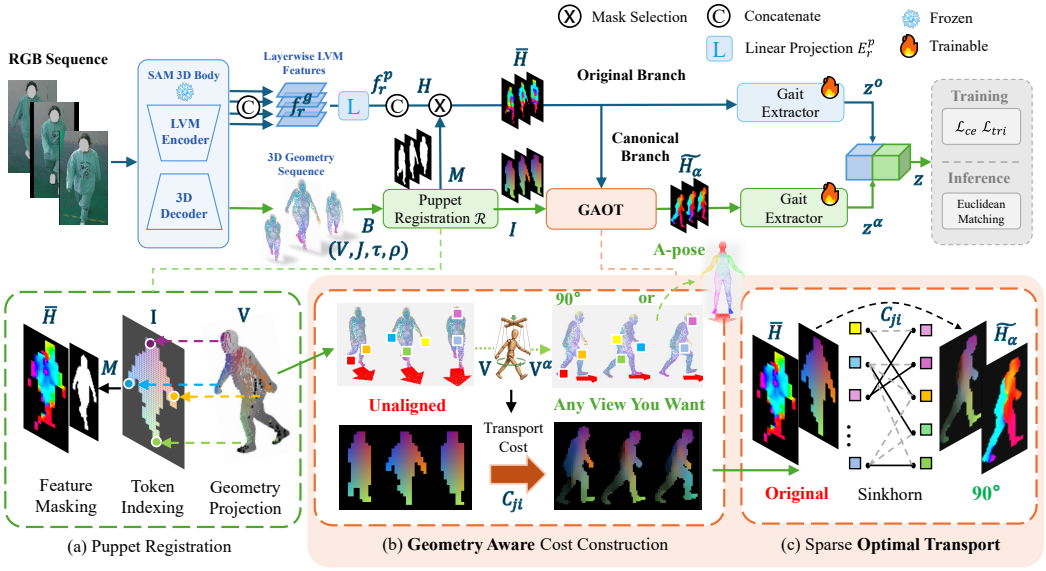


Figure 2: Overview of PuppetGait. A frozen SAM 3D Body model provides layer-wise LVM features and inferred 3D body geometry. The original-view branch encodes masked image-plane features, while the canonical branch anchors tokens to the visible body surface and transports them to a canonical feature layout by Geometry-aware Optimal Transport (GAOT). The original-view and canonical descriptors are fused for recognition.

2.4 Long-term human recognition and feature transport

Gait recognition is related to person re-identification (ReID), especially under clothing changes and long-term retrieval. ReID benchmarks and baselines have studied scalable retrieval, part pooling, transformer features, video aggregation, and re-ranking [57, 25, 12, 53]. Recent clothes-changing and body-shape ReID works further emphasize that identity cues should be disentangled from appearance and clothing, often through gait, pose, shape, or semantic priors [10, 17, 19, 23]. Unlike ReID methods that primarily retrieve single images or short video tracks from appearance, PuppetGait focuses on sequence-level gait features and their body-part correspondence across views. GAOT is inspired by optimal transport, which provides a principled way to match distributions or features under structural constraints [38, 4, 3, 31]. We use it for geometry-aware body-part feature transport rather than for generic domain adaptation.

3 Method

Given an RGB gait sequence, PuppetGait learns a sequence-level identity descriptor for retrieval using identity supervision during training. As illustrated in Fig. 2, PuppetGait converts LVM features into a canonical feature representation with a two-branch pipeline. For each RGB frame, a frozen SAM 3D Body model provides layer-wise LVM features and inferred body geometry. The original-view branch encodes masked image features, while the canonical branch treats the inferred body as a controllable puppet: it anchors valid LVM tokens to the visible body surface by **Puppet Alignment**. Then the **Geometry-aware Optimal Transport (GAOT)** module reconfigures the body under a preset target configuration and transports the anchored features to a canonical feature layout. Finally, both descriptors are fused for final recognition.

3.1 LVM Feature and Geometry Extraction

As shown in the upper-left part of Fig. 2, the frozen encoder supplies multi-layer visual semantics while the frozen MHR decoder infers the 3D human body geometry required by the canonical branch. For each frame $\mathbf{x}$, the frozen SAM 3D Body image encoder provides selected layer-wise LVM token features $\{\mathbf{F}^l\}$, including the final-layer features. The final-layer features also serve as the

encoder's output representation $\mathbf{e}$ used by 3D body decoder. Following BiggerGait [51], we partition the selected layer-wise features into N_g contiguous groups and apply a separate lightweight feature reduction module to each group to obtain a compact feature map. For the r-th group, the concatenated group feature and its reduced feature map are formulated as

$$\mathbf{f}_r^g = \mathrm{Concat}_{l\in\mathcal{G}_r}\,\mathbf{F}^l, \qquad \mathbf{f}_r^{\mathrm{p}} = E_r^{\mathrm{p}}(\mathbf{f}_r^g), \tag{1}$$

where $\mathcal{G}_r$ denotes the layer indices of group r, and E_r^{p} is a lightweight feature reduction module comprising channel reduction, spatial upsampling and sigmoid normalization. Concatenating the reduced group features along the channel dimension yields the compact LVM feature map $\mathbf{H}$ shared by later branches:

$$\mathbf{H} = \mathrm{Concat}_{r=1}^{N_g}\,\mathbf{f}_r^{\mathrm{p}}, \qquad \mathbf{H}\in\mathbb{R}^{C\times H_c\times W_c}. \tag{2}$$

In parallel, given the encoder representation $\mathbf{e}$ of the same frame, the frozen MHR decoder D_{3D} estimates camera-aware 3D human geometry and camera parameters:

$$\mathcal{B} = D_{\mathrm{3D}}(\mathbf{e}) = (\mathbf{V}, \mathbf{J}, \boldsymbol{\tau}, \boldsymbol{\rho}), \tag{3}$$

where $\mathbf{V} = \{\mathbf{v}_m\}_{m=1}^{M}$ are vertex-indexed 3D body surface points, $\mathbf{J}$ are 3D body keypoints, $\boldsymbol{\tau}$ is the predicted camera translation, and $\boldsymbol{\rho}$ is the global body orientation. Both the image encoder and the MHR decoder remain frozen throughout training and inference. The inferred geometry $\mathcal{B}$ serves strictly as a structural scaffold for foreground masking and spatial feature alignment, rather than as an independent geometric identity representation.

3.2 Puppet Alignment

To reorganize image-plane feature into a canonical feature layout via 3D body, Puppet Alignment is designed to establish this feature-to-body association. We need to associate 2D feature locations with body-surface points on the inferred mesh.

As illustrated in Fig. 2(a), given the decoded body vertices $\mathbf{V}$, camera translation $\boldsymbol{\tau}$, and camera intrinsics $\mathbf{A}$, we render the vertices onto the 2D feature map to obtain its 3D association. Meanwhile, the same projection process naturally provides a 2D silhouette mask, which is used to remove identity-irrelevant background and retain human features.

$$\mathbf{I}, \mathbf{M} = \mathcal{R}(\mathbf{V}, \boldsymbol{\tau}, \mathbf{A}), \tag{4}$$

where $\mathcal{R}(\cdot)$ denotes z-buffered vertex rendering. When multiple vertices are projected to the same grid cell, we keep the vertex with the closest camera depth z along the viewing ray. The resulting index map $\mathbf{I}$ records the visible vertex corresponding to each grid cell: $\mathbf{I}(i) = m$ means that feature location i is associated with body vertex $\mathbf{v}_m$. The body mask $\mathbf{M}$ indicates features that belong to human body, i.e., $\mathbf{M}(i) = 1$ if at least one body vertex is associated to cell i, and $\mathbf{M}(i) = 0$ otherwise.

We apply the mask to the LVM feature map as $\bar{\mathbf{H}} = \mathbf{H} \odot \mathbf{M}$.

3.3 Geometry-aware Optimal Transport

Once features are anchored to visible 3D mesh vertices via Puppet Alignment, we can control the vertices of each frame in 3D space to reorganize features into a unified canonical layout. Geometry-aware Optimal Transport (GAOT) accomplishes this in two stages, as illustrated in Fig. 2(b,c): geometry-guided cost construction, which establishes 3D-routed spatial correspondence to the canonical grid, and sparse optimal transport, which resolves projection collisions to softly transport LVM features into the canonical layout.

Geometry-guided cost construction. Let Ω^s and Ω^c denote the source and canonical feature grids. A source token is indexed by $i \in \Omega^s$ with 2D coordinate $\mathbf{u}_i^s \in [-1, 1]^2$, and a canonical target cell is indexed by $j \in \Omega^c$ with coordinate $\mathbf{u}_j^c \in [-1, 1]^2$. Mesh vertices are indexed by $m \in \{1, \ldots, M\}$ with 3D coordinates $\mathbf{v}_m$. Puppet Alignment provides $\mathbf{I}$ where $\mathbf{I}(i) = m$ indicates that source token i is anchored to visible vertex m.

The recovered 3D body is represented by a parametric human mesh template with a fixed vertex list, each vertex consistently representing the same anatomical body location. This property allows us

to construct a canonical puppet template by retaining the subject's inferred body-shape parameters while resetting the pose and camera viewpoint to a predefined canonical state α:

$$\mathbf{V}^{\alpha} = \mathcal{P}\left(\mathcal{S}(\mathbf{V}); \boldsymbol{\omega}_{\alpha}, \boldsymbol{\pi}_{\alpha}\right), \tag{5}$$

where $\mathcal{S}(\mathbf{V})$ denotes the inferred body shape, $\boldsymbol{\omega}_{\alpha}$ sets the target viewing direction (e.g., yaw and pitch), and $\mathcal{P}(\cdot)$ reconstructs the canonical vertex $\mathbf{V}^{\alpha} = \{\mathbf{v}_m^{\alpha}\}_{m=1}^{M}$ under an optional fixed pose $\boldsymbol{\pi}_{\alpha}$, e.g. A-pose. In Fig. 2(b), we illustrate a 90° unified view while keeping the original pose from input sequence.

Since $\mathbf{V}$ and puppet $\mathbf{V}^{\alpha}$ share the exact same vertex indexing, any vertex m on the observed body directly corresponds to $\mathbf{v}_m^{\alpha}$ on the canonical puppet. This shared index bridges arbitrary real-world camera viewpoints and walking poses to the unified canonical state. To translate this 3D correspondence back to the 2D feature plane for LVM alignment, we look up the anchored vertex $m_i = \mathbf{I}(i)$ of each valid source token i on $\mathbf{V}^{\alpha}$ and project it onto the canonical feature grid:

$$m_i = \mathbf{I}(i), \qquad \mathbf{v}_{m_i}^{\alpha} = \mathbf{V}^{\alpha}[m_i], \qquad \mathbf{p}_i^{\alpha} = \Pi^{c}\left(\mathbf{v}_{m_i}^{\alpha}\right), \tag{6}$$

where Π^{c} projects the 3D canonical vertex onto continuous 2D coordinates $\mathbf{p}_i^{\alpha} \in [-1, 1]^2$, specifying the ideal destination of source token i on the canonical plane. Rendering $\mathbf{V}^{\alpha}$ with Π^{c} simultaneously yields the canonical foreground mask $\mathbf{M}^{c}$ ($\mathbf{M}_j^{c} = 1$ if target cell j is covered by the puppet).

With the destination $\mathbf{p}_i^{\alpha}$ of each source token determined, a straightforward implementation is *Direct Relocation*—placing each token into its nearest grid cell. However, when the visible 3D body surface is rotated to the canonical view and re-projected onto the 2D grid, it inevitably creates overlaps (multiple $\mathbf{p}_i^{\alpha}$ falling into the same discrete target cell) and holes (originally invisible surface on puppet receiving no token), causing loss of rich LVM semantics (discarding over 40% of visible foreground tokens under large view gaps; see Appendix C). To prevent overlapping tokens from being overwritten and allow them to fill nearby void, we relax hard relocation into soft weighted transport. Specifically, we measure the spatial transport cost between any target grid cell j and source token i by the squared Euclidean distance between the cell center $\mathbf{u}_j^{c}$ and the token's projected destination $\mathbf{p}_i^{\alpha}$:

$$\mathbf{C}_{ji} = \left\|\mathbf{u}_j^{c} - \mathbf{p}_i^{\alpha}\right\|_2^2. \tag{7}$$

A smaller cost $\mathbf{C}_{ji}$ indicates that source token i contributes more to target cell j on the canonical grid through Optimal Transport (OT).

Sparse optimal transport. We formulate canonicalization as localized sparse optimal transport (*Sparse OT*), as illustrated in Fig. 2(c).

Directly applying standard OT (*Dense OT*) without constraints allows source tokens to contribute to arbitrary target cells, over-smoothing discriminative LVM semantics across unrelated body parts. Specifically, we mask out invalid connections outside the foreground masks $\mathbf{M}$, $\mathbf{M}^{c}$, a spatial radius δ, and restrict each canonical target cell j to its top-k nearest valid source candidates:

$$\mathcal{S}_j = \mathrm{TopK}_i\left(-\mathbf{C}_{ji}, k\right), \qquad k = 4. \tag{8}$$

This localized support ensures that colliding source tokens are softly redistributed only among adjacent canonical cells. We then build a masked transport kernel on this support:

$$\mathbf{K}_{ji} = \exp\left(-\mathbf{C}_{ji}/\varepsilon\right) \cdot \mathbb{I}\left[i \in \mathcal{S}_j\right] \cdot \mathbb{I}\left[\mathbf{C}_{ji} < \delta^2\right] \cdot \mathbf{M}_i \cdot \mathbf{M}_j^{c}, \tag{9}$$

where ε is the transport temperature. Sinkhorn normalization yields the sparse transport plan:

$$\mathbf{P} = \mathrm{Sinkhorn}(\mathbf{K}). \tag{10}$$

The canonicalized feature at target location j is obtained by

$$\tilde{\mathbf{h}}_j^{\alpha} = \sum_{i \in \mathcal{S}_j} \mathbf{P}_{ji} \bar{\mathbf{h}}_i. \tag{11}$$

GAOT does not hallucinate features for largely unobserved body regions: any canonical cell j lacking valid source candidates within radius δ (i.e., $\mathcal{S}_j = \emptyset$) remains empty. After reshaping all target locations back to the canonical grid, we obtain the canonical feature map $\tilde{\mathbf{H}}^{\alpha} \in \mathbb{R}^{C \times H_c \times W_c}$, producing a body-aligned feature layout for sequence encoding.

3.4 Sequence Encoding and Optimization

The original-view and canonical feature sequences are encoded by two GaitBase-style sequence encoders and fused by concatenation:

$$\mathbf{z}^o = G_\theta^o\left(\left\{\bar{\mathbf{H}}_t\right\}_{t=1}^T\right), \qquad \mathbf{z}^\alpha = G_\theta^\alpha\left(\left\{\tilde{\mathbf{H}}_t^\alpha\right\}_{t=1}^T\right), \qquad \mathbf{z} = \mathrm{Concat}\left(\mathbf{z}^o, \mathbf{z}^\alpha\right). \tag{12}$$

The model is trained with the standard triplet and identity classification losses:

$$\mathcal{L} = \mathcal{L}_{\mathrm{tri}} + \mathcal{L}_{\mathrm{ce}}, \tag{13}$$

while the 3D foundation model remains frozen. During inference, probe and gallery descriptors are compared using Euclidean distance.

4 Experiments

4.1 Datasets and Evaluation Protocols

We evaluate PuppetGait on four RGB gait benchmarks in both within-domain and direct cross-domain settings. CCPG [19] is the primary clothing-changing benchmark; we report Rank-1 on full clothing change (CL), upper-clothing change (UP), pants change (DN), and bag carrying (BG), excluding identical-view matches. For CCGR [59], we use the official CCGR-MINI split, denoted as CCGR*, and report Rank-1/Rank-5. SUSTech1K [33] and CASIA-B* [54] are additional transfer targets; we report SUSTech1K overall Rank-1/CL Rank-1 and CASIA-B* NM/BG/CL Rank-1.

4.2 Implementation Details

Unless otherwise specified, PuppetGait uses SAM 3D Body as the frozen LVM-based 3D human foundation model. We use its image encoder for layer-wise LVM features and its decoder for inferring body geometry. The last 16 encoder layers are split into four groups and projected to a 64×32 feature map. For GAOT, the temperature is 0.01, the distance threshold is 0.2, top-$k = 4$, and sparse Sinkhorn runs for 4 iterations. The canonical target configuration is A-pose rendered at 0° yaw.

All models use triplet and identity classification losses and are optimized by SGD with learning rate 0.1, momentum 0.9, and weight decay 5×10^{-4}. Using four 48GB NVIDIA A40 GPUs, PuppetGait is trained for 20k iterations on CCPG and 30k on CCGR*, with learning-rate decay at 10k/15k and 15k/25k iterations, respectively. Each batch contains 8 identities and 4 sequences per identity; we sample 30 frames for training and use up to 250 frames for evaluation.

4.3 Main Results

Table 1 compares PuppetGait with representative silhouette-based, multimodal, and RGB-based methods across four benchmarks. Cross-domain entries are omitted when target-domain modalities or annotations are unavailable. For evaluations that match the LVM backbone, foreground masking, and analysis on computational budget, see Tab. 3.

Cross-domain evaluation. The cross-domain columns of Tab. 1 demonstrate the strong generalization ability of PuppetGait across both transfer directions. *Trained on CCPG*, PuppetGait yields consistent improvements on every target benchmark (CCGR*, SUSTech1K, and CASIA-B*), reaching 23.2% Rank-1 on CCGR* (+7.5% over BiggerGait-DINOv2-L) and improving the challenging clothing-change (CL) condition on CASIA-B* by +10.4%. *Trained on CCGR**, PuppetGait raises the overall row average from 76.7% to 83.4% and achieves 63.3% mean Rank-1 on CCGR*-to-CCPG (+15.2% average improvement).

These substantial gains between CCPG and CCGR* reflect their underlying domain gap: CCGR introduces 33 camera views including extreme overhead angles and 53 walking conditions that entangle clothing, carrying and speed covariates [59]. Although unseen camera views and compound covariates severely distort pedestrian layouts in the 2D image-plane, PuppetGait explicitly normalizes these viewpoint variations into a consistent canonical body layout. Backbone-matched ablations (Tab. 2a) confirm that the cross-domain benefit is gained primarily from our canonical alignment

Table 1: Main comparison across source and target datasets. Yellow cells are within-domain results; others are cross-domain results. Best and second-best results are bolded and underlined. CCGR* denotes CCGR-MINI, SAM 3D Body* denotes the frozen SAM 3D Body foundation model [49], and Avg is computed over available entries.

Train Dataset	Input	Method	LVM	Venue	CCPG [19]				CCGR* [59]		SUSTech1K [33]		CASIA-B* [54]			Avg
					CL	UP	DN	BG	R-1	R-5	CL	R-1	NM	BG	CL	Avg
CCPG	Sil.	GaitSet [2]	–	PAMI'22	60.2	65.2	65.1	68.5	2.4	6.9	8.2	12.8	47.4	40.9	25.8	36.7
	Sil.	GaitPart [6]	–	CVPR'20	64.3	67.8	68.6	71.7	2.4	6.9	8.1	13.5	51.2	41.9	26.0	38.4
	Sil.	GaitGL [22]	–	ICCV'21	68.3	76.2	67.0	76.7	3.3	8.4	25.4	33.6	63.1	58.5	46.3	47.9
	Sil.	GaitBase [7]	–	CVPR'23	71.6	75.0	76.8	78.6	2.8	7.3	9.5	16.8	59.1	52.7	30.4	43.7
	Sil.	DeepGaitV2 [9]	–	TPAMI'25	78.6	84.8	80.7	89.2	3.7	9.1	27.0	38.4	74.6	67.2	50.2	54.9
	Sil.	FoundationGait [52]	–	ECCV'26	80.1	86.0	83.8	90.8	8.5	17.9	29.7	36.5	67.3	59.4	42.6	54.8
	Sil.+Skel.	BiFusion [30]	–	MTA'23	62.6	67.6	66.3	66.0	–	–	–	–	–	–	–	–
	Sil.+Skel.	SkeletonGait++ [8]	–	AAAI'24	79.1	83.9	81.7	89.9	–	–	–	–	–	–	–	–
	Sil.+Par.	XGait [56]	–	MM'24	72.8	77.0	79.1	80.5	–	–	–	–	–	–	–	–
	Sil.+Par.+Flow	MultiGait++ [15]	–	AAAI'25	83.9	89.0	86.0	91.5	–	–	–	–	–	–	–	–
	RGB+Sil.	GaitEdge [20]	–	ECCV'22	66.9	74.0	70.6	77.1	–	–	8.9	19.6	66.5	58.7	44.8	–
	RGB	BigGait [50]	DINOv2-S [28]	CVPR'24	82.6	85.9	87.1	93.1	7.4	16.3	43.7	56.4	77.4	71.5	33.6	59.5
	RGB	Gait-X [45]	–	ICCV'25	82.0	86.0	90.7	95.7	–	–	47.9	73.4	81.8	76.9	28.5	–
	RGB	BiggerGait [51]	DINOv2-S	NeurIPS'25	89.8	92.1	93.7	97.5	15.5	26.9	70.9	79.6	93.0	90.8	55.6	73.2
	RGB	BiggerGait [51]	DINOv2-L	NeurIPS'25	92.8	94.6	95.7	**98.2**	15.7	26.9	**76.4**	85.3	95.4	93.7	62.2	76.1
	RGB	LMGait [46]	DINOv2-S	AAAI'26	84.8	87.0	88.5	93.6	–	–	45.6	59.3	81.6	75.4	35.4	–
	RGB	GaitMax [13]	DINOv3-S	CVPR'26	86.6	88.2	90.2	93.2	–	–	55.0	–	85.6	86.9	46.2	–
	RGB	PuppetGait	SAM 3D Body*	Ours	**95.5**	**95.7**	**97.0**	98.0	**23.2**	**37.0**	74.4	**86.8**	**96.1**	**93.8**	**72.6**	**79.1**
CCGR*	Sil.	GaitBase [7]	–	CVPR'23	20.8	31.3	38.3	70.2	21.1	37.8	28.7	48.0	66.3	57.7	33.9	41.3
	Sil.	DeepGaitV2 [9]	–	TPAMI'25	23.0	37.1	36.5	69.9	26.4	45.2	32.1	51.7	72.0	62.0	36.6	44.8
	RGB	BigGait [50]	DINOv2-S	CVPR'24	22.9	42.2	25.0	80.5	88.0	95.9	71.9	85.6	90.1	87.9	58.4	68.0
	RGB	BiggerGait [51]	DINOv2-S	NeurIPS'25	23.2	44.5	29.5	86.7	85.7	94.1	82.1	93.8	97.2	96.4	66.7	72.7
	RGB	BiggerGait [51]	DINOv2-L	NeurIPS'25	25.0	47.9	31.0	88.5	93.0	**97.7**	**94.1**	97.5	97.4	96.6	74.6	76.7
	RGB	PuppetGait	SAM 3D Body*	Ours	**44.7**	**72.2**	**38.7**	**97.5**	**94.0**	**97.7**	94.0	**98.1**	**98.4**	**98.1**	**83.9**	**83.4**

design rather than the SAM 3D Body LVM itself: simply replacing BiggerGait's DINOv2-L backbone with SAM 3D Body raises CCPG-to-CCGR* Rank-1 only from 15.7% to 17.4%, whereas PuppetGait achieved 23.2%.

Within-domain recognition. Beyond cross-domain transfer, PuppetGait preserves top source-domain performance, as shown by the yellow cells in Tab. 1. On CCPG, PuppetGait achieves 96.5% mean Rank-1, surpassing BiggerGait-DINOv2-L on CL by +2.7% and outperforming the multimodal method MultiGait++ by +8.9% using RGB inputs alone, confirming that 3D-guided alignment extracts strong identity cues without requiring silhouettes, parsing, skeletons, or optical flow at inference. On CCGR*, PuppetGait achieves 94.0% Rank-1, outperforming BigGait, BiggerGait-DINOv2-S, BiggerGait-DINOv2-L, and the backbone-matched BiggerGait-SAM-3D-Body baseline (Tab. 3b) by +6.0%, +8.3%, +1.0%, and +4.0%, respectively. When training and test splits share a similar distribution (e.g., matched viewpoints), the original-view branch can directly exploit unmodified domain-specific features, making canonical alignment largely neutral in-domain while mainly benefiting cross-domain generalization.

4.4 Visualization

Figure 3 visualizes the feature reorganization produced by GAOT. Source LVM tokens contain rich semantics but follow the observed camera layout, so body parts can shift or disappear under viewpoint changes, carrying, and occlusion. After transport, tokens are reorganized into a canonical feature layout, making corresponding body regions more comparable across sequences. The branch responses further show the intended complementarity: the original-view branch keeps discriminative image-plane evidence, while the canonical branch produces more stable aligned body-part activations.

4.5 Ablation Study

Table 2 summarizes four design checks under CCPG training. These ablations separate the source of LVM appearance features, the quality of inferred body geometry, the transport strategy, and the canonical target layout, verifying that the transfer gain comes from 3D canonical organization rather than only from a stronger LVM backbone or a specific template choice.

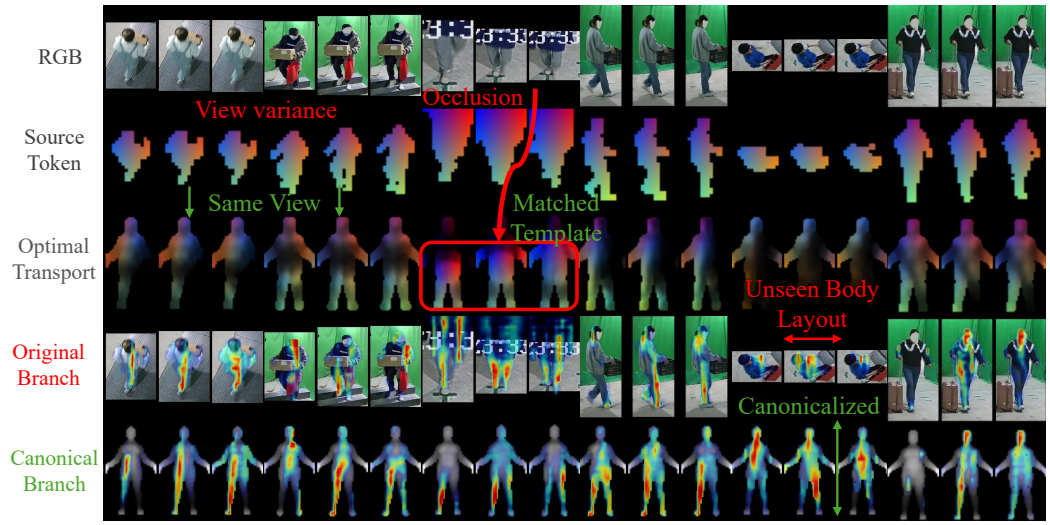


Figure 3: Visualization of PuppetGait. Rows show RGB frames, color-coded source LVM tokens, tokens after Geometry-aware Optimal Transport (GAOT), original-view branch responses, and canonical branch responses. Source tokens remain tied to the observed image layout and can become spatially inconsistent under occlusion or view changes. GAOT routes visible tokens through Puppet Alignment and transports them to a canonical feature layout.

Table 2: Ablations under CCPG training. CCPG denotes the source-domain mean Rank-1 over CL/UP/DN/BG, and CCGR* denotes direct cross-domain Rank-1 on CCGR-MINI.

(a) LVM Models and Canonical Branch

Method	LVM source	Canonical	CCPG	CCGR*
BiggerGait	DINOv2-S	✗	93.3	15.5
PuppetGait	DINOv2-S	✓	92.3	17.9
BiggerGait	SAM 3D Body	✗	95.9	17.4
PuppetGait	SAM 3D Body	✓	**96.5**	**23.2**

(b) 3D Body Geometry

Geometry	CL	UP	DN	BG	CCPG
ROMP [39]	93.4	93.5	96.2	97.7	95.2
SAM 3D Body	**95.5**	**95.7**	**97.0**	**98.0**	**96.5**

(c) Sparse Optimal Transport

Transport	OT	TopK4	CCPG	CCGR*
Direct	✗	✗	96.4	22.9
Dense OT	✓	✗	95.9	21.9
Sparse OT	✓	✓	**96.5**	**23.2**

(d) Canonical Target Layout

Layout	CL	UP	DN	BG	CCPG	CCGR*
w/o layout	**95.7**	95.6	96.3	97.9	96.4	18.7
w/o A-pose, 0°	95.4	**95.9**	96.8	97.9	96.5	23.0
w/o A-pose, 90°	95.5	95.6	96.3	**98.1**	96.4	23.1
A-pose, 0°	95.5	95.7	**97.0**	98.0	**96.5**	**23.2**

LVM models and canonical branch. Tab. 2(a) isolates that the cross-domain generalization gain comes from canonical alignment. With DINOv2-S features, enabling the canonical branch raises CCPG-to-CCGR* Rank-1 from 15.5% to 17.9%, and with SAM 3D Body features it further improves transfer from 17.4% to 23.2% (or from 18.7% when the baseline also uses projected mask from inferred 3D body, i.e., *w/o layout* in Tab. 2d). This shows that reorganizing LVM features into a canonical body layout consistently improves cross-domain generalization across different feature backbones and transfer directions (see Tab. 3b and Appendix C).

3D body geometry. Tab. 2(b) replaces the default SAM 3D Body geometry with ROMP [39] while keeping the recognition pipeline fixed. The ROMP variant remains strong with 95.2% CCPG average Rank-1, indicating that PuppetGait is not tied to a single mesh estimator. SAM 3D Body improves the average to 96.5%, as its shared backbone and more consistent foreground boundary coverage better retain contour tokens during Puppet Alignment. Meanwhile, perturbation experiments in Appendix B verify that within a covered foreground, the mesh acts as a coarse structural scaffold that remains highly tolerant to local pose, orientation, and shape estimation errors.

Sparse optimal transport. Tab. 2(c) compares hard relocation, dense OT, and the sparse TopK4 transport used in the full model. Direct relocation already reaches 22.9% CCGR* Rank-1 because

Table 3: End-to-end computational complexity and backbone-matched Rank-1 attribution. All SAM 3D Body variants share the exact same frozen image encoder.

(a) End-to-End Computational Complexity

Method	LVM	Frozen / Train. / Total (M)	GFLOPs	Peak Mem. (GiB)	Runtime: ms/f (30 frames)
BiggerGait	DINOv2-L	304.3 / 32.6 / 337.0	225.4	4.3	10.2 (307)
BiggerGait	SAM 3D Body	840.6 / 33.6 / 874.2	451.7	5.5	13.8 (415)
w/o canon.	SAM 3D Body	1285.4 / 41.3 / 1326.7	927.0	6.0	29.6 (888)
PuppetGait	SAM 3D Body	1285.4 / 69.5 / 1355.0	949.2	6.3	34.8 (1046)

(b) Backbone-Matched Rank-1 Performance under SAM 3D Body (%)

Evaluation Setting	BiggerGait	w/o canon.	**PuppetGait** (Full)	Net Gain (vs. w/o canon.)	Total Gain (vs. BiggerGait)
CCPG (In-domain)	95.9	96.4	**96.5**	+0.1	+0.6
CCPG → CCGR*	17.4	18.7	**23.2**	+4.5	+5.8
CCGR* (In-domain)	90.0	92.2	**94.0**	+1.8	+4.0
CCGR* → CCPG	62.1	62.6	**63.3**	+0.7	+1.2

it uses 3D geometry to canonicalize source tokens, but dense OT drops to 21.9%, indicating that unrestricted matching can over-mix unrelated body regions. Sparse OT recovers the best source-domain average and improves transfer to 23.2%, as localized soft assignment resolves projection collisions to retain more foreground tokens, particularly under large viewpoint gaps (see view-gap and sensitivity analyses in Appendix C).

Canonical feature layout. Tab. 2(d) evaluates whether the result depends on a particular target pose or yaw. Without any canonical target layout, the SAM 3D Body LVM baseline keeps strong CCPG performance but only reaches 18.7% CCGR* Rank-1. Adding a canonical layout raises transfer to 23.0–23.2%, while keeping the walking pose without A-pose reset and changing the target yaw give close results. Thus, the main benefit comes from establishing a consistent canonical feature layout, rather than from a specific pose or view template.

Backbone-matched attribution and computational cost. To disentangle the contribution of the foundation model from our alignment mechanism across both training domains, Tab. 3 profiles end-to-end inference complexity (batch size 1, 30 frames, FP16 AMP on an NVIDIA RTX 6000 Ada GPU) alongside strictly backbone-matched Rank-1 accuracy under SAM 3D Body. Across all four within- and cross-domain benchmarks, PuppetGait consistently outperforms both the backbone-matched BiggerGait baseline and the no-canonical control (*w/o canon.*, which uses the same MHR-projected foreground mask). While the higher baseline cost relative to DINOv2-L stems primarily from the frozen MHR mesh decoder, our canonical branch and GAOT module add only 2.1% total parameters (+28.2M) and 2.4% FLOPs (+22.2G), yet account for the vast majority of the cross-domain gain (+4.5 of the +5.8 points on CCPG→CCGR*).

5 Conclusion

We presented PuppetGait, a 3D body-aligned LVM framework for generalizable RGB gait recognition. By anchoring image-plane LVM tokens to downstream 3D body geometry and transporting them to a canonical feature layout, PuppetGait preserves rich visual semantics while introducing explicit body-part correspondence. Experiments across four benchmarks show strong within-domain recognition and consistent cross-domain gains, validating 3D-guided canonical alignment as an effective complement to stronger LVM backbones.

Limitations. PuppetGait depends on LVM-based 3D body recovery and projection, which introduces additional computation cost (Tab. 3a). In addition, extreme viewpoint gaps, occlusion, or severe body truncation can reduce visible surface overlap and affect geometry correspondence, as examined in Appendices B.2 and C. More efficient and robust body estimators deserve future exploration.

Acknowledgments and Disclosure of Funding

The team of Southern University of Science and Technology was supported by the National Natural Science Foundation of China (Grant 62476120) and Ant Group. Chao Fan was supported by the National Natural Science Foundation of China (Grant 62506236), Guangdong Basic and Applied Basic Research Foundation (Grant 2026A1515010144), and Shenzhen Science and Technology Program (Grant QNXMC20250701094318026).

References

[1] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[2] Hanqing Chao, Kun Wang, Yiwei He, Junping Zhang, and Jianfeng Feng. GaitSet: Cross-view gait recognition through utilizing gait as a deep set. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3467–3478, 2022.

[3] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865, 2016.

[4] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[5] Sai Kumar Dwivedi, Yu Sun, Priyanka Patel, Yao Feng, and Michael J Black. Tokenhmr: Advancing human mesh recovery with a tokenized pose representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1323–1333, 2024.

[6] Chao Fan, Yunjie Peng, Chunshui Cao, Xu Liu, Saihui Hou, Jiannan Chi, Yongzhen Huang, Qing Li, and Zhiqiang He. GaitPart: Temporal part-based model for gait recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14225–14233, 2020.

[7] Chao Fan, Junhao Liang, Chuanfu Shen, Saihui Hou, Yongzhen Huang, and Shiqi Yu. OpenGait: Revisiting gait recognition towards better practicality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9707–9716, June 2023.

[8] Chao Fan, Jingzhe Ma, Dongyang Jin, Chuanfu Shen, and Shiqi Yu. SkeletonGait: Gait recognition using skeleton maps. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1662–1669, 2024.

[9] Chao Fan, Saihui Hou, Junhao Liang, Chuanfu Shen, Jingzhe Ma, Dongyang Jin, Yongzhen Huang, and Shiqi Yu. OpenGait: A comprehensive benchmark study for gait recognition towards better practicality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[10] Xinqian Gu, Hong Chang, Bingpeng Ma, Shutao Bai, Shiguang Shan, and Xilin Chen. Clothes-changing person re-identification with rgb modality only. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1060–1069, 2022.

[11] Jinguang Han and Bir Bhanu. Individual recognition using gait energy image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(2):316–322, 2005.

[12] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15013–15022, 2021.

[13] Zhanbo Huang, Dingqiang Ye, Xiaoming Liu, and Yu Kong. Unlocking motion from large vision models with a semantic and kinematic duality for gait recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28379–28390, 2026.

[14] Anil K. Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[15] Dongyang Jin, Chao Fan, Weihua Chen, and Shiqi Yu. Exploring more from multiple gait modalities for human identification. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*, pages 4120–4128, 2025.

[16] Dongyang Jin, Chao Fan, Jingzhe Ma, Jingkai Zhou, Weihua Chen, and Shiqi Yu. On denoising walking videos for gait recognition. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12347–12357, 2025.

[17] Xin Jin, Tianyu He, Kecheng Zheng, Zhiheng Yin, Xu Shen, Zhen Huang, Ruoyu Feng, Jianqiang Huang, Zhibo Chen, and Xian-Sheng Hua. Cloth-changing person re-identification from a single image with gait prediction and regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14278–14287, 2022.

[18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[19] Weijia Li, Saihui Hou, Chunjie Zhang, Chunshui Cao, Xu Liu, Yongzhen Huang, and Yao Zhao. An in-depth exploration of person re-identification and gait recognition in cloth-changing conditions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13824–13833, 2023.

[20] Junhao Liang, Chao Fan, Saihui Hou, Chuanfu Shen, Yongzhen Huang, and Shiqi Yu. GaitEdge: Beyond plain end-to-end gait recognition for better practicality. In *Computer Vision – ECCV 2022*, 2022.

[21] Rijun Liao, Shiqi Yu, Weizhi An, and Yongzhen Huang. A model-based gait recognition method with body pose and human prior knowledge. *Pattern Recognition*, 98:107069, 2020.

[22] Beibei Lin, Shunli Zhang, and Xin Yu. Gait recognition via effective global-local feature representation and local temporal aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14648–14656, 2021.

[23] Feng Liu, Minchul Kim, ZiAng Gu, Anil Jain, and Xiaoming Liu. Learning clothing and pose invariant 3d shape representation for long-term person re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19617–19626, 2023.

[24] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023.

[25] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019.

[26] Yasushi Makihara, Ryusuke Sagawa, Yasuhiro Mukaigawa, Tomio Echigo, and Yasushi Yagi. Gait recognition using a view transformation model in the frequency domain. In *European Conference on Computer Vision*, pages 151–163. Springer, 2006.

[27] Mark S Nixon and John N Carter. Automatic recognition by gait. *Proceedings of the IEEE*, 94 (11):2013–2024, 2006.

[28] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

[29] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019.

[30] Yunjie Peng, Kang Ma, Yang Zhang, and Zhiqiang He. Learning rich features for gait recognition by integrating skeletons and silhouettes. *Multimedia Tools and Applications*, 83(3):7273–7294, 2024.

[31] Gabriel Peyré and Marco Cuturi. *Computational optimal transport: With applications to data science*. Now Foundations and Trends, 2019.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763, 2021.

[33] Chuanfu Shen, Chao Fan, Wei Wu, Rui Wang, George Q Huang, and Shiqi Yu. LidarGait: Benchmarking 3d gait recognition with point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1054–1063, 2023.

[34] Chuanfu Shen, Shiqi Yu, Jilong Wang, George Q. Huang, and Liang Wang. A comprehensive survey on deep gait recognition: Algorithms, datasets and challenges. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2024.

[35] Chuanfu Shen, Rui Wang, Lixin Duan, and Shiqi Yu. Lidargait++: Learning local features and size awareness from lidar point clouds for 3d gait recognition. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6627–6636, 2025.

[36] Soyong Shin, Juyong Kim, Eni Halilaj, and Michael J Black. Wham: Reconstructing world-grounded humans with accurate 3d motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2070–2080, 2024.

[37] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

[38] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

[39] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11179–11188, 2021.

[40] Noriko Takemura, Yasushi Makihara, Daigo Muramatsu, Tomio Echigo, and Yasushi Yagi. Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition. *IPSJ Transactions on Computer Vision and Applications*, 10, 12 2018. doi: 10.1186/s41074-018-0039-6.

[41] Jilong Wang, Saihui Hou, Yan Huang, Chunshui Cao, Xu Liu, Yongzhen Huang, Tianzhu Zhang, and Liang Wang. Free lunch for gait recognition: A novel relation descriptor. In *European Conference on Computer Vision*, pages 39–56. Springer, 2024.

[42] Ming Wang, Xianda Guo, Beibei Lin, Tian Yang, Zheng Zhu, Lincheng Li, Shunli Zhang, and Xin Yu. Dygait: Exploiting dynamic representations for high-performance gait recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13424–13433, 2023.

[43] Zengbin Wang, Saihui Hou, Man Zhang, Xu Liu, Chunshui Cao, and Yongzhen Huang. GaitParsing: Human semantic parsing for gait recognition. *IEEE Transactions on Multimedia*, 2023.

[44] Zengbin Wang, Saihui Hou, Man Zhang, Xu Liu, Chunshui Cao, Yongzhen Huang, Peipei Li, and Shibiao Xu. Qagait: Revisit gait recognition from a quality perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5785–5793, 2024.

[45] Zengbin Wang, Saihui Hou, Junjie Li, Xu Liu, Chunshui Cao, Yongzhen Huang, Siye Wang, and Man Zhang. Gait: Exploring x modality for generalized gait recognition. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13259–13269. IEEE, 2025.

[46] Zhengxian Wu, Chuanrui Zhang, Shen'Ao Jiang, Hangrui Xu, Zirui Liao, Luyuan Zhang, Li Huaqiu, Peng Jiao, and Haoqian Wang. Language-guided and motion-aware gait representation for generalizable recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 10871–10878, 2026.

[47] Chi Xu, Yasushi Makihara, Xiang Li, and Yasushi Yagi. Occlusion-aware human mesh model-based gait recognition. *IEEE Transactions on Information Forensics and Security*, 2023.

[48] Shaopeng Yang, Jilong Wang, Saihui Hou, Xu Liu, Chunshui Cao, Liang Wang, and Yongzhen Huang. Bridging gait recognition and large language models sequence modeling. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3460–3469, 2025.

[49] Xitong Yang, Devansh Kukreja, Don Pinkus, Anushka Sagar, Taosha Fan, Jinhyung Park, Soyong Shin, Jinkun Cao, Jiawei Liu, Nicolas Ugrinovic, et al. Sam 3d body: Robust full-body human mesh recovery. *arXiv preprint arXiv:2602.15989*, 2026.

[50] Dingqiang Ye, Chao Fan, Jingzhe Ma, Xiaoming Liu, and Shiqi Yu. BigGait: Learning gait representation you want by large vision models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[51] Dingqiang Ye, Chao Fan, Zhanbo Huang, Chengwen Luo, Jianqiang Li, Shiqi Yu, and Xiaoming Liu. BiggerGait: Unlocking gait recognition with layer-wise representations from large vision models. In *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

[52] Dingqiang Ye, Chao Fan, Kartik Narayan, Bingzhe Wu, Chengwen Luo, Jianqiang Li, and Vishal M Patel. Silhouette-based gait foundation model. In *European Conference on Computer Vision*, pages 458–478. Springer, 2026.

[53] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven CH Hoi. Deep learning for person re-identification: A survey and outlook. *IEEE transactions on pattern analysis and machine intelligence*, 44(6):2872–2893, 2021.

[54] Shiqi Yu, Daoliang Tan, and Tieniu Tan. A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In *18th International Conference on Pattern Recognition (ICPR)*, volume 4, pages 441–444. IEEE, 2006.

[55] Jinkai Zheng, Xinchen Liu, Wu Liu, Lingxiao He, Chenggang Yan, and Tao Mei. Gait recognition in the wild with dense 3d representations and a benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[56] Jinkai Zheng, Xinchen Liu, Boyue Zhang, Chenggang Yan, Jiyong Zhang, Wu Liu, and Yongdong Zhang. It takes two: Accurate gait recognition in the wild via cross-granularity alignment. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 8786–8794, 2024.

[57] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, pages 1116–1124, 2015.

[58] Zheng Zhu, Xianda Guo, Tian Yang, Junjie Huang, Jiankang Deng, Guan Huang, Dalong Du, Jiwen Lu, and Jie Zhou. Gait recognition in the wild: A benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14789–14799, 2021.

[59] Shinan Zou, Chao Fan, Jianbo Xiong, Chuanfu Shen, Shiqi Yu, and Jin Tang. Cross-covariate gait recognition: A benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7855–7863, 2024.

[60] Shinan Zou, Chengyu Long, Fan Guo, and Jin Tang. 3d gallery for cross-covariate gait recognition. *Knowledge-Based Systems*, page 115576, 2026.

A Implementation and Evaluation Details

Input and training settings. SAM-based models use 512×256 RGB inputs. The last 16 layers of LVM form into four feature groups with 64×32 feature maps. The LVM encoder and body decoder remain frozen. All models are trained for 20k iterations on CCPG (learning-rate decay at 10k/15k), 30k on CCGR* (15k/25k), and 6k on CASIA-B* (3k/5k). Each training batch contains eight identities and four sequences per identity, with 30 frames per sequence. Evaluation uses up to 250 ordered frames per sequence.

Variant definitions in evaluation. Table 4 distinguishes the ablation settings that isolate foreground masking, the canonical branch, and transport under the same SAM 3D Body backbone. BiggerGait uses its self-supervised learned mask [51] with the encoder only, whereas PuppetGait derives a projected mask from the decoded 3D body.

Table 4: Differences in ablation study implementations under SAM 3D Body.

Variant	Mask	MHR Decoder	Canonical	Transport
BiggerGait	Learned	✗	✗	–
PuppetGait w/o canonical	Projected	✓	✗	–
PuppetGait (Direct)	Projected	✓	✓	Nearest Depth
PuppetGait (Dense OT)	Projected	✓	✓	Dense OT
PuppetGait (Sparse OT)	Projected	✓	✓	Top-4 OT

Transport settings. Sparse OT selects at most $k = 4$ valid source candidates per target within $\delta = 0.2$ ($\varepsilon = 0.01$) before 4 Sinkhorn iterations. Dense OT removes Top-k pruning (8 iterations). Direct uses no OT, keeping only the single source token with closest depth.

B Robustness to Recovered Geometry

B.1 Geometry Perturbations

Since PuppetGait uses the inferred 3D body as a structural scaffold for feature alignment rather than an exact identity representation, it does not rely on highly precise mesh recovery. To verify this on a CCGR* subset, we perturb the SAM 3D Body reconstruction of 3D body—joint rotations, global orientation, body shape, and scale to affect canonical branch. Specifically, rotation perturbations add uniform random noise $\mathcal{U}(-b, b)$ to the estimated 3D joint or global angles; shape noise adds zero-mean Gaussian noise $\mathcal{N}(0, \sigma^2)$ to shape coefficients. Perturbation severity is measured by 2D projection IoU between the perturbed and clean masks.

Table 5: Geometry perturbations on CCGR*. IoU and Rank-1 are in percent; Δ is in percentage points.

Perturbation	IoU (%)	Rank-1	Δ
Clean	100.00	94.58	0.00
Joint rotation, $b = 15^\circ$	71.75	94.24	-0.34
Joint rotation, $b = 30^\circ$	58.79	93.62	-0.96
Joint rotation, $b = 45^\circ$	51.72	93.26	-1.33
Global orientation, $b = 15^\circ$	81.69	94.31	-0.28
Global orientation, $b = 30^\circ$	70.47	93.94	-0.64
Global orientation, $b = 45^\circ$	63.18	93.52	-1.06
Shape, $\sigma = 0.5$	97.42	94.64	$+0.05$
Shape, $\sigma = 1.0$	94.95	94.55	-0.03
Source projection, $0.75\times$	56.29	91.43	-3.16
Source projection, $1.25\times$	62.62	94.33	-0.26

Perturbing joint rotations or global orientation up to 45° (reducing IoU to 51.72% and 63.18%) lowers Rank-1 by 1.33 and 1.06 points (Tab. 5), confirming strong tolerance to pose and shape errors. Shrinking the projection ($0.75\times$) is more harmful than expanding it ($1.25\times$, -3.16 vs. -0.26 points),

indicating that the preservation of boundary foreground tokens matters more than the exact alignment to the mesh.

B.2 Input Degradation and Occlusion

We evaluate CCGR*-trained models on a 1,050-sequence CCGR* subset. Low-resolution inputs are downsampled and resized back to 512×256; pose jitter adds per-frame Gaussian noise to recovered rotations.

Table 6: Low resolution and recovered-pose jitter on CCGR*. Scores are Rank-1 (%); changes are percentage points.

Test condition	Rank-1	Δ
Clean	96.13	0.00
Low resolution $\times 2$	95.87	-0.27
Low resolution $\times 4$	94.40	-1.73
Recovered-pose jitter 5°	95.33	-0.80
Recovered-pose jitter 10°	94.27	-1.87

Downsampling by $2\times/4\times$ reduces Rank-1 by 0.27/1.73 points, while $5^\circ/10^\circ$ pose jitter reduces it by 0.80/1.87 points (Tab. 6), supporting recovered geometry as a coarse scaffold that need not be exact.

Synthetic occlusion. We mask a centered, full-width horizontal band covering 40%, 60%, or 80% of the person-crop height.

Table 7: Recognition accuracy (%) under synthetic occlusion on CCGR*. Both variants use geometry-projected masking.

	w/o canonical		Full method	
Test condition	Rank-1	Rank-5	Rank-1	Rank-5
Clean	94.93	98.40	**96.13**	**98.53**
Occlusion 40%	**66.27**	**76.53**	64.40	74.93
Occlusion 60%	**34.53**	**46.67**	33.65	45.73
Occlusion 80%	1.47	**5.20**	1.47	4.40

Severe occlusion causes large degradation in both settings, as both branches miss a large proportion of RGB input. The complete method with the canonical branch degrades by at most 1.87 Rank-1 points when 40% of the input is truncated (Tab. 7), showing limited sensitivity due to severe occlusion.

B.3 Temporal Averaging and Natural Covariates

Temporal averaging. Sequence-level averaging of recovered body-shape parameters (retaining frame-specific pose, orientation, and camera) changes Rank-1 from 96.13% to 96.07% on the CCGR* subset, showing that independent per-frame recovery is sufficiently stable.

Natural covariates. We train on CCPG and compare the no-canonical control with the full method on SUSTech1K under clothing, carrying, and natural occlusion conditions.

The canonical branch improves every covariate subset by 7.78–9.31 points (Tab. 8), confirming that inferred geometry remains an effective scaffold under natural variations.

C Additional Transport Strategy Analysis

Viewpoint gap and coverage. During canonicalization, multiple source tokens may project to the same canonical cell: Direct keeps only one, whereas Sparse OT redistributes colliding tokens to nearby empty cells. Table 9 evaluates accuracy, canonical coverage (fraction of valid canonical cells

Table 8: CCPG-to-SUSTech1K Rank-1 (%) under natural covariates.

Condition	w/o canonical	Full method	Gain
Bag	79.14	**87.87**	+8.73
Clothes changing	65.68	**74.41**	+8.73
Object carrying	80.52	**88.30**	+7.78
Occlusion	81.28	**89.09**	+7.81
Uniform	68.59	**77.90**	+9.31
Umbrella	72.14	**81.16**	+9.02
Overall	79.14	**86.81**	+7.67

receiving features), and source utilization (fraction of visible foreground tokens retained) across view gaps $\Delta\theta$ relative to the 0° canonical view.

Table 9: View-gap analysis on CASIA-B* and CCGR*. (a) Rank-1 accuracy (%). (b) Canonical cell coverage (%) and source token utilization (%) on CASIA-B*.

(a) Rank-1 (%) by View Gap

Transport	$\leq 36^\circ$	36°–72°	$> 72^\circ$	Over.→Front.
Direct	96.65	93.37	90.51	85.25
Sparse OT	**96.70**	**94.02**	**91.42**	**86.39**
Gain	+0.05	+0.65	+0.91	+1.14

(b) Coverage & Utilization (%)

	Coverage		Utilization	
View Gap	Direct	Sp.OT	Direct	Sp.OT
$\leq 36^\circ$	52.44	**99.79**	72.64	**98.91**
36°–72°	41.53	**96.18**	58.60	**97.40**
$> 72^\circ$	43.20	**93.99**	58.08	**97.11**

As the view gap exceeds 72°, Direct utilization drops from 72.64% to 58.08% due to grid collisions, whereas Sparse OT retains $> 97.1\%$ of observed tokens and yields up to $+1.14$ points higher Rank-1 under extreme overhead-to-frontal shifts (Tab. 9).

Additional transfer directions and GAOT sensitivity. Table 10(a) evaluates backbone-matched variants across eight within- and cross-domain pairs trained on CCPG and CASIA-B*: all six cross-dataset transfers improve, confirming that the benefit is not specific to CCPG→CCGR*. Table 10(b) shows that Sparse OT is insensitive to hyperparameter variations around the default configuration ($k = 4, \delta = 0.20$, 4 iterations, $\varepsilon = 0.01$).

Table 10: Additional transfer and sensitivity analyses. (a) Backbone-matched transfer Rank-1 (%). (b) Sparse OT hyperparameter sensitivity on CCPG→CCGR* (bold denotes defaults).

(a) Multi-Direction Transfer

Train Dataset	Target Dataset	w/o canonical	PuppetGait (Full)	Gain (pp)
CCPG	CCPG	96.40	**96.50**	+0.10
CCPG	CCGR*	18.70	**23.20**	+4.50
CCPG	SUSTech1K	79.14	**86.81**	+7.67
CCPG	CASIA-B*	87.07	**87.50**	+0.43
CASIA-B*	CASIA-B*	98.92	**98.98**	+0.06
CASIA-B*	CCPG	42.54	**44.71**	+2.17
CASIA-B*	CCGR*	13.69	**15.40**	+1.71
CASIA-B*	SUSTech1K	91.70	**92.20**	+0.50

(b) Sparse OT Hyperparameter Sensitivity

Hyperparameter	Tested Values	Rank-1 (%)	ΔR-1 (pp)
Top-k support (k)	1 / 2 / **4** / 8	24.10 / 24.14 / **24.25** / 24.02	−0.15 / − 0.11 / **0.00** / − 0.23
Distance threshold (δ)	0.02 / **0.20** / ∞	23.54 / **24.25** / 24.31	−0.71 / **0.00** / + 0.06
Sinkhorn iterations	1 / 2 / **4** / 8	20.39 / 23.99 / **24.25** / 24.31	−3.85 / − 0.26 / **0.00** / + 0.06
Temperature (ε)	0.001 / **0.01** / 0.1	24.31 / **24.25** / 24.25	+0.06 / **0.00** / + 0.00

D Responsible Use and Societal Impact

Gait recognition supports security and forensic retrieval at a distance without active subject cooperation, but carries privacy risks if deployed without oversight. All experiments in this work are conducted exclusively on established public benchmarks collected with informed consent. Code and model weights are released under a research-only license prohibiting unauthorized surveillance. Furthermore, automated retrieval outputs should not be used as sole evidence for high-stakes identification without human verification, particularly given the performance degradation observed under severe occlusion (Tab. 7).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims are stated in the Abstract and Introduction and are supported by the method description and experiments in the Method and Experiments sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in the Limitations paragraph, including dependence on 3D body recovery and additional computation from geometry estimation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For all theoretical components in Sec. 3, we provide relative experimental results in Sec. 4.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The Sec. 3 and Sec. 4 sections describe the architecture, datasets, evaluation protocols, training schedule, hyperparameters, and inference procedure needed to reproduce the main results, and all the source code will be released.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [N/A]

   Justification: While all the source code will be released, it is not included in the paper now.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Sec. 4 includes all evaluation protocols, training schedule and hyperparameters.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: This paper use Rank-1 as main results, which is following prior works [51, 50]

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Sec. 4.2 described the compute resources.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We follow the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Gait recognition helps security applications, which is mentioned in Sec. 1

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and models used in this paper are public and cited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: No new human subject assets are collected in this study. All human subject assets are exclusively from publicity available datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.