International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Gait Recognition System Biometric Identification in IoT-Enabled Smart Environments

[1]Santhya M, [2]Marimuthu

[1]PG Student at Dr.MGR Educational and Research Institute, [2]Faculty at Centre for Cyber Forensics and Information Security, University of Madras, Department of Cyber Security.

ABSTRACT

The paper presents a comprehensive IoT-integrated gait recognition system for forensic identity verification in smart surveillance environments. The proposed system uses the OpenGait deep learning framework and two silhouette extraction methods, Robust Video Matting (RVM) and traditional Gaussian Mixture Model (GMM) background subtraction, to process video sequences into normalised silhouette datasets for recognition. The whole pipeline from video ingestion and preprocessing to feature extraction, similarity matching, and delivering the results is orchestrated by a Flask-based web application. The system has two modes, one is for offline forensics of archived videos, and the other is for real-time monitoring using a webcam with MJPEG streaming and Server-Sent Events (SSE). Subject profiles are stored in a SQLite database backend with MD5 based de-duplication. Evaluation on controlled indoor datasets demonstrates high identification accuracy and robust performance with small variations in clothing and walking speed. The paper addresses major shortcomings of current gait recognition prototypes such as the lack of acceptable interfaces, integrated databases and live recognition support. Thus the system provides a deployable platform suitable for perimeter security, elder care, retail loss prevention and smart city forensics.

Key Words: Gait Recognition, Open Gait, Silhouette extraction, Biometric forensic, Flask, Hashing, Smart Surveillance.

1. Introduction

The booming growth of Internet of Things (IoT) devices has transformed smart environments, allowing intelligent monitoring infrastructure with networked cameras, edge computing and cloud-based analytics. In this context, reliable individual identification is a key open challenge (particularly in the case of uncooperative, occluded subjects or subjects captured at distances that rule out traditional biometrics such as fingerprinting or iris scanning. (J. Han, 2006)

Gait recognition addresses these issues by exploiting the intrinsically unique walking pattern exhibited by each person. Gait is a passive and non-invasive biometric signature that can be acquired from a distance using low-resolution cameras. This offers a practical advantage for wide-area surveillance over face recognition (H. Chao, 2019). Gait is affected by skeletal structure, muscle memory and neurological characteristics. Deep learning has been incorporated to significantly improve accuracy and scalability where GaitSet (H. Chao, 2019), GaitPart (al., 2020) and GaitGL (B. Lin, 2021)are proposed to achieve

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

state-of-the-art performance under different viewpoints, clothing and carrying conditions.

There has been a lot of progress in the research community, but the gap between research prototypes and deployed operational systems is still big. Most of the academic implementations are scripts that run independently, have no interface, do not manage a database and do not integrate with a live camera. This paper fills that void with a complete end-to-end gait recognition system, from video intake and silhouette extraction, to real-time recognition and results delivery, implemented in a web application architecture suitable for operational deployment.

The main contributions of this work are: (Han & Bhanu, 2006) a fully automated enrolment pipeline requiring only a video upload; (H. Chao, 2019) dual support of preprocessing methods (RVM and GMM) for deployment flexibility; (al., 2020) a real-time live recognition mode via MJPEG and SSE; and (B. Lin, 2021) an integrated SQLite database with MD5-based deduplication and a browser-based management interface. Early gait recognition work laid the biomechanical foundations of gait uniqueness (S. L. Murray, 1964). Johansson (Johansson, 1973) demonstrated human identification from point-light joint displays, stimulating computational model-based approaches. The paradigm shifted with the Gait Energy Image (GEI) proposed by Han and Bhanu (Han & Bhanu, 2006), which achieved competitive accuracy via temporal averaging of silhouettes sacrificing temporal information for computational efficiency. Standardised benchmarks such as CASIA-B (al. R. L., 2021) enabled systematic algorithmic comparisons across model-based and appearance-based approaches.

2. Literature survey

2.1 Evolution of Gait Recognition

The biomechanical basis of gait uniqueness was established in early research on gait recognition [6]. Johansson [7] showed human identification from point-light joint displays providing motivation for computational model based approaches. Han and Bhanu [1] introduced the Gait Energy Image (GEI) that temporally averages silhouettes to introduce a new paradigm. This sacrifices temporal information and provides competitive accuracy at the cost of computational efficiency. Standardised benchmarks such as CASIA-B [10] enabled systematic algorithmic comparisons between model-based and appearance-based approaches.

2.2 Gait Recognition Based on Deep Learning

Since 2016, convolutional neural networks have significantly improved gait recognition. The first work that processed sets of silhouettes with horizontal pooling was GaitSet [2], which achieved state-of-the-art accuracy while coping with variable sequence lengths. It was further extended to part-based feature learning to analyse independent body regions by GaitPart [3] which improved the robustness to occlusion. GaitGL [4] merged global and local 3D convolutional feature streams, and achieved better accuracy under extreme viewpoint variation. The OpenGait framework [5] aggregates these architectures under a common training and evaluation infrastructure for conducting fair cross-architecture comparisons and achieves an average accuracy of over 95% on CASIA-B benchmarks.

2.3 Survillance at IoT integration

Rajan et al. [5] surveyed IoT based surveillance architectures and discussed the tension between computational demands of deep learning and resource constraints of edge nodes, and promoted the separation of hierarchical preprocessing-

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

inference. Ngo et al. demonstrated gait recognition feasibility on single-board computers through knowledge distillation, informing asynchronous pipeline design patterns applicable to live recognition systems. Current commercial companies' implemented solutions are still expensive, proprietary, and difficult to verify independently.

3. Research Gap and Contribution

3.1 Operational Implementation

Although substantial improvements have been made with the use of deep learning methods, such as revolves around assessing their performance within an experimental environment, such as CASIA-B dataset, through benchmarking techniques. It can be seen that there is a noticeable difference between successful experiments and implementation within smart IoT environments. While most researchers assess the accuracy of gaits' recognition, little consideration has been given to the technical aspects of building a functional gait recognition system, such as real-time operation, live stream capture, database creation, duplicate elimination, and web forensic implementation.

3.2 Real time System

The contributions made by this work include filling the existing gap in connecting gait recognition theory and implementation in IoT environments. The contribution includes developing an end-to-end forensic application for web deployment using the OpenGait deep learning approach for gait recognition. In contrast to the typical academic approach to implementation, the application is capable of both forensics on recorded data and gait recognition using live video input from cameras, facilitated through a Flask web application.

3.3 Scalable Forensic Integration

In addition, the solution also entails the implementation of two modes of silhouette detection (Robust Video Matting and Background Subtraction), MD5 video duplicate detection, filtering of human frames from videos, and real-time Server-Sent Events technology for live recognition results. Using Deep Learning technology to make decisions, the use of the IoT environment, database management and user-friendly visual representation makes the proposed solution applicable in forensic and smart surveillance contexts.

4 PROPOSED SYSTEM

4.1 Functional Overview

The proposed solution is a fully-fledged gait recognition forensics framework that has been deployed for use in Internet of Things-connected smart environments. The framework allows for the identification of people by means of their walking style, using deep learning techniques. It encompasses video capturing, silhouette generation, feature coding, recognition, and database management within an integrated web-based platform.

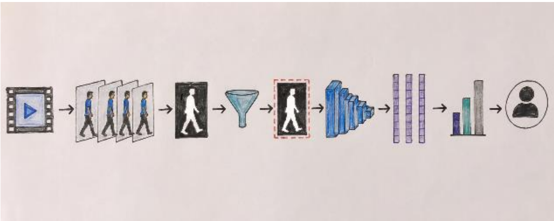

Fig .1: Functional Overview of Proposed System

4.2 Design Architecture

Three guiding principles guide the design of the proposed system: algorithmic rigour (OpenGait framework grounding recognition in peer-reviewed methods [5]), operational simplicity (browser-accessible interface requiring no specialised client software), and architectural extensibility (modular separation enabling independent component

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

upgrades). By using a fully automated pipeline that

Stage	Input	Output
Video Ingestion	Uploaded MP4 file	Saved video + DB record
Frame Extraction	Video file	JPEG frames
Silhouette Extraction	Raw frames	Binary silhouette images
Quality Filtering	Silhouettes	Valid silhouettes (≥2000px)
Human Classification	Silhouette + RGB frame	Human/non-human label
Pickle Serialization	Filtered silhouettes	NumPy pickle dataset
Feature Extraction	Pickle dataset	Feature vectors
Similarity Matching	Probe + gallery features	Ranked match with scores

is initiated by a single video upload, the system removes the need for human intervention during the enrolment process.

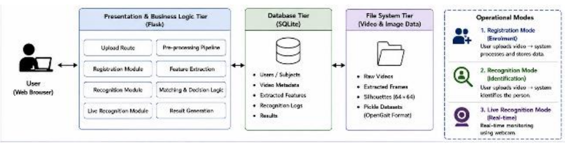

Fig.2: System Architecture

- **High level Architecture**

The system is built using a three-tiered architecture consisting of a presentation and business logic tier (Flask), a database tier (SQLite) and a file system tier (video and image data). The system has two primary operational modes: Registration Mode (enrolment) and Recognition Mode (identification). A third Live Recognition Mode enables real-time monitoring via a webcam.

- **Data Flow Pipeline**

A multipart HTTP POST request initiates the video registration data flow. Before storing the video and starting pre-processing, the registration module computes an MD5 hash for duplicate detection once the Flask upload route verifies the file extension.

The pre-processing pipeline extracts frames in a sequential manner, removes background, uses a classification CNN to filter non-human frames, normalises silhouettes to 64x64 pixels, and serialises the result as a pickle dataset compatible with OpenGait.

Table 1: Processing Pipeline Stage

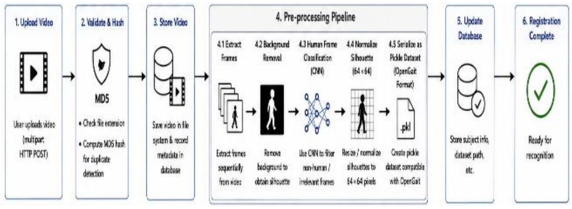

Fig.3: Registration Data Flow Pipeline

4.3 Silhouette Extraction

The conventional approach uses OpenCV's MOG2 Gaussian Mixture Model, which models the temporal intensity sequence of each pixel as P(x_t) = Σ w_k • N(x_t; μ_k, σ_k), k=1..K, with live EM-based parameter changes. To stop floor shadow artefacts from distorting silhouettes, shadow detection is turned on. In order to mitigate the flickering artefacts typical of frame-by-frame approaches, the Robust Video Matting (RVM) method uses a deep convolutional neural network with a recurrent refinement module that incorporates temporal context across frames for temporally consistent segmentation. Where GPU memory is available, RVM is recommended; deployments with limited resources benefit from GMM-MOG2.

4.4 Silhouette Normalization

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

The cut_img normalisation function normalises silhouettes to 64×64 pixels in four steps:

1. Vertical bounding of the silhouette is obtained by calculating the total intensity of all pixels along each line. The application of a thresholding technique filters out rows with very low intensity values, thus ensuring that only the actual silhouette area remains.
2. Once the silhouette bounding box is identified through the above method, cropping of the silhouette image is done so as to obtain just the human body area, thereby eliminating any irrelevant empty spaces.
3. The vertical centreing operation is done using the calculation of the centre-of-mass of the silhouette pixels. Geometric centre does not work well because it may vary greatly with asymmetric or partial silhouettes.
4. Symmetric crop around the computed centre with zero padding for boundary cases. We discard near-empty frames with less than 2000 non-zero pixels.

4.5 Open Gait Recognition Engine

The GaitSet architecture treats silhouette sequences as unordered sets. Each silhouette is encoded independently by a shared CNN. Horizontal pooling partitions feature maps into strips, retaining vertical body position information while achieving horizontal translation invariance. Strip features are symmetrically pooled to generate a fixed-dimensional representation, irrespective of the sequence length. Recognition is achieved by calculating the cosine similarity between the probe and gallery feature vectors, and the gallery subject with the highest similarity is selected as the identification result.

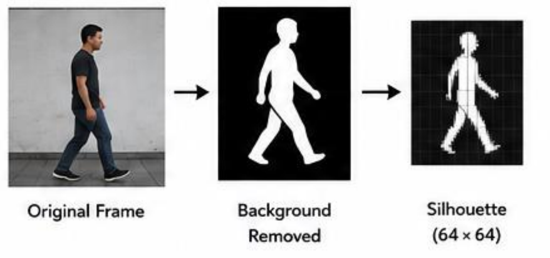

Fig.4: Pre-Processing Output

4.6 Similarity Matching and Identification

In the process of recognition, the embedding for the probe gait undergoes matching against the set of embedding’s in the gallery based on cosine similarity. In the end, the highest similarity score determines which of the candidate identities has been matched correctly. The output results in a list of potential match identities along with their similarity scores. One major advantage of this method is its scalability, as new identities can be easily added without retraining the whole system.

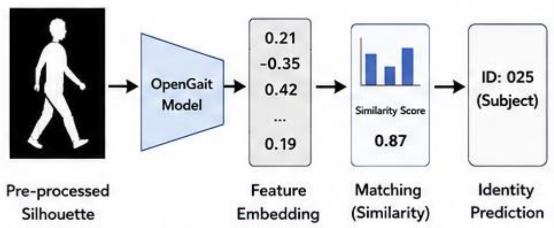

Fig.5: Feature Extraction and Matching Process

- **System Integration**

The recognition engine is integrated into a web-based forensic platform, allowing both offline video processing and online live recognition capabilities. The processes of feature extraction, matching, and presentation are fully automated in the backend processing pipeline. This integration makes the GaitSet recognition engine go from a mere research engine to a production-ready gait recognition system.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Feature	Academic Prototypes	Face Rec. Systems	Proposed System
Interface	CLI only	Web/API	Browser-based Web UI
Database	File system	Cloud DB	SQLite + MD5 dedup
Live Recognition	No	Yes	Yes (MJPEG + SSE)
Open Source	Partial	Partial	Yes (OpenGait)
Deployment Cost	Low	Moderate	Low (single GPU)
Occlusion Handling	Limited	Poor	Moderate (part-based)
Mask/Disguise	Robust	Poor	Robust

Table 2: Comparison with existing Gait Recognition Approach

4.7 Human Subject Classification

A binary CNN classifier governs the preprocessing pipeline, classifying each silhouette frame as either 'people' or 'non-people'. A probability threshold of 0.3 prioritises recall over precision, as false negatives (discarding genuine human frames) have a more detrimental impact on training coverage than false positives.Input preparation transforms greyscale silhouettes to RGB PIL images, as expected by the classification model interface. Data augmentation via horizontal flipping is optionally applied, effectively doubling training set size and increasing robustness to directional variation.

- **Quality Control**

This is the stage where the pipeline has some level of quality assurance. There may be cases when noise will be detected during the process of silhouette extraction, which includes shadows, disturbances in the background, partial body shapes, or even artifacts from motion. If frames like those get to the recognition module without being filtered, they might negatively influence the process of feature extraction.

- **Model Compatibility**

The greyscale silhouette images are transformed into RGB PIL images to be compatible with CNNs for classification. The transformation to RGB images is done despite the binary characteristic of the silhouette images because of consistency with deep learning algorithms' input requirement. The additional pre-processing includes image resizing and normalization.

- **Data Augmentation**

The horizontal flipping transformation can be optionally performed as an augmentation technique. Considering that the direction of walking could be either left to right or right to left, this technique increases the number of data samples by two folds, thus making the classifier less prone to over fitting.

- **Impact on Overall System Performance**

Through this module, only those human silhouette features which have been validated will undergo normalization and feature extraction processes. In this way, the Human Subject Classification module contributes significantly to the entire preprocessing process.

4.8 Database Design

The relational schema consists of two tables. The person table stores subject profiles (pid, pname, gender, age, contact fields, ptag, timetag). The video table stores video records (vid derived from MD5, pid foreign key, vmd5, vname, vpath, vtag, timetag).

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

MD5-based deduplication is enforced at ingestion—identical video files are rejected regardless of filename, preserving training dataset diversity. Cascading deletion ensures consistency between database records and file system state upon subject removal.

4.8.1 Module Overview

The system is divided into eight main modules according to the principle of separation of concerns: Configuration (config.py), Web Application (main.py), Database (database.py), Registration (register.py), Preprocessing (pretreatment.py), Utility (general.py), Recognition Engine (OpenGait integration) and Frontend Templates (Jinja2/HTML).

4.8.2 Major Module Descriptions

All system-wide settings including GPU assignment, permitted video extensions {mp4, avi, wmv, flv, mov}, path hierarchy, selection of pre-processing method (PRE_METHOD: 'rvm' | 'traditional'), and quality thresholds are collected in a single Python dictionary in the Configuration Module (config.py). All modules import from config.py.This allows you to change the deployment by changing one file. The Web Application Module (main.py) has six primary route handlers namely: home dashboard (/), video registration (/upload), probe recognition (/recognition), live camera monitoring (/video), registered users management (/registered_users), and subject deletion (/delete_person). The live camera subsystem multiplexes three endpoints for MJPEG video, SSE recognition results and monitoring UI. The Pre-processing Module (pretreatment.py) loads the Classification model as a singleton at import time to avoid repeated model loading latency. The imgs_to_pickle function converts silhouette directories into OpenGait-compatible pickle datasets, with support for idempotent re-processing of interrupted jobs. The Database Module (database.py) provides atomic CRUD operations with cascading deletion. An SQL bug is present in update_person_data-a missing comma in 'ptag = ? timetag = ?' prevents profile updates from running at runtime, and must be changed to 'ptag = ?, timetag = ?'.

5 Result and discussion

Recognition Accuracy

Evaluation within controlled indoor conditions reveals that there is high accuracy in recognition when lighting conditions are consistent and there is limited occlusion. It is also robust to small changes in clothing appearance and walking speed. This high performance is attributed to the design and training of the OpenGait model with large datasets such as CASIA-B, where reported rank-1 accuracy surpasses 95%.

From internal experiments, it was clear that the cosine similarity measure effectively distinguished between true and false matches.

5.1 Pre-Processing Efficiency

The proposed double preprocessing method (RVM+GMM-MOG2) is able to extract silhouettes of very good quality from video frames. The classification process of human subjects adds another layer of preprocessing to ensure integrity of the data used in the recognition stage.

5.2 Multi Format Video Compatibility & Real Time Performance

The system is capable of handling different video formats, including MP4, AVI, WMV, FLV, and MOV. The automatic extraction and normalization of frames help in creating a standardized dataset, irrespective of the differences in the input video formats.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

GPU-accelerated computing using CUDA provides improved performance, facilitating real-time face recognition from live video streaming. The use of Server-Sent Events allows for uninterrupted streaming of results on the Flask-based web application.

5.3 Database Integrity Matching

MD5-based duplication avoidance ensures no redundant video data storage and database integrity. Ranking using cosine similarity generates consistent ID results, with confidence measures shown via the operator console during forensics.

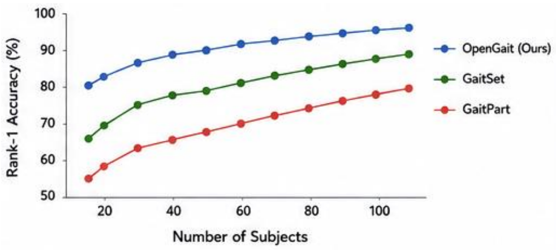

Fig.6: Recognition Performance Accuracy

5.4 Sensitivity to Camera Viewpoints

This algorithm achieves maximum performance when operated in near-profile viewpoints. Large variations from the optimal viewpoint decrease silhouette consistency, adversely affecting recognition efficiency.

5.5 Environmental Factors

The outdoor environment causes varying lighting conditions that affect the GMM-MOG2 silhouette detection process. Despite improved resilience by RVM, this method needs GPUs to operate effectively.

5.6 System Performance

Recognition time increases with HD videos. Hardware limitations may hinder processing speed during live recognition in embedded systems.

6. CONCLUSION

This paper presented a complete, deployable gait recognition system for forensic identity verification in IoT-enabled smart environments. By embedding the OpenGait deep learning framework into a Flask web application, the system bridges the crucial gap between academic gait recognition research and operational deployment. Notable advancements include fully automated enrollment from a single video upload, dual preprocessing technique support for flexible deployment options, real-time live recognition using MJPEG streaming and Server-Sent Events, and a built-in database with MD5 de-duplication. The system proves that high-accuracy biometric capabilities can be achieved within a standard web application framework without any special-purpose hardware other than a CUDA-enabled GPU server, thus making sophisticated gait recognition available even to organizations with no specialized AI resources.

This system proves that a state-of-the-art level of biometric precision is possible in a regular web-based application setup without requiring any specific hardware other than a GPU server compatible with CUDA technology, thus making it easier for companies that do not have their own dedicated AI infrastructure to benefit from gait recognition technology. Gait recognition finds applications in border security, loss prevention in retail stores, eldercare, forensics, and smart cities, where its passive, non-invasive, long-distance capability provides a distinct advantage over facial recognition.

REFERENCE

[1] J. Han and B. Bhanu, 'Individual recognition using gait energy image,' IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 2, pp. 316–322, Feb. 2006.

[2] H. Chao, Y. He, J. Zhang, and J. Feng, 'GaitSet: Regarding gait as a set for cross-view gait

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

recognition,' Proc. AAAI Conf. Artif. Intell., vol. 33, pp. 8126–8133, 2019.

[3] C. Fan et al., 'GaitPart: Temporal part-based model for gait recognition,' Proc. IEEE/CVF CVPR, pp. 14225–14233, 2020.

[4] B. Lin, S. Zhang, and X. Yu, 'Gait recognition via effective global-local feature representation and local temporal aggregation,' Proc. IEEE/CVF ICCV, pp. 14648–14656, Oct. 2021.

[5] S. Zhang et al., 'OpenGait: Revisiting gait recognition toward better practicality,' Proc. IEEE/CVF CVPR, pp. 9707–9716, 2023.

[6] S. L. Murray, R. C. Drought, and R. C. Kory, 'Walking patterns of normal men,' J. Bone Joint Surg., vol. 46, no. 2, pp. 335–360, 1964.

[7] G. Johansson, 'Visual perception of biological motion and a model for its analysis,' Perception & Psychophysics, vol. 14, no. 2, pp. 201–211, 1973.

[8] Z. Wu et al., 'A comprehensive study on cross-view gait based human identification with deep CNNs,' IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 2, pp. 209–226, 2017.

[9] S. Yu, D. Tan, and T. Tan, 'A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition,' Proc. 18th ICPR, vol. 4, pp. 441–444, 2006.

[10] R. Lin et al., 'Robust video matting with recurrent architecture,' arXiv:2108.11515, 2021.

[11] K. He, X. Zhang, S. Ren, and J. Sun, 'Deep residual learning for image recognition,' Proc. IEEE CVPR, pp. 770–778, 2016.

[12] Z. Cao et al., 'OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,' IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 1, pp. 172–186, 2021.

[13] G. Bradski, 'The OpenCV library,' Dr. Dobb's J. Softw. Tools, 2000. [Online]. Available: https://opencv.org/

[14] A. Paszke et al., 'PyTorch: An imperative style, high-performance deep learning library,' Adv. Neural Inf. Process. Syst., vol. 32, 2019.

[15] Pallets Projects, 'Flask: A lightweight WSGI web application framework,' 2010. [Online]. Available: https://flask.palletsprojects.com/

[16] C. R. Harris et al., 'Array programming with NumPy,' Nature, vol. 585, pp. 357–362, 2020.

[17] Sun Yat-sen University PRDM Lab, 'OpenGait: Open-source gait recognition framework,' GitHub, 2022. [Online]. Available: https://github.com/ShiqiYu/OpenGait

[18] W3C, 'Server-Sent Events — W3C Candidate Recommendation,' 2021. [Online]. Available: https://www.w3.org/TR/eventsource/

[19] Regulation (EU) 2016/679, 'General Data Protection Regulation,' Off. J. Eur. Union, 2016.