SIMPLE AND EFFICIENT METHODS FOR GAIT RECOGNITION USING POSE INFORMATION

VÍTOR CÉZAR DE LIMA

SIMPLE AND EFFICIENT METHODS FOR GAIT RECOGNITION USING POSE INFORMATION

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: William Robson Schwartz

Belo Horizonte

Junho de 2021

VÍTOR CÉZAR DE LIMA

SIMPLE AND EFFICIENT METHODS FOR GAIT RECOGNITION USING POSE INFORMATION

Thesis presented to the Graduate Program in Computer Science of the Universidade Federal de Minas Gerais in partial fulfillment of the requirements for the degree of Master in Computer Science.

Advisor: William Robson Schwartz

Belo Horizonte

June 2021

Lima, Vítor Cézar de

B732s Simple and efficient methods for gait recognition using pose information [manuscrito]. / Vítor Cézar de Lima. — Belo Horizonte, 2021.
xxvi, 47 f. : il. ; 29cm

Orientador: William Robson Schwartz.

Dissertação (mestrado) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação.
Referências: f. 41-47

1. Computação – Teses. 2. Visão por computador - - Teses. 3. Biometria – Teses. 4.– Reconhecimento de padrões (Computadores) – Teses. I. Schwartz, William Robson. II. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da computação. III. Título.

CDU 519.6*85(043)

Ficha catalográfica elaborada pela bibliotecária Irénquer Vismeg Lucas Cruz - CRB 6ª Região nº 819.

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Simple And Efficient Methods For Gait Recognition Using Pose Information

VÍTOR CÉZAR DE LIMA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. WILLIAM ROBSON SCHWARTZ - Orientador
Departamento de Ciência da Computação - UFMG

PROF. GUILLERMO CÁMARA CHÁVEZ
Departamento de Computação - UFOP

PROF. APARECIDO NILCEU MARANA
Departamento de Computação - Universidade Estadual Paulista

Belo Horizonte, 26 de Julho de 2021.

I dedicate this thesis to my mother, to my aunt and to the friends and brilliant scientists of Smart Sense Laboratory

Acknowledgments

I would like to thank my family for its support, professor William Robson Schwartz for the guidance on my master's degree, Victor Melo and Antonio Nazare for their assistance in works that contributed to this thesis and Smart Sense Laboratory for the friendship and insights to this work.

I also would like to thank the National Council for Scientific and Technological Development – CNPq (Grants 438629/2018-3 and 309953/2019-7), the Minas Gerais Research Foundation – FAPEMIG (Grants APQ-00567-14 and PPM-00540-17), the Coordination for the Improvement of Higher Education Personnel – CAPES (DeepEyes Project) and Petrobras (Grant 2017/00643-0).

"Simplicity is the ultimate sophistication."
(Leonardo da Vinci)

Resumo

Gait é um tipo de biometria que diferencia os indivíduos pela forma como andam. Pesquisas relacionadas a essa biometria estão ganhando evidência devido à vantagem de *gait* ser discreto e poder ser capturado a distância, o que é desejável em cenários de vigilância. A maioria dos trabalhos da literatura foca em usar silhueta humana como representação de *gait*; no entanto, elas sofrem de diversos fatores, como movimento de pessoas nas cenas, condições de carga e uso de roupas diferentes. Para evitar esses problemas, esse trabalho propõe um método de estimativa de pose, denominado PoseDist, para recuperar coordenadas de articulações e transformá-las em sinais e histogramas de movimento. Depois disso, essas informações são processadas usando uma fusão de *Subsequence Dynamic Time Warping* e distância euclidiana para comparar as sequências de *gait* da consulta com as da galeria. Esse método é avaliado em todas as visualizações de CASIA Dataset A e comparado com trabalhos existentes, demonstrando sua eficácia. No entanto, como seu custo algorítmico é alto, ele só é adequado para ambientes com poucos indivíduos; e dessa forma, um novo método denominado PoseFrame é desenvolvido para reconhecimento de *gait*, treinando uma rede neural multicamadas para classificar as poses a partir de quadros individuais e agregando os resultados por votação majoritária. PoseFrame é testado em CASIA Dataset A, tendo precisão acima dos outros trabalhos baseados em modelo, incluindo PoseDist; e em CASIA Dataset B, alcançando precisão estado-da-arte quando a amostra tem a mesma visualização da galeria e tendo alguns dos melhores resultados em validação cruzada. Finalmente, um estudo de ablação também é realizado para descobrir quais partes do corpo são as mais importantes para reconhecimento de *gait* e de acordo com os resultados, os braços e os pés são as localizações mais importantes.

Palavras-chave: Reconhecimento de Gait, Biometria, Visão Computacional.

Abstract

Gait is a biometry that differentiates individuals by their walking manner. Research on this topic has gained evidence since it is unobtrusive and available at distances, which is desirable in surveillance scenarios. Most of the previous works have focused on the human silhouette as representation; however, they suffer from many factors such as movement on scene, clothing and carrying conditions. To avoid such problems, this work employs a pose estimation method, called PoseDist, to retrieve the coordinates of body parts and transform them into signals and movement histograms. These features are processed using a fusion of Subsequence Dynamic Time Warping and Euclidean distance to compare gait sequences from the probe with those in the gallery. This method is evaluated on all views of CASIA Dataset A and compared to existing ones, demonstrating its efficacy. However, as its algorithmic cost is high, it is only suitable for environments with few individuals; and this way, a new method called PoseFrame is employed for gait recognition, training a multilayer perception to classify poses from individual frames and aggregating its results by majority voting. PoseFrame is tested on CASIA Dataset A, having accuracy above other model-based works, including PoseDist; and on CASIA Dataset B, achieving state-of-the-art accuracy on same-view condition and having some of the best results on cross-view. Finally, an ablation study is also performed to find which body parts are the most important for gait recognition and according to the findings, the arms and feet are the most important locations.

Palavras-chave: Gait Recognition, Biometry, Computer Vision.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Biometry is characterized as a trait or a personal characteristic that persists over time and is capable of identifying an individual. It is used on many applications, like forensics, access control, workers monitoring and identification of shoplifters.

Biometrics such as fingerprint, face and iris are the most used historically. However, they have limits that prevent them from being applied in some situations. The first is that some of these biometrics require the cooperation of the person to be identified, which is not desired in surveillance scenarios. In addition, these biometrics cannot be extracted remotely, because their information is unavailable or degraded at distance.

An alternative comes with gait, which is a biometry that identifies individuals by the way they walk. Experiments like Murray [1967] and Johansson [1973] serve as its basis, where the former work shows that the movement of limbs and joints gives each person a consistent and unique way of walking, and the latter indicates that it is possible to visually recognize from bright spots the motion of human walking. Different of others biometrics, gait has the advantages of being unobtrusive, available at distances – far before a face can be clearly seen – and can also be collected at night by infrared cameras [Tan et al., 2006].

The features of gait recognition are created following two approaches, model-based and model-free, where the former is characterized by modeling the human body structure or motion and the latter represents the human gait as a whole, without concerning the underlying structure of body or movement. As the methods of model-free approach are simpler and present a lower computational requirements compared with the model-based ones (modeling the pattern of human motion is a costly operation), the former approach is more common on gait literature. However, model-free works are limited by confounding gait information with the appearance present on silhouettes, and for this reason just model-based methods are developed on this thesis.

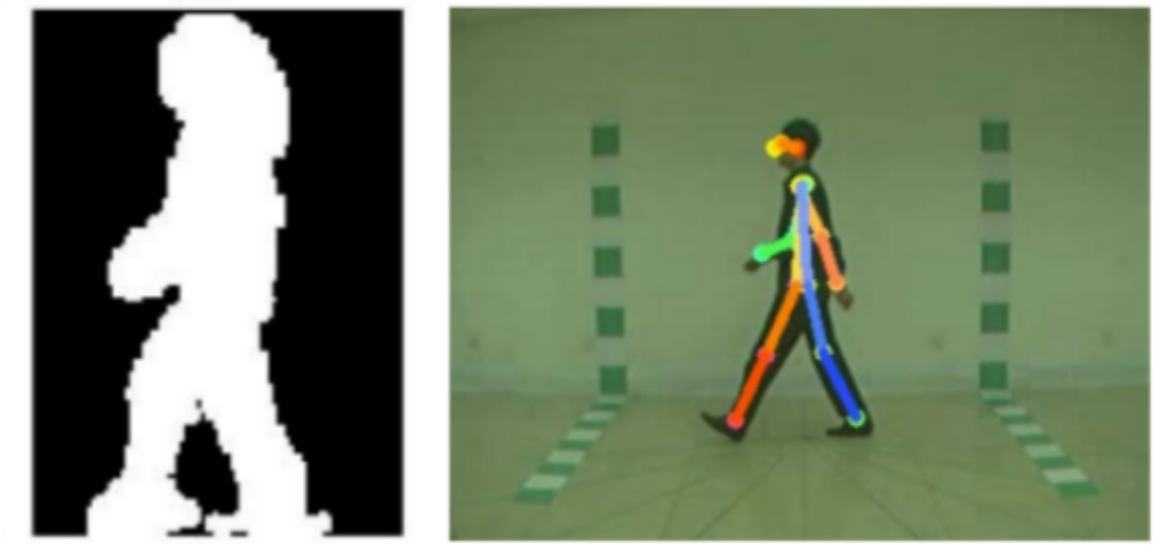

Figure 1.1. Silhouettes (left) are the most popular representation for model-free approach, while poses (right) are the most common for model-based.

The skeleton of poses and silhouettes are the most popular representation for model-based and model-free approach respectively, being represented on Figure 1.1. After their creation, they are passed to classifiers for recognition. Some classifiers used on literature are Euclidean distance [Liu and Sarkar, 2004], Hidden Markov Models [Chen et al., 2006], Linear Discriminant Analysis [Boulgouris and Chi, 2007], and more recently, deep learning methods [Chao et al., 2019; An et al., 2020; Elharrouss et al., 2020].

Although gait has greatly evolved with the use of deep learning, there are still some challenges that limit its application. Some of them are variation on carrying, clothing and view: a person walks differently when it carries different weights and its appearance is also affected, impacting the methods of both approaches; variation on clothing affects recognition the same way, but in a more accentuated manner; and variation on view transforms the appearance of the individuals and their direction of movement. This work aims to improve the results of gait recognition in these conditions, using pose as feature and creating two gait recognition methods.

PoseDist is the first method, following the premise that the gait movement is a signal with sufficient discrimination for recognition. Without using machine learning algorithms, PoseDist only fuses two signal processing methods - Subsequence Dynamic Time Warping and Euclidean distance - to recognize the probe by comparing its signals with those in the gallery. As it is not based on machine learning techniques, PoseDist is a ready-to-use method that does not require training, unlike the works on literature.

It also has the advantage of being based on pose estimation, allowing its application in environments where most works – which have the silhouette as a resource – cannot be applied.

The second method, called PoseFrame, considers that the pose in each frame has enough information for discrimination. This way, different of the works from literature, that train a classifier for a complete walking sequence or cycle, PoseFrame innovates by using just the information of individuals frames to train a multilayer perceptron for recognition. Giving that the focus is to classify an entire walking sequence, the output of the multilayer perceptron is aggregated by majority voting over multiple frames to obtain a final result. Compared with other methods, PoseFrame gains in relation to its simplicity, by using simple features and a very small neural network for training and inference; and given that it uses individuals frames for classification, PoseFrame can also be applied on sequences with occlusion by just ignoring the frames with invalid poses.

Both methods are tested on CASIA Dataset A [Wang et al., 2003] and compared with existing works. Although PoseDist neither uses machine learning techniques nor deep learning-based approaches, it can achieve similar accuracy to the best works on lateral and oblique views and the same accuracy of the best work on the frontal view. PoseFrame improves the accuracy of PoseDist on all views, having state-of-the-art results on this dataset for model-based works. Experiments with PoseFrame are also carried out on CASIA Dataset B [Yu et al., 2006] (PoseDist is not evaluated on this dataset due to its algorithmic complexity). A cross-view experiment is performed following the protocol of Chao et al. [2019], showing that PoseFrame is better than existing works on the small-training configuration with normal sequences. In other configurations GaitSet [Chao et al., 2019] is better, but this thesis shows that GaitSet uses the shape of silhouettes, relying on features with information not directly related to gait. The same-view protocol from Liao et al. [2020] is also evaluated and PoseFrame is compared with other works achieving state-of-the-art results. Finally, an ablation study is also performed to understand the role each body plays in gait recognition.

1.1 Motivation

Surveillance systems are increasingly common in our lives, being present in airports, government buildings, commercial locations and other places. These systems are mostly operated by humans, but studies show that in a short time the concentration of human operators is lost, as this activity is routine and monotonous [Smith, 2004]. Because

of this, these systems are being automated and artificial intelligence technologies are applied.

The recognition of people by biometrics is an important step in identifying the actors in surveillance scenes. Face is a popular biometry, however it cannot be identified at distance and in places with low lighting. Also, masks are often used in criminal actions and facial information is not available. In these cases, gait can be considered the ideal biometry to use, as it is consistent, non-obstructive, and available at distance.

Due to the mentioned aspects and the advantages of gait as biometry, this area has received many contributions in the last decade [Bashir et al., 2010b; Zheng et al., 2011; Iwama et al., 2012]. However, as many problems have not been solved, gait recognition has much room for further advances. Model-free works that use silhouettes are more explored because the creation of model-based representations by modeling the human movement is expensive computationally, and it is just becoming more common in recent years after the creation of efficient pose estimators [Cao et al., 2018; Xiu et al., 2018; Sun et al., 2019]. The problem is that the model-free representations carry appearance information, and we believe that gait can only evolve using features that are related to the human movement. For this reason, only model-based methods are developed in this thesis with the use of recent pose estimators.

1.2 Contributions

This work contributes to the gait literature as follows: the creation of a method based on signals called PoseDist, which does not require training and can be applied in challenging environments where silhouettes are not easily extracted; the development of PoseFrame, which achieves state-of-the-art results for model-based works, using a much simpler neural network than those found in the literature; the conduction of the first model-based ablation study that aims to identify which locations of pose are the most important for gait recognition.

During the development of this work, a technical paper entitled *"Gait Recognition Using Pose Estimation And Signal Processing"* containing contributions for this thesis was published at Iberoamerican Congress on Pattern Recognition (CIARP) [de Lima and Schwartz, 2019], presenting the PoseDist method. Also, the journal paper *"Simple and Efficient Pose-based Gait Recognition Method for Challenging Environments"* was published at Springer Pattern Analysis and Applications Journal (PAAA) [de Lima et al., 2020], presenting the PoseFrame method and performing the ablation study presented in this thesis.

1.3 Work Organization

The remaining of this work is organized as follows. Chapter 2 introduces some background concepts that are useful to understand the proposed methods. Chapter 3 provides a review of gait recognition, indicating the results obtained by model-free and model-based approaches and presenting the novelty of this work. Chapter 4 presents PoseDist and PoseFrame methods, discusses the extraction and normalization of poses from walking sequences, and gives the details of the algorithms used for classification. Chapter 5 compares the methods with existing works and performs an ablation study to assess which parts of the body are most important for gait recognition. Finally, Chapter 6 provides conclusions and guidelines for future works.

Chapter 2

Background Concepts

This chapter contains background concepts that are useful to understand the methods proposed in this thesis. Section 2.1 presents the Subsequence Dynamic Time Warping algorithm and its cost function, which is used to compare the alignment of two signals on PoseDist method proposed in Chapter 4. Section 2.2 presents the multilayer perceptron architecture used on PoseFrame (method also proposed in Chapter 4) and how it is trained, also showing the forward and backpropagation steps. Finally, Section 2.3 describes briefly the pose estimator used on both methods of this thesis to extract features of gait.

2.1 Subsequence Dynamic Time Warping

Because the distance of body parts to the neck in a walking movement is a time-varying value, the proposed PoseDist method represents gait as signals, which generally have different sizes and start at different phases in a walking cycle. For these reasons, the gait sequences cannot be directly compared using a norm distance, requiring a function that also considers their alignment. The solution is the use a variant of Dynamic Time Warping (DTW) [Müller, 2007], called Subsequence Dynamic Time Warping (SDTW), that is able to calculate the cost of alignment of two sequences.

SDTW works by comparing two sequences: $X = (x_1, x_2, ..., x_N)$ of length N and $Y = (y_1, y_2, ..., y_M)$ of length M, where $N \leq M$. X can be warped and the first and the last elements of Y can be ignored (Figure 2.1). The sequences X and Y are represented by features sampled at equidistant points in time and a cost $c(x_n, y_m)$ is defined to compare the features x_n and y_m, for $1 \leq n \leq N$ and $1 \leq m \leq M$, being small if x_n and y_m are similar to each other, and large otherwise.

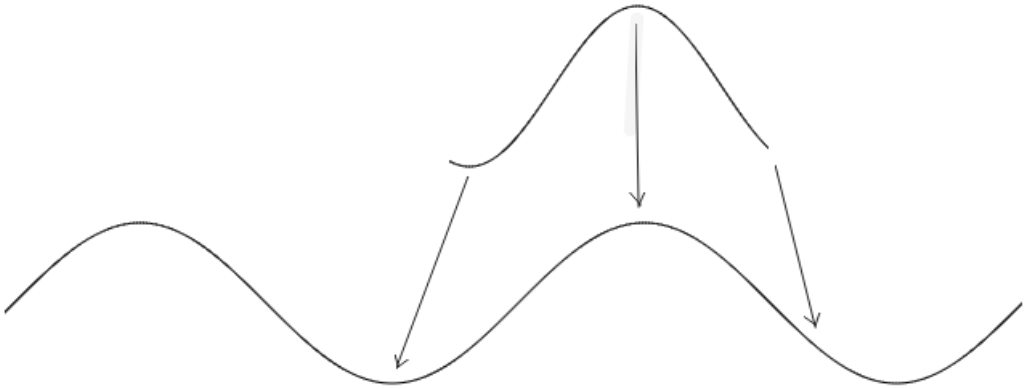

Figure 2.1. SDTW searches for an optimal alignment of two sequences (the image was created in this work). The arrows represent the correspondences on the two signals. The start (on the left of the first arrow) and the end (right of the last arrow) of the second signal are ignored.

A possible alignment of X and Y is represented by a sequence $p = (p_1, ..., p_L)$ of length L, with $p_l = (n, m) \in [1 : N] \times [1 : M]$. p is monotonic ($n_i \leq n_{i+1}$ and $m_j \leq m_{j+1}$), meaning that the alignment follows just one direction on X and Y; $p_1 = (1, m)$ for $m \in [1 : M]$ and $p_L = (N, m)$ for $m \in [1 : M]$, because the path p must start and end at the first and last elements of X; and $p_l - p_{l-1} \in \{(1, 0), (0, 1), (1, 1)\}$ for $l \in [1 : L]$, forcing that no element of X and Y is ignored while the signals are being aligned.

The goal is to find the lowest cost t_p for a path p that ends in (N, M), where $t_p = \sum_{(i,j)\in p} c(x_i, y_j)$. To this end, SDTW creates a matrix D to save the optimal cost t_p^* of alignment, associating $D(n, m)$ with the cost t_p^* of the optimal p^* ending in (n, m). For the first column, the values of D can be easily computed by walking on each element of X and comparing with y_1, defining $D(n, 1) = \sum_{k=1}^{n} c(x_k, y_1)$ for $n \in [1 : N]$. D can also be easily computed for the first line, by comparing the first element of X with the m^{th} element of Y, having $D(1, m) = c(x_1, y_m)$ for $m \in [1 : M]$ (this calculation assumes that the first $m - 1$ elements of Y are not used on the alignment). The value of D at position (n, m) can be found by iterating on D line by line, taking the values on the neighboring positions ($D(n - 1, m - 1)$, $D(n - 1, m)$ and $D(n, m - 1)$) and adding the cost $c(x_n, y_m)$ of transition to $D(n, m)$ (except on the transition from $D(N, m - 1)$ to $D(N, m)$, where the last positions of Y are ignored on the alignment). This is presented below:

$$D(n,m) = \begin{cases} \min\{D(n-1,m-1) + c(x_n, y_m), D(n-1,m) + c(x_n, y_m), D(n,m-1)\}, \\ \qquad \text{for } n = N, 1 < m \leq M \\ \min\{D(n-1,m-1), D(n-1,m), D(n,m-1)\} + c(x_n, y_m), \\ \qquad \text{for } 1 < n < N, 1 < m \leq M. \end{cases} \tag{2.1}$$

Using Equation 2.1, it is possible to calculate the cost of aligning two signals without being influenced by the size of their sequences. In gait application, if two signals can be aligned at a low cost on D, they must be from the same person. This fact is used on the PoseDist algorithm on this thesis for comparing the signals of probe and gallery and give a result for recognition.

2.2 Multilayer Perceptron

A multilayer perceptron is a neural network composed of layers l^i for $1 \leq i \leq L$, with N^i neurons on each layer l^i (Figure 2.2). The layers are fully-connected, meaning that there is a connection to every pair of neurons of neighboring layers. Each neuron on layer l^i is represented by n^i_j, with $1 \leq j \leq N^i$. There is an input layer l^1, where each neuron receives a value from the environment, and an output layer l^L, whose neurons represent the values outputted to the environment. A matrix W^i also exists for each layer in $2 \leq i \leq L$ (only the input layer l^1 does not have a corresponding matrix), where each position w^i_{jk} represents a value of influence, that multiplies the value generated on n^{i-1}_k before it reaches the neuron n^i_j.

The multilayer perceptron is trained with the steps of forward propagation, back-propagation and weights update. In forward propagation, the values inserted on the input neurons flow toward the outputs, passing through internal neurons. The value o^{i-1}_k transmitted from neuron n^{i-1}_k is multiplied by w^i_{jk} before it reaches n^i_j. For the input neurons, o^1_i is the same value inserted on the i-th neuron by the environment. The other neurons sum the values obtained from the previous layer $s^i_j = \sum_{k=1}^{N^{i-1}} w^i_{jk} o^{i-1}_k$ to generate $o^i_j = f^i_j(s^i_j)$, basing on its activation function f^i_j. The values o^L_i of the output neurons are compared with the expected results on a cost function $c(\vec{e}, \vec{o})$, where $\vec{o}$ is the vector with the output values and $\vec{e}$ is the vector with the expected value e_i for the neuron n^L_i.

The objective of training is to find the values w^i_{jk} that minimize the cost function c, which are updated based on their gradients. This way, $\partial c / \partial w^i_{jk}$ must be calculated:

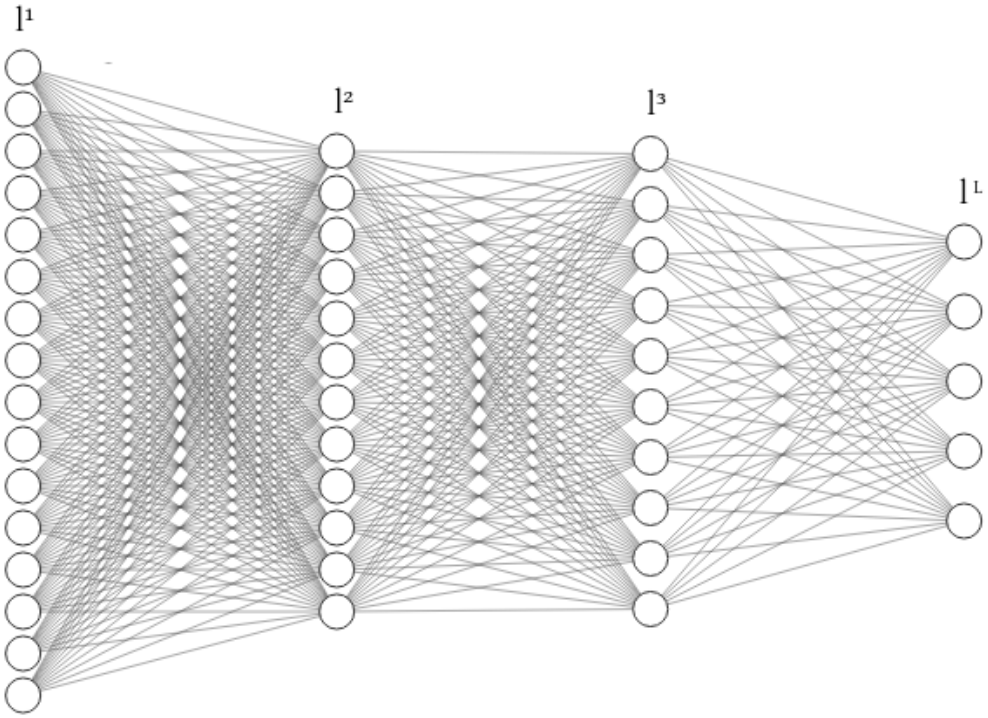

Figure 2.2. Example of multilayer perceptron (image generated from http://alexlenail.me), with input layer l^1, that receives values from the environment; and output layer l^L, whose resulting values are outputted to the environment. All layers are fully-connected.

$$\frac{\partial c}{\partial w_{jk}^i} = \frac{\partial s_j^i}{\partial w_{jk}^i} \cdot \frac{\partial o_j^i}{\partial s_j^i} \cdot \frac{\partial c}{\partial o_j^i} = o_k^{i-1} \cdot \frac{\partial o_j^i}{\partial s_j^i} \cdot \frac{\partial c}{\partial o_j^i},$$

where $\partial o_j^i / \partial s_j^i$ is dependent of the derivative of f_j^i for neuron n_j^i, so it can easily be acquired. $\partial c / \partial o_j^i$ is calculated directly on the output neurons by the derivative of the cost function; and for the other neurons,

$$\frac{\partial c}{\partial o_j^i} = \sum_{k=1}^{N^{i+1}} \frac{\partial s_k^{i+1}}{\partial o_j^i} \cdot \frac{\partial o_k^{i+1}}{\partial s_k^{i+1}} \cdot \frac{\partial c}{\partial o_k^{i+1}} = \sum_{k=1}^{N^{i+1}} w_{kj}^{i+1} \cdot \frac{\partial o_k^{i+1}}{\partial s_k^{i+1}} \cdot \frac{\partial c}{\partial o_k^{i+1}}.$$

As $\partial c / \partial o_j^i$ depends on $\partial c / \partial o_k^{i+1}$ from the neurons of the next layer, the information for retrieving the gradients of the neurons flows backwards, in a step called backpropagation. First $\partial c / \partial o_j^L$ output values are calculated directly by the derivative on the cost function, and for the neurons of other layers, $\partial c / \partial o_j^i$ is calculated using the partial derivatives $\partial c / \partial o_k^{i+1}$ of the next layer, multiplying them by $w_{kj}^{i+1} \cdot \partial o_k^{i+1} / \partial s_k^{i+1}$. Having $\partial c / \partial o_k^{i+1}$, the gradient $\partial c / \partial w_{jk}^i$ is easily calculated by multiplying this value with the

derivative of activation function f_j^i and o_k^{i-1} generated on neuron n_k^{i-1}.

With this information, the next step updates the weights. There are multiple methods of update that use gradient [Bottou, 2012; Zeiler, 2012; Dozat, 2016]. However, just the Adam optimizer [Kingma and Ba, 2014] is presented here, given that just this method is used on this thesis. For its usage, the parameters α (step size), $\beta_1 \in [0, 1)$ and $\beta_2 \in [0, 1)$ (exponential decay for the moment estimates) are defined, and ϵ (update parameter) is fixed as 10^{-8}. Two variables are also created for each neuron n_{jk}^i and set to 0 initially: m_{jk}^i that will save the first moment of the gradient $\partial c/\partial w_{jk}^i$ and v_{jk}^i that will save the second moment. A counter t is also created being incremented on each iteration of training. These variables are updated at each iteration by:

$$m_{jk}^i \leftarrow \beta_1 \cdot m_{jk}^i + (1 - \beta_1) \cdot \frac{\partial c}{\partial w_{jk}^i}$$

$$v_{jk}^i \leftarrow \beta_2 \cdot v_{jk}^i + (1 - \beta_2) \cdot \left(\frac{\partial c}{\partial w_{jk}^i} \right)^2$$

$$\hat{m_{jk}^i} \leftarrow m_{jk}^i / (1 - \beta_1^t)$$

$$\hat{v_{jk}^i} \leftarrow v_{jk}^i / (1 - \beta_2^t)$$

$$w_{jk}^i \leftarrow w_{jk}^i - \alpha \cdot \hat{m_{jk}^i} / (\sqrt{\hat{v_{jk}^i}} + \epsilon),$$

where β_1^t and β_2^t are β_1 and β_2 to the power t, and $\hat{m_{jk}^i}$ and $\hat{v_{jk}^i}$ are temporary variables that compute bias-corrected first and second moment estimate, respectively.

The variable t is incremented and the training continues, only stopping after convergence or if a predefined condition (like maximum number of iteration) has been met. After it, the multilayer perceptron is ready to be evaluated or used for inference on video sequences to identify individuals by their gait.

2.3 Pose Estimation

The gait recognition methods developed in this thesis use coordinates of body parts extracted from pose estimation [Cao et al., 2017, 2018]: PoseDist uses this information to create signals of gait movement, while PoseFrame uses these coordinates as the features of a multilayer perceptron employed for recognition.

To create pose estimators, two types of information are extracted from the frames. The first information is the confidence maps, which are defined by a Gaussian function centered in an annotation, indicating the pixels on the image containing a body part of some person (Figure 2.3). The maximum value of the confidence maps for each body

part is used as the ground-truth and non-maximum suppression on confidence maps is performed to obtain body part candidates. Given a set of detected body parts, the second type of information is extracted from the frame to indicate if each limb (a pair of body parts) belongs to the same person, creating a feature vector representation called part affinity fields, which consists of unit vectors on the locations within a defined distance threshold from the line segment that connects the body parts of a limb (Figure 2.4).

Figure 2.3. Example of confidence maps for elbows and shoulders (image from Cao et al. [2017]).

Figure 2.4. Part affinity fields encoding position and orientation of limbs for different people (image from Cao et al. [2017]).

Both confidence maps and part affinity fields are trained by a Convolutional Neural Network (CNN) [Albawi et al., 2017] with two branches – one for each feature –, which uses the results of the last prediction and the original image features on each training stage. Having the information of confidence maps and part affinity fields from the neural network, the poses are then constructed. Non-maximum suppression [Neubeck and Van Gool, 2006] is applied on the confidence maps to obtain candidates of body part detection for multiple people. A graph is created, having the body part candidates as nodes and the possible connections as edges, and weighting each edge by the part affinity aggregate (a function over part affinity fields that measures their association). Finding the optimal matching on this graph is be reduced to the maximum weight bipartite graph matching [West et al., 2001] problem, which is solved by applying the Hungarian algorithm [Kuhn, 1955]. Finally, the pose of multiple people is found by choosing a minimal number of edges to obtain a spanning tree skeleton and determine the matching in adjacent tree nodes. With this information, the connection candidates are obtained and assembled into poses.

PoseDist method of this thesis uses the resulting poses to extract the location of body parts and normalize them by their distance to the neck position on each coordinate. Signals and histograms of movement are created with this information to represent gait features. PoseFrame also normalizes the poses to process the frames individually, also using the confidence of pose extraction to train the multilayer perceptron for gait recognition.

Chapter 3

Related Works

The literature of gait recognition is divided by the way the features are generated from raw data, having methods from model-based and model-free approaches [Lee et al., 2014]. The following sections discuss their characteristics and the main works on each category, also presenting how they try to address the main problems that affect gait recognition.

3.1 Model-free Approaches

According to Wan et al. [2018], model-free works represent human gait as a whole without knowing the underlying structure of the human body. Some of these features are texture information of optical flow [Hu et al., 2012], spatio-temporal histogram of oriented gradients [Kawai et al., 2012], symmetry of human motion [Hayfron-Acquah et al., 2003], Fourier descriptors [Mowbray and Nixon, 2003] and silhouettes [Wang et al., 2003]. This last representation is by far the most used feature, and many works [Zhang et al., 2007; Bashir et al., 2010b; Preis et al., 2012] refer to model-free works as the ones based on silhouettes.

The reason for the popularity of silhouettes can be related to the low computational requirement for their extraction by using background subtraction algorithm [Gross and Shi, 2001; Wang et al., 2003; Sarkar et al., 2005] and the efficiency of the intermediate representations that are created by processing the silhouettes, creating features like Gait Energy Image [Man and Bhanu, 2006], Gait History Image [Liu and Zheng, 2007] and Gait Entropy Image [Bashir et al., 2009]. By using Gait Energy Image (GEI), which is just the mean of silhouettes over a gait cycle, many works achieved impressive results on gait recognition.

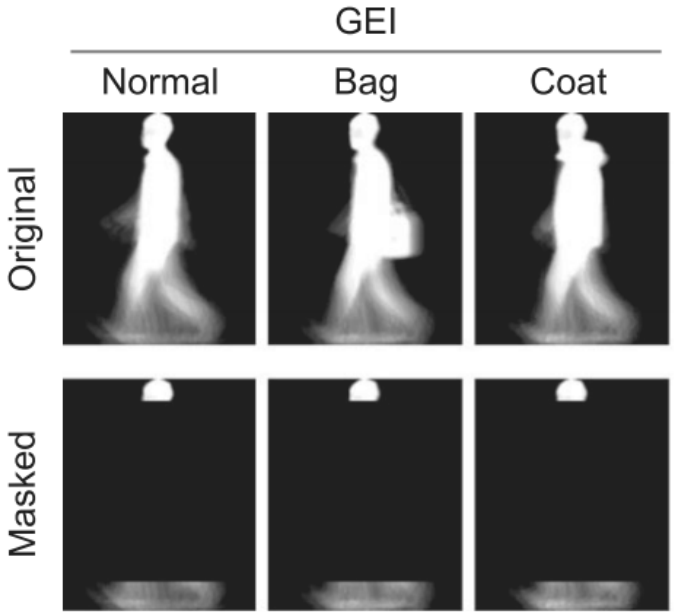

Figure 3.1. According to Isaac et al. [2017] the head and the feet are the most important locations inside GEI for gait recognition, due to their robustness on carrying and clothing conditions.

Bashir et al. [2010a] proposed a cross-view approach that does not require prior information regarding the probe angle, using a Gaussian process for view angle estimation and correlation strength for canonical correlation analysis. The algorithm was tested on the CASIA Dataset B [Yu et al., 2006] and performed better than existing works. However, the results still need to be improved, especially when other conditions are tested, such as carrying and clothing. A more efficient work was proposed by Isaac et al. [2017], which employed a genetic template segmentation to select the silhouette parts more appropriate for classification. For each angle, the genetic algorithm finds boundary positions and selects the parts to use, from which a view-estimator can determine the probe angle and select the suitable view-specific classifier for recognition. The results obtained are different from those in this thesis, because according to Isaac et al. [2017], just the head and the feet must be used to improve the results on clothing and carrying condition (Figure 3.1), while this thesis concludes that all body parts must be used and that the arms are also very important. Although this work is not so recent, it still has the best results on same-view experiments on CASIA Dataset B, having accuracy above 92% in all cases.

Because of the success of deep-learning in many applications, current model-free methods are shifting toward using neural networks to directly process the silhouettes of a gait sequence, not relying on intermediate representations. VGR-Net [Thapar et al.,

2018] is one of these methods, where a three-dimensional convolutional neural network (CNN) for multi-view gait recognition is used on stereo images, achieving some of the best results on the normal sequences of CASIA Dataset B. Another work was proposed by Chao et al. [2019] creating the GaitSet method, in which a CNN is used to extract frame-level features from each silhouette of a set independently, achieving some of the best results on the cross-view situation of CASIA Dataset B.

Despite the high accuracy, the model-free methods carry appearance information, which is not directly related to gait. This way, the methods from this thesis are model-based, using coordinates of poses to have only gait information.

3.2 Model-based Approach

Different of model-free, model-based works are characterized by modeling the human body structure or motion. This way, this approach is computationally more expensive, containing fewer works than the model-free approach. Some of these works are Wang et al. [2004], that uses as dynamic feature a tracking operation that calculates joint-angle trajectories of the main lower limbs; and Wagg and Nixon [2004], that uses anatomical data to generate shape models consistent with regular human body proportions and create a prototype adapted to fit each subject. The former was evaluated on CASIA Dataset A [Wang et al., 2003], having an accuracy of 87.5% on lateral view; and the latter on a Southampton HiD [Shutler et al., 2004] database, with accuracy 84% indoors and 64% outdoors.

In the last years, most model-based works are based on poses, creating their pose estimators or using publicly available methods. In the first group, there is Sokolova and Konushin [2018], where a pose estimator based on an optical flow of five regions is created and a residual network is trained for classification. Their method was tested in side-view sequences of TUM-GAID [Hofmann et al., 2014], CASIA Dataset B [Yu et al., 2006] and OU-ISIR Large Population Dataset [Iwama et al., 2012], obtaining rank-1 accuracy of 99.78%, 92.95% and 94.9% in normal gait sequences, respectively. Feng et al. [2016] is another work that created its pose estimator, by using Human3.6M database [Ionescu et al., 2013]. After extracting the heat-map of pose estimation, this information is fed to a long-short term memory (LSTM) to classify normal gait sequences from different angles. Although its results are good when the angles of the probe and gallery are close, in the case where the angles are distant, their method is not efficient. A custom pose estimator and a LSTM are also used in Liu et al. [2016] to create a method that can achieve good results even on the challenging conditions

of clothing and carrying on CASIA Dataset B. However, its results are not good on CASIA Dataset A, having mean accuracy of just 89.2%.

On the group of works that use publicly available pose estimators, there is Liao et al. [2020], which extract pose coordinates using OpenPose, place the neck at the origin of the plane coordinate system and normalize size of the skeleton. After the normalization, it extracts three-dimensional poses and passes them to a convolutional neural network for classification, improving baseline results [Yu et al., 2006]. Other work is Sheng and Li [2020], which proposes a skeleton-based model called Siamese Denoising Autoencoder (Siamese DAE), that uses coordinates from OpenPose. Its method automatically learns to remove noise, recover missing skeleton points and correct outliers in joint trajectories, achieving good results on TUM-GAID.

Different of Wang et al. [2003, 2004]; Wagg and Nixon [2004]; Shutler et al. [2004] and following the other aforementioned model-based works, the methods of this thesis also use features of pose estimation. Similarly to Liao et al. [2020], the coordinates of the neck are placed on the origin and the size of the skeleton is normalized; however, just two-dimensional information is used and deep-learning techniques are not applied. By using only signal processing algorithms and a multilayer perceptron, the methods of this thesis show that the creation of complex deep-learning architectures is not necessary, and that gait can be recognized by simple methods.

Chapter 4

Proposed Methods

Currently, most works use silhouettes as input to extract gait representations [Wan et al., 2018]. However, this representation is limited because it carries appearance information that affects the results on recognition. Furthermore, it is not robust on cases that are common on uncontrolled scenes, such as occlusion and different carrying and weighting conditions. With this in mind, the model-based approach is more promising [Liao et al., 2020]. Therefore, this work proposes a novel gait recognition method that uses features from pose estimation.

In this chapter, the proposed approach for gait recognition based on pose is described. Given a video sequence, pose estimation is first employed to estimate the coordinates of body parts. Then, pose estimation is used as input for two different approaches for gait recognition, PoseDist and PoseFrame.

PoseDist, discussed in Section 4.1, extracts feature descriptors from the pose locations which are then used for nearest neighbor classification with two different distances, namely, SDTW and Euclidean. PoseFrame, described in Section 4.2, takes the body part coordinates as input and first normalizes them using the neck coordinate as reference. These normalized body parts are then presented to a multi-layer perceptron that learns representations for recognition. Finally, each frame is classified individually and temporal aggregation is performed to predict the identity of a probe sample.

4.1 PoseDist

The PoseDist method considers the movement of limbs to be a signal that can be used to differentiate individuals on a gait sequence. This way, it extracts signals and histograms of movements from pose features by calculating the distance of body parts to the neck position (Figure 4.1) and processes them using two signal processing meth-

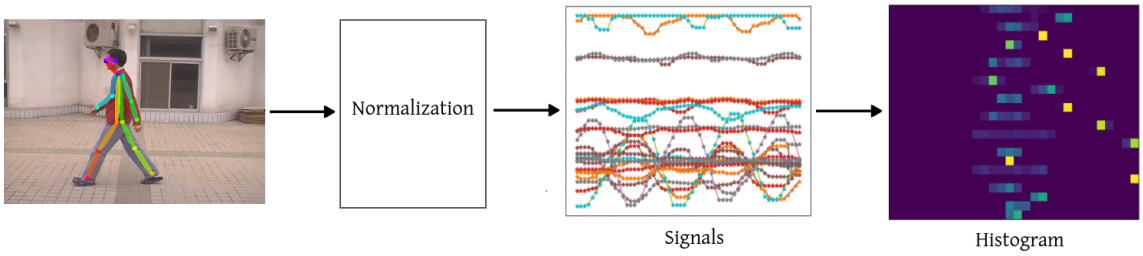

Figure 4.1. The pose coordinates are normalized by their distance to the neck position, creating signals and histograms of movement. These features are used by Subsequence Dynamic Time Warping and Euclidean distance to compare the gait sequences on gallery and probe.

ods - Subsequence Dynamic Time Warping and Euclidean distance -, comparing the sequences of the probe with those in the gallery.

Given the body coordinates extracted using a pose estimator [Cao et al., 2017], $P^i_{b,t}$ is returned for each frame t on gait sequence i and body part indexed by b. $1 \leq t \leq T^i$, where T^i is the total number of frames on sequence i, $P^i_{b,t}$ is a coordinate (x, y) and x and y values are referenced by $P^i_{b,t}.x$ and $P^i_{b,t}.y$, respectively. The coordinates have non-negative values, except when the body part is not found due to occlusion. In this case, they have the invalid values $(-1, -1)$.

The coordinates returned by the pose estimator are from 18 body parts: neck, nose, ears, wrists, elbows, hips, knees, ankles, shoulders and eyes. While ears, eyes and nose are ignored because their positions provide little information for gait recognition, the remaining 13 body parts, indexed by b ranging from 1 to 13, are used. The index b for the neck is 1 and $P^i_{1,t}$ is the neck coordinate at the t-th frame and in the i-th sequence.

The invalid coordinates generated by the pose estimator due to occlusion may interfere with the results. So, to avoid interference, a tracking of invalid body parts is performed, creating the noise indexing N^i for sequence i, that saves the indexes of all body parts whose percentage of invalid coordinates is higher than a defined threshold. It is used on classification to eliminate noisy body parts on the Subsequence Dynamic Time Warping and the Euclidean distance calculation. N^i is defined as

$$N^i = \left\{ b : \frac{\#\{P^i_{b,t}.x = -1 \ \ \forall t \in [1, 2, ..., T^i]\}}{T^i} > \gamma \right\}, \tag{4.1}$$

where γ is a parameter that represents the noise tolerance.

4.1.1 Feature Extraction

Before recognition, PoseDist first requires to extract features. This work proposes the extraction of two feature descriptors, namely, body part signals and movement histograms. The former captures dynamic information, while the latter models static information to be used by PoseDist.

4.1.1.1 Feature based on body part signals

Body part signals are used to capture dynamic information on gait sequence, differentiating individuals by the way their body locations vary on time relative to neck position.

Signals to represent gait movement from sequence i are created, represented by S^i. Each b from 2 to 13 (the neck is used on the formula, but signals for it are not created because they would have only zeros) will generate two lines on S^i: one for its x coordinate value and the other for y. S^i has 24 lines and T^i columns and it is obtained using

$$S^i_{2b-3,t} = \begin{cases} -1, & \text{if } P^i_{b,t}.x = -1 \\ \frac{P^i_{b,t}.x - P^i_{1,t}.x}{max_j P^i_{j,t}.y - P^i_{1,t}.y}, & \text{otherwise} \end{cases} \text{ and} \tag{4.2}$$

$$S^i_{2b-2,t} = \begin{cases} -1, & \text{if } P^i_{b,t}.y = -1 \\ \frac{P^i_{b,t}.y - P^i_{1,t}.y}{max_j P^i_{j,t}.y - P^i_{1,t}.y}, & \text{otherwise} \end{cases} . \tag{4.3}$$

According to the Equations 4.2 and 4.3, the person position on frame is not relevant because the distances are relative to neck position. The denominator of S^i is the vertical distance of the neck to one of the feet, making the signals invariant to the person distance on the video. This makes the assumption that one foot is always on the floor (so maximum y is from it), which is based on the observation that the majority of gait recognition datasets comprise persons in walking pace. However, such assumption might not hold for other conditions, such as someone running.

After the signal creation, each line of S^i has its invalid values removed using linear interpolation. Median filter is also applied to reduce the noise inherent of pose estimation that persists after creating the signals. The window width is selected by randomly choosing ten signals and searching for the size that best decreases the amount of abnormal peaks and valleys without impacting the characteristic of the signals. With this experiment, five is selected as the size of window width. Figure 4.2 shows examples of the signals from three sequences of CASIA Dataset A [Wang et al., 2003] with lateral (0° from the image plane), oblique (45°) and frontal (90°) view, respectively.

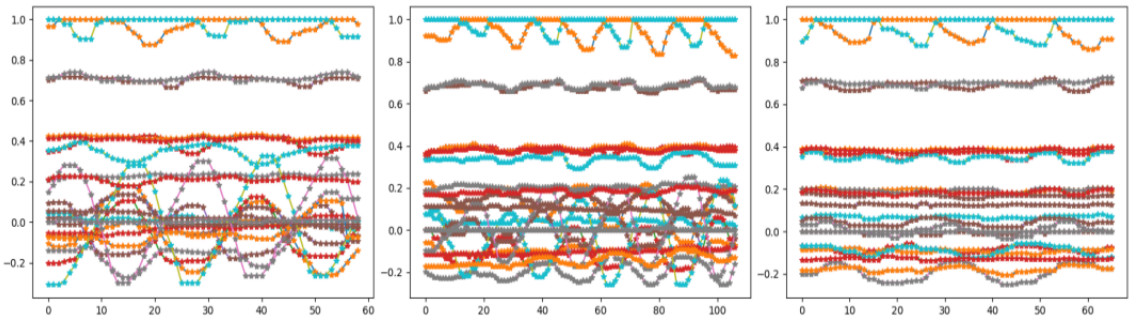

Figure 4.2. Examples of signals from the lateral, oblique and frontal view of CASIA Dataset A, respectively. The lines correspond to x and y distances of body parts to the neck position on coordinates, which vary with time, changing their amplitude. The higher two lines are the y coordinates from the feet, which reach the maximum possible value of 1 when their y coordinates are at the maximum vertical distance from the neck position. The negative values are from x coordinates of the body parts that are on the left of the neck position on the frame. Although the signals of the lateral view are more discriminate (their amplitude varies more comparing with the other views), they are more affected by occlusion (it will be shown in the experimental section).

4.1.1.2 Feature based on movement histograms

After creating and processing S^i, as described in the previous section, the movement histogram H^i is created to also capture the static information on the gait sequence i and improve the recognition results. Because of normalization by the maximum vertical distance (the denominators on Equations 4.2 and 4.3), the values on S^i are limited on the interval $(-1, 1]$, having -1 as lower bound and 1 as upper bound. $(-1, 1]$ is divided on $nVals$ sub-intervals with size $2/nVals$. The movement histogram is then created by having $nVals$ bins that correspond to each of these sub-intervals. Each occurrence of a value belonging to a sub-interval on S^i increments the corresponding bin on the movement histogram. This is mathematically represented by

$$H^i_{l,j} = \frac{\#\left\{\left\lceil \frac{nVals(S^i_{l,t}+1)}{2} \right\rceil = j \;\; \forall t \in [1, 2, ..., T^i]\right\}}{T^i}, \tag{4.4}$$

for $1 \leq l \leq 24$ and $1 \leq j \leq nVals$. Each value on H^i is also divided by T^i, making the values on histogram invariant to the sequence size.

Increasing $nVals$ makes it easier to differentiate individuals. The problem is that if $nVals$ is extremely high, the bins of H^i will be sparse and the recognition will be affected. Therefore, it is necessary to find an optimum value of $nVals$ that

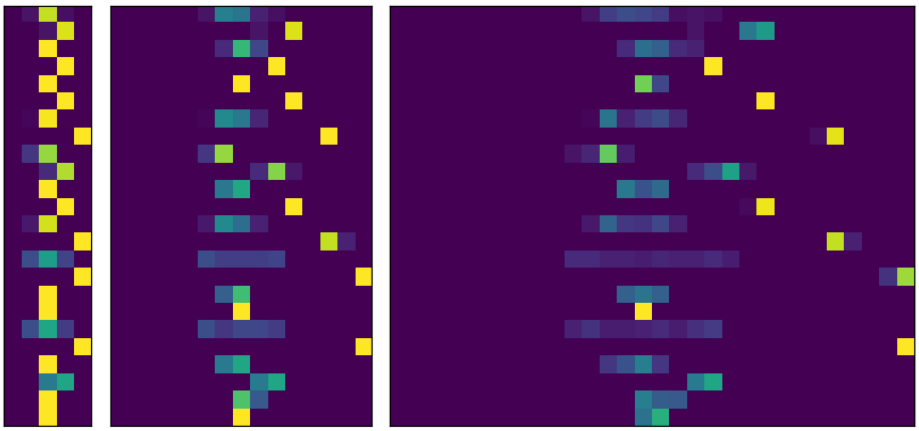

Figure 4.3. Histograms with $nVals$ equal to 5, 15 and 30, respectively. Each line is related to one coordinate of a body part and each column corresponds to a bin on the histogram. The bins with hot colors (yellow) have more elements than the ones with cold colors (blue). The images show that increasing $nVals$ increases the distribution of signal values on the histogram making it more discriminative, but causing sparsity within some bins.

can differentiate individuals without decreasing recognition (see experimental results section). Figure 4.3 shows movement histograms for different values of $nVals$.

4.1.2 Gait Recognition

This section presents the two methods used for recognition and their fusion. They use the features of body part signals and movement histograms presented on the previous sections and find their parameters empirically. In this way, no machine learning techniques are applied.

The individuals are divided in the gallery set, that contains the features extracted from each known person g; and the probe set, whose each person p is unknown and will have its features compared with those on the gallery. The goal is to find the person g from the gallery that minimizes the cost functions $D^{p,g}_{sdtw}$, $D^{p,g}_{dist}$ or $D^{p,g}_{fusion}$ (these functions are defined below) for the person p of probe. The union operation is applied on the indexes of body parts on N^p (noise indexing of person p) and N^g (noise indexing of person g), creating $N^{p,g}$. S^{i*}, S^{p*}, H^{i*} and H^{p*} are created from S^g (signals of person g), S^p (signals of person p), H^g (histogram of person g) and H^p (histogram of person p), removing the lines corresponding to the body parts indexed on $N^{p,g}$.

The following paragraphs present how $D^{p,g}_{sdtw}$ is calculated using Subsequence Dynamic Time Warping and $D^{p,g}_{dist}$ using Euclidean distance between the movement histograms, respectively. Finally, the last section discusses the fusion $D^{p,g}_{fusion}$ of the

results.

Subsequence Dynamic Time Warping. On this work, signal matching is applied, using Subsequence Dynamic Time Warping (SDTW) to find S^{g*} from the gallery where a subsequence within S^{p*} is optimally fitted using squared distance. This subsequence is composed of C consecutive columns, where C is the size of a gait cycle, which is delimited by the time when the same foot starts having the maximum y value. In this work, the left foot was used as reference and 26 was found as the value for C. The result of SDTW is also normalized by the number of lines of S^{g*}. This operation is important, giving the variation on number of lines of S^{g*} for different persons from gallery, because of N^g information. The cost function of SDTW is defined in the equation

$$D^{p,g}_{sdtw} = \frac{SDTW(S^{p*}_{:,t_0:t_f}, S^{g*})}{\sqrt{num\_lines(S^{g*})}}. \tag{4.5}$$

The function is only applied to a interval of frames within S^{p*}, ranging from frame $C/2$ to $3C/2-1$ (13 to 38 in this work). It starts on $C/2$ to ignore the initial frames, since their signals are noisy when the person is entering the scene. The interval contains C frames to have the information of a complete cycle.

Euclidean distance. To recognize gait using movement histograms, the two-dimensional H^{p*} and H^{g*} are vectorized and passed as features to a distance function. In this work, Euclidean distance is the only function evaluated, although others are also possible [Qian et al., 2004; de Souza and De Carvalho, 2004; Samworth, 2012]. It is defined as

$$D^{p,g}_{dist} = \frac{\|vec(H^{p*}) - vec(H^{g*})\|}{\sqrt{num\_lines(H^{g*})}}. \tag{4.6}$$

The distance is used to rank the individuals from the gallery based on their distance from the probe. The result is also normalized according to the number of lines on H^{g*} because H^{g*} is impacted by N^g similarly to S^{g*}.

Score fusion. Fusion of score is a common operation used to improve recognition on biometrics applications [Vatsa et al., 2008; Eskandari et al., 2013; Fakhar et al., 2016], and in this work it is used to fuse the results from SDTW and Euclidean distance, creating the PoseDist method.

The score fusion is applied, combining results of the two methods as

$$D^{p,g}_{fusion} = \alpha D^{p,g}_{dist} + (1-\alpha) D^{p,g}_{sdtw}, \tag{4.7}$$

$$\text{with } 0 \leq \alpha \leq 1,$$

in which increasing α favors the results of Euclidean distance and decreasing it favors SDTW.

4.2 PoseFrame

The PoseFrame method considers that the pose in each frame has enough information for discrimination. This way, different of PoseDist and the works from literature, that use a complete walking sequence or a gait cycle as the feature, PoseFrame uses just the information of individual frames to train a multilayer perceptron to recognize gait. This section presents this method and its steps of normalization, learning and recognition.

4.2.1 Feature Normalization

PoseFrame uses OpenPose [Cao et al., 2018] as the pose estimator, which returns a matrix $F = [\mathbf{x}, \mathbf{y}, \mathbf{c}]$ for each frame of a video sequence, where $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{c}$ are column vectors of dimension $(n \times 1)$, with $\mathbf{x} = (x_1, x_2, ..., x_{n-1}, x_n)$, $\mathbf{y} = (y_1, y_2, ..., y_{n-1}, y_n)$, $\mathbf{c} = (c_1, c_2, ..., c_{n-1}, c_n)$ and n as the number of body parts of the pose estimator. For the i^{th} body part, x_i and y_i represent its position on x and y axis, respectively, while c_i is the confidence of the estimator.

Using the matrix F directly is problematic, because the $\mathbf{x}$ and $\mathbf{y}$ coordinates on a video sequence can be different while the poses are similar. It can occur when a person walks on the scene or get closer to the camera, as illustrated on Figure 4.4. Thus, it is necessary to process F and create a representation where the coordinates of similar poses are close to each other. Two processing operations are performed to this end. The first operation uses the neck as reference point and makes the position of all body parts equivalent to its distance from the neck. Using *neck* as the index for neck, this operation creates $F^{(1)} = [\mathbf{x}^{(1)}, \mathbf{y}^{(1)}, \mathbf{c}]$, where $\mathbf{x}^{(1)} = (x_1 - x_{\text{neck}}, x_2 - x_{\text{neck}}, ..., x_{n-1} - x_{\text{neck}}, x_n - x_{\text{neck}})$ and $\mathbf{y}^{(1)} = (y_1 - y_{\text{neck}}, y_2 - y_{\text{neck}}, ..., y_{n-1} - y_{\text{neck}}, y_n - y_{\text{neck}})$. Afterwards, the second operation is performed to normalize the distances. It is done calculating the maximum vertical difference $y^{\max} = \max(\{y_i^{(1)} : 1 \leq i \leq n\})$ and producing $F^{(2)} = [\mathbf{x}^{(2)}, \mathbf{y}^{(2)}, \mathbf{c}]$, with $\mathbf{x}^{(2)} = \mathbf{x}^{(1)}/y^{\max}$ and $\mathbf{y}^{(2)} = \mathbf{y}^{(1)}/y^{\max}$.

The final representation $F^{(2)}$ has the desired characteristic of similar poses having close coordinates. This way, $F^{(2)}$ is vectorized and the entries corresponding to the neck are removed, because they would always be filled with zeros, resulting in a vector $\mathbf{v}$ to be used as feature for the classifier presented on the next section.

It can be noted the described processing step is similar to the one used on PoseDist, but as PoseFrame processes gait information per-frame basis and not rely

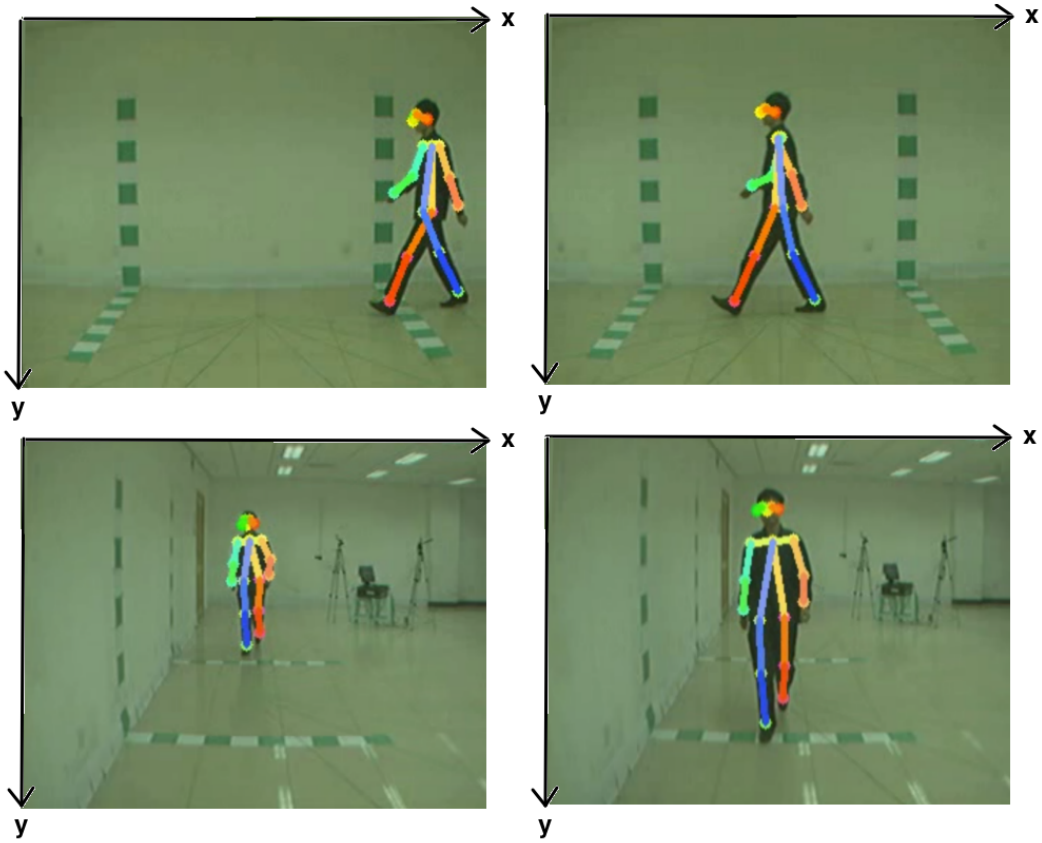

Figure 4.4. The poses from the first and the second image are almost similar, although the **x** location of their body parts are very different. It also occurs with the third and fourth image, which also have differences on their **y** location. This situation evinces the need to process F and create a better representation for the pose coordinates.

on features to represent a whole walking sequences, signals are not created after the normalization. Besides that, noise is not indexed as in PoseDist, because neural networks (the classifier used on PoseFrame is a multi-layer perceptron) are known for their robustness to noise.

4.2.2 Learning and Recognition

The gait methods on literature generally use features extracted from a walking cycle or a video sequence; but this work, instead, make use of information of individual frames for training and inference. For each frame, the feature vector **v** is extracted and used as the input of a neural network, whose output is represented by an one-hot encoding corresponding to the person who walked on the scene.

The neural network employed is a shallow multi-layer perceptron (MLP), com-

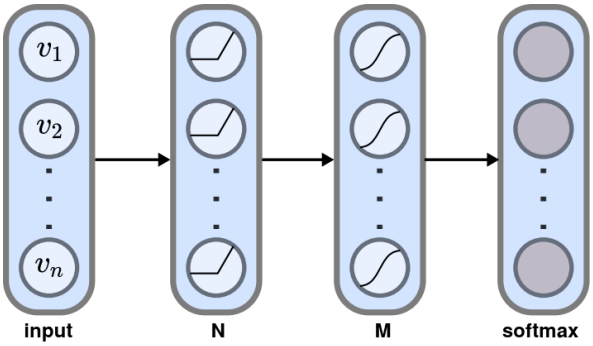

Figure 4.5. PoseFrame architecture, which is a multi-layer perceptron comprising of an input layer, ReLU and sigmoid hidden layers, and a softmax output layer.

prising only three dense layers, where the first layer has N neurons with rectified linear unit [Nair and Hinton, 2010] activation, the second has M sigmoid [Han and Moraga, 1995] neurons and the last layer is a softmax output (Figure 4.5). N and M are determined on the experimental section.

Although the neural network classifies individual frames, gait recognition applications are focused on a whole walking sequence. This way, all $\mathbf{v}$ vectors of a gait sequence are classified and temporal aggregation based on majority voting is performed on the outputs for a final result. During tests, the results of the multilayer perceptron on the sequences are used to verify the efficiency of this model on gait recognition application.

Chapter 5

Experimental Results

This chapter describes the experiments performed using PoseDist and PoseFrame methods on CASIA Dataset A and CASIA Dataset B, focusing on evaluating these methods in challenging conditions. It starts by presenting these datasets in Section 5.1, showing their challenges for gait recognition and the number of individuals and walking sequences they contain. Section 5.2 describes the experiments of PoseDist on CASIA Dataset A, showing the results obtained from different parameters of the methods and comparing the results with existing works. Section 5.3 presents the experiments of PoseFrame on CASIA Dataset A and CASIA Dataset B and discusses the ablation study that was employed to evaluate which body parts are more important for gait recognition. Finally, Section 5.4 discusses the overall results of the proposed methods and how they relate to the state-of-the-art works.

5.1 Datasets

The proposed approaches require color images to estimate pose skeletons, which hinders the usage of large datasets (such as the OU-MVLP [Takemura et al., 2018]), that contain only information of silhouettes. Thus, the experiments are conducted on two publicly available datasets which provide images, being them CASIA Dataset A [Wang et al., 2003] and the CASIA Dataset B [Yu et al., 2006], which are described in the following subsections.

5.1.1 CASIA Dataset A

CASIA Dataset A is a gait dataset containing 20 individuals, whose walking sequences are recorded outdoors in three different views: lateral (0° from the image plane), oblique

(a) Lateral (b) Oblique (c) Frontal

Figure 5.1. CASIA Dataset A has sequences from 20 individuals on three different views (images from http://www.cbsr.ia.ac.cn). On the lateral view (a), the individuals walk parallel to the camera plane, in oblique (b) they walk with an angle of 45° with the plane, and in frontal view (c), they walk perpendicular to the camera.

(45°), and frontal (90°). Each individual has four gait sequences for each view, where two of these sequences have the same walking direction, and on the other half, the walking direction is reversed. For instance, when the individual is walking on the lateral view, the direction is from left to right on two of the four sequences, and from right to left on the others. The dataset has a total of $20 \times 4 \times 3 = 240$ video sequences.

5.1.2 CASIA Dataset B

CASIA Dataset B [Yu et al., 2006] is a multiview gait database collected in an indoor environment, comprising of 124 individuals. Eleven cameras placed 18° apart record its sequences, making the views range from 0° to 180°. The view 0° (180°) is perpendicular to the image plane and records the individual walking toward (moving away from) the camera; the view 90° records the individuals walking laterally; and the other views record oblique movement (Figure 5.2). For each view, the individuals are recorded in six normal (NM) sequences, two sequences with a coat (CL), and two with bag (BG) (Figure 5.3). It has a total of $124 \times 11 \times 10 = 13640$ video sequences, being one of the largest gait datasets available in the literature.

5.2 PoseDist Evaluation

This section presents the experiments of PoseDist on CASIA Dataset A and compares its results with existing works. For each pair of videos with the same configuration of person, view, and walking direction, one video is used to find the best parameters of the method and the other for evaluation. First, the results obtained by varying the

Figure 5.2. CASIA Dataset B contains 11 different views, ranging from 0° to 180° (images from Yu et al. [2006]).

(a) Normal (NM) (b) Wearing a coat (CL) (c) Carrying a bag (BG)

Figure 5.3. CASIA Dataset B contains three walking conditions for each individual (images from Yu et al. [2006]), which are normal (a), carrying a bag (b) and wearing a coat (c).

parameters are presented. Then they are compared with the state-of-the-art works from both gait recognition approaches (model-based and model-free approach).

5.2.1 Noise Tolerance

The noise tolerance γ defines which body parts will not be used with SDTW and Euclidean distance because of occlusion. This parameter is tested on the SDTW method, varying from 0.05 to 1. According to the results showed in Figure 5.4, occlusion has a great impact on recognition and the best results are found when γ is between 0.05 and 0.2. Given that lower values remove more body parts on calculations and make the computation simpler as consequence, 0.05 has been chosen to be used in the remaining

experiments for PoseDist.

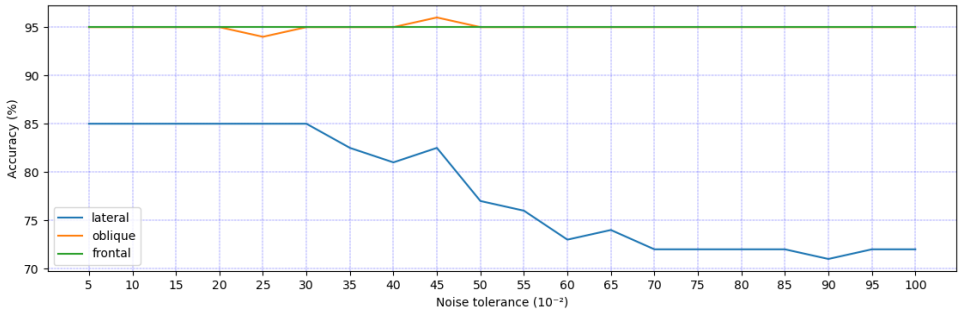

Figure 5.4. Accuracy on D_{sdtw} varying the noise tolerance γ. The best results are found when γ is between 0.05 to 0.2.

5.2.2 Number of Intervals

This experiment evaluates the number of intervals $nVals$ on the movement histogram. It is responsible for the size of intervals, which are used to distinguish gait on sequences. $nVals$ varies from 5 to 100. According to Figure 5.5, the best results are achieved by $nVals$ equals to 85, which maximizes accuracy on D_{dist}. This value is used in the remaining experiments.

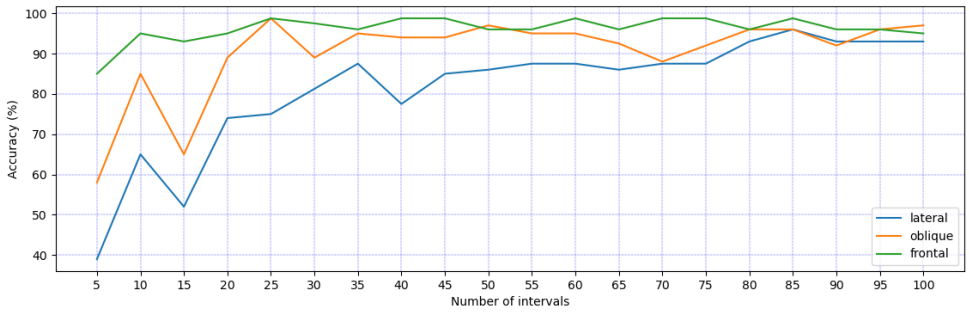

Figure 5.5. Accuracy on D_{dist} varying the number of intervals $nVals$. The value 85 gives the best results.

5.2.3 Weight of Score Fusion

The score weight α is used to fuse results from SDTW and Euclidean distance. It ranges from 0 to 1 and, when α presents higher values, it favors Euclidean distance and when it has lower values, SDTW is favored. In the experiments α varies from

0.05 to 1. According to Figure 5.6, the best results are achieved with α equals 0.75, meaning that the Euclidean distance is more important than SDTW.

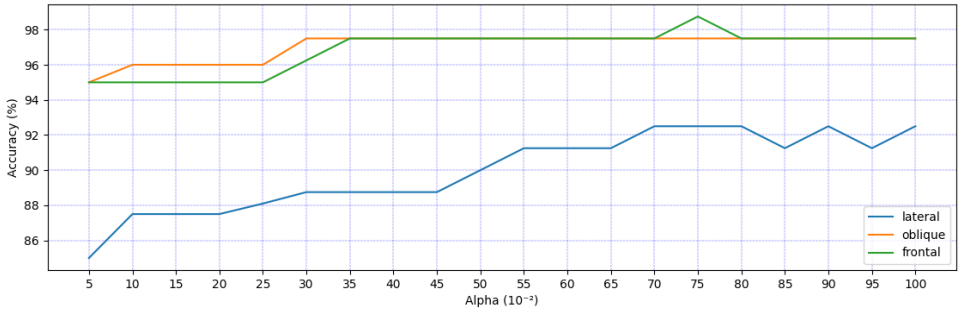

Figure 5.6. Accuracy on D_{fusion} varying the score fusion weight α. The value 0.75 gives the best results.

5.2.4 Comparison with Literature

The best results of PoseDist have been achieved when γ is 0.05, $nVals$ is 85 and α is 0.75, achieving an accuracy of 92.5%, 97.5%, and 98.75% on lateral, oblique, and frontal views, respectively. The lower accuracy on lateral view is probably caused by occlusion, which is more common on that view. Euclidean distance method is more accurate than SDTW, showing that static information is more accurate than dynamic on gait recognition. However, the accuracy increases when dynamic and static gait information from SDTW and Euclidean distance are fused.

The proposed method is compared to others that use the same dataset. It should be noted that different of PoseDist, all these works use the leave-one-sequence-out protocol, where one sequence is left for evaluation, and the training occurs on the others. As PoseDist uses a cost function to compare the sequences from gallery and probe instead of training a descriptor, the protocol described at the start of this section is used instead, being equivalent statistically.

According to the results showed in Table 5.1, PoseDist has some of the best results on lateral, the second-best on oblique, and the best on the frontal view (together with Kusakunniran et al. [2009]). It is also interesting to note that the method is better than Liu et al. [2016], which is a recent model-based method that uses deep learning. These results demonstrate the good performance of PoseDist despite its simplicity, showing the viability of applying signal processing algorithms for gait recognition.

Table 5.1. Rank-1 recognition on CASIA Dataset A, comparing the methods from literature with PoseDist.

	View		
	Lateral	Oblique	Frontal
Liu et al. [2016]	85%	87.5%	95%
Nizami et al. [2010]	**100%**	-	-
Kusakunniran et al. [2009]	**100%**	**100%**	**98.75%**
SDTW distance	85%	95%	95%
Euclidean distance	92.5%	96.25%	97.5%
PoseDist (fusion)	92.5%	97.5%	**98.75%**

5.3 PoseFrame Evaluation

This section evaluates PoseFrame comparing its results with existing works, and also performs an ablation study to assess which body parts are the most important for gait recognition. For each sequence, the poses are extracted using OpenPose and processed as described in Section 4.2.1. The MLP of PoseFrame is trained using cross-entropy loss function with Adam [Kingma and Ba, 2014] optimizer on its default settings ($\alpha = 0.001$, $\beta^1 = 0.9$, $\beta^2 = 0.999$ and $\epsilon = 10^{-8}$) and batch size set to 4000, which corresponds to adding the whole dataset in memory, allowing faster convergence. The network is trained by 512 epochs and fine-tuned by 40 epochs. N (number of neurons on the rectified linear unit layer) and M (the number of neurons on the sigmoid layer) were determined through grid-search and 1024 presented the best results for both parameters.

The evaluation of PoseFrame on CASIA Dataset A follows the leave-one-sequence-out protocol. On CASIA Dataset B, the protocol from Liao et al. [2020] is used on the same-view experiment, consisting of training on half the individuals and using the other half for testing. After the method is trained, fine-tuning occurs in the first four normal sequences of the individuals in the test set, and evaluation is done in the remaining sequences.

Cross-view experiments follow the protocol from Chao et al. [2019] using leave-one-angle-out, which trains on all angles except one, which is left for prediction. Besides that, the dataset is divided into three different configurations: small-sample (ST), medium-sample (MT), and large-sample training (LT). The first 24 individuals are used to train the model on the ST configuration, 62 are used on MT, and 74 on LT. The remaining individuals are used on evaluation, having their first four normal sequences used as the gallery (where fine-tuning occurs) and the rest as the probe.

Table 5.2. Rank-1 recognition on CASIA Dataset A in comparison with model-based approaches.

	View			
	Lateral	Oblique	Frontal	Average
Liu et al. [2016]	85.0%	87.5%	95.0%	89.16%
PoseDist (ours)	92.5%	**97.5%**	98.75%	96.25%
PoseFrame (ours)	**97.5%**	96.25%	**100%**	**97.97%**

5.3.1 Evaluation on CASIA Dataset A

PoseFrame is evaluated on CASIA Dataset A and compared with PoseDist and another model-based method [Liu et al., 2016]. The results are presented in Table 5.2. It can be seen that both approaches yield better results than Liu et al. [2016], despite the latter having a more complex architecture that uses a recurrent layer to model temporal information, while PoseFrame uses a simple temporal aggregation approach and a shallow MLP.

PoseFrame is more accurate than PoseDist on lateral and frontal views, while only 1.25% worse in oblique view. In addition to having better accuracy, PoseFrame is also faster than PoseDist, given that the latter requires the comparison of templates from gallery and probe using Subsequence Dynamic Time Warping, which is a costly operation with a time complexity of $O(NM)$ for sequences of length N and M. This way, PoseFrame is more suitable than PoseDist to be applied on gait recognition, and this fact is valid even on datasets with few walking sequences like CASIA Dataset A.

5.3.2 Evaluation on CASIA Dataset B

This section describes the experiments conducted on the CASIA Dataset B. As the PoseFrame approach performs better in most scenarios and PoseDist is not feasible to be tested on CASIA Dataset B due to its algorithmic complexity, just PoseFrame is evaluated. Its robustness on same-view and cross-view conditions is analyzed. An ablation study is also performed to evaluate the importance of each joint of a pose.

5.3.2.1 Same-view Recognition

Table 5.3 presents the results obtained by PoseFrame when it is evaluated on normal sequences and compares it with recent works that follow the same protocol. According to the table, PoseFrame achieves state-of-the-art accuracy, having the best results for the most angles. It is interesting to note that it improves the results obtained by

Table 5.3. Accuracy of different works on normal sequences under same-view.

Angle\Work	Elharrouss et al. [2020]	Liao et al. [2020]	PoseFrame
0°	94	96.0	**99.2**
18°	95	96.8	**99.2**
36°	97	96.0	**99.2**
54°	97	96.8	**98.4**
72°	**98**	96.0	96.8
90°	**98**	97.6	97.6
108°	98	97.6	**98.4**
126°	98	94.4	**99.2**
144°	**97**	96.8	96.8
162°	95	**97.6**	**97.6**
180°	93	97.6	**99.2**

Table 5.4. Accuracy of PoseFrame and Liao et al. [2020] on carrying and clothing sequences under same-view.

BG sequence	0°	18°	36°	54°	72°	90°	108°	126°	144°	162°	180°	Mean
Liao et al. [2020]	74.2	75.8	77.4	76.6	69.4	70.2	71	69.4	74.2	65.3	60.5	71.3
PoseFrame	**83.1**	**88.7**	**91.1**	**89.5**	**90.3**	**87.0**	**87.0**	**83.9**	**83.9**	**85.5**	**80.7**	**86.4**
CL sequence	0°	18°	36°	54°	72°	90°	108°	126°	144°	162°	180°	Mean
Liao et al. [2020]	**46.8**	48.4	57.3	**61.3**	58.1	**56.5**	**59.7**	**54.8**	**55.7**	**58.1**	39.5	**54.2**
PoseFrame	41.9	**59.7**	**64.5**	**61.3**	**58.9**	50.0	53.2	40.3	45.2	37.9	**45.2**	50.7

Liao et al. [2020], which is a deep-learning model-based work that uses 3D coordinates calculated from the 2D coordinates of OpenPose, showing that it is not necessary to have a complex method to achieve good results on recognition. Because PoseFrame is also better than the recent model-free work of Elharrouss et al. [2020], it can be seen that the developed method is state-of-the-art for same-view recognition on normal sequences for works from both approaches.

PoseFrame is compared further to Liao et al. [2020], evaluating both methods on carrying and clothing condition (Elharrouss et al. [2020] is not evaluated because it does not report its results on these conditions). Table 5.4 shows that the developed method is much better at carrying, increasing the mean accuracy by 15.1 percentage points, but a little worse on clothing, losing 3.5 percentage points.

5.3.2.2 Cross-view Recognition

Table 5.5 presents the results for cross-view recognition, reporting each experimental setting with ST, MT and LT. Results reported by other approaches that follow the same experimental setup [Hu et al., 2013; Kusakunniran et al., 2014; Wu et al., 2016;

Table 5.5. Cross-view recognition on CASIA-B. Average rank-1 accuracies under three experimental settings (ST, MT, LT) using leave-one-angle-out. PoseFrame is compared with ViDP [Hu et al., 2013], CMCC [Kusakunniran et al., 2014], CNN-LB [Wu et al., 2016], AE [Yu et al., 2017], MGAN [He et al., 2019], CNN-3D [Wu et al., 2016], CNN-Ens [Wu et al., 2016] and GaitSet [Chao et al., 2019].

		Gallery	NM#1-4 leave-one-angle-out										
		Probe	0°	18°	36°	54°	72°	90°	108°	126°	144°	162°	180°	mean
ST (24)	NM	ViDP	–	–	–	59.1	–	50.2	–	57.5	–	–	–	–
		CMCC	46.3	–	–	52.4	–	48.3	–	56.9	–	–	–	–
		CNN-LB	54.8	–	–	77.8	–	64.9	–	76.1	–	–	–	–
		GaitSet	**64.6**	83.3	**90.4**	**86.5**	80.2	75.5	80.3	86.0	**87.1**	81.4	59.6	79.5
		PoseFrame (Ours)	62.5	**97.9**	87.5	64.6	**93.8**	**95.8**	**93.8**	**97.9**	70.8	**91.7**	**75.0**	**84.7**
	BG	GaitSet	**55.8**	70.5	**76.9**	**75.5**	69.7	63.4	68.0	75.8	**76.2**	70.7	**52.5**	**68.6**
		PoseFrame (Ours)	52.1	**70.8**	58.3	43.8	**79.2**	**81.2**	**77.1**	**77.1**	66.7	**77.1**	52.1	66.9
	CL	GaitSet	**29.4**	**43.1**	**49.5**	**48.7**	**42.3**	40.3	44.9	47.4	**43.0**	35.7	25.6	40.9
		PoseFrame (Ours)	22.9	29.2	35.4	33.3	39.6	**62.5**	**52.1**	**52.1**	33.3	**43.8**	**33.3**	39.8
MT (62)	NM	AE	49.3	61.5	64.4	63.6	63.7	58.1	59.9	66.5	64.8	56.9	44.0	59.3
		MGAN	54.9	65.9	72.1	74.8	71.1	65.7	70.0	75.6	76.2	68.6	53.8	68.1
		GaitSet	**86.8**	**95.2**	**98.0**	**94.5**	**91.5**	89.1	91.1	95.0	**97.4**	**93.7**	**80.2**	**92.0**
		PoseFrame (Ours)	66.9	90.3	91.1	55.6	89.5	**97.6**	**98.4**	**97.6**	89.5	69.4	68.5	83.1
	BG	AE	29.8	37.7	39.2	40.5	43.8	37.5	43.0	42.7	36.3	30.6	28.5	37.2
		MGAN	48.5	58.5	59.7	58.0	53.7	49.8	54.0	61.3	59.5	55.9	43.1	54.7
		GaitSet	**79.9**	**89.8**	**91.2**	**86.7**	**81.6**	76.7	**81.0**	**88.2**	**90.3**	**88.5**	**73.0**	**84.3**
		PoseFrame (Ours)	45.2	66.1	60.5	42.7	58.1	**84.7**	79.8	82.3	65.3	54.0	50.0	62.6
	CL	AE	18.7	21.0	25.0	25.1	25.0	26.3	28.7	30.0	23.6	23.4	19.0	24.2
		MGAN	23.1	34.5	36.3	33.3	32.9	32.7	34.2	37.6	33.7	26.7	21.0	31.5
		GaitSet	**52.0**	**66.0**	**72.8**	**69.3**	**63.1**	**61.2**	**63.5**	**66.5**	**67.5**	**60.0**	**45.9**	**62.5**
		PoseFrame (Ours)	13.7	29.0	20.2	19.4	28.2	53.2	57.3	52.4	25.8	26.6	21.0	31.5
LT (74)	NM	CNN-3D	87.1	93.2	97.0	94.6	90.2	88.3	91.1	93.8	96.5	96.0	85.7	92.1
		CNN-Ens	88.7	95.1	98.2	96.4	**94.1**	91.5	93.9	97.5	98.4	95.8	85.6	94.1
		GaitSet	**90.8**	**97.9**	**99.4**	**96.9**	93.6	91.7	95.0	97.8	**98.9**	**96.8**	**85.8**	**95.0**
		PoseFrame (Ours)	66.2	93.9	88.5	56.1	79.7	**98.0**	**98.6**	**99.3**	81.8	80.4	70.3	83.0
	BG	CNN-LB	64.2	80.6	82.7	76.9	64.8	63.1	68.0	76.9	82.2	75.4	61.3	72.4
		GaitSet	**83.8**	**91.2**	**91.8**	**88.8**	**83.3**	81.0	**84.1**	**90.0**	**92.2**	**94.4**	**79.0**	**87.2**
		PoseFrame (Ours)	48.6	69.6	56.1	41.9	56.8	**84.5**	80.4	83.1	65.5	58.1	48.0	63.0
	CL	CNN-LB	37.7	57.2	66.6	61.1	55.2	54.6	55.2	59.1	58.9	48.8	39.4	54.0
		GaitSet	**61.4**	**75.4**	**80.7**	**77.3**	**72.1**	**70.1**	**71.5**	**73.5**	**73.5**	**68.4**	**50.0**	**70.4**
		PoseFrame (Ours)	16.2	28.4	21.6	23.6	28.4	50.7	54.1	49.3	28.4	25.0	20.3	31.4

Chao et al., 2019] are also included.

First considering the ST setting, it can be seen that PoseFrame performs better for NM, with an average of 84.7% across all angles. The angles that provide the best results are 18°, 72°, 90°, 108° and 126°. Except for 18°, it is believed it occurs because the lateral view is more robust, and its information can be easier recovered by the surrounding angles. PoseFrame also obtains competitive results in comparison with GaitSet on BG and CL, especially on angles 90°, 108° and 126°. In the MT setting, the proposed method is better on 90°, 108° and 126° in normal sequences (NM) and on 90° in carrying (BG). Similarly, PoseFrame has the best accuracy on these angles in NM and BG in the LT setting. Overall, PoseFrame is not accurate on clothing

Table 5.6. Accuracy of GaitSet using the original and the pose-based silhouettes.

Features	NM	BG	CL
Original silhouettes	95.6	89.0	73.2
Pose-based silhouettes	97.2	82.2	70.0

conditions. The reason for this is that the coat and jacket affect the pose estimation, making it difficult to search for the location of body parts.

In general, GaitSet is more accurate in most of the challenging environments of CASIA Dataset B when compared with PoseFrame. However, it must be noted that the former method uses silhouettes that contain attributes (*e.g.* the shape of the head) not directly related to gait and that are invariant to carrying and clothing condition [Isaac et al., 2017]. To evaluate that, an experiment is performed to verify the performance of GaitSet when it uses features without appearance information. To this end, a new feature called pose-based silhouette is developed on this thesis, being represented as a binary image whose body parts are built joining the coordinates from OpenPose and ignoring the head (Figure 5.7). GaitSet is evaluated following the LT configuration, having as inputs the original silhouettes for one test and the pose-based on another test. Just sequences with 0° are considered because the sequences from other angles contain many poses with occlusion and their pose-based silhouettes could affect the method. The results are presented in Table 5.6, where it is visible that GaitSet is more accurate when the original features are used, containing appearance information. When pose-based silhouettes are used, GaitSet is affected by having results close to the ones obtained by model-based methods (already presented in Table 5.4).

It also shows that the results of PoseFrame can be improved if appearance information is also used, but as the focus of the present work is just using gait information, the methods of this thesis are limited to features from pose estimation. Although only GaitSet is evaluated, it is believed that the conclusions are valid for all model-free works because they also use appearance information.

5.3.2.3 Ablation Study

Experiments are conducted to determine the role each body part plays in model-based gait recognition and different combinations of joints are tested, as depicted in Figure 5.8. The combinations that use joints from just one kind of limb are considered simple, while the others (except the configuration that uses all joints) are considered mixed. The experiments are conducted in the MT setting, training and testing on all angles.

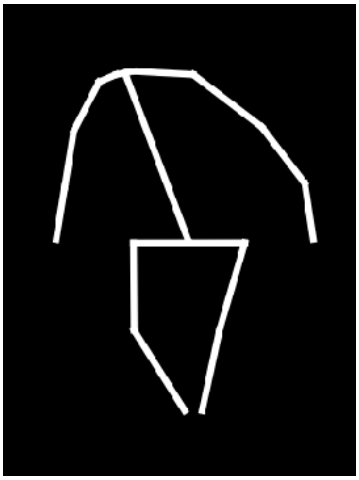

Figure 5.7. Example of a pose-based silhouette.

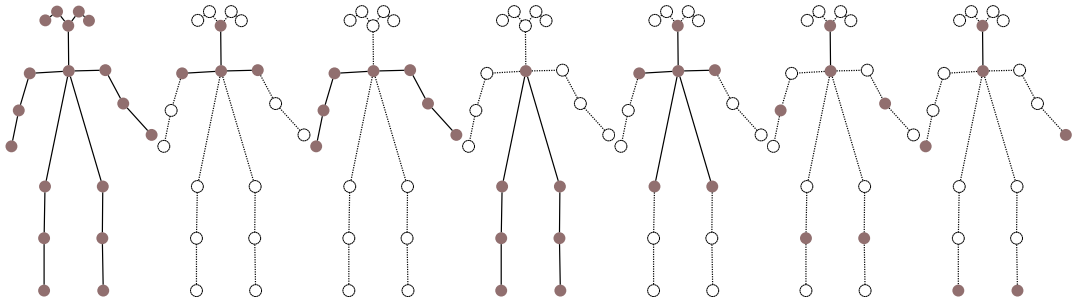

Figure 5.8. Joint configurations of the ablation study. Filled circles represent the joints used: (a) all joints; (b) nose and shoulders; (c) arms (shoulders, elbows, wrists); (d) legs (hips, knee, ankles); (e) beginning-joints (nose, shoulders, hips); (f) middle-joints (nose, elbows, knees); and (g) end-joints (nose, wrists, ankles). The configurations (b), (c) and (d) are considered simple, because they use joints from just one kind of limb, different of (e), (f) and (g), that are considered mixed.

According to the results presented in Table 5.7, for simple configuration, arms are the most important joints for recognition, having an accuracy of 97.65%, which is 9.17 percentage points greater than the one achieved using legs. However, as the individuals of CASIA Dataset B use the arms to carry bags, the legs are less affected on carrying conditions, with an accuracy of 18.04 percentage points higher than the arms. Nose and shoulders are inaccurate compared to other locations, highlighting the importance of limbs on gait recognition.

Regarding mixed configuration, the end-joints are the most robust. It is believed that this is related to the fact these joints have more variation on gait sequences, allowing the methods to have better discrimination. However, they are less accurate

Table 5.7. Rank-1 accuracy of the ablation study, evaluating the importance of joints on recognition.

Joint	NM	BG	CL
a. All joints	99.36	85.91	49.09
b. Nose and shoulders	61.65	44.57	15.76
c. Arms	97.65	49.26	20.89
d. Legs	88.48	67.30	23.24
e. Beginning-joints	86.14	60.77	15.61
f. Middle-joints	96.33	66.78	31.96
g. End-joints	97.21	62.31	43.10

than middle-joints on carrying conditions, because the bags affect the movement of the arms, as discussed in the last paragraph. The beginning joints have the worse results on all conditions, given that they contain less information of movement (nose and shoulders are almost static on gait sequences after normalization). They are also the most affected by the usage of coat/jacket because it is harder to find the correct position of covered hips and shoulders.

The results obtained by this ablation study are different from previous model-free works. Isaac et al. [2017] found that just the location of head and feet on silhouettes must be used to recognize gait on clothing and carrying variations; but differently, the present work shows that recognition is impacted if some parts are removed. According to Sarkar et al. [2005], the lower 30% of the silhouette is the most important location, being responsible for 75% of accuracy on identification. Table 5.7 shows that the arms are also very important, being the only part with accuracy above 90% on normal sequences with both sides and having results above the feet on a single side. These facts show that the conclusions for the model-free approach are not valid on model-based works, evincing the importance of the performed experiment.

5.4 Discussion

PoseDist and PoseFrame were tested on CASIA Dataset A and CASIA Dataset B. The experiments with noise tolerance of PoseDist showed that the inherent noise of pose estimation affects the recognition results. This way, noisy body parts must be filtered, or a method robust to noise, such as the MLP used on PoseFrame, must be used. Besides that, the best results are found using the histogram of movement for PoseDist on CASIA Dataset A, showing that static information is more accurate than dynamic. This fact is reinforced by the results of PoseFrame, which is intrinsically static for not

using any feature of movement.

PoseFrame is much faster than PoseDist, whose complexity is $O(NM)$. Because of this fact, just the former method was tested on CASIA Dataset B. Despite its simplicity, PoseFrame is very accurate compared with existing works. In same-view recognition, PoseFrame achieved state-of-the-art results for most situations, except on some views on clothing condition, where it is worse than the three-dimensional model of Liao et al. [2020]. In the cross-view experiment, PoseFrame had the best results for the ST setting, showing that this method is better than the state-of-the-art when the number of walking sequences for training is limited. For other settings, its results were worse than model-free methods, specially GaitSet. However, another experiment demonstrates that it occurs because these methods use the appearance information of silhouettes, and their accuracy decreases when this information is removed.

Chapter 6

Conclusions

This work presented two approaches for gait recognition based on 2D pose information, namely, PoseDist and PoseFrame. PoseDist is composed of hand-crafted feature descriptors that encode spatial and temporal information for each pose, which is then subsequently presented to Subsequence Dynamic Time Warping and Euclidean distance to compare probes with the gallery. PoseFrame is a multi-layer perceptron (MLP) with only three layers that take as input pose coordinates normalized by the neck coordinate and vertical distance. Classification is then performed per-frame basis, whose predictions are aggregated temporally using majority voting.

Experiments were performed on all views of CASIA Dataset A and the current methods achieved state-of-the-art results for model-based works, in which PoseDist obtained an accuracy of 92.5%, 97.5%, and 98.75% on lateral, oblique, and perpendicular view, respectively; while PoseFrame yielded an accuracy of 97.5%, 96.25%, and 100%.

PoseFrame was evaluated for cross-view recognition on CASIA Dataset B to understand how it performs for unseen angles, and it was observed that it was better for angles closer to 90°, which are the most common on gait literature. The results also showed its efficiency in normal and carrying walking sequences under small-sample training settings, but low accuracy on other others in comparison with existing methods. To justify why it happens, an experiment was performed on GaitSet to show that model-free methods are more robust in other settings because they use the shape of silhouettes besides gait, and when this information is removed they are also affected.

Same-view recognition was also evaluated on CASIA Dataset B using PoseFrame and following the protocol from Liao et al. [2020]. According to the results, PoseFrame achieved state-of-the-art accuracy on normal sequences compared with methods from both approaches. It was further compared with Liao et al. [2020] on other conditions, indicating that PoseFrame is better on recognition by 15.1 percentage points on

carrying, but 3.5 percentage points worse on clothing.

Ablation experiments were also performed, showing the importance of redundancy of joints to avoid the negative influence of object and clothing distractors – such as bags, coats, and jackets. It was seen that end-joints – parts in the body extremities such as wrist and ankle – yield better results since they are more visible even under a jacket or a coat, which allows a more accurate pose estimation. Finally, the results of the ablation experiment were compared with model-free works, showing that the conclusions for one approach are not valid for the other, and thus justifying the importance and novelty of this experiment.

In general, the achieved results from this work and literature show that gait is limited, especially on carrying and clothing conditions, requiring the use of appearance to improve recognition as it is done in model-free works. For this reason, it is intended in a future work to test the use appearance on PoseDist and PoseFrame. Furthermore, as gait was only tested in controlled environments, its results can greatly decrease in real case scenarios, and other biometrics might be necessary to have good recognition. To verify this fact, a new challenging dataset with outdoor sequences will be created to verify the robustness of gait methods in uncontrolled environments and how their recognition is improved using other biometrics.

Bibliography

Albawi, S., Mohammed, T. A., and Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1--6. Ieee.

An, W., Yu, S., Makihara, Y., Wu, X., Xu, C., Yu, Y., Liao, R., and Yagi, Y. (2020). Performance evaluation of model-based gait on multi-view very large population database with pose sequences. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2(4):421--430.

Bashir, K., Xiang, T., and Gong, S. (2009). Gait recognition using gait entropy image.

Bashir, K., Xiang, T., and Gong, S. (2010a). Cross view gait recognition using correlation strength. In *Bmvc*, pages 1--11.

Bashir, K., Xiang, T., and Gong, S. (2010b). Gait recognition without subject cooperation. *Pattern Recognition Letters*, 31(13):2052--2060.

Bottou, L. (2012). Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421--436. Springer.

Boulgouris, N. V. and Chi, Z. X. (2007). Gait recognition using radon transform and linear discriminant analysis. *IEEE transactions on image processing*, 16(3):731--740.

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., and Sheikh, Y. (2018). OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, volume 1, page 7.

Chao, H., He, Y., Zhang, J., and Feng, J. (2019). Gaitset: Regarding gait as a set for cross-view gait recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8126--8133.

Chen, C., Liang, J., Zhao, H., and Hu, H. (2006). Gait recognition using hidden markov model. In *International Conference on Natural Computation*, pages 399--407. Springer.

de Lima, V. C., Melo, V. H., and Schwartz, W. R. (2020). Simple and efficient pose-based gait recognition method for challenging environments. *Pattern Analysis and Applications*, pages 1--11.

de Lima, V. C. and Schwartz, W. R. (2019). Gait recognition using pose estimation and signal processing. In *Iberoamerican Congress on Pattern Recognition*, pages 719--728. Springer.

de Souza, R. M. and De Carvalho, F. d. A. (2004). Clustering of interval data based on city–block distances. *Pattern Recognition Letters*, 25(3):353--365.

Dozat, T. (2016). Incorporating nesterov momentum into adam.

Elharrouss, O., Almaadeed, N., Al-Maadeed, S., and Bouridane, A. (2020). Gait recognition for person re-identification. *The Journal of Supercomputing*, pages 1--20.

Eskandari, M., Toygar, Ö., and Demirel, H. (2013). A new approach for face-iris multimodal biometric recognition using score fusion. *International Journal of Pattern Recognition and Artificial Intelligence*, 27(03):1356004.

Fakhar, K., El Aroussi, M., Saidi, M. N., and Aboutajdine, D. (2016). Fuzzy pattern recognition-based approach to biometric score fusion problem. *Fuzzy Sets and Systems*, 305:149--159.

Feng, Y., Li, Y., and Luo, J. (2016). Learning effective gait features using lstm. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 325--330. IEEE.

Gross, R. and Shi, J. (2001). The cmu motion of body (mobo) database.

Guo, G., Wang, H., Bell, D., Bi, Y., and Greer, K. (2003). Knn model-based approach in classification. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 986--996. Springer.

Han, J. and Moraga, C. (1995). The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks*, pages 195--201. Springer.

Hayfron-Acquah, J. B., Nixon, M. S., and Carter, J. N. (2003). Automatic gait recognition by symmetry analysis. *Pattern Recognition Letters*, 24(13):2175--2183.

He, Y., Zhang, J., Shan, H., and Wang, L. (2019). Multi-task GANs for view-specific feature learning in gait recognition. *IEEE Transactions on Information Forensics and Security*, 14(1):102--113.

Hofmann, M., Geiger, J., Bachmann, S., Schuller, B., and Rigoll, G. (2014). The tum gait from audio, image and depth (gaid) database: Multimodal recognition of subjects and traits. *Journal of Visual Communication and Image Representation*, 25(1):195--206.

Hu, M., Wang, Y., Zhang, Z., Little, J. J., and Huang, D. (2013). View-invariant discriminative projection for multi-view gait-based human identification. *IEEE Transactions on Information Forensics and Security*, 8(12):2034–2045. ISSN 1556-6021.

Hu, M., Wang, Y., Zhang, Z., Zhang, D., and Little, J. J. (2012). Incremental learning for video-based gait recognition with lbp flow. *IEEE transactions on cybernetics*, 43(1):77--89.

Ionescu, C., Papava, D., Olaru, V., and Sminchisescu, C. (2013). Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325--1339.

Isaac, E. R., Elias, S., Rajagopalan, S., and Easwarakumar, K. (2017). View-invariant gait recognition through genetic template segmentation. *IEEE signal processing letters*, 24(8):1188--1192.

Iwama, H., Okumura, M., Makihara, Y., and Yagi, Y. (2012). The ou-isir gait database comprising the large population dataset and performance evaluation of gait recognition. *IEEE Transactions on Information Forensics and Security*, 7(5):1511--1521.

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 14(2):201--211.

Kawai, R., Makihara, Y., Hua, C., Iwama, H., and Yagi, Y. (2012). Person re-identification using view-dependent score-level fusion of gait and color features. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 2694--2697. IEEE.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83--97.

Kusakunniran, W., Wu, Q., Li, H., and Zhang, J. (2009). Automatic gait recognition using weighted binary pattern on video. In *Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*, pages 49--54. IEEE.

Kusakunniran, W., Wu, Q., Zhang, J., Li, H., and Wang, L. (2014). Recognizing gaits across views through correlated motion co-clustering. *IEEE Transactions on Image Processing*, 23(2):696–709. ISSN 1941-0042.

Lee, T. K., Belkhatir, M., and Sanei, S. (2014). A comprehensive review of past and present vision-based techniques for gait recognition. *Multimedia tools and applications*, 72(3):2833--2869.

Liao, R., Yu, S., An, W., and Huang, Y. (2020). A model-based gait recognition method with body pose and human prior knowledge. *Pattern Recognition*, 98:107069.

Liu, D., Ye, M., Li, X., Zhang, F., and Lin, L. (2016). Memory-based gait recognition. In *BMVC*.

Liu, J. and Zheng, N. (2007). Gait history image: a novel temporal template for gait recognition. In *2007 IEEE International Conference on Multimedia and Expo*, pages 663--666. IEEE.

Liu, Z. and Sarkar, S. (2004). Simplest representation yet for gait recognition: Averaged silhouette. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 4, pages 211--214. IEEE.

Man, J. and Bhanu, B. (2006). Individual recognition using gait energy image. *IEEE transactions on pattern analysis and machine intelligence*, 28(2):316--322.

Mowbray, S. D. and Nixon, M. S. (2003). Automatic gait recognition via fourier descriptors of deformable objects. In *International Conference on Audio-and Video-Based Biometric Person Authentication*, pages 566--573. Springer.

Müller, M. (2007). Dynamic time warping. *Information retrieval for music and motion*, pages 69--84.

Murray, M. P. (1967). Gait as a total pattern of movement: Including a bibliography on gait. *American Journal of Physical Medicine & Rehabilitation*, 46(1):290--333.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807--814.

Neubeck, A. and Van Gool, L. (2006). Efficient non-maximum suppression. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 850--855. IEEE.

Nizami, I. F., Hong, S., Lee, H., Lee, B., and Kim, E. (2010). Automatic gait recognition based on probabilistic approach. *International Journal of Imaging Systems and Technology*, 20(4):400--408.

Preis, J., Kessel, M., Werner, M., and Linnhoff-Popien, C. (2012). Gait recognition with kinect. In *1st international workshop on kinect in pervasive computing*, pages 1--4. New Castle, UK.

Qian, G., Sural, S., Gu, Y., and Pramanik, S. (2004). Similarity between euclidean and cosine angle distance for nearest neighbor queries. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 1232--1237.

Samworth, R. J. (2012). Optimal weighted nearest neighbour classifiers. *The Annals of Statistics*, 40(5):2733--2763.

Sarkar, S., Phillips, P. J., Liu, Z., Vega, I. R., Grother, P., and Bowyer, K. W. (2005). The humanid gait challenge problem: Data sets, performance, and analysis. *IEEE transactions on pattern analysis and machine intelligence*, 27(2):162--177.

Sheng, W. and Li, X. (2020). Siamese denoising autoencoders for joints trajectories reconstruction and robust gait recognition. *Neurocomputing*.

Shutler, J. D., Grant, M. G., Nixon, M. S., and Carter, J. N. (2004). On a large sequence-based human gait database. In *Applications and Science in Soft Computing*, pages 339--346. Springer.

Smith, G. J. (2004). Behind the screens: Examining constructions of deviance and informal practices among cctv control room operators in the uk. *Surveillance & Society*, 2(2/3).

Sokolova, A. and Konushin, A. (2018). Pose-based deep gait recognition. *IET Biometrics*, 8(2):134--143.

Sun, K., Xiao, B., Liu, D., and Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In *CVPR*.

Takemura, N., Makihara, Y., Muramatsu, D., Echigo, T., and Yagi, Y. (2018). Multi-view large population gait dataset and its performance evaluation for cross-view gait recognition. *IPSJ Transactions on Computer Vision and Applications*, 10(1):4.

Tan, D., Huang, K., Yu, S., and Tan, T. (2006). Efficient night gait recognition based on template matching. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 1000--1003. IEEE.

Thapar, D., Nigam, A., Aggarwal, D., and Agarwal, P. (2018). Vgr-net: A view invariant gait recognition network. In *2018 IEEE 4th international conference on identity, security, and behavior analysis (ISBA)*, pages 1--8. IEEE.

Vatsa, M., Singh, R., and Noore, A. (2008). Improving iris recognition performance using segmentation, quality enhancement, match score fusion, and indexing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(4):1021--1035.

Wagg, D. K. and Nixon, M. S. (2004). On automated model-based extraction and analysis of gait. In *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pages 11--16. IEEE.

Wan, C., Wang, L., and Phoha, V. V. (2018). A survey on gait recognition. *ACM Computing Surveys (CSUR)*, 51(5):1--35.

Wang, L., Ning, H., Tan, T., and Hu, W. (2004). Fusion of static and dynamic body biometrics for gait recognition. *IEEE Transactions on circuits and systems for video technology*, 14(2):149--158.

Wang, L., Tan, T., Ning, H., and Hu, W. (2003). Silhouette analysis-based gait recognition for human identification. *IEEE transactions on pattern analysis and machine intelligence*, 25(12):1505--1518.

West, D. B. et al. (2001). *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River.

Wu, Z., Huang, Y., Wang, L., Wang, X., and Tan, T. (2016). A Comprehensive Study on Cross-view Gait Based Human Identification with Deep CNNs. *IEEE transactions on pattern analysis and machine intelligence*, 39(2):209--226.

Xiu, Y., Li, J., Wang, H., Fang, Y., and Lu, C. (2018). Pose Flow: Efficient online pose tracking. In *BMVC*.

Yu, S., Chen, H., Wang, Q., Shen, L., and Huang, Y. (2017). Invariant feature extraction for gait recognition using only one uniform model. *Neurocomputing*, 239:81--93.

Yu, S., Tan, D., and Tan, T. (2006). A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 4, pages 441--444. IEEE.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zhang, R., Vogler, C., and Metaxas, D. (2007). Human gait recognition at sagittal plane. *Image and vision computing*, 25(3):321--330.

Zheng, S., Zhang, J., Huang, K., He, R., and Tan, T. (2011). Robust view transformation model for gait recognition. In *2011 18th IEEE International Conference on Image Processing*, pages 2073--2076. IEEE.