View-Invariant Gait Person Re-identification with Spatial and Temporal Attention

Babak Rahi

Supervisor: Prof Maozhen Li

Department of Electronics and Computer Engineering

College of Engineering Design and physical sciences

Brunel University London

May 2021

ABSTRACT

Person re-identification at a distance across multiple none overlapping cameras has been an active research area for years. In the past ten years, Short term Person Re-Id techniques have made great strides in terms of accuracy using only appearance features in limited environments. However, massive intraclass variations and inter-class confusion limit their ability to be used in practical applications. Moreover, appearance consistency can only be assumed in a short time span from one camera to the other. Since the holistic appearance will change drastically over days and weeks, the technique, as mentioned above, will be ineffective. Practical applications usually require a long-term solution in which the subject appearance and clothing might have changed after a significant period has elapsed. Facing these problems, soft biometric features such as Gait have been proposed in the past. Nevertheless, even Gait can vary with illness, ageing and changes in the emotional state, changes in walking surfaces, shoe type, clothes type, objects carried by the subject and even clutter in the scene. Therefore, Gait is considered a temporal cue that could provide biometric motion information. On the other hand, the shape of the human body could be viewed as a spatial signal which can produce valuable information. So, extracting discriminative features from both spatial and temporal domains would be very beneficial to this research. Therefore, this thesis focuses on finding the best and most robust method to tackle the gait

human Re-identification problem and solve it for practical applications. In real-world surveillance scenarios, the human gait cycle is primarily abnormal. These abnormalities include but not limited to temporal and spatial characteristics changes such as walking speed, broken gait phase and most importantly, varied camera angles. Our work performed an extensive literature study on spatial and temporal gait feature extraction methods with a focus on deep learning. Next, we conducted a comparative study and proposed a spatial-temporal approach for gait feature extraction using the fusion of multiple modalities, including optical-flow, raw silhouettes and RGB images. This approach was tested on two of the most challenging publicly available datasets for gait recognition TUM-GAID and CASIA-B, with excellent results presented in chapter 3.

Furthermore, a modern spatial-temporal attention mechanism was proposed and tested on CASIA-B and OULP datasets which learns salient features independent of the gait cycle and view variations. The spatial attention layer in the proposed method extracts the spatial feature maps using a two-layered architecture that are fused using late fusion. It can pay attention to the identity-related salient regions in silhouette sequences discriminatively using the spatial feature maps. The temporal attention layer consists of an LSTM that encodes the temporal motion for silhouette sequences. It uses the encoded output vectors in the temporal attention architecture to focus on the most critical timesteps in the gait cycle and discard the rest. Furthermore, we improved the performance of our method by mapping our extracted spatial-temporal gait features to a discriminative null space for use in our Siamese architecture for cross-matching. We also conducted an element removal experiment on each segment of our spatial-temporal attentional network to gain insight into each component's contribution

to the performance. Our method showed outstanding robustness against abnormal gait cycles as well as viewpoint variations on both benchmark datasets.

ACKNOWLEDGMENT

First and foremost, I would like to thank my supervisor **Prof Maozhen Li** who tirelessly supported me through my years at Brunel University. His guidance went above, and beyond and I could not have asked for a better teacher and mentor.

Also, I would like to thank my family specially my mother, **Esmat Zabihi**, and my aunt **Azam Zabihi**, who supported me financially and emotionally throughout my years of study, even at great personal cost.

Finally, I would like to thank the great staff of the postgraduate programmes office at Brunel university for their help and support during my years of study.

STATEMENT OF ORIGINALITY

I hereby declare that this thesis submitted to the Brunel university London, is entirely my original work prepared under the supervision of Prof Maozhen Li. I have duly acknowledged all the sources of information which have been used in the thesis. I hereby declare that the content mentioned in this paper has not been submitted for the acquisition of a degree in any other schools or places and I shall be solely responsible if any evidence is found against my declaration.

Babak.H.Rahi

Babak Rahi

Brunel University London

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

The concern over the safety and security of people is continuously growing in recent years. It is no secret that governments are severely concerned with the security of public places such as metro stations, shopping malls, airports. Protecting the public is an expensive and taxing endeavour. Consequently, governments seek the help of private companies and scientists to alleviate this pressure and provide better security solutions. With the rise of the COVID-19 pandemic, the need for better security solutions is even more evident than before. Video surveillance systems and CCTV cameras play a crucial role in optimising such efforts.

The abundance of security cameras and surveillance systems in public areas is valuable for tackling various security issues, including crime prevention. All the recordings from these video surveillance systems must be analysed by surveillance operators, which can be daunting. These operators need to analyse these surveillance videos in real-time and identify various categories of anomalies while looking for a ”person of interest” who could easily change their appearance and be unidentifiable to a human with naked eyes. Intelligent video surveillance systems (IVSS) automate

the process of analysing and monitoring hours of acquired videos and help the operators understand and handle them. Person Re-Identification (Re-Id) is one of the most challenging problems in IVSS, which uses computer vision and machine learning techniques to achieve automation.

In the world of computer vision and surveillance systems, person re-identification refers to recognising a person of interest at different locations using multiple non-overlapping cameras, in other words identifying an individual over a massive network of video surveillance systems with non-overlapping fields of view [11, 12]. Person re-identification can also be defined as matching individuals with samples in publicly available datasets which may have various positions, view angles, lighting or background conditions. This process can be performed on an image sequence or video frames that have been prerecorded or in real-time.

Person Re-Id problems arise when the subject moves from one camera view to another in a network of cameras since moving to another camera view could change their position as well as lighting and background conditions. Even the distance and the angle of the camera view, along with numerous other factors such as unidentified objects and areas from one camera view to another, can affect the outcome of Re-ID.

Furthermore, Person Re-Id is used in security and forensics applications to help the authorities and government agencies find a person of interest. There are three general steps to every Person re-identification solution, segmentation, which determines which parts of the frames need to be segmented and focused on. The signature generation finds invariant signatures to compare these parts, and finally, Comparison finds an appropriate method to compare these signatures.

There are multiple methods to person Re-Id, but an image or sequence of the

subjects is usually given as a Query or Probe. The individuals recorded by the camera network as a template gallery are given as the frames. Descriptors are generated for both the template gallery and the Probe. Consequently, they are compared, and the system gives a ranked list or percentage of similarity based on the probability and similarity of images or sequences to the Probe. Figure 1.1 shows a very primitive

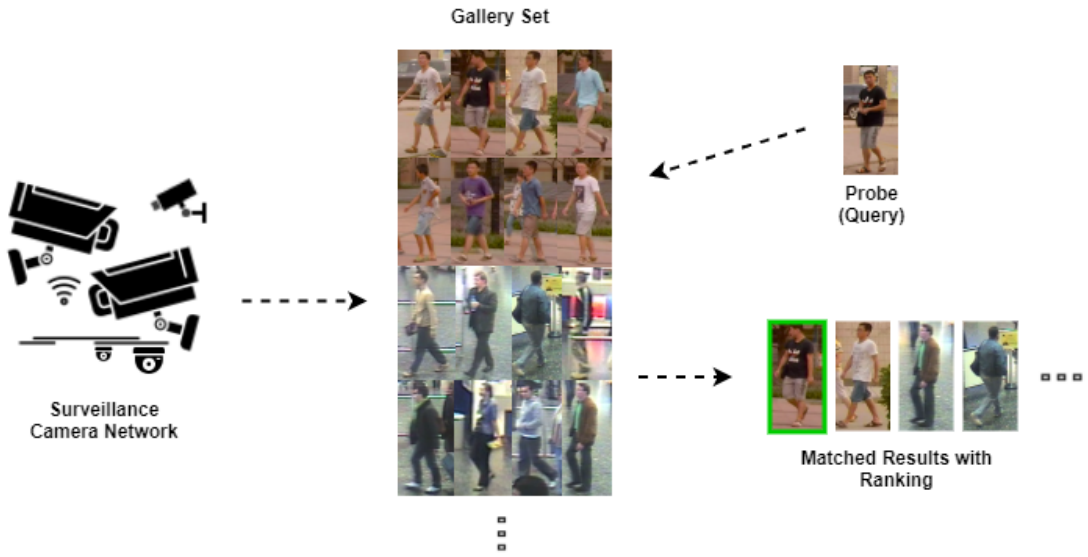


Figure 1.1: A Basic Person Re-Id framework

person Re-Id pipeline in which the system tries to find the corresponding images for a given probe in a gallery of templates. Creating the gallery relies directly on how the re-identification solution has been set up, and [13] categorises this as single-shot and multiple shots, which indicates one or more than one template per frame, respectively. In the case of multiple shots, the new image will be used in the continuous Person Re-Id solution, and each time the new image will be used as a probe for the next level.

Deep learning methods, in particular, Convolutional Neural Networks (CNNs), are used in Person Re-Id solutions as a way to learn from a large number of data [14, 15, 16]. A CNN uses many filters to look at an image through a smaller window and create feature maps at each layer. Features maps are essentially the reported results

of findings after the image has been run through the filter. A particular combination of low-level features can be an indication of more complex features, and feature maps can gradually capture higher-level features at each layer of the network [17]. There are various ways of training a Deep Neural Network (DNN) model. Depending on the problem and the availability of adequate labelled data, these approaches can be categorised as Supervised Learning, Semi-Supervised Learning and Unsupervised Learning. In the problem of Person Re-Id, mainly for security and anti-terrorism applications, only a small amount of training data is available, so semi-supervised or unsupervised models might result in inaccurate solutions. Another categorisation of DNN person Re-Id approaches is made based on their learning methodologies. Single, pairwise and triplet feature matching methods are expanded and explained in chapter two of this thesis.

These methods can also be categorised into short-term and long-term Person Re-identification according to the time interval. Figure 1.2 shows an example of a short term person Re-ID in which camera-1 and camera-2 monitor the same walking path with non-overlapping surveillance views.

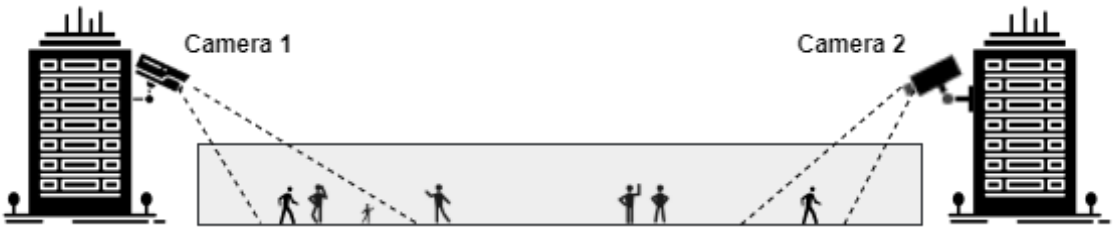


Figure 1.2: An example of short term Person Re-Id

When the person of interest walks from camera view one to camera view two, short term person Re-Id can bridge the gap between these two surveillance views. Given a video sequence (Probe) captured by camera one, Person Re-Id tries to identify the same person while crossing the field of view of camera two. It then publishes a ranked

list of images with descending probability of being the person of interest similar to the primitive Person Re-Id framework in Figure 1.1. Accomplishing this task requires four independent steps: human detection, Human tracking, feature extraction, and Classification. Together these four steps create a complete Person Re-Id pipeline illustrated in figure 1.3.

Figure 1.3: Pipeline of Person Re-Id system

The two first steps of this timeline are independent fields of research which are achieved by numerous methods; however, they are utilised in research to validate the results [18, 19, 20, 21, 22, 23, 24, 25, 26]. Feature extraction refers to learning specific properties that make training samples unique, and Classification is the process of matching the feature variables between the training data and the Probe.

Most of the methods focus on the short-term Person Re-Id and the use of hand-crafted or metric features and a single deep neural network solution. In this case, since the area between the two camera views is small, most likely the appearance features will stay the same, and the task of person Re-Id can be achieved by only relying on the appearance or metric features. Figure 1.4 shows a typical person re-id system with an incorporated Human detection and feature extraction step. This system contains a training phase in which a gallery set of feature vectors is generated. This gallery set includes features of all the individuals walking the path. In the testing phase, one descriptor for the person of interest is produced using the same method in the training phase. When the probe person enters the camera network, his feature vector is generated the same way as the training phase. Then it will be compared against the

gallery using a classification algorithm or similarity matching technique. The system will then output the ID of the best matched re-identified person.

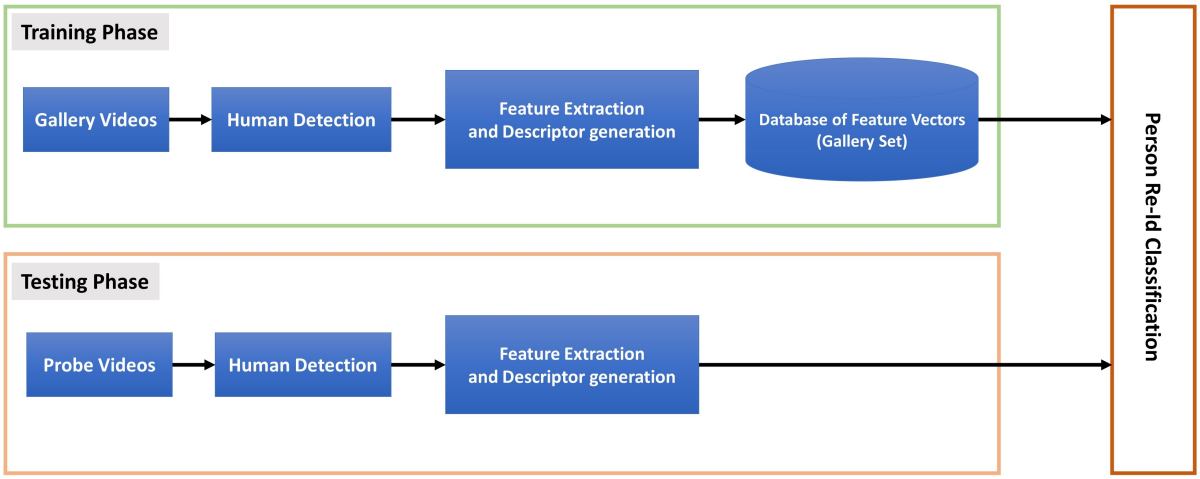
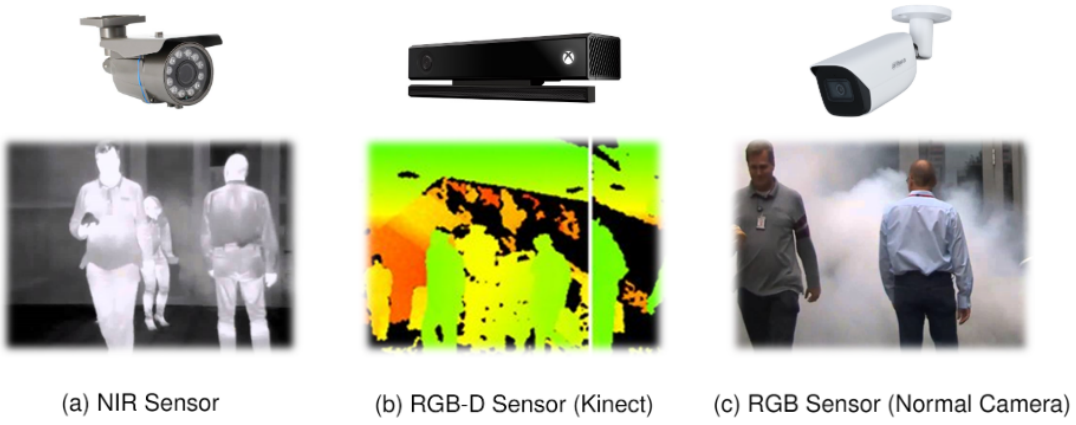


Figure 1.4: Classic Person Re-Id Systems

Another consideration is the type of sensors used for person Re-Id. Figure 1.5 shows the most popular sensors used in Person Re-Id research in the past. Near-Infrared (NIR) cameras are primarily used in situations with low lighting, such as dark indoor conditions or at night [27]. RGB-D Sensors or Kinects can acquire depth information which can be helpful for 3D reconstruction or depth perception but is used in very particular Person Re-Id applications [28, 29]. In practice, most surveillance systems work with the standard RGB cameras, so naturally, researchers focused on reducing the inter-class and intra-class variations on this type of sensors [30, 31]. In Short Term Re-Id, RGB-D cameras are proven to have less performance and cost more than the standard RGB cameras, so they are not practical for widespread use in the real world.

Person Re-Id systems could also be characterised based on the type of inputs into image-based and video-based categories. In image-based, the system receives random frames as input and focuses on appearance features such as colour and texture since these features remain the same in short periods of time [32, 33, 34, 35, 36, 37].



(a) NIR Sensor (b) RGB-D Sensor (Kinect) (c) RGB Sensor (Normal Camera)

Figure 1.5: Most popular sensors for person Re-Id

Video-based systems receive a video clip or a collection of successive frames as input, but in addition to the appearance attributes, they explore movement data to improve the performance of the system [38, 39, 40, 41, 42, 43, 44, 45].

Short term person Re-Id methods have achieved great accuracy on publicly available datasets, but due to several inter-class and intra-class variations, they suffer from low performance and high cost. Subsequently, researchers try to reduce these variations and their effects. In the early years, the main focus was on handcrafting descriptors from images or videos. This is including but not limited to histograms of different colour spaces such as HSV, YCbCr, LAB and LBP which are extracted from overlapping features and then concatenated into single feature vectors, Local patterns in binary form [46, 47, 48, 49], a collection of local features [50, 51], maximal Occurance representation of local features [35], HOG3D [44], STFV3D [43], features learnt by Convolutional Neural Networks [52, 32, 53], or a combination of the features above. Metric learning classification methods such as KISSME, Local Fisher and Marginal Fisher analysis, Top Push Distance Learning, Nearest Neighbour, NFST and Quadratic discriminant analysis are also used to discriminate between the mentioned features [34, 35, 44, 48, 54, 55, 56, 57].

1.1 Motivation

According to a projection published by IHS Markit, over one billion surveillance cameras will be in operation globally by 2022. At the moment of writing this thesis, there are 770 million cameras installed around the world [58]. After China, the UK has one of the most substantial numbers of CCTV cameras globally. In 2015 the British Security Industry Association (BSIA) estimated that between four to six million security cameras are installed in the country. London has the highest number of CCTV cameras in the UK. By that estimation, an average Londoner could get caught on camera three hundred times a day [59]. These cameras are used in places from home and shop surveillance to public areas such as airports, shopping centres, metro stations, and other forms of public transportation.

The main reasons behind this radical increase in the use of CCTV are reduction in the price of cameras and effectiveness in crime prevention. Utilising CCTV cameras in real-time, the police and security agencies can prevent incidents by detecting suspicious behaviour or gathering evidence such as identifying suspects, witnesses, and vehicles after a crime has been committed. Accordingly, the task of threat detection and person re-identification is left entirely to the human operators. These security operators need to possess a high level of visual attention and vigilance to react to rarely occurring incidents. Moreover, many human resources are required to analyse the millions of hours of collected videos, thereby making this task very costly. To search for a person of interest in thousands of hours of prerecorded videos is even more taxing and time-consuming and requires experts forensics specialists.

Automatic video analysis considerably reduces these costs, and for this reason, it became a critical field of research. This research field tackles problems including

but not limited to object detection and recognition, object tracking, human detection, person re-identification, behaviour analysis and violence detection. Solutions to these problems have applications in numerous domains like robotics, entertainment, and to no small extent, video surveillance and security.

Person Re-Id is different from the classic detection and identification tasks since Person detection is to distinguish a human from the background, and Identification is the process of determining a person's identity in a picture or video. Detection indicates whether there is a person in the provided image, and Identification tells us who it is. However, Person Re-Id determines whether an image over video clip belongs to the same individual who previously appeared in front of the camera.

Usually, the assumption is that the subject wears the same clothing in different camera views, and the appearance stays the same for the re-identification task. This premise produces a significant limitation on the job since people can change their appearance and especially their clothing over the course of days, hours or even minutes. These Alterations makes re-identification based on appearance unlikely after a certain period of time. The hypothesis is that biometric features like faces or Iris are not always available in CCTV videos, especially after the rise of the COVID-19 pandemic.

Illumination changes, position variations, viewpoints and inter-object occlusions make appearance-based Person Re-ID a notable problem. Most recent models use different features like the colour of clothes and texture to improve their performance. Typically they generate feature histograms, concatenate and finally weigh them according to their importance and distinguishing power [60, 61]. These features can be learnt through multiple methods such as boosting, measuring distance metrics and rank learning [62, 63]. The downside of these methods is their lack of scalability since

the learning process needs constant supervision as the subjects change. It is a better practice not to bias all the weights to global features and give selective weights to more individually unique features such as saliant appearance features or gate features such as walking speed and direction and the flow of movement. The human visual attention is studied in [30] and the results imply that attentional competition between features could take place not only based on the global features but also the salient features in individual objects.

To overcome the issues mentioned above, soft biometrics, such as Gait, has been used in the past. Gait is a biometric feature that focuses on a person's walking characteristics and motion features. Other biometrics such as Iris and face could be altered using a pair of contact lenses or a simple surgical mask after the COVID-19 pandemic. Gait's advantages in video surveillance include the ability to extract features non-invasively from a distance, property acquisition from low resolution, and the ability to extract features even in the dark using different modality cameras. The two key metrics in Gait analysis are spatial and temporal parameters. Spatial and temporal features can describe the state of an object over time or a position in space. Moreover, in the past few years, several publicly available datasets have been published, which can be used to validate our research. A complete literature review of Gait spatial-temporal methods, along with these datasets, are introduced in chapter 2 of this thesis.

In short, working on Gait as a biometric feature for Person recognition and re-identification could be a massive help in the war against crime and even prevent terrorist attacks by recognising well-known terrorists and dangerous repeat criminals.

1.2 Applications and Challenges of Person Re-Id

1.2.1 Applications

Person Re-Id methods have much potential in a wide range of practical applications from security to health care and even retail. For instance, cross camera person tracking could understand a scene through computer vision, track people across multiple camera views, perform analysis on the crowd movement and do activity recognition. When the person of interest moves from one camera view to the other, the track will be broken; Therefore, Person Re-Id is used to establish connections between the tracks to accomplish cross camera tracking.

Another form of person Re-Id is tracking by detection, which uses person detection methods to perform tracking of a subject. This task includes modelling diverse data in videos, detecting people in the frames, predicting their motion patterns and performing data association between multiple frames. When person Re-Id is associated with the recognition task, a specific query image will be introduced and searched in an extensive database to perform person retrieval. This method will usually produce a list for similar images or frame sequences in the database.

Person Re-Id is also used to observe long-term human behaviour and activity analysis, for example, observing customers' shopping trends, analysing their activities and altering products and even shopping floor layouts to maximise sales. In health care, it can be used to analyse patient behaviour and habits to assist hospital staff with a higher standard of care.

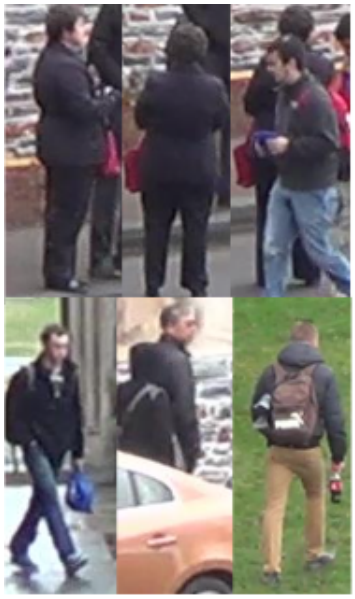
1.2.2 Challenges

Appearance inconsistency and clothing variations

Several challenges have to be overcome to solve Person Re-Id's problem and use it in the above applications. Matching a person across different scenes requires dealing with class variations and confusions. The same person can undergo significant changes in appearance from one scene to another, or two different individuals can have similar appearance features across multiple camera views. Some of these variations include Illumination variation, Camera Viewpoint Variation, Pose Variation, Low Resolution, Similar clothing, Partial Occlusion, Real-time constraint, clothing change, Accessories change, camera settings, a small training set and data labelling cost. Moreover, most models support short-term Re-ID in which they leverage colour and texture and the object carried by the subject. One of the most common challenges of appearance-based methods is assuming that colours could be assigned to the same object under various lighting conditions. Achieving colour consistency under such conditions is not an easy task [64].

Another limitation is appearance consistency, which can only be assumed in a short time span from one camera to another. Since the holistic appearance will change drastically over days and weeks, the technique, as mentioned above, will be ineffective. Practical applications usually require a long term solution in which the subject's appearance and clothing might have changed after a significant period of time has lapsed [65]. It is irrefutable that appearance-based Short term Person Re-Id techniques have made great strides in terms of accuracy in the past years, but problems such as massive intraclass variations and inter-class confusion caused by the conditions mentioned above make this a challenging and worthwhile field of research.

Some of these challenges are shown in figure 1.6 on four of the benchmark Person Re-Id datasets namely PRID2011 [66], MARS [42], iLIDS-VID [67], DukeMTMC-reID [68].



(a) DukeMTMC-reID

(b) PRID2011

(c) iLIDS-VID

(d) MARS

Figure 1.6: Variations in four Person Re-Id Datasets

Insufficient datasets

Insufficient datasets is another challenge. Several publicly available datasets are out there, but none of them is large enough regarding the number of camera angles, num-

ber of subjects, or recorded period of time. Most datasets are usually recorded using two cameras and a small sample of subjects, and since deep learning models need an extensive training set and validation set, building realistic datasets would help further the progress of Person Re-Id research.

1.2.3 Tackling the challenges using generated data

When training a model on one data set and testing on another, more often than not, the performance drops significantly. To overcome this challenge, data augmentation techniques such as Generative Adversarial Network (GAN) [69] has been designed in recent years to introduce large scale datasets [70, 71] or to expand sample data [68, 72]. Another important research direction is the unsupervised Person Re-Id models. Unlike supervised learning, these models do not train using labelled data in the same environment and have a lower annotation cost. [73, 74, 75].

Since these features on their own are not robust enough, some authors try to include other modalities such as depth and thermal data in their models using deep learning. These models are known as multi-modal methods and are particularly challenging in real life because a framework has to be developed that can handle multiple variations such as subject position and view obstructions [76, 77]. Building architectures of deep learning models for person re-id is a very time-consuming task, so some researchers are using Neural Architecture Search methods (NAS) to automate the process of architecture engineering [78, 79]. The most crucial challenge in NAS methods is that there is no guarantee how appropriate the CNN would be after NAS has chosen it.

1.2.4 Tackling challenges Using Biometric features

Facing all these problems, long term approaches using Biometric features have been suggested by researchers in the past. Biometrics is the science of person identification based on their physical and behavioural traits like body measurements and calculations related to human characteristics, which can be used to describe and label them [80]. Biometric surveillance identifies a person of interest by extracting features of all the people in the camera network and comparing them to the gallery set. The most commonly used biometrics are categorised as Hard Biometrics and Soft Biometrics. Some Hard biometrics are fingerprint, iris, face, voice and palm print which do not change over time and are primarily used in access control systems. The acquisition of these biometrics demands a controlled environment and invasive measures. In video surveillance scenarios, people move more freely and without supervision, making the gathering of Hard Biometrics impractical. To tackle the problem of long term Re-Id, Soft Biometrics [81] like anthropometric measurements, body size, height or Gait are used with better success. These Biometrics are more reliable for long term re-id solutions; however, they are challenging. Compared to Hard Biometrics, Soft Biometrics lack strong indicators of an individual's identity. Nevertheless, the non-invasive nature of Soft Biometrics and the ability to acquire information from a distance without the need for subject cooperation makes them a strong candidate for tackling Person Re-Id.

Gait as a soft biometric feature

Among Soft Biometrics, Gait is the most widespread feature for Person Re-Id in surveillance networks. Other Soft Biometrics such as 3D face recognition is also considered,

but Gait is more popular because first, it does not require contact with the subject, and the cues gathered from Gait are unique to each individual and are extremely hard to fake [82]. Gait can also work in low-resolution videos. Gait Re-Id gathers and labels distinctive measurable features of human movement just like any other Biometric system. In Psychology and neuroscience, focusing on Gait is an essential subject in humans' perception of others. For example, it is a known fact that in cases of prosopagnosia or face blindness, "the patients tend to develop individual coping mechanisms to allow them to identify the people around them, mostly via non-facial cues such as voice, clothing, gait and hairstyle recognition" [83, 84, 85]. Moreover, Gait Analysis is a valuable tool for the diagnosis of several neurological disorders such as stroke, cerebral palsy, Parkinson's, and sclerosis [86, 87, 88, 89].

Person Re-Id using Gait

Person Re-Id based on Gait has received substantial attention in the past ten years, especially from the biometric and computer vision community due to the advantages mentioned above, but it is not without its disadvantages. Gait as a Soft Biometric can vary with illness, ageing, changes in the emotional state, walking surfaces, shoe type, clothes type, objects carried by the subject, and even clutter in the scene. Moreover, Gait based Person Re-Id problem should not be mistaken with Gait based Person Recognition since they are applied in entirely different scenarios. Recognition is employed in heavily controller environments, often with a single camera, and the operator can influence conditions including but not limited to background, subject pose, angle of the camera and occlusion. On the other hand, the conditions in Person Re-Id are entirely out of control. Since a large camera network is used to solve this type of prob-

lem, variables like lighting, number of people and occlusion, the angle and direction of the walk are unknown. Gait allows us to analyse a person of interest in various standpoints and poses. What makes Gait more attractive is the tendency to be used in long-term person Re-Id, relying on more than only spatial signals and appearance features. Gait is considered a type of temporal cue that could provide biometric motion information. Combinations of appearance and soft Biometrics have been used in the past to solve the problem of Person Re-Id [34]. On the other hand, the shape of the human body could be considered a spatial signal which can produce valuable information. Several works have tried to extract discriminative features from both spatial and temporal domains [43, 90, 91].

Model based vs model free Gait Recognition

Early research on Person gait recognition and Re-Id attempted to model the human body because Gait is essentially analysing the motion of distinct parts of the human body [92, 93, 94]. Specific characteristics pertaining to various body parts are extracted from each image in the sequence to form a model of the human body. Then the parameters will be updated over time for each silhouette in the sequence and used to create a gait signature. These characteristics are usually metric parameters such as the length and width of a body part like arms, legs, torso, shoulders and head and their position in the images. They can define the walking trajectory or the euclidean distance between the limbs. Twenty-two such features are introduced in [95]. [96] proposed a posture-based method for gait recognition which learned posture characteristics by considering the displacement of all joints between two consecutive frames and a fixed centre of body coordinate system for all the joints. A two-point gait repre-

sentation was introduced in [94] which modelled the motion of limbs regardless of the body shape.

Other works such as [97] create a motion model by considering the motion of hips and knees in various phases of a gait cycle. In this approach, the features are extracted from 2D images. Gait parameters, namely magnitude and phase of the Fourier components, are extracted using a Viewpoint rectification stage. However, Because of the low quality of surveillance images captured in the real world, it is implausible to calculate a model robust enough for widespread practical usage.

For this reason a model free approach has been employed in this thesis which relies on the spatial-temporal motions of the body. In a spatial-temporal model-free approach, Gait is represented by mapping the motions through time and space [98]. There are different spatial-temporal methods proposed in the literature which have been presented in Chapter 2 of this thesis.

In conclusion, large models achieve the best accuracy, but when applied to existing video surveillance systems, they may consume a lot of memory and time, which directly impacts their efficiency, so this trade-off between accuracy and efficiency must be considered in Person Re-Id models.

1.3 Research gap and contributions

Gait recognition is an attractive human recognition technology, especially after the rise of the COVID 19 pandemic, which rendered most face recognition and re-identification solutions moot. However, existing gait recognition methods mainly focus on the regular gait cycles, which ignore the irregular situation. In real-world surveillance, human Gait is almost irregular, which contains arbitrary dynamic characteristics (e.g., duration,

speed, and phase) and various viewpoints.

This irregularity of Gait poses new challenges for the research community to identify a person and understand people's movements. Human gait is one of the most attractive biometric features, which can identify a person from others in a remote and non-invasive manner. Gait recognition in surveillance systems has become a desirable research topic over the past few years because of its across-the-board applications, including social security [99], Person Re-Id [100, 101], and analysis of health status [102]. However, due to the complex surveillance environments and diverse human behaviours, human Gait is always irregular.

First of all, massive irregular gait sequences contain more than or less than one complete gait cycle, varied lengths, asynchronous paces, different size strides, and inconsistent phases. Moreover, gait appearances are dramatically altered due to camera viewpoints, clothing, and carry objects. Among these, irregular Gait and change of viewpoints are two significant challenges. The periodic motion cues of the irregular gait sequences are complicated to extract due to the difficulty of precisely detecting the gait cycle. Moreover, the change of viewpoints will seriously magnify the intra-class variations. Additionally, many different subjects with similar gait appearances have unclear inter-class differences. Therefore, they are challenging to be accurately identified. Some examples are shown in Figure 1.6 of this chapter.

This thesis proposes the Attentional Spatial-Temporal system to solve the above challenges in irregular gait recognition. Moreover, existing approaches on gait recognition have been extensively investigated. Most of them exploit Gait Energy Image (GEI) [2] to represent the temporal information of gait sequence, which leads to the loss of the vast dynamic information and only represents the external movement cues.

Recently, some deep learning-based gait recognition methods have been proposed to extract the robust gait features [57, 103], [104]. They are well known for their superiority in as opposed to traditional methods.

Although existing methods have provided a better avenue for learning gait signatures and improved the gait recognition accuracy to a certain extent, most of them need to detect at least one complete gait cycle precisely and are not robust to the viewpoint changes. Therefore, irregular and specially view-invariant gait recognition still needs particular attention, which has been accomplished in this thesis by discovering a novel solution for reducing the effect of irregular Gait on Person Re-Id problems.

Summary of contributions

Within the research gap presented in this section the main focus of my thesis is to address the problem of how to disentangle the contributions of the viewpoint factor from other Person descriptors, as a way to increasing the features discriminative power. The above conditions correspond to the following specific contributions achieved in this work:

- Achieving a more efficient exploitation of the available labelled data by proposing a new approach based on spatial-temporal architecture with multiple modality fusion for gait feature extraction which uses the best network modality combination for Gait recognition.

- Improving the efficiency of the training stage by designing a novel A Spatial-Temporal Attention framework for Gait feature extraction which extracts the discriminative sequence-level features for representing the periodic motion cues of irregular gait sequences. The designed dual spatial attention mechanism can

concentrate on the discriminative identity-related semantic regions from the spatial feature maps. The proposed mechanism for temporal attention can automatically assign adaptive weights (attention) to enhance the discriminative gait timesteps and suppress the redundant ones.

- A new and robust approach to improve the accuracy of cross-view gait recognition and reduce the effect of viewpoint variations. A combination of the Siamese structure and learned Null Foley-Sammon transform (LNFST) was employed to obtain the view-invariant gait features from irregular gait sequences. This robust approach is not limited to our attention mechanism and can be implemented to other spatial-temporal feature extraction.

1.4 Thesis outline

The thesis studies the Gait based Person re-ID task by using two different methods and addresses several research problems, including but not limited to appearance, pose, carry and view variations. Furthermore, it introduces two solid approaches for the feature extraction method and cross-view person matching on three challenging datasets. The rest of this thesis is organised as follows:

- **Chapter 2:** Introduces some basic knowledge about the methods used in Gait Person Re-Id and reviews the literature surrounding the subject by dividing the Person Re-Id algorithms into three different paradigms. An overview of available datasets in motion analysis and feature extraction using deep learning methods is also provided in this chapter.

- **Chapter 3:** Provides a Comparative spatial-temporal study with different deep

learning feature extraction and motion encoding techniques using multiple modality fusion for gait feature extraction and proposes a different approach for feature extraction and fusion of modalities, including optical flow, RGB images and silhouette sequences. This approach is then tested on two of the most challenging publicly available datasets, CASIA-B and TUM-GAID.

- **Chapter 4:** Proposes a spatial-temporal Attention framework for Gait feature extraction with a Long short term memory mechanism to encode the motion information in a silhouette gait sequence. The proposed approach uses an elaborate attention mechanism to solve the problems mentioned earlier and reduce the inter and intraclass variations faced in more complex datasets focusing on view variations. The technique is then tested using a Siamese architecture for cross-matching gait signatures, achieving excellent results on CASIA-B and OULP datasets.

- **Chapter 5:** Proposes improvements on the previous chapter's technique to reduce further the effects of viewpoint variations on the spatial-temporal attention mechanism. It further analyses the robustness of our method against abnormal gait cycles and performs an ablation study by mixing and matching different components of the network.

- **Chapter 6:** This chapter concludes the thesis and provides guidance for possible future work based on the research findings.

1.5 List of Publications

- **Published:** Zhang, B., Huang, Z., Rahi, B.H., Wang, Q. and Li, M., 2019. Online semi-supervised multi-person tracking with gaussian process regression.

- **Published:** Zhang, B., Li, S., Huang, Z., Rahi, B.H., Wang, Q. and Li, M., 2018. Transfer learning-based online multiperson tracking with Gaussian process regression. Concurrency and Computation: Practice and Experience, 30(23), p.e4917.

- **In Process:** A novel architecture for human re-identification with a two-stream neural network and attention mechanism

- **In Process:** View-Invariant Gait Person Re-identification with spatial and temporal attention

Chapter 2

LITERATURE REVIEW

The Re-Identification of humans in digital surveillance systems will be discussed in the following chapter. The methods employed to tackle this problem be listed and compared with a focus on methods to help elevate the irregular gate problem, which is the main focus of our research. Critically assessing these methods will help exploit the available labelled data from the publically available datasets more efficiently and improve the training stage's efficiency in irregular gait Person Re-Id. It also shows a significant research gap in areas relating to irregular gait recognition and Re-Identification, which we face in practical scenarios rather than closed lab environments.

It can be assumed that Aristotle in De Motu Animalium (On the Gait of Animals) was the modern gait analysis pioneer. A series of papers were published on the biomechanics of Gait for humans under unloaded and loaded conditions by Christian Wilhelm in the 1980s [105].

Capturing frame sequences that reveal details of animal and human movements unnoticed by the naked eye became possible with cinematography and photography's

progression. In 1900s [106] and [107] pioneered these developments. For example, the horse gallop sequence was normally distorted in painting before the discovery was exposed by aerial photography. The production of video camera systems in the 1970s helped begin the extensive research and practical application of gait analysis on people with pathological diseases such as Parkinson's and Cerebral palsy within a realistic time frame and with low cost. Based on the results obtained by gait analysis, orthopaedic surgery made a significant advance in the 1980s. Many orthopaedic hospitals worldwide are currently equipped with gait labs to suggest and develop regimens and plans for doctors' treatment and schedule follow-ups. Nowadays, human identification from a distance or non-intrusive human recognition has gained much interest in the computer vision research community. The Gait Recognition study area addresses the identification of a person by the way they walk. Gait Recognition proposes quite a few exclusive characteristics compared to other biometric methods. One characteristic that is most attractive to researchers is the unobtrusive nature of science. The subjects do not need to cooperate or pay attention to be identified. Also, no physical information is needed for capturing human gait characteristics from a far distance.

In this research area, complications may arise when multiple cameras are used to monitor an environment. For example, a person's position is known to us when they are within a single camera view, but problems might accrue as soon as the subject moves out of one view and enters another. This is the essence of the human Re-Identification problem. How can the system detect the same person in another camera view earlier? The purpose of this chapter is to evaluate the gait-based methods of re-identification.

The literature review is structured in this chapter in the following manner:

- Section 2.1 goes through a taxonomy of some of the more recent approaches

to Gait Person Re-Id, which use other methods rather than deep learning. This section categorises and critically assesses these approaches.

- Section 2.2 categorises the best and most recent deep learning approaches to Gait Person Re-Id in the literature.

- Section 2.3 introduction the fundamental concepts of deep learning, namely Convolutional Neural Network (CNN), Siamese and triplet neural networks, Long-Short Term Memory (LSTM) which are the building block of all approaches tried and tested in this thesis.

2.1 A taxonomy of previous work on Gait Person Re-Id

To best grasp the state of the art methods used in Gait person Re-Id, we classify the past algorithms into three different paradigms. Figure 2.1 shows the overall classification of these algorithms. Finally, we will discuss these approaches and their strengths and drawbacks to better understand Gait Re-Id's challenges.

2.1.1 Approaches to Data Acquisition

The number and types of cameras used in the raw data acquisition directly impact the Person Re-Id algorithm. We can categorise these approaches into approaches using a single camera acquired dataset and approaches which use datasets gathered using multiple cameras.

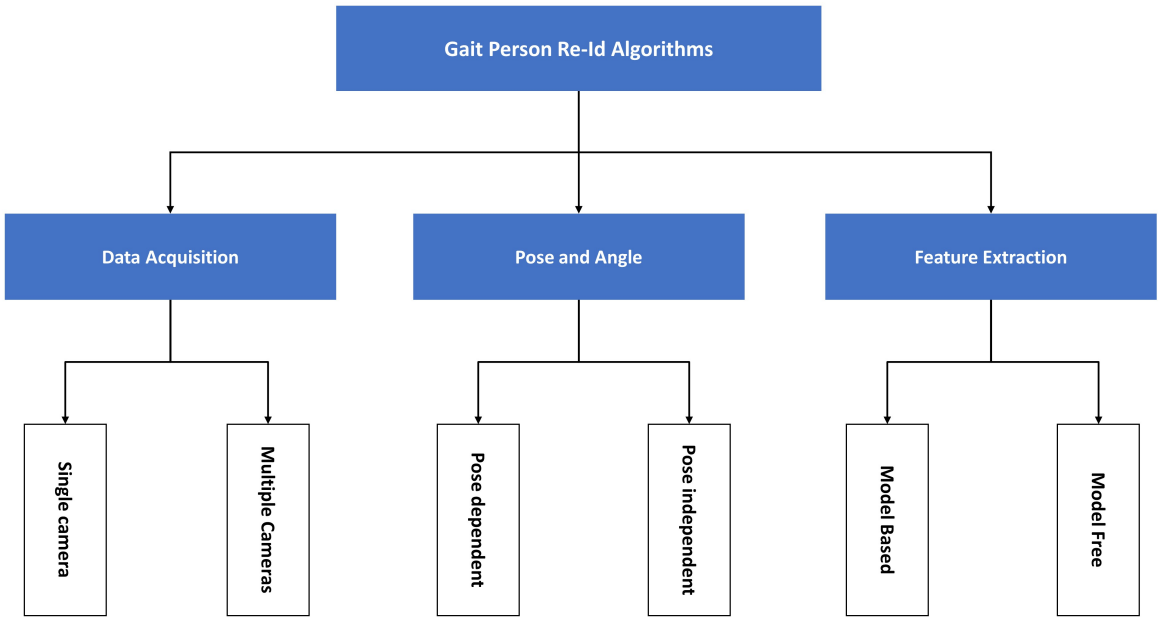


Figure 2.1: An overall classification of Gait Person Re-Id methods

Single Camera Approaches

Depending on the type of camera used, the algorithm could exploit two dimensional (2D) or three dimensional (3D) information. Motion Capture (MOCAP) systems or depth sensors such as Kinect can directly acquire a 3D representation of the environment [108, 109, 110, 111]. However, datasets generated from 2D cameras are the standard in most previous works [12, 97]. A Re-Id scenario usually consists of multiple variations, including camera viewpoints, clothing, lighting and walking speed. When there are multiple cameras in the network, some could be used for the gallery and some for the probe set. Other variations could be used in the same way to compare their effect on an algorithm. In most cases, the data used for gait recognition and Re-Id is a short video clip or a sequence of consecutive frames containing gait cycles. Some of the most popular publicly available Gait datasets are listed in Table 2.1. Previous works usually use different combinations of camera views to simulate a real-world scenario even though datasets such as CASIA-B and SAIVT have multiple overlap-

ping and non-overlapping camera views [112]. Only one camera is used for the probe when running the system, and the rest are used to create the gallery. This approach was used in [113, 114] on PKU, SOTON and CASIA, which contain multiple camera viewpoints. [97] used the i-LIDS dataset with two different camera views; They used one camera for the probe and the other for the gallery set. [39] used a random fifty-fifty split for each sequence in the dataset regardless of the camera view and tested this approach on iLIDS-VID, HDA+ and PRID2011 datasets.

Multiple Camera Approaches

Some previous works used overlapping camera views, but due to constraints in calibration and installation of multiple cameras simultaneously, such works are scarce in the literature. One example of such works is [115] in which they used 16 overlapping cameras to create a 3D gait model of a human. The 3D gait models were generated by the volume intersection of silhouettes extracted from walking frame sequences. Then random viewpoints from these synthetic images were chosen to create the gallery. Using Kinect and MOCAP system will considerably reduce the amount of computation and complexity of the problem since they can acquire the 3D data directly and use the same camera for both probe and gallery sets [116, 117, 118, 110]. In these works, people walk in arbitrary directions or side to side in front of the same Kinect 3D sensor. [109] and [108] use MOCAP systems for person Re-Id. In [108] they used the dataset of the Carnegie Mellon University. This dataset was collected using 12 Vicon infrared MX-40 cameras with people wearing marked black clothes. The markers were only visible in infra-red and were used to produce the 3D information.

Although In the past few years there was a paradigm shift towards methods with

overlapping viewpoints with Kinect or MOCAP systems, there is still an overwhelming amount of research on 2D surveillance networks since the use of such sophisticated devices which can acquire direct 3D motion information is not yet incorporated in the real-world.

2.1.2 Approaches to pose and angle

In automatic surveillance systems, The pose of a person or human pose is determined by the direction of the walk and the camera viewpoint. Depending on the pose, the dynamic and static information acquired from the image sequence can change when switching between camera views or when a period of time has passed. Conditional on the human pose, Pose-dependant and Pose-independent approaches are reported in the literature. In pose-dependant, the data applied to the Person Re-Id system is restricted to only one direction or camera view, but in Pose-independent, any arbitrary viewpoint or direction could be used as the input of the system.

Pose-dependent Approaches

Pose-dependant approaches are most useful for indoor scenarios where walking directions will not change in relation to the camera, such as station entrances and shopping centre corridors. Numerous publicly available datasets, including iLIDS-VID, CASIA, PRID2011 and TUM-GAID, support the Pose-dependant approach. [117, 118, 119] use a single camera viewpoint approach in their work. In [117] both side and top views are used, and in [118, 119] they use frontal view, but most past research is focused on the lateral (side) human view because it enables a more clear observation of the human Gait and provides a consistent amount of self-occlusion in each frame

Table 2.1: Details of some of the well known publicly available datasets for gait recognition

Name	Subjects	Location	Image Dimension	Views	Variations
TUM-GAID	305	Inside	640x480	1	Normal Back pack Shoes
CASIA Dataset A	20	Outside	352 x 240	3	Clothing
CASIA Dataset B	124	Inside	320 x 240	11	Normal Back pack Coat Lighting
SOTON (Large)	115	Inside/Outside	20 x 20	2	Clothing
CMU MoBo	25	Inside (Treadmill)	640 x 480	6	Slow Walk Incline Walk Ball Walk
OU-MVLP	10,307	Inside	1280 x 980	7	-
OU-ISIR, Treadmill A	34	Inside (Treadmill)	88 x 128	1	Speed
OU-ISIR, Treadmill B	68	Inside (Treadmill)	88 x 128	1	Clothing
OU-ISIR, Treadmill D	185	Inside (Treadmill)	88 x 128	1	Occlusion
i-LIDS (MCT)	119	Inside	576 x 704	5	Clothing Occlusion
iLIDS-VID	300	Outside	576 x 704	2	Lighting Occlusion
PRID2011	245	Outside	64 x 128	2	Lighting Occlusion
KinectREID	71	Inside	Vary	1	-
MARS	1261	Outside	1080 x 1920	6	Lighting
HDA	85	Inside	2560 x 1600	13	-
USF	25	Inside	640 x 480	6	32 Variations
Vislab KS20	20	Inside	Only depth data	1	-

[91, 120]. Some works choose the system inputs based on the viewpoints provided by the dataset [114, 113] for example, lateral for CASIA and TUM-GAID and frontal and back for PKU.

Pose-independent Approaches

Pose-independent or cross-view are among the most practical methods for real-world applications because the human walk's direction is most likely arbitrary in an uncontrolled data acquisition setup. The viewpoint variation puts a higher computational strain on the Person Re-Id system. On the other hand, the input data must be of a higher quality than approaches dependent on only one view. The gallery set in these approaches is constructed from random data collected from arbitrary walking directions, which will be put against a random probe set for testing. For example, [114] produces the gallery using all the 11 viewpoints in CASIA Dataset B and then tests the algorithm by choosing a random probe image from the same datasets. The same technique is used in [39, 121] and [12, 97] where the gallery set is created from random viewpoints. [97] also provided a view transformation model (VTM), which exploits projection techniques to tackle pose variation problems. Models such as VTM are utilised to transform multiple data samples into the same angel. Similarly, sparsified representation-based cross-view model and discriminative video ranking model were used by [112] and [39] respectively.

Furthermore, 3D data can reduce the computational cost of view alignment using simple geometrical transformation like [115], which creates 3D models from 2D images that multiple overlapping cameras have acquired. Synthesising these models generated virtual images with random viewpoints and constructed a gallery. A

probe was chosen from authentic images collected by a single view camera to test the algorithm. MOCAP was also employed by [109] and [122] to collect 3D data for a Pose-independent Person Re-Id system. Since Kinect can provide a 3D volumetric representation of the human body, some works such as [110] use skeleton coordinates provided by Kinect to demonstrate the impact of viewpoint variation on Gait Person Re-Id systems. They use these findings in [116], and [123] in which they proposed a context-aware gait Person Re-Id method that used viewpoints as the context in their study.

2.1.3 Approaches to feature extraction

Human Gait Cycle

Gait Signature or Gait feature is the subject's unique characteristics extracted from a sequence of sample images over a Gait Cycle or Stride. To understand the Gait Cycle, we need to analyse, isolate and quantify a unique short and repeated task when someone walks. A gait cycle can be measured from any gait event to the same gait event on the same foot. However, it is implied in the literature that a Gait Cycle should be measured from the strike of one foot to the ground to the next strike of the same foot in a person's walking pattern. The Gait Cycle is considered the fundamental unit of Gait, so by measuring temporal and spatial aspects of it, we can extract gait signatures for a particular person of interest.

There are two primary phases of the gait cycle: the ”swing” phase and the ”stance” phase. These phases alternate and repeat when a person walks from point A to point B. The stance phase is the duration when the foot is on the ground, and the swing phase is the whole time when the foot is in the air. Consequently, observing the move-

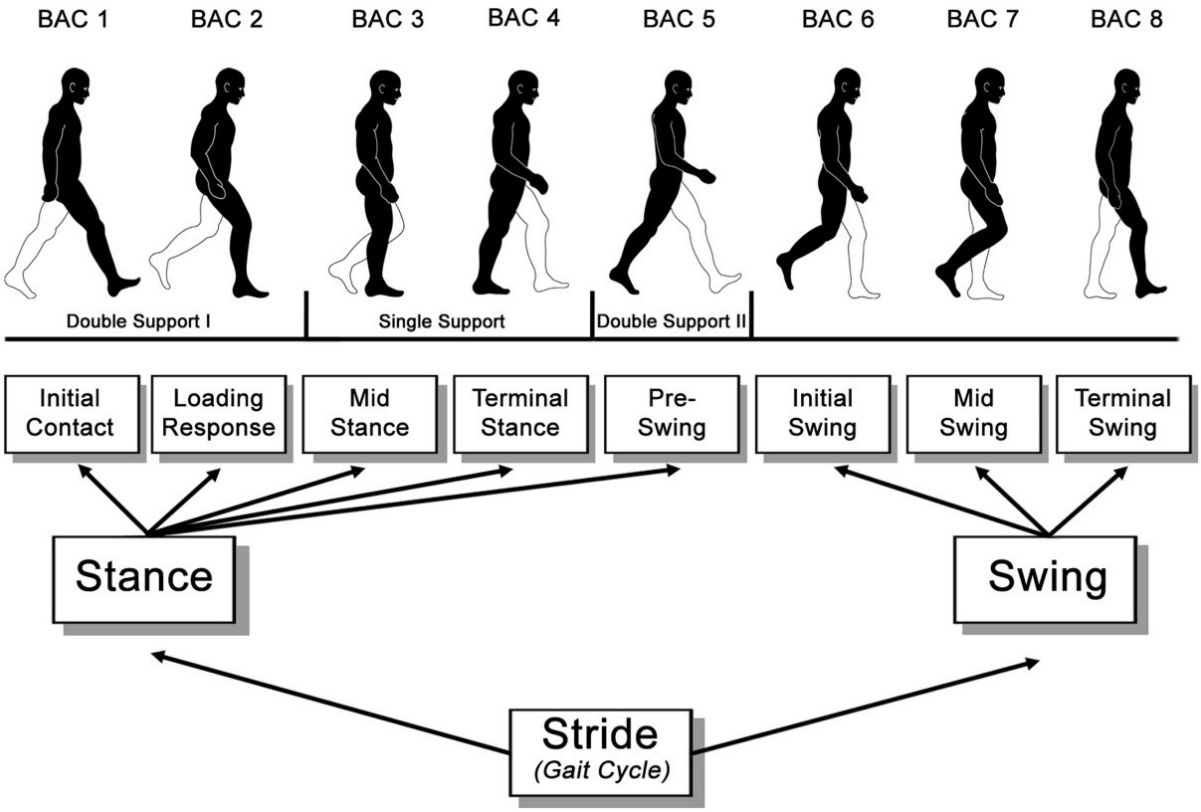


Figure 2.2: A Gait Cycle broken down into phases [1]

ments of two lower limbs is imperative in extracting spatial and temporal features. Figure 2.2 shows a breakdown of a gait cycle. As the figure illustrates, when both limbs are in the stance phase simultaneously, the legs are in bipedal or double support, and when only one leg is in the stance phase, it is in uni-pedal or single support sub-phase. Thus, according to [124], the swing phase has four sub-phases. **(1) Pre-swing** which is when the foot is pushed off the ground, and the transition between stance phase and swing phase happens, **(2) Initial swing**, when the foot clears off the ground. The **(3) mid swing** in which advancement of the foot continues, and **(4) terminal swing** that the foot is back in the position it was in the beginning of the gait cycle on the ground.

The traditional feature extraction algorithms in the literature are divided into two categories model-based, and model-free (motion-based) approaches. The model-based methods use the kinematics model of human Gait, and model-free or motion-based

approaches find correspondence between sequences of images (optical flow or silhouette) of the same Person and extract a gait signature. Table 2.2 shows examples of different works using model-based and model-free approaches. Another feature extraction category is based on deep learning, which has become more prevalent due to better performance and less complexity in discovering gait cycles in real-world situations.

Model-based Approaches based on the human body

Early research on Person gait recognition and Re-Id attempted to model the human body because Gait is essentially analysing the motion of distinct parts of the human body [92, 93, 94]. In these approaches, the silhouettes are obtained from 2D images by background subtraction and Binarisation. Specific characteristics pertaining to various body parts are extracted from each image in the sequence to form a model of the human body. Then the parameters will be updated over time for each silhouette in the sequence and used to create a gait signature. These characteristics are usually metric parameters such as the length and width of a body part like arms, legs, torso, shoulders and head and their position in the images. They can define the walking trajectory or the euclidean distance between the limbs. Twenty-two such features are introduced in [95]. [96] proposed a posture-based method for gait recognition which learned posture characteristics by considering the displacement of all joints between two consecutive frames and a fixed centre of body coordinate system for all the joints. A two-point gait representation was introduced in [94] which modelled the motion of limbs regardless of the body shape. Other works such as [97] create a motion model by considering the motion of hips and knees in various phases of a gait cycle. In

Table 2.2: Details of some traditional methods in the literature

Method	Approach	Feature	Feature Analysis Technique
[108]	Model-based	3D joint info	MMC PCA LDA
[97]	Model-based	Motion Model	Haar-like template for localization and magnitude and phase of fourier components for gait signature and KNN classifier
[119]	Model-free and Model-based	Fusion of depth information	Soft biometric cues and point cloud voxel-based width image for recognition; LMNN classifier
[43]	Model-free	2D Silhouettes	Gait and appearance features combined
[120]	Model-free	Silhouettes	STHOG and colour fusion
[118]	Model-free	Optical Flow	HOFEI
[116]	Model-Based	3D joint info	Context based ensemble fusion and SFS feature selection
[121, 39]	Model-Free	2D Silhouettes	ColHOG3D
[113]	Model-Free	3D joint info	Swiss system based cascade ranking
[91]	Model-Free	2D Silhouettes	Virtual 3D sequential model generation
[112]	Model-Free	2D Silhouettes	GEI-FDEI HSV
[115]	Model-Free	2D Silhouettes	Virtual 3D sequential model generation
[117]	Model-Free	Point Cloud	Frequency response of the height dynamics

this approach, the features are extracted from 2D images. Gait parameters, namely magnitude and phase of the Fourier components, are extracted using a Viewpoint rectification stage.

Because of the low quality of surveillance images captured in the real world, it is implausible to calculate a model robust enough for widespread practical usage. In recent years depth sensors (i.e. Kinect) and MOCAP systems made modelling the human body more manageable. With the help of a Kinect [125] extracted comprehensive gait information from all parts of the body using a 3D virtual skeleton. Other works such as [119] employed this method in gait recognition and automatic Re-Id systems. Two Kinects RGB-D cameras were used in [119] to acquire information from the person's frontal and back view. Since Kinects have a limited range for sensing depth, they each captured only a part of a subject's gait cycle, so they fused the information from all the sensors. The authors used the Kinect software development kit to compute a set of soft biometric features for Person Re-Id when the subject moves from one Kinect to the next. Then feature vectors called width images were constructed at the granularity of small fractions of a gait cycle.

Model free Approaches

The model-free approaches can again be categorised into Sequential and spatial-temporal motion-based methods. In sequential Gait is presented as a time sequence of the Person's poses. The sequential model-free approach was first proposed in [126] in which temporal templates represent the motion. The Temporal templates spot the motion and record a history of these movements. In a spatial-temporal model-free approach, Gait is represented by mapping the motions through time and space [98].

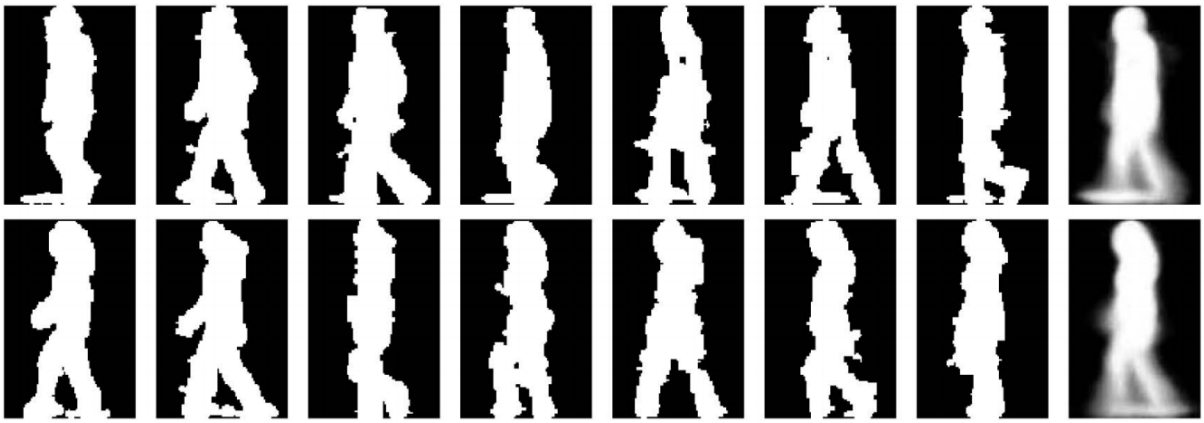


Figure 2.3: Gait Energy Images (GEI) generated over a gait cycle [2]

Different spatial-temporal methods are proposed in the literature, and almost all of them use silhouette sequences. Gait Energy Image (GEI), first introduced in [2] is a spatial-temporal representation of Gait that characterises the human walk properties for individual gait recognition. It produces a single image template by taking the average of all binary silhouettes over the entire gait cycle.

Figure 2.3 shows GEIs generated from sequences of human walk silhouettes. For binary gait silhouette images $B_t(x, y)$, the Gray level GEI is computed using the below formula where N is the number of frames in a sequence, t is the moment in time (frame number) and (x, y) are the position coordinates in a 2D image. GEIs and their variations became the primary feature for many Gait Person Re-Id research, including [112]. In this work, GEI and its variation called Frame Difference energy image (FDEI) were used as gait features. Other works such as [118], and [91] also used Histogram of Flow Energy Image (HOFEI) and Pose Energy Images, respectively (PEI). In [43] Gait Energy Images and appearance features such as HSV histograms were fused to deliver a spatial-temporal model.

$$G(x, y) = \frac{1}{N} \sum_{t=1}^{N} B_t(x, y), [2]$$

Figure 2.4: Optical Flow Images (GFI) generated over a gait cycle [3]

Optical Flow is also used as a feature in several past works. [3] proposed Optical Flow Images (GFI) for the first time in 2011. The basis of GFIs were also sequences of binary silhouette images. The process of GFI generation is depicted in Figure 2.4. Variations of optical flow based methods appeared in the literature in the next years. [127] used a Pyramidal Fisher Motion for Multi-view Gait Recognition, a descriptor for motions based on short-term trajectories of points. Other methods such as [128] and [129] are also based on silhouettes. [128] proposed a partial similarity matching method for Gait that could construct partial GEIs from 2D silhouettes and [129] generated virtual views by combining multi-view matrix representation and randomised kernel extreme learning. Spatial-temporal histogram of oriented gradient (STHOG) is proposed in [120]. They suggest that the STHOG feature can represent both Motion and shape data. [115] generated a syntactic 3D volume of the subject from images acquired by multiple overlapping 2D cameras and extracted features from them to be used in gait recognition.

2.2 Deep Learning-Based Approaches

Deep learning is a subarea of Machine Learning (ML) that tries to train computers on learning by example. This technique has been used before and is the key to technolo-

gies such as driverless cars, detecting a stop sign or distinguishing an object from a human being. Other applications can be in voice control devices such as Google Assistant, Apple Siri and Amazon Alexa projects. Deep learning is getting much attention from the research community because its results were not previously possible.

In this method, a computer model learns how to perform classifications using the training it previously got from sound, text, images or video frames and gain unbelievable accuracy that is sometimes better than humans. Vast collections of labelled data and neural networks with various levels are used to train deep learning models. Deep learning receives such impressive recognition because of its higher than before accuracy level. This helps big companies keep the users happy, meet their expectations, and decrease safety concerns in more critical projects like driverless cars. Recent developments show that deep learning transcends human beings in tasks like feature extraction in images.

In recent years deep learning approaches to gait recognition have been progressing fast [100, 103]. The idea of a Siamese Convolutional Neural Network (SCNN) was introduced in [130] and utilised in numerous research after that [131, 132, 133]. A Siamese network uses multiple identical sub-networks with the same weights and parameters to find a relationship between two related things. By mapping Input patterns into a target space and calculating their similarity metrics, they can discriminate between objects. If the objects are the same, the metrics will be small, and if they are different, the metrics are significant. Some applications of Siamese networks are face recognition, signature verification, and in recent years person re-identification [134, 135, 133]. Siamese frameworks are suitable for gait recognition scenarios since there are usually many categories and a small volume of sample data in each cate-

gory. Siamese CNN for gait feature extraction and Person Re-Id was first mentioned in [100] and then in [136]. Based on that [103] proposed a CNN-based similarity learning method for gait recognition, and [137] designed a cycle consistent cross-view gait recognition method by using Generative Adversarial Networks (GANs) to create realistic GEIs. To handle angle variations, authors in [129] proposed localised Grassmann mean representatives with partial least squares regression (LoGPLS) method for gait recognition, and authors in [138] offered an autocorrelation feature which is very close to Gait Energy Images. Multi-channel templates for Gait called Period energy Images (PEI) and Local Gate Energy Image (LGEI) with a Self Adaptive Hidden Markov Model (SAHMM) were introduced in [139] and [140] respectively.

All of the above work used GEIs as input data for their systems. As mentioned before, in the process of generating GEIs from gait sequences, a vast amount of Gait temporal data will be lost, so in more recent work, attention mechanisms [141, 38], and pooling approaches [142, 143] were used. A Sequential Vector of Locally Aggregated Descriptor (SeqVLAD) was proposed in [141] which combined Convolutional RNNs with a VLAD encoding process to combine spatial and temporal information from videos. Sparse Temporal pooling, line pooling and trajectory pooling were also used in various work to extract gait features [142, 143]. Some works, for example, [144] and [145] used raw Silhouette sequences instead of GEI or its variations to preserve the temporal features. In [145] they used ResNet and Long-Short Term Memory (LSTM). GAN based methods were used in recent works on large scale datasets [138, 146, 147]. Researchers in [148] even used RGB image sequences instead of Silhouettes with auto-encoder and LSTM networks. [149] even proposed a model-based gait recognition method with the use of CNNs, which did gait recognition as well as

Table 2.3: Details of most notable deep learning methods in the literature

Method	Feature Analysis Technique	Feature
[149]	Convolutional Nueral Networks (CNNs)	Joints relationship pyramid mapping (JRPM)
[139]	Generative Adversarial Networks (GAN)	Period energy Image (PEI)
[145, 148]	Convolutional Nueral Networks (CNNs) long short-term memory (LSTM)	Silhouette Sequences
[140]	Self-Adaptive Hidden Markov Model (SAHMM)	Local gait energy image (LGEI)
[144]	Convolutional neural network (CNN)	Silhouette Sequences
[146]	Generative Adversarial Networks (GAN)	Gait energy image (GEI)
[129]	Localized Grassmann mean representatives	Partial least squares regression (LoGPLS)
[103, 138]	Localized Grassmann mean representatives	Convolutional neural network (CNN)

predicting the angel. A list of some of the more important deep learning methods and their features and techniques is provided in table 2.3.

It is essential to mention that Various features can be fused to improve a Person Re-Id system's performance. However, noise and irrelevant information have to be eliminated. Many Feature Selection and Dimension Reduction techniques have been carried out in the past to eliminate redundant and irrelevant data collected in feature extraction [150]. Principal Component Analysis (PCA) is one of the most popular techniques for Dimension Reduction[114, 91]. Moreover, Probabilistic Principal Component Analysis (PPCA) [151] and multilinear Principal Component Analysis (MPCA) [109] have been used in Person Re-id. [108] even used a fusion of PCA and Linear Discriminant Analysis (LDA) to achieve better accuracy. [116] and [117] utilised algorithms such as KL-divergence and Sequential Forward Selection to achieve feature selection.

Drawbacks of Deep Learning Methods

Although deep learning was theorised in the 1980s, it was not recognised until recently. This lack of attention was due to two main reasons. Firstly, deep learning algorithms need large quantities of labelled footage or other data. For example, many subjects must be used for training in human recognition before testing the model's accuracy. Additionally, they need significant computing power. The advancements in Parallel architectures for High-Performance Graphical Processing Units (HPGPUs) help this issue immensely. These architectures can be combined with clusters or cloud computing and reduce the training time significantly.

Applications of Deep Learning Methods

Despite the disadvantages mentioned above, there are practical implementations of deep learning in industries such as **(1)** Automated Driving where objects like pedestrians and stop signs are automatically detected, which reduces the possibility of collisions, **(2)** medical research, specifically cancer research where deep learning is used to detect cancerous cells in a human body automatically. UCLA researchers Build a microprocessor that can detect and analyse 36 million images per second using deep learning and photonic time stretch for cancer diagnosis. **(3)** Aerospace and defence projects where using deep learning satellites can identify objects in areas of interest and safe zones for troops deployed in a specific area. **(4)** Industrial Automation, where the workers' safety is improved when working with heavy machinery by detecting when a person is at an unsafe distance from the machines. **(5)** Electronics for speech and voice recognition such as voice-assisted tools that translate speech into words or control devices around the house.

Gap of research in the existing methods

Almost all the approaches introduced in this section are well known for their superiority over the traditional methods, but they focus specifically on ”regular gait cycles”. In this thesis, we tried to close this gap by proposing an Attentional Spatial-Temporal system to solve the challenges in irregular gait recognition. Although existing methods have provided a better avenue for learning gait signatures and improved the gait recognition accuracy to a certain extent, most of them need to detect at least one complete gait cycle precisely and are not robust to the viewpoint changes. Therefore, irregular and specially view-invariant gait recognition still needs particular attention, which has been accomplished in this thesis by discovering a novel solution for reducing the effect of irregular Gait on Person Re-Id problems.

The following section provides an overview of the deep learning techniques employed to develop the approaches in this thesis.

2.3 Employed deep learning methods

Neural networks (NNs) are the basis of deep learning methods. They are sometimes called deep neural networks (DNNs) because of the number of hidden layers between the input and the output. Modern networks can be as deep as 200 layers, while classic neural networks consist of only two to four hidden layers. The models produced for deep learning rely on training by using large datasets of labelled data and architectures of neural networks that learn to extract features or learn from the frames without feature extraction. Figure 2.5 shows a neural network with two hidden layers that includes a set of interconnected nodes.

As stated before, deep learning is a type of machine learning with significant differences in the workflow. In machine learning, the features are extracted manually from images. Later, they create a specific model that labels the object in an image. However, deep learning provides automatic feature extraction as well as ”end to end learning” [152] which means giving a task and the raw data to the network and expecting it to learn how to do the said task automatically. Deep learning algorithms also can be extended with more raw material (data). Hence it reveals one of the critical advantages of deep learning, but co-generation of ”Shallow Learning” [153] might be a problem. Machine learning methods that cannot be improved performance-wise after a certain point are shallow.

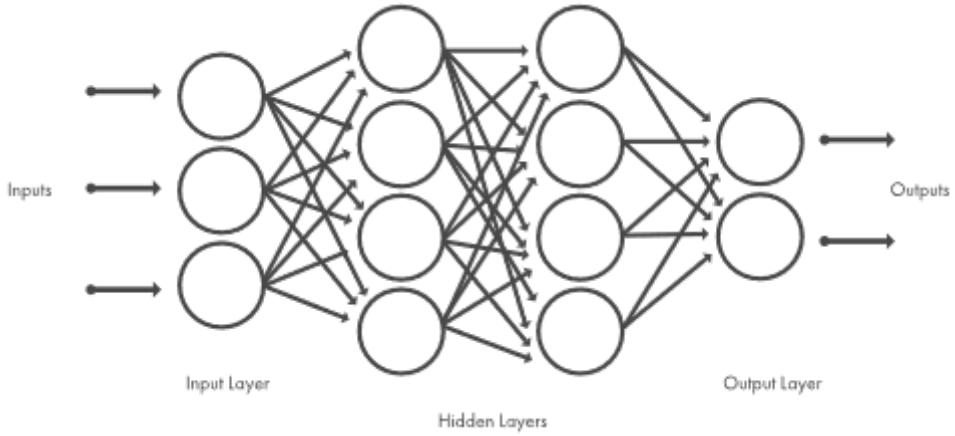


Figure 2.5: A typical Artificial Neural Network (ANN)

Mathematically a Neuron is modelled as the below equation in which x is an input vector, and W is the weight vector. These vectors are added together with a bias b and transformed with a usually non-linear activation function of σ. Figure 2.6 shows some activation functions to choose from. The most chosen functions are the sigmoid, hyperbolic tangent (tanh), or the Rectified Linear Unit (ReLU) function. Since artificial neurons on their own can only perform simple computations, they are commonly arranged in an acyclic graph to perform more complex classification operations.

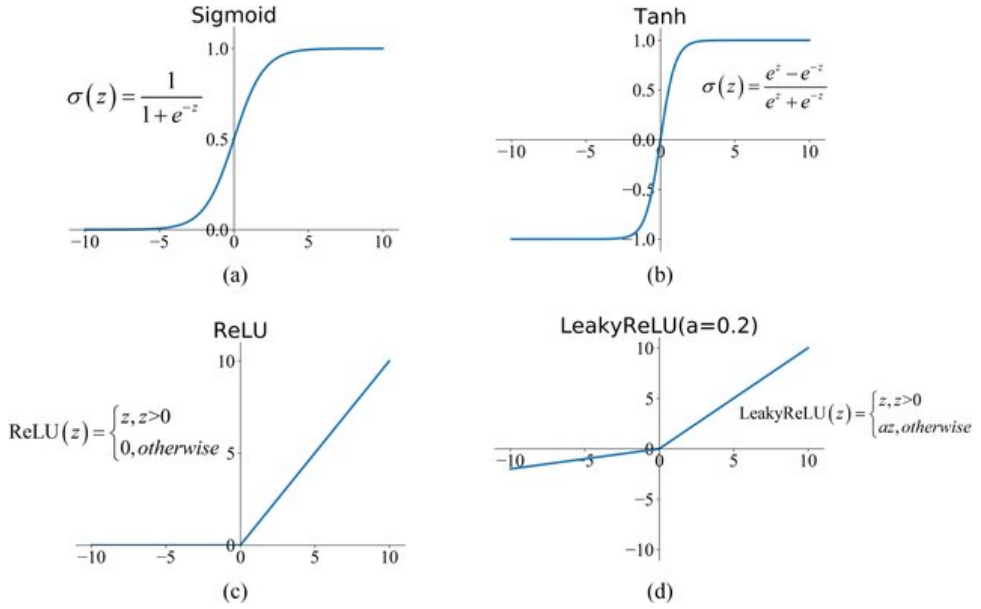


Figure 2.6: Some of the well-known activation functions used in Neural Networks [4]

$$y = \sigma(\sum_{i} w_i x_i + b) = \sigma(w^T x + b) \tag{2.1}$$

An acyclic graph of neurons where the input does not depend on the output is called a Feedforward Neural Network (FFNN) since the activations can be propagated forward. As opposed to the Recurrent Neural Network (RNN), the topology includes cyclic connections. Figure 2.7 illustrates the differences between these two Architectures. Unlike RNNs, there are no feedback loops in Feed Forward Neural Networks, and their basic structure only allows for forwarding connections between the neurons.

Recurrent neural networks have a chain-like structure, making them especially suited for sequences and time series operations. Some applications of RNN are forecasting stocks, analysing DNA sequences, Natural Language processing and, in recent years, gait temporal feature extraction. Typical feedforward networks, including Convolutional Neural Networks (CNNs), only consider the current input and do not remember the past, so they have trouble predicting the next step in the sequence.

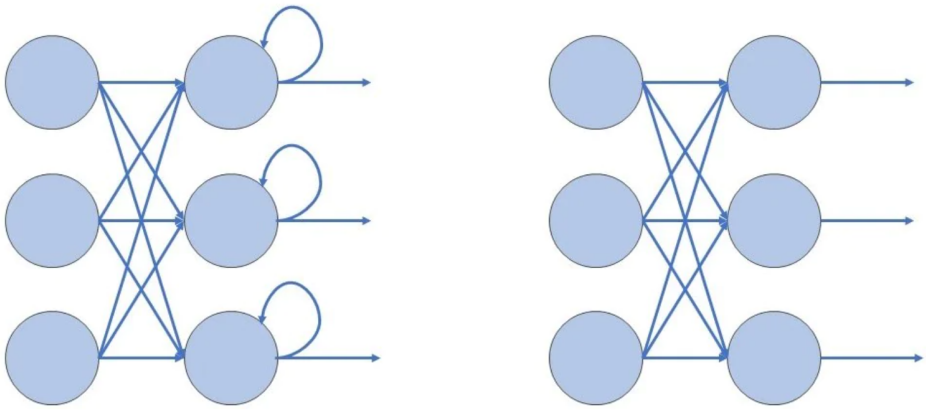


Figure 2.7: Difference of architecture between Feed Forward and Recurrent Neural networks

Back-propagation

A universal algorithm for feedforward neural networks is backpropagation [154]. It is essentially the Gradient Descent Algorithm applied to Neural Networks. Backpropagation can be divided into two steps: **(1)** A complete forward step from the input to the output layer just to compute the activations and **(2)** A backward step from the output to the input layer to update the weights by backpropagating and compute the errors. A cost function E is computed among the computed outputs and their targets. The below equation calculates the gradient of the cost function concerning weights. h_j^l is the output of the l^th neuron of the j^th layer, and g_j^l is the activation function of the l^th neuron of the j^th layer. $W_{j,i}^{k,l}$ is the weights from k^th neuron of the i^th layer and the l^th neuron of the j^th layer.

$$\frac{\partial E}{\partial W_{i,j}^{k,l}} = \frac{\partial E}{\partial h_j^l} \frac{\partial h_j^l}{\partial g_j^l} \frac{\partial g_j^l}{\partial W_{j,i}^{k,l}} \tag{2.2}$$

Same equation can be written as below where σ is the activation function:

$$\frac{\partial E}{\partial W_{i,j}^{k,l}} = \frac{\partial E}{\partial h_j^l} \times \sigma' \times h_i^k \tag{2.3}$$

The input of the activation function of the $j + 1^th$ layer is the weighted sum of the outputs of the neurons in the previous layer. The gradient of the cost function with respect to the outputs can be represented as the recursive formula below where the computation propagates backwards:

$$\frac{\partial E}{\partial h_j^l} = \sum_m \frac{\partial E}{\partial h_{j+1}^m} \sigma'(g_{j+1}^m) W_{j,j+1}^{l,m} \tag{2.4}$$

So the gradient of the Loss concerning all the weights can be calculated using Eq. 2.3. Then weights could be updated by gradient descent with a learning rate of α:

$$\Delta W_{i,j} = -\alpha \frac{\partial E}{\partial W_{i,j}} \tag{2.5}$$

If the coefficient η is used as momentum to integrate the update calculated in previous iterations, Then this process could be accelerated by:

$$\Delta W_{i,j}^t = -\alpha \frac{\partial E}{\partial W_{i,j}^t} + \eta W_{i,j}^{t-1} \tag{2.6}$$

2.3.1 Convolutional Neural Network (CNN)

Convolutional Layer

In computer vision, the input of a neural network is usually an image. Images used as input could have different sizes up to several tens of thousands of pixels and are represented in a computer as Matrices. A fully connected neural network will be able to process this input matrix, but the number of parameters to consider and the computational power needed for training such a network would be very large. To avoid overfitting and overcome the problem of large computations such as these, a Convo-

lutional Layer was introduced in the 1980s by Yann LeCun, a postdoctoral computer science researcher. The neural network with a convolutional layer became the first successfully trainable neural network for images to that date. Two basic ideas were used in COnvolutional Layers: **(1)** Shared wights and **(2)** local receptive fields, both to reduce the complexity of computation. Basically, in a Convolutional Neural Network (CNN), a matrix will be received as input data.

In contrast with a fully connected neural network, the hidden neuron will be connected to only a tiny region of the input layer. This region is a local receptive field for the hidden neuron since the spatial correlation is local. Furthermore, all the connections from input to the hidden layer share the exact weights, which is especially useful to find a particular feature in an image with reduced complexity. When applied, these two essential parameters will transform a fully connected neuron (E.q. 2.1) into a convolutional layer as follows:

$$y = \sigma(W * X + b) \tag{2.7}$$

X is the input matrix, W is the weight matrix, also referred to as a filter or kernel, and $*$ symbol represents the Convolution operation, and b is a bias added to the equation. Also, σ represents the activation function as previously mentioned and could be chosen from one of the functions in Figure 2.6. Since Images are two-dimensional matrices, the convolution operation can be modelled as below:

$$(W * X)(i, j) = \sum_{m} \sum_{n} X(m, n) W(i - m, j - n) \tag{2.8}$$

As shown in Figure 2.8, the output of the A convolutional layer is the dot product of the

weight matrix slid over the input matrix. The resulting matrix is called a ”Feature Map” or activation map. Filters can detect various features in Image processing and perform operations such as edge detection, image sharpening, blurring just by changing the numeric values. The kernel can be learnt automatically with backpropagation, and one layer commonly has more than one kernel, which results in feature maps.

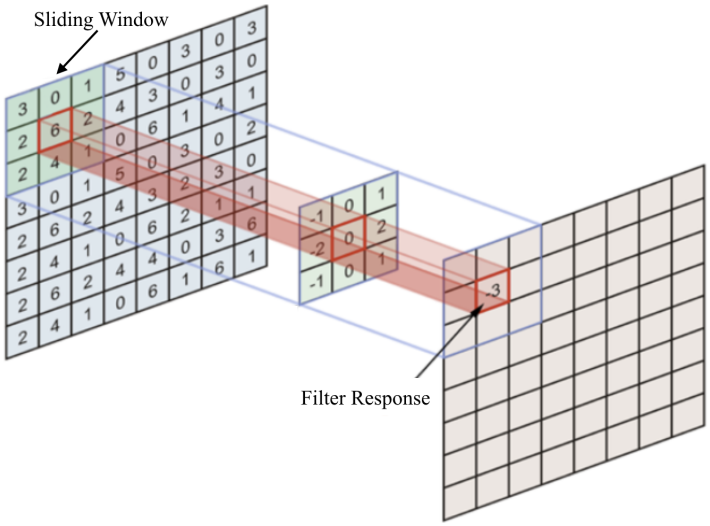


Figure 2.8: Operation of a convolutional layer convolving a 3×3 filter with a 8×8 matrix [5]

CNN architecture

Convolutional Neural Networks (CNN or ConvNet) are the most popular deep neural networks. These kinds of networks are best suited to process image and video data. A CNN convolves or coils together with the input data with filters to learn features during training. Hence, it removes the need to extract the features needed to classify the images manually. The complexity of the features learned from the images increases by the number of hidden layers. For instance, the first layer can detect the edges the last layer can learn to detect more complex objects. CNN is a subcategory of deep neural network with discriminative architecture [155], which have shown highly satisfactory results for processing 2D (two dimensional) data such as images and videos. Animal

visual cortex organisation was the inspiration behind CNN's architecture. The concept of receptive fields was introduced in the 1960s, which proves that an animal's visual cortex that is in charge of light detection has a complex and mappable cell arrangement [156].

CNN was inspired by Time Delay Neural Networks (TDNN), in which the weights are obtained and shared in the time vector (a temporal dimension) that significantly reduces the necessary computations. Convolution is used in the CNNs, instead of the normal matrix multiplications in a standard neural network so that the network can be less complex as the number of weights decreases. Moreover, there is no need for a separate feature extraction process as it can be done during training on the images fed as raw inputs to the CNNs. This detail is a significant improvement over the standard learning algorithms. Because of the successful training in the hidden layers, CNNs are considered the first successful deep learning architecture in decades. To reduce the number of parameters in the network, they use spatial relationships. They also use backpropagation algorithms to improve and optimise performance. As a result, CNNs need minimal to no preprocessing operation. Also, GPU-Accelerated computing techniques have been employed to train the networks with less cost. Some of the practical applications of CNNs are handwriting recognition, face detection, behaviour detection, including violence and fight detection, speech recognition, classification of images and Natural Language processing.

CNN Learning process

For CNN to learn, three key factors should be considered [157]. These factors are **(1) Sparse interaction**, **(2) Parameter Sharing**, **(3) Equivariant representation**. In-

stead of using normal matrix multiplication in classic ANN, CNN reduces the amount of necessary computation with sparse interaction, meaning that kernels will be made smaller in size than the original input image. Parameter sharing means that the network needs to learn only one set of parameters instead of learning a separate set in each location. Parameter sharing increases the performance in CNN. Equivariance is a property of the CNN, which means that the input and the output will change in the same way, and Fewer parameters are required for the network, which reduces the memory needs and improves efficiency.

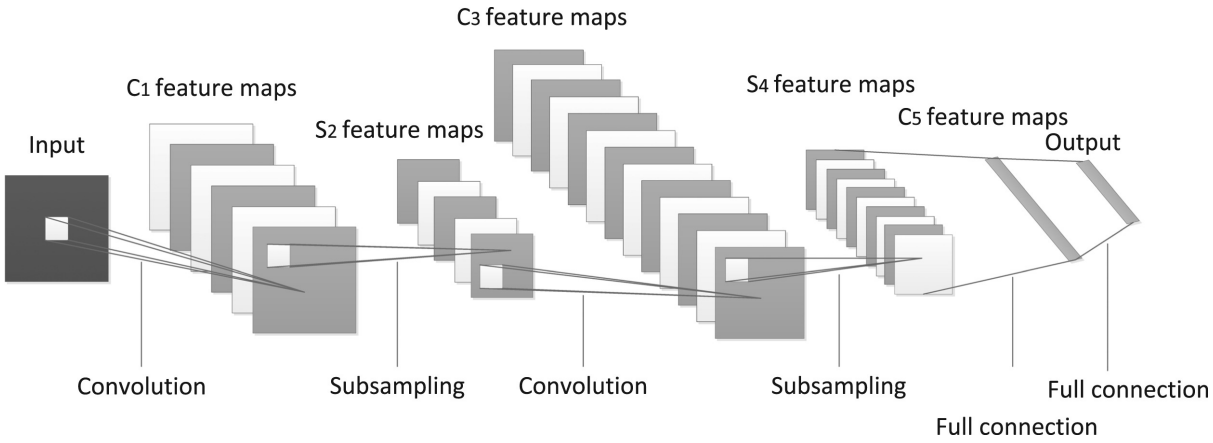


Figure 2.9: Standard schematic structure of Convolutional Neural Network (CNN) [6]

The Architecture of a typical CNN and a standard CNN layer's inner workings are illustrated in Figures 2.5 and 2.6. As it can be seen, in each layer of CNN, one of the two operations might happen, Sub-Sampling or Convolution [6]. Figure 2.9 shows how the input image is convolved with trainable filters to produce the first convolutional layer. Each filter has a layer of weights, typically four pixels from a group in the feature map. Then the pixels go through a Sigmoid activation function and construct the first layer that contains more feature maps with smaller dimensions. This process carries on and produces more feature maps in the successive layers. Lastly, an output vector is generated from the values [155]. As mentioned earlier, in a Convolutional layer,

each neuron's input is connected to the local receptive of the layer before. Features are extracted and passed along this way, and then their relationship will be figured out simultaneously. Fundamentally, a sub-sampling layer is utilised for mapping the features. These layers form a plane by sharing weights. The Sigmoid is used as an activation function because it affects the kernel, which will be done to attain scale invariance.

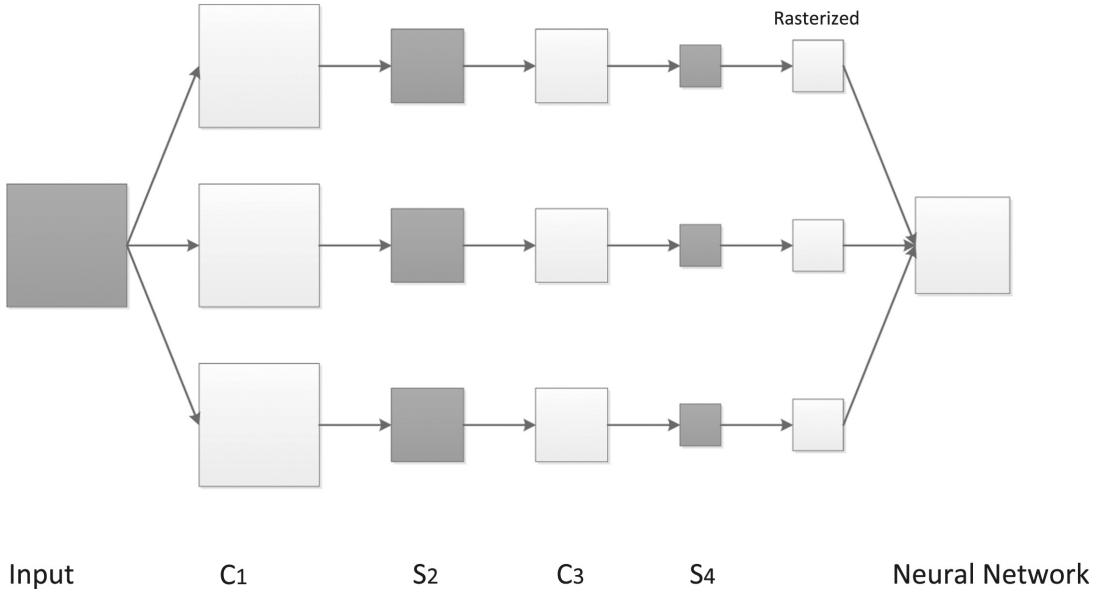


Figure 2.10: Conceptual structure of Convolutional Neural Network (CNN) [6]

There are several notable neural networks worth mentioning in the literature, for example, LeNet [158] for handwritten text recognition, which contains convolutional, pooling and fully connected layers. The convolutional layer is used to learn features from the input image automatically. Pooling layers combine the information in the region to reduce the need for more computation, and the fully connected layers, which look like an ordinary feedforward neural network, perform classification. Another example is the winner of the 2012 ImageNet Large Scale Visual Recognition Competition (ILSVRC). AlexNet [16] with five convolutional layers, and three fully-connected layers achieved the best results on the ImageNet dataset with 15 million high-resolution images from

22,000 categories (classes). The Authors proposed the Rectified Linear Unit (ReLU) activation function. As shown in Figure 2.6, the derivative of Relu is zero when x is equal or less than zero and becomes one when x is greater than zero, which solved the vanishing gradient problem with the Sigmoid function. Visual Geometry Group (VGG) invented the VGG network [159] which is the same as AlexNet, but instead of huge kernels, they used multiple 3×3 filters in a row. More complicated features were learned by increasing the depth of the network, and better performance was achieved. There are multiple types of VGG. VGG16 has sixteen layers, and VGG19 has nineteen layers. Figure 2.11 shows the architecture of VGG16 in which the convolution and pooling operation is repeated multiple times.

Figure 2.11: VGG16 Architecture [7]

Deep Residual Network (ResNet) was introduced by [8]. The authors noticed that by increasing the depth of the network, accuracy becomes saturated and degrade more rapidly. They pointed out that the accuracy of a network should not degrade as it becomes more prominent in size. Deeper neural networks should at least have the same ability as a more shallow neural network since they should quickly learn the identity function. If a network with the ability to predict $f(x) = x$ is attached to an existing network, it must logically output the same results. Therefore, we must keep adding more identity layers and get the same answer. However, when training

an extensive network, the performance degrades. To solve this problem, the authors came up with a side branch or ”skip Connection” shown in Figure 2.12. They proposed to learn a residual function $F(x)$ instead of a direct mapping function $H(x)$ as defined in the equation below, where x is the identity function:

$$F(x) = H(x) - x \tag{2.9}$$

Since $F(x) = x$ for identity functions, e.q. 2.9 could be reformed as:

$$H(x) = F(x) + x \tag{2.10}$$

To expand this idea for the ImageNet dataset, they used a shortcut branch to add the input of one layer to the output after every two layers. Therefore, driving the new layer to learn something more complex from the encoded information. It also solves the vanishing gradient problem using the identity connection. Figure 2.13 shows the architecture used for ImageNet put against the VGG network.

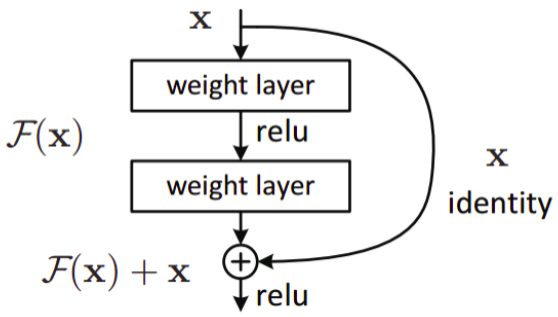


Figure 2.12: A Residual learning block [8]

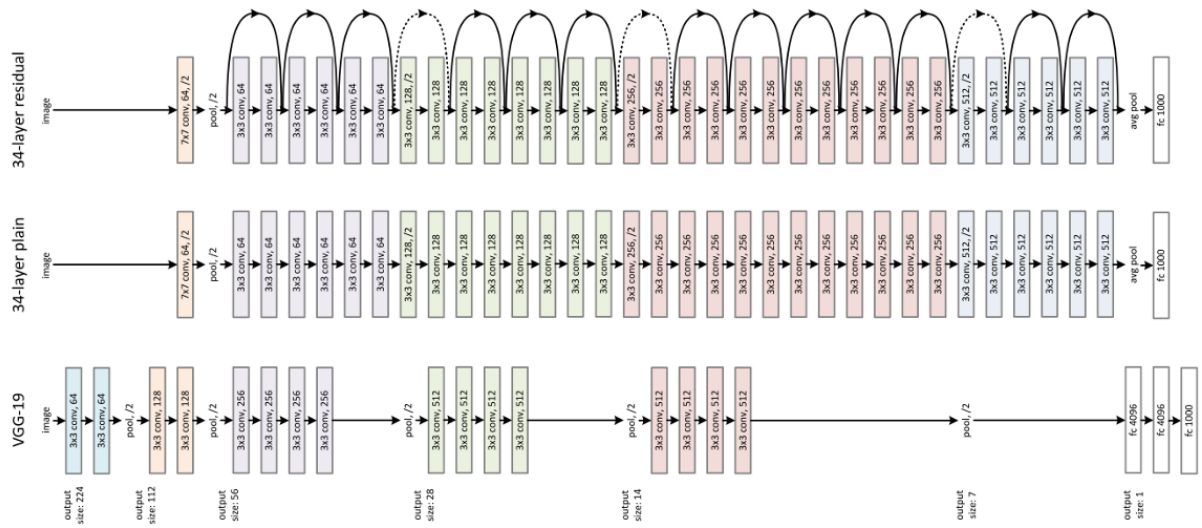


Figure 2.13: The ResNet architecture [8]

2.3.2 Siamese and triplet neural Networks

Siamese Neural Network was first introduced in [134] for signature verification by using Time Delay Neural Network (TDNN). They produce an output vector by receiving two samples at the input. They contain two identical subnetworks. For the network to be called siamese, it is important for the weights to be shared among the subnetworks. They are repeatedly presented with pairs of True and False examples to learn a similarity metric with non-linearity. The sample could be presented from the same or two different classes in a classification problem. The central idea behind siamese networks is that the learnt descriptors could be utilised to compare the respective subnetworks' inputs. Moreover, inputs could be numerical data, Images, with CNNs as subnetworks or data sequential in nature, for example, sentences or time signals with RNNs as subnetworks. The architecture for Siamese is presented in figure 2.14.

When training Siamese networks for Image classification, the idea is to map the input vectors into a non-linear subspace. The Euclidean distance in the subspace should be approximately the same as the input space's semantic distance. So if two images are in the same category, they should have a small distance, and if they are from dif-

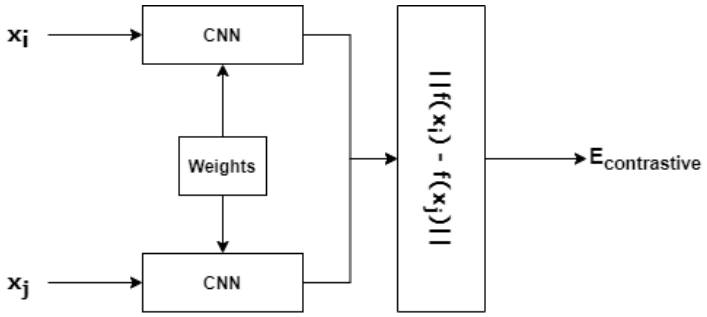


Figure 2.14: The architecture of a Siamese neural network

ferent categories, the distance should be significant. This idea was used by [130] for face classification with Siamese CNNs in 2005. The most popular loss function for Siamese is contrastive Loss. This Loss minimises and maximises the Euclidean distance between similar and different points, respectively. For an input pair of x_i and x_j, the constant margin of m and y as the label of the input pair, the contrastive Loss of E is calculated as below where the y parameter equals 1 for positive, and 0 for negative pairs and f is the projection of the neural network.

$$E_{contrastive} = \frac{1}{N}\sum^{N} y.||f(x_i) - f(x_j)||_2^2 + (1-y).max(m - ||f(x_i) - f(x_j)||, 0)^2 \quad (2.11)$$

The shared weights between subnetworks guarantees that the distance metric is symmetric during the learning process, which means the distance between input a and b is equal either way. The training is done using regular backpropagation, and the gradient is calculated across the subnetworks as below where α represents the learning rate, and W is the weight matrix:

$$\Delta W = -\alpha(\frac{\partial E}{\partial f(x_i)}\frac{\partial f(x_i)}{\partial W} + \frac{\partial E}{\partial f(x_j)}\frac{\partial f(x_i)}{\partial W}) \quad (2.12)$$

As the name suggests in triplet networks, three subnetworks are shown in figure 2.15. the input is inserted as a triplet of images where x_a is an anchor sample, x_p

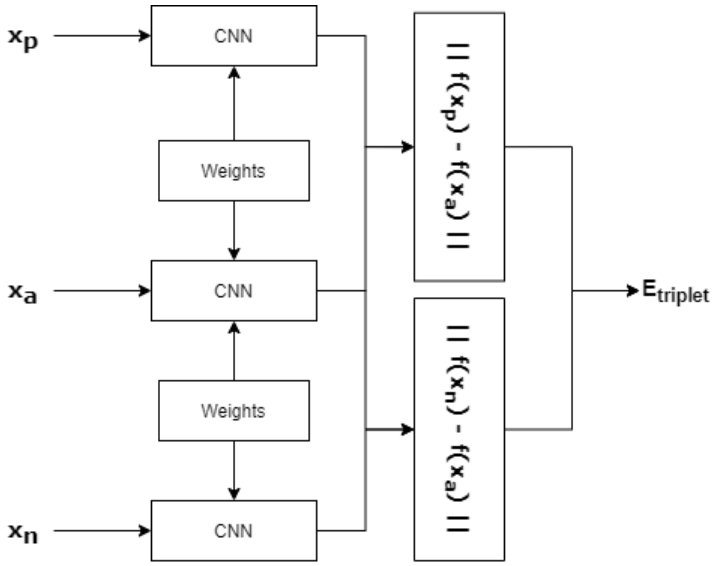


Figure 2.15: The architecture of a Triplet neural network

is a positive sample of the same person, and x_n is a negative sample of a different person. The same principle for the shared weight from Siamese networks apply to triplet networks, but the loss function is calculated based on relative distance rather than Euclidean distance. The loss function is as follows:

$$E_{triplet} = \frac{1}{N}\sum^{N} max(||f(x_a) - f(x_p)|| - ||f(x_a) - f(x_n)|| + m, 0) \tag{2.13}$$

In this equation, m is the constant margin, x_a is the anchor sample input, x_p positive sample input, and x_n is the negative sample input. The network updates itself when the positive image is further away from the anchor than the negative image. Essentially, during training, the loss function pushes the negative sample away from the anchor and pulls the positive samples closer.

2.3.3 Recurrent neural networks (RNNs)

Recurrent Neural Networks (RNNs) use time series or sequential data, as mentioned before in the chapter. They are primarily used to solve temporal problems such as Nat-

ural Language Processing (NLP), Speech Recognition and Gait Recognition. Similar to feedforward and convolutional neural networks (CNNs), recurrent neural networks learn by practising on training data. However, they are distinctive by their ”memory” as they take information from preceding inputs to affect the current input and output. While traditional deep neural networks assume that inputs and outputs are independent of each other, recurrent neural networks' output depends on the sequence's prior elements. While future events would also help determine the output of a given sequence, unidirectional recurrent neural networks cannot account for these events in their predictions. Figure 2.16 shows the architecture of a traditional RNN.

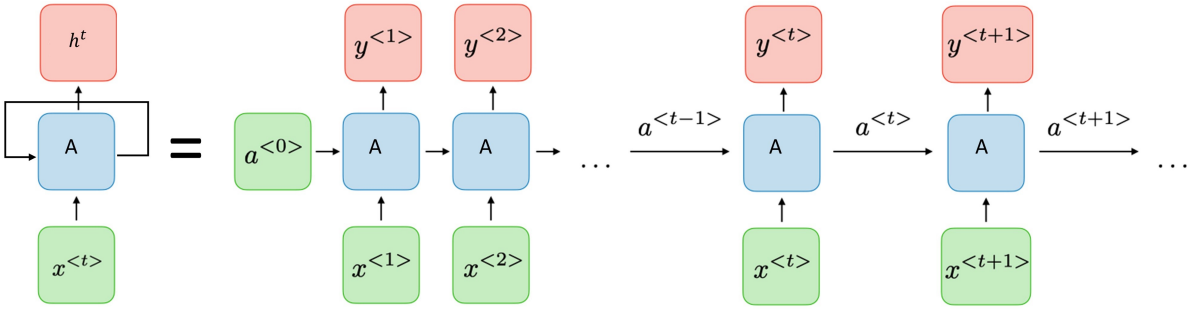


Figure 2.16: The architecture of a traditional RNN [9]

The symbols $a^{<t>}$ and $y^{<t>}$ are the activation and output for each time-step t respectively which are represented as the following equations [9] where g_1 and g_2 are activation functions and W_{ax}, W_{aa}, W_{ya}, b_a and b_y are shared temporary coefficients:

$$a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a) \tag{2.14}$$

$$y^{<t>} = g_2(W_{ya}a^{<t>} + b_y) \tag{2.15}$$

The loss function of all timesteps in RNNs is calculated based on the Loss at every timestep:

$$L(\hat{y}, y) = \sum_{t=1}^{T_y} L(\hat{y}^{<t>}, y^{<t>}) \tag{2.16}$$

2.3.4 Long-Short Term Memory (LSTM) neural networks

Long-Short Term Memory (LSTM) neural networks are a more advanced version of Recurrent neural networks, which have much more success in language modelling, translation, speech recognition, image captioning, and recent years, action recognition. To understand the need for LSTM, we first need to understand why traditional neural networks are not practical for modern problems. As is clear to the reader, the trail of thought in a human is not repeated from scratch every minute. When humans watch a video or read a book, they understand each sentence based on their understanding of the previous frames or sentences. Thus our thoughts are persistent. A major drawback of traditional neural networks is their inability to remember. Recurrent Neural Networks address this issue by using loops that allow for the persistence of information. Figure 2.16 is an unrolled RNN that represents one loop. Their chain architecture implies that they are appropriate for dealing with sequential data, for instance, frames in a video, by connecting the previously extracted information to the present task. The RNNs have the ability to do this, but they suffer from short term memory. If a sequence is very long, they will forget the earlier information and not carry it to the later timesteps. So if we are dealing with a long sequence of images (in a video clip), the RNN will forget lots of important information from the earlier frames. RNNs have the vanishing gradient problem during backpropagation which does not contribute much to the learning process. Naturally, the earlier layers in the chain with minor gradient updates stop learning. As the gap grows, RNN forgets what it had seen

earlier in a long frame sequence. In theory, this problem could be solved by parameter tuning, but as explained in [160, 161], there are fundamental reasons why such an action is impossible.

LSTM is a variant of the RNN model capable of solving this problem by learning long term dependencies. LSTMs were first introduced in [162] and perfected over the years. As shown in Figure 2.17, In an RNN loop, the repeating module contains a single layer, usually with a tanh function. The architecture of LSTM is very close to RNN, with an almost identical chain-like formation. However, the repeating module has a more complicated structure.

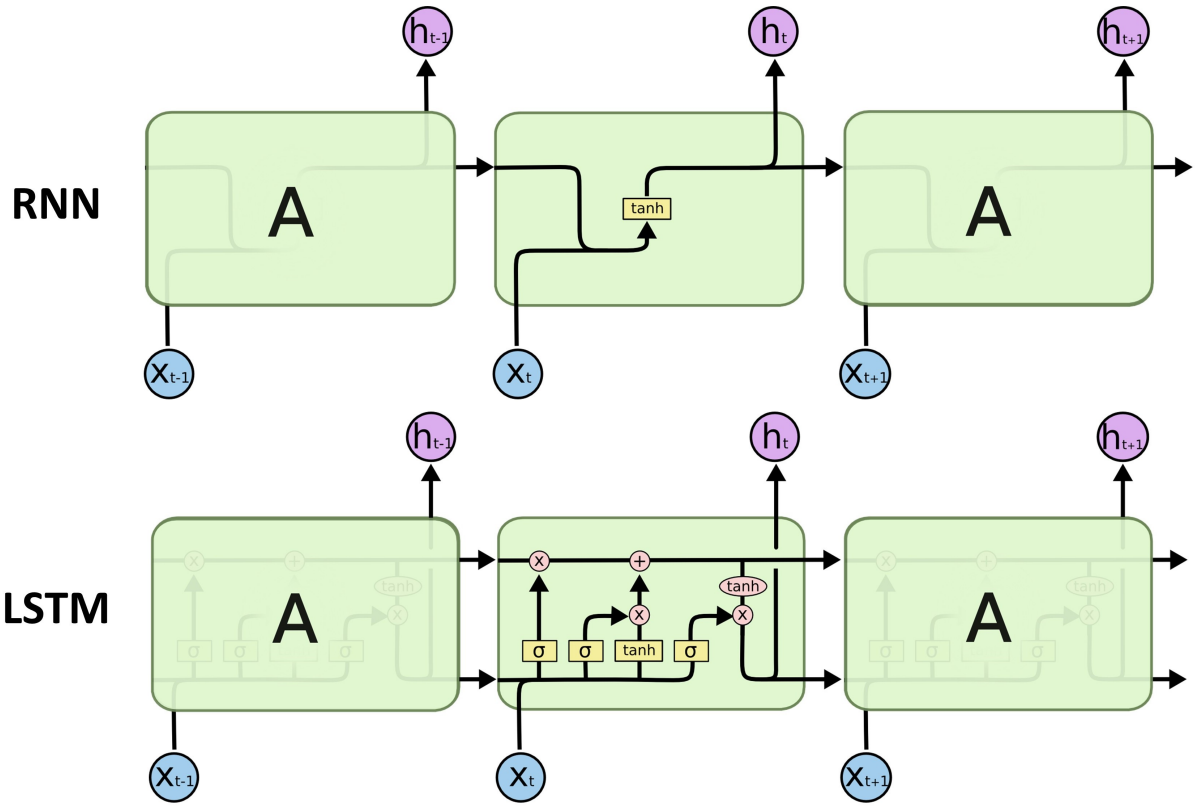


Figure 2.17: A comparison between RNNs and LSTMs

The core idea behind LSTMs is the cell state at time t, which C_t denotes. The cell state is the flow of information going through the chain at each timestep. Using gate structures, LSTM will add or remove information to the cell state optionally. Each gate

comprises a pointwise multiplication with the cell state and a sigmoid activation function σ. The sigmoid activation function acts as a regulator of the information allowed on the cell state. It outputs a number between zero and one, in which one is all the information and zero is none of the information. A typical LSTM like Figure 2.18 has a forget gate, an input gate and an output gate. The symbol X represents scaling of information by pointwise multiplication, +, adds information to the cell state by pointwise addition, σ, is a layer with a sigmoid activation function, tanh, is a layer with a tanh activation function, h_{t-1} and C_{t-1} are the output and memory from the last LSTM unit respectively, and x_t, c_t and h_t are the current input, memory and output variables.

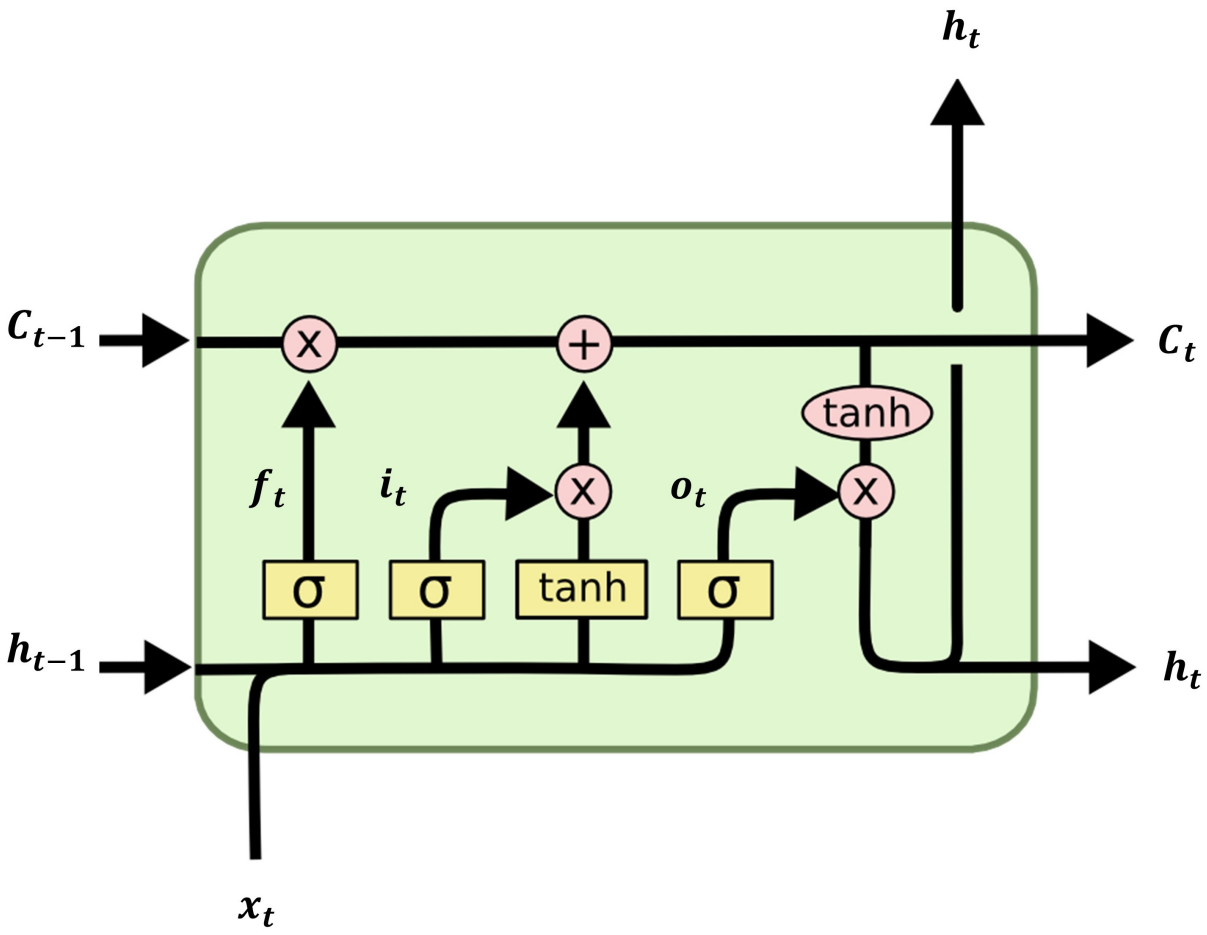


Figure 2.18: Illustration of a typical LSTM

LSTM consists of a forget gate (f_t), an input gate (i_t) and an output gate (o_t) at

each timestep. The forget gate decides what information to throw away (forget) and what information to keep. The previous module's information, h_{t-1}, and the current input, x_t, are passed to the sigmoid function, which produces a number between zero and one for each number in the C_{t-1}. The closer the value to one means to keep, and closer to zero means to forget. The below equation shows how the forget gate works mathematically.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \tag{2.17}$$

The input gate and a tanh layer update the cell state with new information. To decide what new information should be carried to the next level, h_{t-1} and x_t are inserted into the sigmoid input gate, which returns a value between 0 and 1. The same inputs are also inserted into the tanh activation function to regulate the network by outputting a vector of candidate values between -1 and 1 denoted by g_t. Both outputs are then multiplied, and the input gate's sigmoid function will decide what should be updated on the cell state. The below equations represent the output of the input gate and the tanh layer.

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{2.18}$$

$$g_t = tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{2.19}$$

Now that enough information has been gathered, the cell state will be updated from c_{t-1} to c_t. First the old cell state will be multiplied by the output of the forget gate (f_t) and then the pointwise multiplication of the input gate and the the ouput of the tanh layer ($i_t \times g_t$) will be aded to create the new cell state.

$$C_t = f_t \times C_{t-1} + i_t \times g_t \tag{2.20}$$

Finally, the output gate (o_t) decides the next hidden state (h_t), which contains information on previous inputs and is used for predictions. First, the previous hidden state and the current input are passed to the output gate's sigmoid function, and then the new cell state will be passed to a tanh function. Next, these two functions' outputs are multiplied to decide what information should be carried forward to h_t.

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{2.21}$$

$$h_t = o_t \times tanh(C_t) \tag{2.22}$$

The LSTM described in this section is a standard LSTM. However, there are multiple variations of LSTM in the literature, with slight differences in each academic paper. Some of these papers are mentioned in this chapter. In addition, a unique variation of LSTM has been used in chapter 4 of this thesis to improve our gait human recognition system's performance.

2.4 Chapter Summary

In this chapter, the literature surrounding person Re-Id was discussed and taxonomy with three different paradigms based on Data Acquisition, Pose and angle and feature extraction was presented. Moreover, multiple techniques for feature extraction and motion encoding cross-matching techniques in person Re-Id was expanded upon and discussed to be used in the later chapters of this thesis. In the next chapter, a comparative spatial-temporal study with multiple modality fusion for gait feature extraction was conducted, and the effects of each modality on the gait recognition was explored.

Finally, a spatial-temporal approach based on the fusion of modalities was proposed and tested on two challenging publicly available datasets.

Chapter 3

Network-modality cross comparison for a two-stream spatial-temporal architecture in gait feature extraction

3.1 Introduction

This chapter proposes a multiple modality spetial-temporal architecture for gait recognition to efficiently exploit the available labelled data. The experimental study in this chapter is directed towards identifying the best architecture to extract spatial features using 2D information from a frame sequence and temporal features from 3D spatial-temporal information from a subset of video frames with high accuracy.

To design a robust architecture spatial-temporal feature extraction in irregular gait sequences, a combination of 2D and 3D CNNs has been tried out, and a also a Resnet architecture has been proposed in this chapter. Fusion of modalities have been used in the past for handcrafted features, Optical flow maps, grayscale and silhouette im-

age sequences to improve the robustness of gait recognition methods [163]. Similar feature fusion could be performed on different combinations of modalities and networks [164], but it has never been tested on irregular gait sequences and viewpoint variations for robustness. Therefore, a comparative study of state-of-the-art CNN architectures using these modalities as inputs is necessary to test our theory on irregular gait sequences, which is the aim of this thesis. The main objective is to find the best network/modality combination for gait recognition which could perform well on irregular gait sequences generated from publically available datasets and therefore be robust enough in real-world scenarios.

3.2 Overview

Two well-known publicly available datasets, CASIA-B and TUM-GAID, are used to test this architecture. A comparative approach has been used in this chapter on various state of the art networks are tried out, and finally, an original network based on [8] is proposed, which has shown the most promising results for this task. Several types of input modalities are also considered to generate a single gait signature from the input sequences, including handcrafted high-level descriptors, grayscale images, silhouette sequences, and optical flow maps [165]. Moreover, information fusion at different levels has been studied for the final two-stream Spatial-Temporal CNN architecture. The most challenging part of this approach was consolidating temporal features and preserving them as well as the spatial features for use in high layered networks without losing information along the way. All modalities were tried out on the range of our proposed architectures and compared against state of the art. We also attempted to fuse two parallel networks at different architectural levels with various fusion techniques

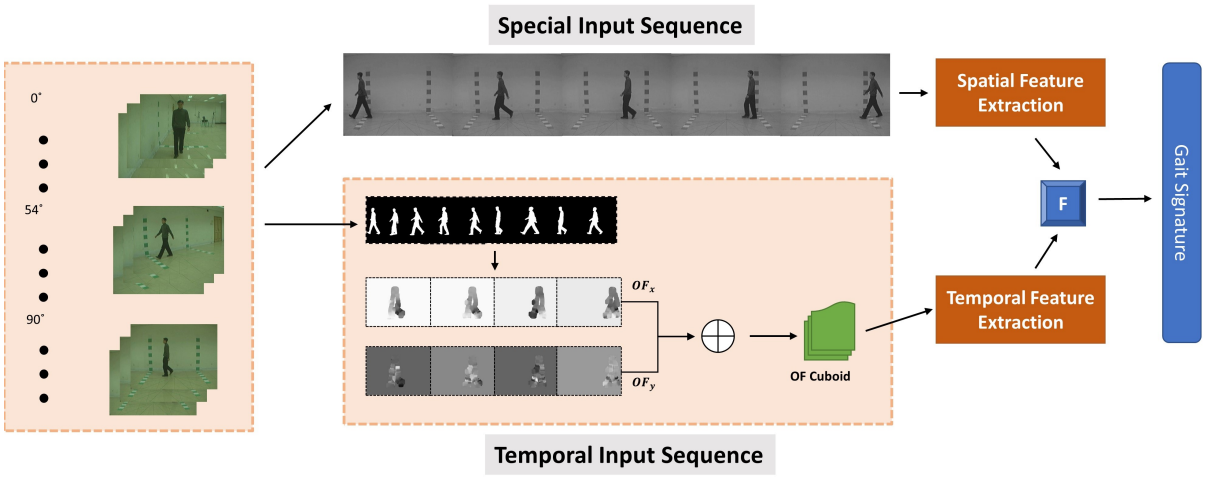


Figure 3.1: High level illustration of our proposed two stream CNN

and appropriate modalities to improve the results. A very high-level example of our proposed approach can be seen in Figure 3.1.

As can be seen, we feed the image sequences of a particular person to the architecture and extract features from two separate modalities, namely Optical Flow (OF) and Grayscale. Using different CNN architectures, we extract a gait signature based on each modality and fuse them at the end for use in a classifier. The Contributions of this chapter are:

- An comprehensive study on low-level and high-level descriptors to be used as input for gait recognition.
- A robust framework for extracting the said low-level descriptors as input of convolutional neural networks.
- A compares 2D and 3D and residual convolutional neural networks with the proposed descriptors extracted from CASIA-B and TUM-GAID datasets.
- Proposing a two-stream architecture and achieving great results with the fusion of modalities.

3.3 The Input data

3.3.1 Hand crafted high-level descriptor Input

This section is dedicated to data preparation for use in the comparison study. The chosen datasets are TUM-GAID and CASIA Database-B, introduced in the previous chapter. The goal is to prepare the data in a way that helps to obtain better recognition results.

Our first attempt was at features which are designed manually and are appropriate for extracting temporal and movement features from surveillance videos. Our approach started with frame by frame noise cancellation by applying Gaussian blur to each frame in the sequence. Since the colour information are not essential for gait recognition, before applying the guassian blur algorithm the RGB images were transformed into grayscale to reduce computation time. The next step after noise cancellation was the computation of low-level descriptors, and the final step was the computation of high-level feature vectors (Gait Signatures) as a global input for any classification algorithm.

Low-Level descriptors are used to summarise and encode the information. They make it easier for the algorithm to search for high-level features in a sequence of images or videos. It is most common to detect more significant input regions and generate low-level descriptors based on these regions. This method can be challenging due to video quality and environmental variations, particularly in real life. Therefore, we adopted approaches to produce a dense grid of points in the input frames. The advantage of such methods is that they can be computed with minimal cost, but since too much information is involved in creating a dense grid, this method can lower the quality of feature detection due to light and angel variations in the real world. This problem

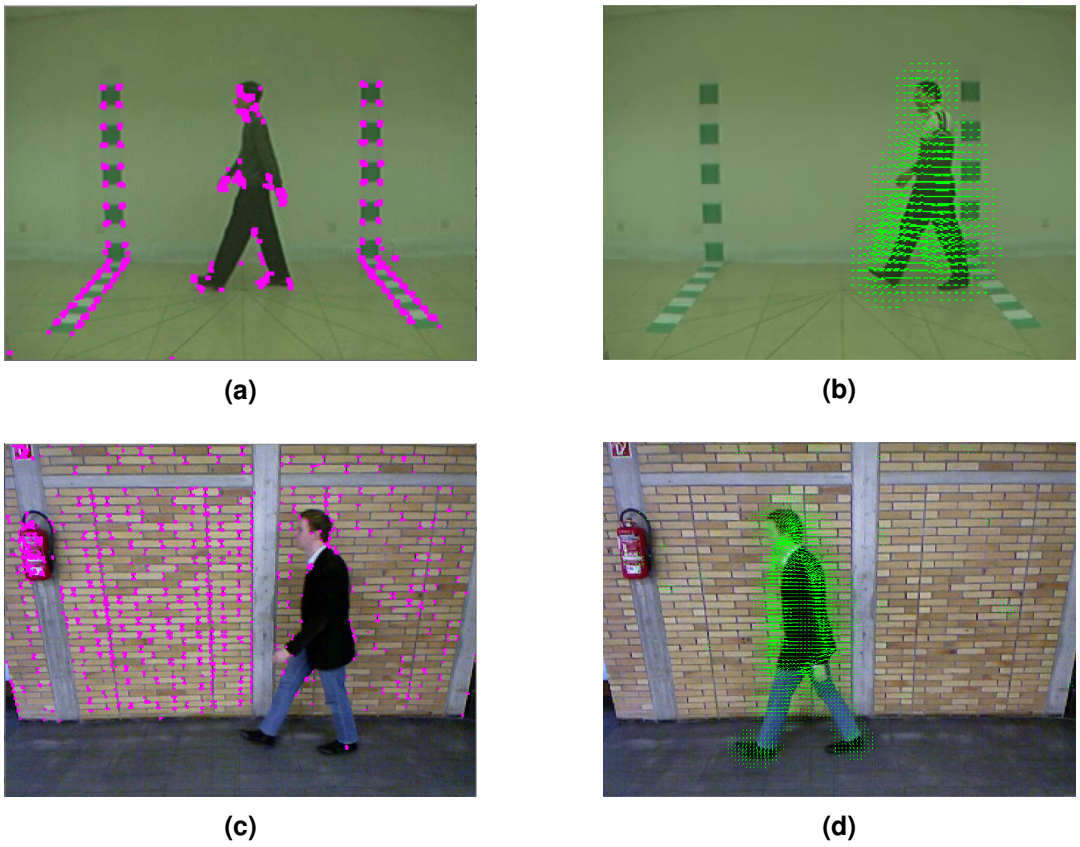


Figure 3.2: Attempts at region detection (a) Harris corner detection on CASIA-B (b) Dense trajectories on CASIA-B (c) Harris corner detection on TUM-GAID (d) Dense trajectories on TUM-GAID

can be mitigated by eliminating some noise in the preprocessing step, but it will lead to more computational cost and less automation. Figure 3.2 shows our different attempts on CASIA-B and TUM-GAID datasets at a 90-degree angle. In (a) and (c), the Harris algorithm for corner detection was applied to single images, which show essential regions at borders of the person, but also other corners in the image are highlighted, which are irrelevant to gait recognition. In (b) and (d), a dense trajectory was created from a sequence of frames based on [165]. The green lines show the motion along the temporal dimension. A simple comparison by looking at the images reveals that a dense grid will provide more accurate and relevant information for gait recognition. It is important to note that in both figure 3.2.b and figure 3.2.d, the grid points have been omitted, and just the movement along the time dimension has been presented.

To calculate the dense trajectories, we used the method introduced in [166] first compute the optical flow fields, OF, using Farneback algorithm on a dense grid and then use a fast median filtering method [167] to track each point from frame t to frame $t+1$. The median filtering used has a kernel of M, and the operation is limited to 15 frames to avoid drifting. Additionally, trajectories with large sudden displacement are removed from the operation since they represent trivial movements (clothing movement), leaving us with only the computed local trajectories. The spatial derivatives for horizontal and vertical optical flow are calculated using movement features to encode low-level descriptors. They are referred to as kinematic features in [166]. Information is collected on physical flow patterns in video clips by considering the divergence, curl and hyperbolic terms. Divergence is related to the axial motion, expansion and scaling effects, curl is the rotation at the image level, and the hyperbolic terms represent the visual sheer of the flow. At every point p of the frame at time t divergence, curl, hyper-

bolic terms and sheer are calculated using the below equation. As described in [166], the features are paired up to calculate the low-level motion descriptor.

$$\begin{cases} div(p_t) = \frac{\partial u(p_t)}{\partial x} + \frac{\partial v(p_t)}{\partial y} \\ curl(p_t) = \frac{-\partial u(p_t)}{\partial y} + \frac{\partial v(p_t)}{\partial x} \\ hyp_1(p_t) = \frac{\partial u(p_t)}{\partial x} - \frac{\partial v(p_t)}{\partial y} \\ hyp_2(p_t) = \frac{\partial u(p_t)}{\partial y} + \frac{\partial v(p_t)}{\partial x} \\ shear(p_t) = \sqrt{{hyp_1}^2(p_t) + {hyp_2}^2(p_t)} \end{cases} \tag{3.1}$$

It is not possible to use low-level descriptors as input for an algorithm since they only provide an abundance of information about corners and edges and, in the case of divergence, curl, sheer (DCS) [166] some very low-level kinematic motion information. This information is usually summarised as an input for a Support Vector Machine (SVM) or other classification algorithms. Our goal is to use this low-level information and generate a high-level descriptor. We represent each frame as a whole to be used as an input for a CNN. Usually, to find the patterns in the raw data, a clustering method is employed and using the collected patterns, a dictionary of gait signature vectors are created. When creating the high-level descriptor, no visual features are considered since the algorithm uses low-level features as input data. Authors such as [168] used fisher vectors (FVs) to create hybrid classification architectures with the help of CNNs. [127] proposes a Pyramidal Fisher Motion descriptor for Multiview Gait Recognition which had promising results on the SVM classification algorithm. It uses densely collected local motion features as low-level descriptors and uses fisher vectors to summarise low-level features and create high-level descriptors for support vector machines. They use the approach from [169] which is an extension of Bag of

Words (BOW) based on the Gaussian Mixture Model (GMM). An image representation is computed by a gradient vector using a generative probabilistic model in Fisher Vectors encoding. In [127] they used the term Fisher Motion to refer to the high-level descriptor generated from low-level motion descriptors. We will also refer to them as such in this thesis.

We create our high level descriptors according to the above method. A video clip (V) can be represented by the below gradient vector equation for T low-level descriptors (x_t), where, $p(x|\lambda)$ is the low-level descriptor independently generated by the Gaussian Mixture Model with $\lambda = \{w_i, \mu_i, i = 1...N\}$ parameters and ∇_λ is the gradient operator.

$$G_\lambda(V) = \frac{1}{T} \sum_{t=1}^{T} \nabla_\lambda log p(x_t|\lambda) \tag{3.2}$$

A fisher kernel is calculated below to compare two video clips V and W. F_λ is the Fisher information matrix described in [170]. As F_λ is symmetric and positive, it has a Cholesky decomposition $F_\lambda = L_\lambda^T L_\lambda$ and $K(V, W)$ can be rewritten as a dot-product between normalized vectors which is then known as the Fisher Vector of video V.

$$K(V, W) = G_\lambda(V)^T F_\lambda G_\lambda(W) \tag{3.3}$$

In this method, the training set is used to compute a dictionary of patterns obtained with the Gaussian Mixture Model, and then a gradient vector is computed in the dictionary to build a feature vector and use them in a classification algorithm.

3.3.2 Optical Flow Input

Our second attempt was based on the assumption that temporal descriptors for gait could be extracted from optical flow [171] and the fact that CNNs can self learn features from optical flow maps. Later in this chapter, we extract appearance features from RGB images and use fusion operation to combine the features extracted from these modalities with optical flow and present the results. Optical flow is essentially the movement of a person from one frame to the next in a sequence of frames. The motion is calculated based on the movement between the camera and the person, giving us valuable temporal information in a gait sequence. A person's movements can be tracked across consecutive frames and estimate their position and even walking velocity using optical flow. Image intensity I is the basis for calculating optical flow. Figure 3.3 shows the optical flow process between two proceeding frames. As can be seen, Image intensity is represented as a function of time and space where t is time or frame number and (x, y) is the position of one pixel in the 2d image, and If the same pixel id is displaced by d_x in the x direction and d_y in the y direction in a period of t then the new image could be described as:

$$I(x + d_x, y + d_y, t + d_t) \tag{3.4}$$

Assuming a constant intensity for pixels in a sequence of images:

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) \tag{3.5}$$

$$I(x, y, t) = I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + ... \Rightarrow \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \tag{3.6}$$

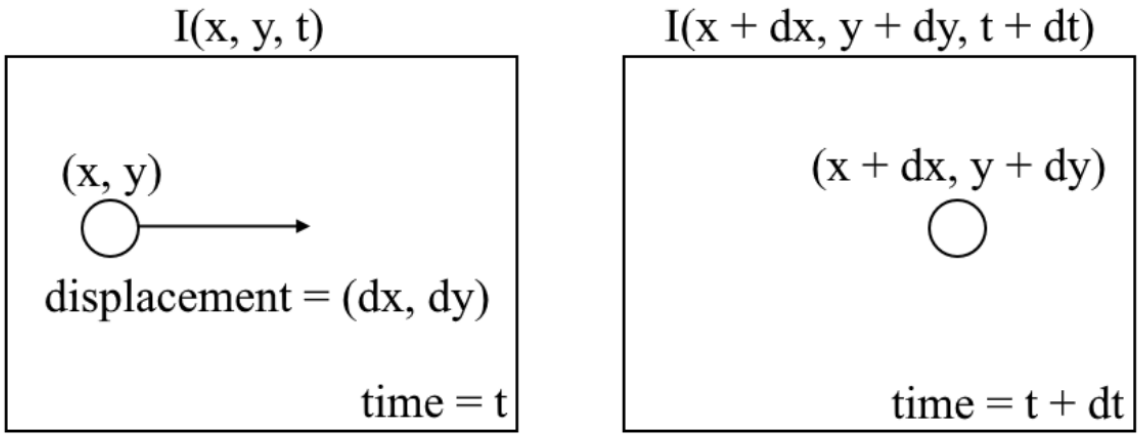


Figure 3.3: Optical Flow between two images [3]

Hence, the optical flow equation is as shown in Eq.3.4 Where $\frac{\partial I}{\partial x}$ is horizontal image gradient along the x axis and $\frac{\partial I}{\partial y}$ is the vertical image gradient along the y axis and $\frac{\partial I}{\partial t}$ is the temporal image gradient. To solve the optical flow equation and determining the motion over time one just needs to solve $u = \frac{d_x}{d_t}$ and $v = \frac{d_y}{d_t}$. As multiple unknown variables are needed to solve this equation, some algorithms have been proposed to address the issue.

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0 \tag{3.7}$$

Optical flow has been used in the past as the input for CNN and archived excellent results for action recognition [172, 173, 174]. There are two types of optical flow, namely Sparse-OF and Dense-OF algorithms. The output of a typical Sparse-OF algorithm is vectors which contain information about edges and corners and some other features of the moving object in the frame. In contrast, Dense-OF algorithms produce vectors for all the pixels in the image like the Farneback algorithm [175]. Some traditional implementations of sparse optical flow are Horn-Schunck [176] and Lucas-Kanade [177] algorithms. Most of the techniques rely on energy minimization

in a coarse-to-fine framework [178, 179] and [180]. A numerical method warps one of the images towards the other from the most coarse level to the finest level and cleans the optical flow with each iteration. however, for motion estimation, normal flow-based methods present a better outcome [181, 182]. Deep learning methods such as FLowNet [183], FlowNet2 [184] and LiteFlowNet [185], use CNNs to estimate the optical flow and are most promising in action recognition problems.

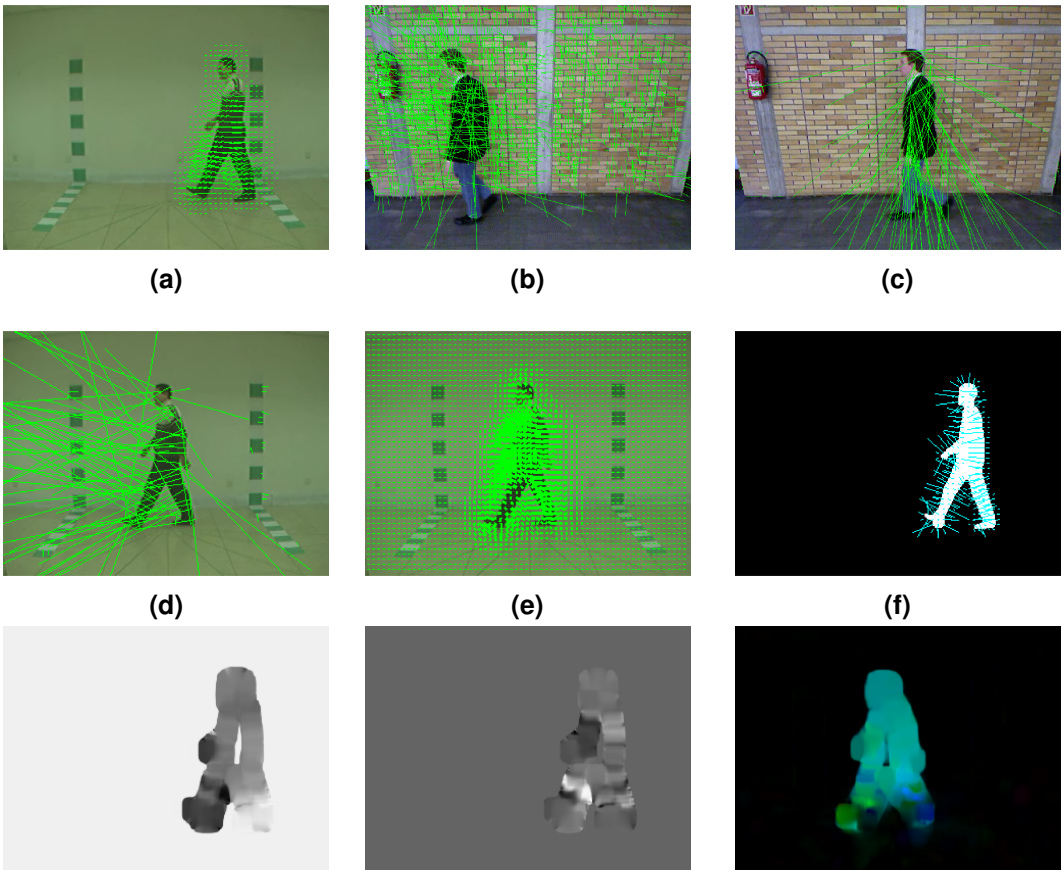


Figure 3.4: Optical Flow representations with different algorithms implemented on CASIA-B and TUM-GAID datasets

In our proposed approach, we are using a stack of optical flow displacement fields for several consecutive frames, much like proposed in [172]. Figure 3.4 shows some of our attempts to extract different optical flow fields considered for the input of our temporal stream. To prepare the data as an input of our CNN optical flow stacking was used [172] to generate a cuboid by stacking optical flow maps. Consider a displace-

ment vector between a pair of frames at times t and $t+1$ denoted by $d_t(x, y)$. Where (x, y) is the location of the pixel in frame t. Therefore, d_t moves the point to the corresponding (x, y) in frame $t+1$ and d_t^{hor} and d_t^{ver} can be seen as image channels which are perfect to use in CNNs for recognition. For a sequence of L consecutive images, the motion is represented by a stack of $d_t^{x,y}$ over $2L$ input channels. To be more clear, if the input sequence of images have the width of w and height of h, an input cuboid for the CNN can be represented as $I_t \in \Re^{(w \times h \times 2L)}$ for frame t in the sequence and calculated as below:

$$I_t(x, y, 2k-1) = d^{hor}_{t+k-1}(x, y) \tag{3.8}$$

$$I_t(x, y, 2k) = d^{ver}_{t+k-1}(x, y) \tag{3.9}$$

Where $1 \leq x \leq w$, $1 \leq y \leq h$ and $1 \leq k \leq L$. So for a random point of (x, y) optical flow map of F_t can be represented as:

$$F_t(x, y, c) = I_t(x, y, c), 1 \leq c \leq 2L \tag{3.10}$$

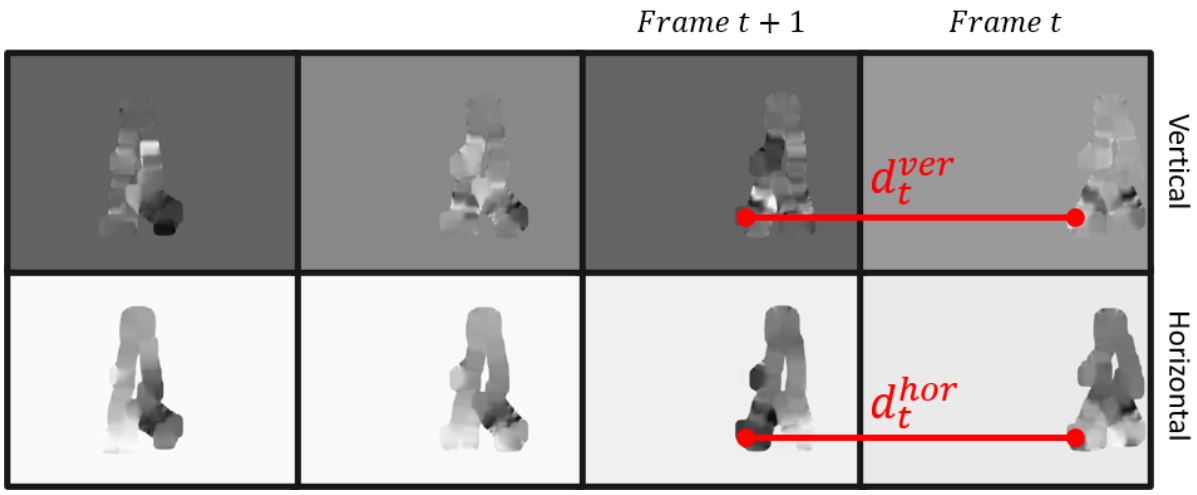


Figure 3.5: The input data for CNN

Since sequence images for each person in real life are extracted from videos as video frames and might differ in different situations, we chose a constant number of frames from each sequence and created the cuboid using those frames. These could be chosen from a random initial point in the sequence, but they must be consecutive. Figure 3.5 shows optical flow examples for consecutive frames in both horizontal and vertical directions. The colour grey denotes no motion, black denotes the maximum amount of motion in the negative direction, and white denotes the maximum motion in the positive direction. A background subtraction step has been implemented to reduce the background noise and extract the silhouettes before extracting the optical flow, significantly reducing the noise. This method works perfectly because we are using optical flow only to capture temporal features and are not concerned with appearance features such as colour.

To create the Optical flow volumes for our experiments, we followed the same method as [171]. Like the authors, the resolution was kept at 80×60. This resolution is chosen to keep the aspect ratio of the original frames in each sequence. Using background subtraction, we extract the silhouettes of subjects and the frame sequences and then use the Farneback algorithm to extract the dense optical flow in horizontal and vertical directions, as shown in figure 3.5. After that, for each direction, we crop the optical flow frames to 60×60 to preserve the height of each subject, keep the subject in the middle of the frame and remove any excessive background noise. Finally, we build an optical flow stack sequence with the length of 25 frames in which 20 frames are used from the previous sequence, and 5 is used from the next. We choose 25 since it is acceptable by most works in the literature as the acceptable number of frames to complete a gait cycle (Figure 2.17) in a surveillance video. Finally, we stack

both directions into the exact $60 \times 60 \times 50$ volume.

3.3.3 Grayscale Input

RGB frames were considered since they are the modality of choice for person re-id [186]. However, These methods are overly dependent on appearance features such as colour, which is generally unreliable for gait recognition. In most of these works, many false positives are due to the person of interest wearing the same colour clothes as other people in the gallery. Moreover, it is known that colour information is not very useful for long-term person re-id. We decided to remove the colour variation in our approach by using grayscale image sequences as inputs. By omitting the colour information, we reduced the computation time and focused the attention of our network towards extracting features that are more relevant to gait Person Re-Id. Some works such as [187] use greyscale and RGB colour images for their architectures for Person Re-Id, but they neglect the temporal information in the process.

To prepare the data for our input, we keep 25 consecutive frames from the sequence at random and crop them to 60×60 for the sake of consistency with the temporal stream input data. Then the 25 frames are stacked, creating a volume of size $60 \times 60 \times 25$. Finally, a normalisation operation based on the dataset characteristics happens before the data is injected into CNN. For example, due to the significant viewpoint variations in CASIA-B, the grayscale volume is normalised between 0 and 1.

3.3.4 Silhouette Input

The silhouette images for datasets were extracted using the method proposed in [188]. The first step to silhouette extraction is the segmentation [189, 190]. First, we perform

a background subtraction using the background image provided with the dataset. To segment binary maps out of the raw images in the sequence. In TUM-GIAD, the background is conveniently provided. However, in real-world scenarios and publicly available datasets such as iLIDS-VID and PRID2011, variations such as occlusion and illumination are present as holes and shadows and missing parts in the image. In such cases, a Least Median of Squares (LMeds) technique is used to model the background from small parts of the image sequence. If I denotes a sequence of N images, the background could be modelled using the below equation. Where p is the brightness value for pixel location of (x, y), med is the median value, and t is time or the frame index in the sequence.

$$b_{xy} = min(med_t(I^t_{xy} - p)^2) \tag{3.11}$$

After extracting the silhouettes for all the image sequences, they are cropped to 60×60 to be consistent as inputs for our networks. The height of the subject is preserved during the cropping process.

3.4 Network Architecture

Since the input of our networks are either optical flow channels, grayscale images or silhouette sequences, we consider two types of convolutional neural networks that were most successful in the past. For the grayscale and silhouette inputs, 2D CNNs seem sufficient. In 2D CNNs, a two-dimensional convolution operation is performed on feature maps from the layer before extracting spatial features. In our experiment, three state of the art 2D CNNs were tried out, two of which [172, 16] are classic 2D CNNs ap-

plied to CASIA-B and TUM-GAID, and the other one is a residual neural network based on ResNet [8]. These results are compared in section 3.1.5 to choose the best network for our spatial stream. In problems such as Person Re-Id, it is necessary to acquire the temporal motion information when going through a continuous frame sequence. 3D CNNs are mainly designed for capturing motion information in videos. They can compute features from both spatial and temporal dimensions in an image sequence by convolving a stack of frames in the form of a cuboid with three-dimensional filters. They connect the feature maps in the convolutional layer to the consecutive images in previous layers to capture motion data. For optical flow inputs, both the horizontal and vertical channels are considered. Hence, we need to employ a 3D convolutional neural network. In our proposed 3D network, which is based on [191] Multiple 3D filters are applied to the input cuboids extracting multiple features from consecutive optical flow frames. There are no shared weights between the connections, which result in many feature maps for the next layer.

Figure 3.6 shows the first of the three aforementioned 2D CNN networks (2D-1) used for the spatial stream experiment. This network consists of five convolutional and three pooling layers with ReLU as their activation function. The first convolutional layer consists of 96 kernels of size 7×7 with a stride of 2 with a batch normalisation followed by a 2×2 pooling layer. The second convolutional layer includes 256 kernels with a size of 5×5 and the same numbers for stride normalisation and pooling. The third and fourth convolutional layers consist of 512, 3×3 kernels with the stride of 1 and no proceeding pooling layer. Finally, the last convolutional layer contains 512, 3×3 kernels a stride of 1, followed by a 2×2 pooling layer. To prevent overfitting and reduce the parameters, both fully connected layers are followed by a dropout [192]. This network

is based on AlexNet [16] and has been used with excellent results in action recognition problems in the past [172].

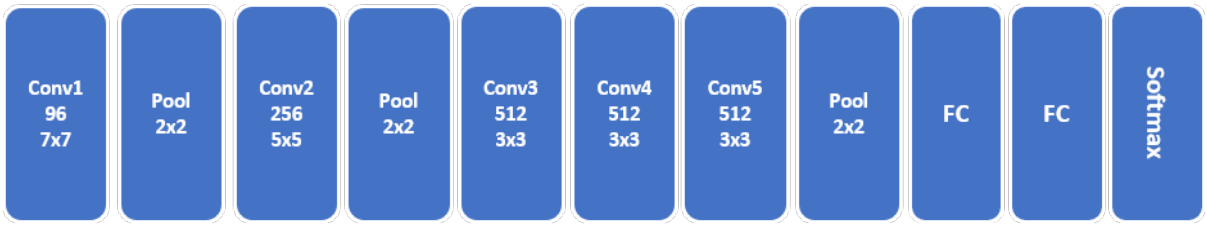


Figure 3.6: A 2D linear CNN with five convolutions and two fully connected layers

Consequently, Figure 3.7 shows our second choice for the spatial stream (2D-2), which was initially used for optical flow gait signature extraction in [171]. As can be seen, the network consists of four convolutional layers, three pooling and two fully connected layers. The first convolutional layer consists of 96 kernels with a 7×7, the stride of 1 and batch normalisation, followed by a 2×2 pooling layer. The second convolutional layer has 192, 5×5 kernels with a stride of 2 followed by a pooling layer of the same size. The third convolutional layer includes 512, 3×3 filters, with a stride of 1 and a pooling layer of 2×2. The last convolutional layer has 4096 kernels of size 2×2, the stride of 1, no pooling layer, and finally, two fully connected layers with 4096 and 2048 neurons and a dropout each. It is worth mentioning that all the gait signatures are extracted from the fully connected layers right before the softmax layer.

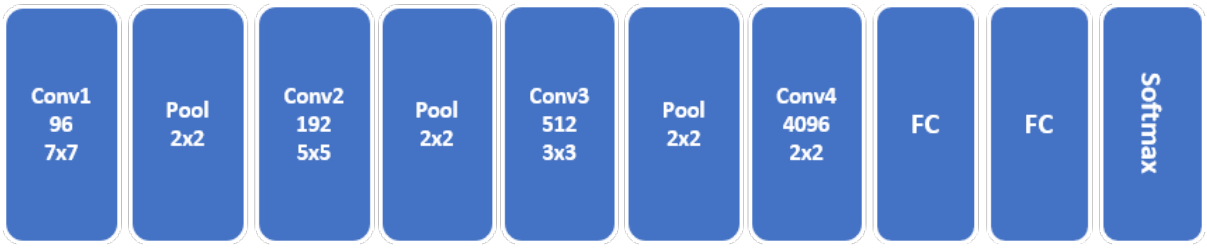
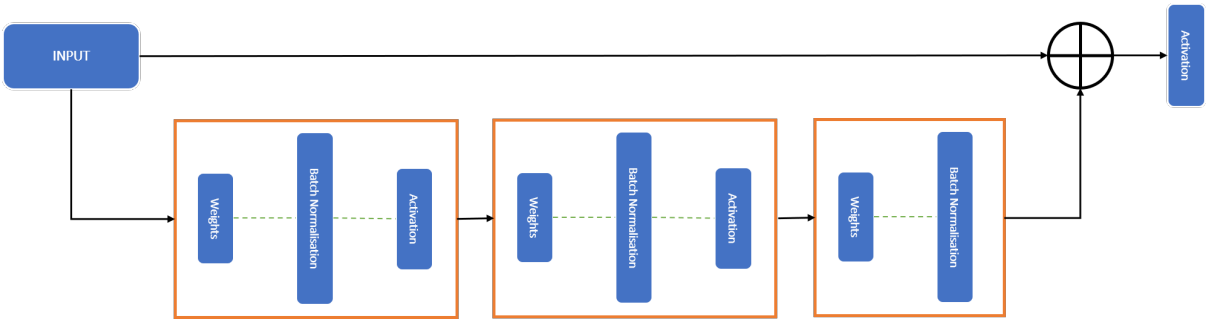
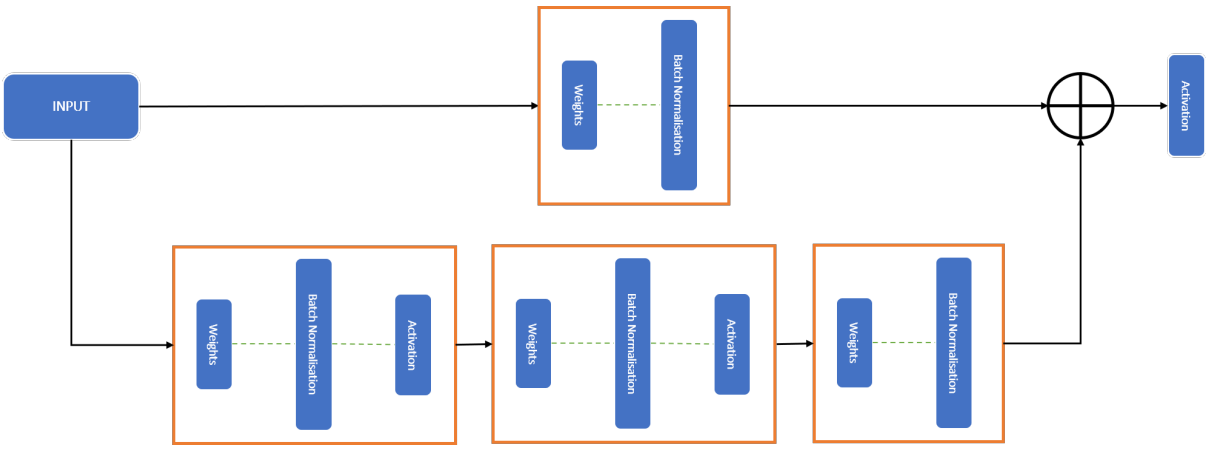


Figure 3.7: A 2D linear CNN with four convolutions and two fully connected layers

The third 2D CNN is based on ResNet [8] which uses residual blocks and identity blocks to extract a gait signature from datasets with high variations like CASIA-B, which



(a) Type-A residual block (Identity Block)



(b) Type-B residual Block (Conv Block)

Figure 3.8: Two types of residual blocks for our ResNet architecture **(a)** Type-A block adds the input directly to the output with a identity shortcut branch and includes three convolution, three batch normalisation and two activation (Relu) in the main branch **(b)** Type-B residual block includes three convolution, three batch normalisation and two activation (Relu) in the main branch and adds the input to the output after one convolution operation and a batch normalisation

has 11 view angles. The architecture of ResNet and Residual learning blocks are described in section 2.1.1. As mentioned before, the performance of a very deep neural network starts to degrade after a certain number of layers and to solve this problem, residual blocks with the use of identity are used. A residual block passes the input to the output of a particular layer directly and uses an Addition function to add it to the output (Figure 2.8).

One crucial detail not apparent from the diagram is that the final activation is performed after the addition. When applying batch normalisation in the main branch, it should appear after the weight matrix and not after the main activation and also, batch

normalisation eliminates the need for bias terms in the dense layer. All the residual blocks in our ResNet occur in the convolutional layers. There are two types of residual blocks in our ResNet, illustrated in Figure 3.8. Each residual block in Figure 3.8 consists of a 1×1, a 3×3 and a 1×1 convolution operation on the main branch. Relu is used for the activation function each time. The Conv block in Figure 3.8.b also has a convolution operation on the shortcut branch. To add the input to $F(X)$, we need to keep the same shape in each block layer by using Same-Mode in the convolution. However, we can use the below equation in the case of exceptions. The input X is multiplied by an identity matrix W_s with zero-padding to make it the same shape as the final output.

$$F(X) + W_s X \tag{3.12}$$

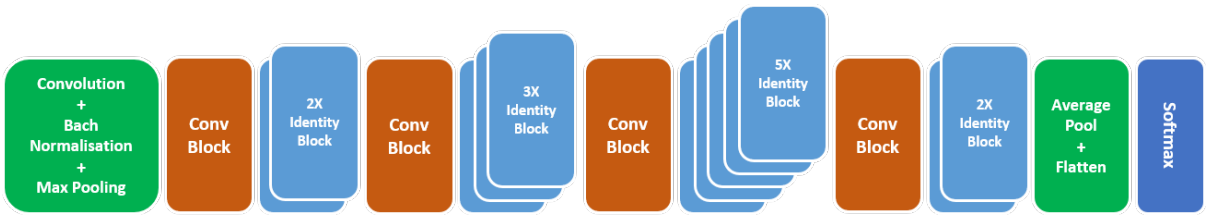


Figure 3.9: A 2D linear residual CNN with Conv and identity blocks

Our proposed ResNet has fifty layers, not including the batch normalisation and pooling layers, although we have to keep in mind that the batch normalisation layer has some parameters of its own. Figure 3.9 shows the ResNet architecture used for our experiment for spatial feature extraction. The first traditional convolutional layer has 64 kernels with 7×7 dimensions and stride of 1. The pooling in this layer is of the size 3×3 and stride of 2. The average pooling layer has a size of 2×2 with a stride of 1. The exciting aspect of ResNet is that it will add no extra parameters. In other words, only the architecture has changed, and we train the same number of parameters as we would have, had there been no residual blocks. Therefore, there would be less

computational cost, less training time on the system and less chance of overfitting with increased parameter size. The ResNet in [8] with 34 layers requires 18% fewer operations than VGG with 19 layers.

We chose a 3D CNN for our temporal stream since they are well suited for temporal feature learning [193]. Our proposed network can model temporal information from optical flow stacks With 3D convolution and 3D pooling operations. When 2D convolution is applied to an image or a series of frames, the result will be an image. Therefore 2D CNNs lose a lot of the temporal information each time a convolution operation is applied to the input sequence. In [172] their attempt to preserve the temporal information by applying a 2D CNN to a series of optical flow images, but all temporal information is lost completely after the second convolutional operation. Even some fusion models like [194] lose the temporal information after the first convolutional operation when using a 2D CNN. The Slow Fusion method proposed in [194] manages to keep the temporal information for three layers using 3D convolution and 3D average pooling. Our CNN is essentially the same CNN used in [193] with slight modifications in dimensions. The method is based on [191] but with a sequence of optical flow images instead of RGB video frames. It also bears some resemblance to [172] and slow fusion model in [194] but with 3D pooling operations after each layer. Our proposed 3D CNN is shown in Figure 3.10. It consists of eight convolutional layers with size of $3 \times 3 \times 3$ and five $3 \times 3 \times 3$ pooling operations. All convolutions are applied with a stride of 1. There are two fully connected layers with 4096 and one with 2048 units, followed by dropouts for each dense layer and a softmax layer.

To help our CNN models achieve convergence faster and increase the system's performance, we used curriculum learning [195] in the training process. Curriculum

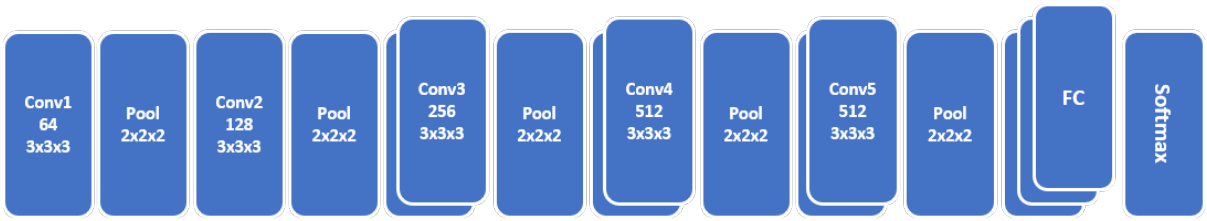


Figure 3.10: A 3D CNN for temporal feature extraction

learning starts with some easy examples of a task and gradually increases the difficulty. This type of training is based on the idea that humans and animals learn better when presented with organised and meaningful examples going from the least to the most complex. In other words, we allocated fewer units per layer and eliminated dropouts in the first training process, and then we used the weights to train a more complex network with more kernels and dropouts after each fully connected layer. This method is used to train our first and second 2D CNNs in figures 3.6 and 3.7 and the 3D CNN shown in figure 3.10. For the 2D CNNs, we used three increments and 0.1 dropouts. For the 3D CNN, we used a 0.5 dropout with three increments. The ResNet was running without curriculum learning due to the nature of the network and computation complexity.

Batch gradient descent can be used for smoother curves and converges directly to the minima, and stochastic gradient descent (SGD) converges faster for large datasets. However, SGD cannot be used independently since it can only be implemented with one example at a time. Hence, Mini-batch stochastic gradient descent (m-SGD) is applied during the training of our models to prevent slowing down the computation. m-SGD is a mixture of batch gradient descent and stochastic gradient descent. Neither all the dataset is handled at a time nor a single example. A fixed batch of the training data is chosen from the dataset and fed into the network in one epoch. Then the mean gradient of the mini-batch is calculated and used to update the weights. The m-SGD

can be described as the below equation. Where the trainable variables of the model is $\theta \leftarrow \theta - \Delta\theta_t$. The learning rate is represented by α, and n is the mini-batch size. The current and previous weights are $\Delta\theta_t$ and $\Delta\theta_{t-1}$. The momentum is γ, and h is the model's output.

$$\Delta\theta_t = \gamma.\Delta\theta_{t-1} + \alpha\frac{1}{n}\sum_{i=1}^{n}(h_\theta(x^{(i)}) - y^{(i)}).x^{(i)} \tag{3.13}$$

The momentum for the first three CNNs is set to 0.90 for the first iteration and 0.95 for the rest. For the ResNet, the momentum is set to 0.95. On TUM-GAID, we ran the models with 20 epochs at first, but then we increased to 30. The learning rate was equal to 10^{-2} initially for the first three networks and 0.1 for ResNet. In CASIA-B, each mini-batch has a size of 150 examples. The initial learning rate was set to 10^{-3} with 30 epochs. For the ResNet on CASIA-B, a mini-batch of 64 samples has been used. The implementation was tried initially on MATLAB with the use of MatConvNet library [196], but due to the limitations of MATLAB, we moved on to python with Tensorflow, PyTorch and Keras libraries which made it easier to make use of CUDA and cuDNN with our NVIDIA Quadro M5000 GPU [197]. After training, the gait signatures are extracted from the last fully connected layer and classified using softmax regression described in the below equation where $\theta = \sum_{i=0}^{k} W_i X_i$ and so $\theta^{(i)}$ is an element of the gait signature also sometimes denoted as x_i.

$$Softmax = P\left(y = j|\theta^{(i)}\right) = \frac{e^{\theta^{(i)}}}{\sum_{j=0}^{k} e^{\theta_k^{(i)}}} \tag{3.14}$$

The Softmax is a form of logistic regression that creates a normalised vector of input values that follows a probability distribution with the sum of one. It predicts if the

trained set of features x with their weights belong to a class of j. The output of softmax is a value between zero and one. This method, also called Maximum Entropy classifier (MaxEnt) or multinomial logistic regression, calculates the exponential of each input x_i divided by the sum of exponential parameters of the input.

3.5 Fusion of modalities

Three modalities have been used in our experiment in this chapter, namely

1. Temporal Features extracted from optical flow maps,

2. Appearance Features extracted from Grayscale frame sequences, and

3. Depth information extracted from Silhouette image sequences.

Since we have three modalities for optical flow, grayscale images and Silhouette images, we need to combine the results and extract a single gait signature. The expectation is that a better classification score can be achieved by fusing more modalities. Two popular fusion methods could be used for our purposes: late fusion and early fusion.

Late Fusion

Late fusion happens after the softmax layer to combine the output of all the CNNs. For this purpose, we can take the product or the sum of all softmax vectors extracted at the end of each CNN. So if we have n probability vectors at the end of each CNN for m different modalities (two modalities at this point), we can use any of the two equations below to produce a result vector that shows the probability of the person

in the sequence s having the same identity as the person in class c. The symbol $0 < \rho < 1$ represents a weight related to the modality and is assigned based on experiments.

$$Result = \prod_{i=1}^{n} P_i(m_i = c) \tag{3.15}$$

$$Result = \sum_{i=1}^{n} \rho P_i(m_i = c) \tag{3.16}$$

The two-stream architecture in [172] employs late fusion, but since the fusion happens using the softmax layers, it neglects the correlation between temporal and spatial features at a pixel level. Also, since spatial CNN works on one image at a time and the temporal CNN on a stack of ten optical flow images, a lot of the temporal information is ignored.

Early Fusion

On the other hand, early fusion can happen at any layer of the CNN architecture as long as it is before the softmax layer. If the fusion is performed in a convolutional layer, the descriptor is a matrix, and if it is performed on a fully connected layer, it is a vector. Fusing at a convolutional layer is only possible if the two networks have the same spatial resolution at the location of the layers intended to be fused. We can stack (overlay) the layers from one network to the other so the channel responses at the same pixels in the temporal and spatial stream can correspond. If we presume that separate channels in the spatial stream are responsible for different parts of the human body (Leg, Foot, Head) and a channel in the temporal stream is in charge of the

motion information (moving the foot forward), After stacking the channels, the kernels in the next layer needs to learn this correspondence as weights.

Fusing of layers

There are several ways of fusing layer between two networks. If f is a fusion function that fuses X_t^a and X_t^b which are two feature maps at time, t then f has the output of $y_t \in \Re^{H'' \times W'' \times C''}$ for $x_t^a \in \Re^{H \times W \times C}$ and $x_t^b \in \Re^{H' \times W' \times C'}$. The width, height and channel numbers are represented as W, H and C respectively and the fusion function is represented as:

$$f : X_t^a, X_t^b \Rightarrow y_t \tag{3.17}$$

To decide where and how to perform the fusion, we first go through a range of feasible methods introduced in the literature. We assume the same dimensions for X_t^a, X_t^b and y_t as we discussed before. So for $x^a, X^b, y \in \Re^{H \times W \times D}$, **Sum Fusion** sums the feature maps at the same location in space indicated by i, j and with c number of channels.

$$y_{(i,j,c)} = X_{(i,j,c)}^a + X_{(i,j,c)}^b \tag{3.18}$$

Since the channel numbers are chosen at random, the correspondence between the networks is arbitrary. Therefore, more training is necessary over the following layers to make this correspondence helpful. The same rule applies to **MAX Fusion** which takes the maximum of two feature maps.

$$y_{(i,j,c)} = max\{X_{(i,j,c)}^a, X_{(i,j,c)}^b\} \tag{3.19}$$

Concatenation Fusion stacks the feature maps at location (i, j) for channels c and outputs $y \in \Re^{H \times W \times 2C}$ but it does not define correspondence so they need to be defined in the next layers by learning new kernels.

$$y_{(i,j,2c)} = X^a_{(i,j,c)} \quad \text{and} \quad y_{(i,j,2c-1)} = X^b_{(i,j,c)} \tag{3.20}$$

The feature maps can be stacked as shown in 3.16 and succeeded by a convolution on the result with a kernel $k \in \Re^{1 \times 1 \times 2C \times C}$ and bias of $b \in \Re^C$, to solve the problem of concatenation fusion. Note that the number of output channels is C and the dimensions for the kernel is $1 \times 1 \times 2C$. For early fusion in the fully connected layers of 2D and 3D CNNs and also to fuse the gait signatures obtained by each modality, we perform a concatenation operation followed by another three fully connected layers with dropout before the softmax layer. This fusion is best performed after the last fully connected layer since the signature is already extracted at that point. In ResNet, we will not add fully connected layers after concatenation since it might lead to overfitting. Note that the Relu function is used for all activations in our networks, so for the extra layers, it remains the same.

$$Convf = y * k + b \tag{3.21}$$

Adding fusion significantly affects the number of parameters used in the computation. Moreover, we also needed to consider the fusion process over time (temporally). Our first consideration was to average all the feature maps x_t over time t which was used in [172], but this way, only two-dimensional pooling is possible, which loses lots of temporal information. Next we considered using 3D pooling [198] on a stack of feature

maps $x \in \Re^{H \times W \times T \times C}$ over time t. This layer applies max pooling to a stack of data of size $\Re^{W' \times H' \times T'}$. It is also possible to perform a convolution with a fusion kernel before performing 3D pooling, much like [193]. Figure 3.11 illustrates this fusion technique.

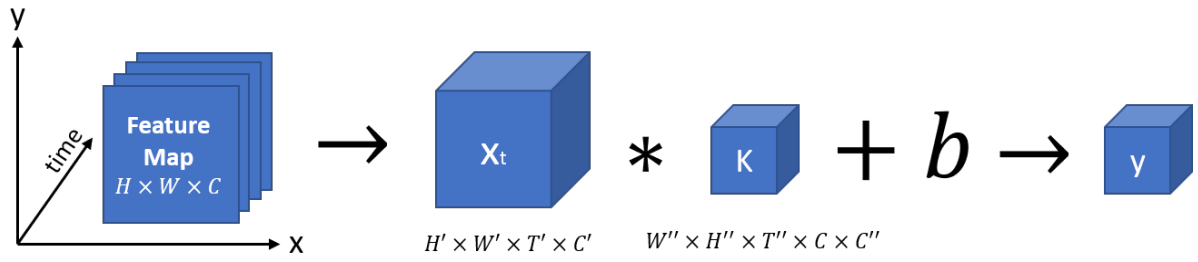


Figure 3.11: 3D pooling operation on features maps for fusion

3.6 Experimental Results

This section presents the results obtained for two publicly available datasets on different spatial and temporal streams modalities. Next, we will exhibit the results obtained by the most reliable fusion approach and explain how we achieved great results on both datasets.

Our two datasets separate the experimental results presented in this section. First, both datasets have been introduced in detail, and then the results are presented. For each dataset, scenarios are presented for testing the robustness of the networks and fusion method combinations. First, the results for single modalities on each network described in section 3.4 have been obtained to find the most suitable network for each modality. Then various combinations of modalities have been tried out to find the best combination of networks and fusion methods. Finally, these results have been compared against state of art.

3.6.1 Datasets

We first go through the two benchmark datasets used in our experiment in more detail to present the results. Both CASIA-B [199, 200] and TUM-GAID [111] datasets have been introduced in table 2.1 in chapter 2.

CASIA Dataset

The Institute of Automation Chinese Academy of Sciences has provided three datasets A, B and C, for CASIA. Dataset A contains twenty subjects with twelve image sequences (four sequences for each of the three directions to the image plane). The length of each sequence is not identical for the variation of the walker's speed, but it must range from 37 to 127.The CASIA dataset includes $19,139$ images and having size $2.2GB$.

CASIA-B is an extensive Multiview gait database, which was created in January 2005. There are 124 subjects, and the gait data was captured from eleven camera views. Three variations, namely view angle, clothing and carrying condition changes, are separately considered. The 124 subjects were videoed for three to five seconds in the same location indoors with the resolution of 320×240. The video frame rate is around $25fps$, and subjects are either with no extra condition, carrying a bag or backpack or wearing a coat or jacket. Ten videos were captured from each view, six with normal conditions, two with a jacket or coat and two carrying a bag or backpack. It is essential to mention that we are using all the viewpoints included in this dataset when extracting the gait signature. Figure 3.12 shows some of the frames used in the experiment.

Figure 3.12: CASIA-B dataset sample Frames from the same person with three different view points

TUM-GAID Dataset

The TUM Gait from Audio, Image and Depth (GAID) dataset consists of 3370 sequences of 305 individuals walking from left to right and right to left in an indoor environment. The dataset was recorded with a Microsoft Kinect sensor in January and April of the same year. In the January session, the subjects wore winter boots and coats and in the April session in which people wore regular clothes. The video recorded had a resolution of 640×480 pixels with a frame rate of 30fps. For each 350 subjects, the dataset is presented based on different modalities. For each of the modalities (image, depth and audio), a subfolder contains the data.

Additionally, tracked image data are provided in the folder named ”croppedFrames”,

and tracked depth data are provided in the folder named ”croppedDepth”. Each of these folders has 305 subfolders, named ”p001”- ”p305”. Those subfolders each contain the data for one subject. For example, in the p001-p305 folder, there is a subfolder for each recording of the related subject. For the subjects p001-p032, these are n01-n12 which are the recordings of the normal type. n01-n06 are from the first recording session, and n07-n12 are from the second recording session. b01-b04 are the recordings with a backpack. b01 and b02 are from the first recording, and b03 and b04 are from the second recording. s01-s04 are the recordings with coating shoes, whereby s01 and s02 are from the first recording, and s03 and s04 are from the second recording. For the subjects p33-p305, only the recordings n01-n06, b01-b02, and s01-s02 are available. The sub-folders for the recordings contain the actual audio data, which is one file, ”audio.wav”. In addition, the image folder contains several ”jpg” video frames (named starting from 000.jpg). Besides, one background image per recording session is provided called ”background.jpg” and is in the folder ”back” and ”back2” (when the person took part in two recording sessions). Figure 3.13 shows sample images of this dataset.

3.6.2 The Experiment On TUM-GAID Dataset

Splitting the dataset for training and testing

Our experiment on the TUM-GAID dataset is designed based on the identity experiment in [111], which is the original paper authored by the creators of TUM-GAID. We break the dataset into three groups. The training set used during the learning process to fit the parameters [201], validation set to fine-tune the hyperparameters and the test set. We used an approximate split of 50%-50% for development and test sets. For

Figure 3.13: TUM-GAID dataset sample Frames from the same person in three different conditions

the training set, we used 100 subjects combined with 50 subjects for the validation set, creating a development set of 150 people for our re-identification experiment. The test set consists of the remaining 155 subjects.

Experiment Scenarios with different variations

To take into account different variations, we used all the available sequences for each subject. Since n07-n12, b03-b04 and s03-s04 are not available for the subjects p33 to P305 and only 32 subjects participated in both recording sessions, we split them between development and test sets, so 16 subjects are included in the development set and 16 subjects in the test set. The development set can be applied for training and fine-tuning the parameters. Also, it can help us build a recognition system that is invariant of the clothing alterations and uses all 3400 frame sequences. Building such

Table 3.1: Re-Id Experiment scenarios on TUM-GAID dataset

Scenario-1						**Scenario-2**					
Gallery			Probe			Gallery			Probe		
n01-n04	b01	s01	n05-n06	b02	s02	n01-n04	b01	s01	n07-n08	b03-b04	s03-s04

a development set helps build a robust and covariate[1]-invariant recognition system.

Experiment Scenarios

For our identification experiment, only the test set is used. For this, we will follow a structure based on [111], which is splitting the sequences of the test set into another training and test set of 65% to 35% respectively. In **Scenario-1**, we take the first four normal walking sequences (n01-n04) and also one backpack (b01) and one shoe sequence (s01) for the gallery, and used two normal-walk (n05-n06), one backpack (b02) and one shoe (s02) scenario for the probe.

In **Scenario-2** we use the sequences from the 32 subjects who participated in the second recording. The gallery sequences will stay the same structure as scenario-1 (n01-n04, b01, s01), but the probe will be chosen from the second recording session (n07-n12, b03 and s03). To summarise, the experiment is as shown in table 3.1, where gallery and probe are identified. Note that the experiment results in [111] for TUM-GAID are reported separately for all the different conditions, but for our experiment, we used different variations in both gallery and probe, which will increase the number of samples and avoid overfitting on the ResNet network when training.

[1] covariates are characteristics of the participants in an experiment. If data is collected on characteristics before running an experiment, it could be used to see how the procedure affects different groups of people [202].

Performance Evaluation

To evaluate the performance of the models, we use the average identification accuracy percentage as a metric. We calculated the accuracy by dividing the number of correct predictions by the number of all predictions and multiply the result by 100. In [111] they used a Rank system to evaluate the identification model, so we base our evaluation on the same system in which Rank 1 accuracy corresponds to the scenario in which the top one identity is the correct one. Rank 5 is when the correct identity is in the top five results, and Rank 10 is when the correct identity is in the top 10 results. Note that this form of evaluation is more practical, particularly in a real-world scenario with uncontrolled environmental variations.

Moreover, it is essential to mention that since we split the video sequences in random 25 frame gait cycles, and they are classified by softmax independently, we cannot rely on softmax for a subject's identity. All the signatures extracted for all sub-sequences of the same sequence need to be combined to give us a person's identity. To combine the results and obtain one gait signature, we use the product of all softmax vectors for a single sequence. The softmax vectors which we are combining are the probability (P) of assigning an individual identity to a person in a specific sequence, where, $0 \leq P \leq 1$.

Single Modality Results

To obtain comprehensive results first, we try our networks individually on optical flow, grayscale and silhouette inputs and extract the gait signatures. This way, we can evaluate which low-level feature is more convenient for re-identification scenarios and assess our CNNs with each input type. Then we combine our networks using vari-

Table 3.2: Cross comparison between networks and modalities on TUM-GAID dataset. Each row represents a different CNN with all three input modalities and the columns correspond to Scenario-1 (Short-Term Re-Id) and Scenario-2 (Long-Term Re-Id). The accuracy is calculated using the Rank system introduced in this section for Rank-1 (R1), Rank-5 (R5) and Rank-10 (R10).

Network	**Modality**	**Scenario-1**			**Scenario-2**		
		R1	R5	R10	R1	R5	R10
2D-CNN-1	OF	95.2	98.5	99.8	59.9	70.2	73.2
	Gray	96.5	97.4	99.1	28.2	64.9	70.3
	Silhoutte	96	96.2	97.9	25	36.1	45.7
2D-CNN-2	OF	97.74	99.8	100	60.4	87.5	89.7
	Gray	99.4	99.8	100	34.4	64.6	73.2
	Silhoutte	99.6	100	100	44.2	44.9	48.2
3D-CNN	OF	96.97	97.86	98.52	68.76	86.46	89.92
	Gray	98.3	99.56	99.8	36.5	37.7	42.5
	Silhoutte	99.3	99.3	99.6	72.2	75	75
ResNet	OF	98.8	99.3	100	39.9	45.2	62.4
	Gray	99.4	100	100	48.2	62.5	75
	Silhoutte	99.8	99.8	100	78.5	79.9	90.1

ous fusion techniques to find the best combination of modalities for feature extraction. Finally, we compare our results with state of the art on the same datasets.

Network cross-referencing

To start, we cross-reference each one of our networks (2D-CNN-1, 2D-CNN-2, 3D-CNN and ResNet) with three individual modalities on TUM-GAID following the scenarios mentioned above. Then we calculate the Rank 1, Rank 5 and Rank 10 accuracy metrics for both scenarios and present the results in Table 3.2 for each modality and each network. By focusing on Rank 1 in both scenarios, It can easily be deducted that the best results belong to the 3D-CNN and ResNet. Moreover, since Silhouette and optical flow yield the best results in the long-term scenario, they are the most dominant modalities. It clearly shows that the temporal information is preserved longer using 3D-CNN with 3D pooling and ResNet because of its residual blocks. This idea becomes stronger by examining Rank 5 and Rank 10 results in which Silhouette and optical flow show the most reliable outcomes.

If we focus on scenario-1, we can see that in 3D-CNN and ResNet, grayscale images perform better than optical flow and Silhouette due to the low variability of the appearance features. In scenario-2, the results are less impressive for grayscale modality in 3D-CNN, which shows that appearance features are not very useful over-time when the training and test samples have high variability like in a long-term Re-Id scenario. On the other hand, ResNet shows less impressive results on optical flow for scenario-2 and better results using silhouettes. So considering our purposes for gait person Re-Id, the best results can be achieved using ResNet and 3D-CNN, which are better for temporal scenarios. Thus we can conclude that the best modalities for

long-term scenarios are Silhouette and optical flow, respectively.

Fusion of modalities with the best networks

The next step in our experiment is to fuse the data from different modalities. To fuse the information, we employ the techniques introduced in the previous section. The late fusion approaches are fairly simple to implement after the softmax layer using the product sum or the weighted sum of softmax vectors.

Fusion Positions

The early fusion could be done in various positions in the network. These positions are catagorised as **signature level** and **convolution level**. The signature level fusion happens at the last fully connected layer before the softmax layer, but the convolutional fusion can happen in any convolutional layer.

Since there are numerous combinations for the early fusion, late fusion and fusion between modalities, we focus our experiment on modalities that have shown better performance according to table 3.2, namely Silhouette and optical flow. We also limited our experiment to the 3D-CNN and the ResNet, which have shown promising results on scenario-2. In 3D-CNN (Figure 3.10), we chose four positions to perform convolution level fusion. **Pos-1**, in conv2 layer, **Pos-2** in the second conv3 layer, **Pos-3** in the second conv4 layer and **Pos-4** in the second conv5 layer and finally **Pos-5** is allocated to the signature level fusion. In ResNet (Figure 3.9), the convolutional level fusion happens on the last convolutional layer of each conv blocks. **Pos-1** to **Pos-4** belong to the four conv blocks and **Pos-5** corresponds to the last layer before the softmax.

Table 3.3: Early fusion positions for ResNet and 3D-CNN

Network	**3D-CNN**			**ResNet**		
Modality	OF-SIL	SIL-Gray	OF-Gray	OF-SIL	SIL-Gray	OF-Gray
Pos-1	85	90.3	92.1	85.2	90.01	84.1
Pos-2	79.1	91.6	92	85.6	92.1	88.5
Pos-3	70.04	89.1	88	85.5	92.8	87.6
Pos-4	90.0	92.7	91.3	91.1	94.2	92
Pos-5	**90.02**	**94.8**	**93.4**	**91.4**	**96.3**	**92.9**

Results of the fusion in TUM-GAID

Table 3.3 shows the fusion results in all positions for both silhouette and optical flow modalities on ResNet and 3D-CNN. The results are presented in Rank 1 only since Rank 5 and Rank 10 would be redundant.

By studying the results, we can deduct that for 3D-CNN, The best performance for any modality combinations belongs to pos-5, which is the gait signature layer. For ResNet, the best results appear in pos-5. The last position is optimal since it is just a one-dimensional vector, and it reduces the complexity when performing the fusion operation. It is also essential to notice that the Silhouette and grayscale modalities deliver the best results on both network types when combined, and ResNet has the most consistent results on this combination of modalities. After finding the best position for early fusion, we can experiment on the best method of fusion (Late or Early) in both architectures. Table 3.4 shows a comparison for both late and early fusion results on the 3D-CNN and ResNet in both scenarios. When we fuse all modalities using equation 3.13 for weighted sum, the value of ρ is 0.6 for optical flow and silhouette inputs and 0.3 for grayscale.

Table 3.4: Comparison of early and late fusion strategies

Scebario-1									
Fusion	Sum-Product (Late Fusion)			Weighted-Sum (Late Fusion)			Signature-Level (Early Fusion)		
Modality	OF-SIL	SIL-GRAY	OF-GRAY	OF-SIL	SIL-GRAY	OF-GRAY	OF-SIL	SIL-GRAY	OF-GRAY
3D-CNN	89.5	82.6	89.6	93.8	90.65	94.1	90.1	**95**	93.2
ResNet	75.6	77.5	69.6	62.2	60.2	59.1	92.1	**96.1**	93

Table 3.5: Comparison with previous work on TUM-GAID

Modality	**Method**	**N**	**B**	**S**	**TN**	**TB**	**TS**
Optical Flow	PFM [203]	75.8	70.3	32.3	50.0	40.6	25.0
	OF-CNN-NN [171]	99.7	98.1	95.8	62.5	56.3	59.4
	OF-ResNet-B [163]	99	95.5	97.4	65.6	62.5	68.8
	MTaskCNN-7NN [204]	99.7	97.4	99.7	59.4	62.5	68.8
	3D-CNN-SMP [164]	98.7	97.1	94.5	71.9	68.8	65.6
	Ours-3DCNN	**100**	**100**	**100**	70.5	88.9	85.9
Silhoette	GEI [111]	99.4	27.1	52.6	44.0	6.0	9.0
	SEIM [205]	99.0	18.4	96.1	15.6	3.1	28.1
	GVI [205]	99.0	47.7	94.5	62.5	15.6	62.5
	Ours-ResNet	99.8	99.8	99.6	89.9	90.5	90.1
Fusion	DGHEI + GEI [111]	99.4	51.3	94.8	66.0	3.0	50.0
	2D-CNN [164]	99.4	98.4	98.7	56.3	53.1	46.9
	3D-CNN [164]	100	99.4	99.4	75	62.5	62.5
	Ours-3DCNN (All)	100	99.8	99.89	82.2	81.0	82
	Ours-ResNet (All)	100	100	100	98.9	99.5	99.5

Table 3.6: Cross comparison between networks and modalities on CASIA-B dataset 90° viewpoint

Network	Modality	Scenario-1								
		Normal			Bag			Clothing		
		R1	R5	R10	R1	R5	R10	R1	R5	R10
2D-CNN	OF	97.74	99.8	100	60.4	87.5	98.7	48	66.2	67.2
	Gray	99.4	99.8	100	34.4	64.6	73.2	51	70.1	72.1
	Silhoutte	99.6	100	100	44.2	44.9	48.2	48	65.1	68.9
3D-CNN	OF	96.9	97.8	98.5	68.7	86.4	89.9	62.4	68.9	70.1
	Gray	98.3	99.6	99.8	36.5	37.7	42.5	46.2	48	50.6
	Silhoutte	99.3	99.3	99.6	72.2	75	75	40.1	45.2	49.7
ResNet	OF	98.8	99.3	100	39.9	45.2	62.4	37	39.9	45.6
	Gray	99.4	100	100	48.2	62.5	75	44.1	50.2	54.9
	Silhoutte	99.8	99.8	100	78.5	79.9	90.1	40.1	41.1	43.2

To compare our results to state of the art, we ran our experiment precisely as instructed in [111] for TUM-GAID. This way, we can compare our results with other methods in the literature which used the same setup. For identification, the gallery is exactly as determined in table 3.1, and the probe is chosen from the range in table 3.1. The only difference is that we use normal walk, backpack and shoes separately in our two scenarios. Table 3.5 shows this comparison.

3.6.3 The Experiment On CASIA-B Dataset

Since for each of the 124 subjects in CASIA-B, we have eleven view angels from 0° to 180°; we followed the same structure suggested in the [199, 200]. We used the first four normal walking sequences for training and the rest for testing, which gives us 5456 sequences only for training. Using normal walking along with backpack and coat scenarios in the test set, we can evaluate the robustness of our model when facing

clothing variations as well as lighting and view angle.

For the experiment, we use the common setup in the state of the art approaches such as [103, 203, 164]. We use all cameras as the gallery and use the 90° angle for the probe set. We follow the same structure as in our experiment on TUM-GAID. First, we compare the models for the optical flow, grayscale and silhouette modalities. We do this in two scenarios similar to the TUM-GAID experiment. In **Scenario-1**, we obtain the identification results only for the 90° viewpoint and in **Scenario-2**, for all existing eleven viewpoints in CASIA-B dataset. The results are presented in table 3.6 and 3.7, respectively. By analysing the results, it is obvious that the clothing variations have the most effect on accuracy. Also, by comparing tables 3.6 and 3.7, adding more angels to the experiment significantly increases the performance of all three networks. By focusing on the clothing variations that is the weak link of our experiment, the most promising results belong to the ResNet network using the Silhouette sequence modality. ResNet does not provide state of the art results on carrying bag, and we can observe better results using grayscale images.

Analysing the data in this chapter gives us great insight into feature extraction using different modalities. However, since we obtained the most promising results from the silhouette sequences on both CASIA-B and TUM-GAID datasets, we will shift our focus to a more modern gait feature extraction method in the next chapter, which employs attention mechanisms to improve our performance using the silhouette sequence modality. This approach is chosen based on the fact that view variations play a significant role in gait signature extraction. Moreover, in practical long term person Re-Id scenarios, a gait cycle rarely gets completed from one camera to the other.

Table 3.7: Cross comparison between networks and modalities on CASIA-B dataset all eleven viewpoints

Network	Modality	Scenario-2								
		Normal			Bag			Clothing		
		R1	R5	R10	R1	R5	R10	R1	R5	R10
2D-CNN	OF	97.74	99.8	100	60.4	87.5	98.7	38.1	41.2	44.2
	Gray	99.4	99.8	100	91.4	93.6	96.2	39.1	45.1	45.1
	Silhoutte	99.6	100	100	44.2	44.9	48.2	39.8	45.1	46.4
3D-CNN	OF	96.97	97.86	98.52	90.7	93.4	99.9	46.6	49.5	49.5
	Gray	98.3	99.56	99.8	88.5	89.7	93.5	45.3	48.1	48.1
	Silhoutte	99.3	99.3	99.6	72.2	75	75	47.1	48.5	52.1
ResNet	OF	98.8	99.3	100	92.9	95.2	95.4	44.1	48.3	48.9
	Gray	99.4	100	100	93.2	94.5	94.8	43.2	45.3	45.3
	Silhoutte	99.8	99.8	100	98.5	98.9	99.6	49.9	59.5	60.2

3.7 Chapter Summary

This chapter proposed an approach for modality fusion using 2D CNN, 3D CNN and resent for spatial-temporal feature extraction. We essentially manipulated low-level data blocks in convolutional neural networks to better preserve spatial and temporal information. Using the extracted spatial and temporal information, we tried to improve the performance of the entire architecture by providing better inputs to the convolutional neural networks.

The method proposed in this chapter achieved great results on both benchmark datasets, but at high computational and preprocessing cost, it could be inefficient for practical use. Therefore, in the next chapter, we will shift our focus to a more modern spatial-temporal gait feature extraction architecture which uses LSTM and spatial and temporal attention interfacing to model the gait motion information and reduces the need for data prepossessing.

Chapter 4

A Spatial-Temporal Attention framework for Gait feature extraction

In the last chapter, we took a traditional approach to gait recognition. By comparing different modalities on different architectures, we gained insight into how the use of low-level data blocks representing spatial and spatial-temporal information can improve the impact of CNN architectures. We experimented with the fusion of extracted features using different modalities with excellent results on Optical flow and silhouette images from TUM-GAID and CASIA-B datasets by performing an extensive comparative study. Our experiment in the previous chapter showed notable results in TUM-GAID but less than optimal in CASIA-B, which indicates that the view variation plays an essential role in gait recognition. The experiment also showed that spatial and spatial-temporal features alongside each other would present us with better outcomes.

4.1 Introduction

As video surveillance becomes more widespread and video data becomes more available, the need for a better and more robust Person Re-Id framework becomes more evident. Furthermore, pedestrians in surveillance videos are the central area of concern in real-world security systems. Hence, new challenges are presented to the research community to identify a person of interest and understand human motion. Our study on the human gait provides for a non-invasive and robust way of feature extraction which can be used for gait recognition in surveillance systems in a remote and non-invasive manner.

In real-world scenarios, the subjects passing through an elaborate surveillance network cannot be expected to act predictably. We might not even get a complete gait cycle from a person of interest in most cases. As demonstrated in the last chapter, Clothing variation or carry bags considerably impact the system's performance in re-identification problems. Other abnormalities, including the camera angle, significantly aggravate the intra-class variation. Moreover, the similarity between gait appearances of different people extracted from low-level information introduces inter-class variations, resulting in similar gait signatures in more complex cases.

Due to the complex surveillance environments, the changes in camera viewpoints and diverse human behaviours, we could assume that human gait is always irregular. There are massive irregular gait sequences with varied lengths, asynchronous walking paces, different size strides, and inconsistent phases which can be longer or shorter than a complete gait cycle. Moreover, viewpoint variations play an influential role in irregular gait recognition. Firstly, extraction of the periodic motion cues becomes very challenging due to the difficulty in acquiring an entire Gait cycle. Secondly, it will mag-

nify the intra-class variations as mentioned before. Additionally, other pedestrians in the scene might have similar Gait patterns, which could significantly skew the variables due to inter-class confusion.

Since human motion is the main subject of this research area, a spatial-temporal approach seems essential. Unfortunately, most existing methods use shallow motion cues by employing Gait Energy Images (GEIs) to represent temporal data, leading to the loss of a vast amount of dynamic information. Although creating a deep learning-based gait recognition option that can robustly extract gait features has been tried in some of the literature provided previously [100, 103], [104], most of them need to detect at least one complete gait cycle and are not robust to the viewpoint changes. Therefore, irregular gait recognition concerning viewpoint variations still needs particular attention.

4.1.1 Attentional interfacing

When watching a video clip, we focus on some frames more than others. Depending on what we are looking for, there might be areas of each frame that attract more attention. This behaviour is also extended to other applications that involve a continuous sequence such as transcription, translation [206], parsing text [207] or voice recognition [208]. While transcribing an audio recording, we listen to the section we are writing down more carefully.

By using an attention mechanism, neural networks could mimic the same behaviour by focusing on essential sections of the information sequence. This can be achieved using a recurrent neural network that watches over another RNN using an attention distribution interface. The attending RNN can focus on different positions of the at-

tended RNN at each timestep by regulating its focus to different extents. This way, the RNN could learn where to focus deferentially. The attention distribution in such systems is usually based on content. The attending RNN generates a query. The query is combined (dot product) with each input item to generate a score. These scores are usually fed into a softmax layer to generate the attention distribution.

Attention can also interface between a CNN and a traditional RNN or LSTM network. This method can be employed in image processing to concentrate on diffident parts of the image at each timestep. Figure 4.1 shows a sample from our attention mechanism described in chapter 4. One of the most popular uses of this method is described in [209] for image captioning, where a CNN extracts the high-level features, and then an RNN provides a description of that image. To generate a caption, the RNN should concentrate on different parts of the image provided by the CNN.

Figure 4.1: Sample result from our attention interface at three consecutive timesteps

Another example of interfacing CNN and RNN with an attention mechanism is diagnosing diseases from medical images. In [210] the authors use an attention mechanism to diagnose thorax disease from chest x-rays. Their attention guided convolutional neural network (AG-CNN) consists of three branches. The global and local branches use ResNet-50 as a backbone. A fusion branch combines the pooling outputs of the global and local branches. In [211], The input image is encoded by a dense CNN and decoded using an LSTM network to capture mutual dependencies between

the labels.

Soft Attention vs Hard Attention

There are two types of attention known as ”Hard Attention” and ”Soft Attention”. The main idea behind attention is to change the weight of certain features according to some externally or internally provided weights. When these weights are provided internally, the process will be called ”Self Attention”. Soft attention uses continuous weights, and hard attention uses binary weights. For example, [210] is an example of hard attention since it crops part of the image using its global branch. So the cropped section weights one, and the rest of the image weights zero. The downside of hard attention is the fact that it is none-differentiable[1], therefore unable to train end-to-end.

Soft attention is used in several works in order to train attention weights. One example of soft attention is ”Squeeze and excitation blocks” introduced in [212] which re-weight the responses at certain levels of the network to model dependencies between the channels of the features extracted by the CNN. After each convolution, the squeeze operation aggregates the global feature responses at a spatial level. Then the excitation operation produces a channel-wise weight response. Finally, the convolution operation and the excitation operation outputs are multiplied (channel-wise) and passed into the next convolutional layer. These blocks include the global information in the network’s decision-making by accumulating the information from the entire receptive field. However, a typical convolution only looks at local spatial information. The excitation operation’s generated weights are similar for different classes in lower

[1] In calculus, a differentiable function of one real variable is a function whose derivative exists at each point in its domain. In other words, the graph of a differentiable function has a non-vertical tangent line at each interior point in its domain. A differentiable function is locally well approximated and smooth as a linear function at each interior point and does not contain any break, angle, or cusp.

layers of the network, but they become more discriminative in higher layers. An important point to notice is that most excitation weights become one at the last stage of the network, so the squeeze and excitation blocks should not be used at this level. Moreover, these blocks can be integrated into popular networks such as ResNet at particular stages without considerable overhead to the training parameters.

Self-Attention

Self-attention (intra-attention) finds relationships between different positions of a single sequence. This technique is used in machine reading with the help of LSTM [213]. In the [209] first, a CNN encodes the image and extracts features, then an LSTM is used to decode the convolutional features, and the weights are learned through attention. A Stand-alone self-attention has been introduced in [214], which asks whether attention could be used instead of convolution or serve as an interface to support convolutional models. The authors introduce a self-attention layer that can reduce the training parameters while replacing a convolutional layer. The results presented by the authors in this article are obtained by replacing convolutional layers in ResNet with a self-attention block on the ImageNet dataset. It is noticeable that the first convolutional layers and the 1×1 convolutional layers remain untouched. With this method, they reduced the parameters by 29% compared to ResNet-50. They introduced a memory block similar to convolution works on a small area at position (i, j).

Global Attention vs Local Attention

Another categorisation of attention is the ”Global” and ”Local” attention. Global Attention is a blend of soft attention, and local attention is a mixture of hard and soft

attention. [215] proposed a local attention mechanism that essentially improves hard attention and makes it differentiable. Global/Soft attention attends to the whole space [209] on the input sequence, but Local/Hard attention attends to only part of the input space [215].

4.1.2 Tackling Challenges

This chapter proposes a novel approach using attention mechanism to solve the problems mentioned earlier and reduce the inter and intraclass variations faced in more complex datasets such as CASIA-B and OULP. We use these two datasets because of their tremendous complexity, providing a robust solution to the viewpoint variations. Although the TUM-GAID dataset was appropriate for the experiment in the first chapter to give us insight on clothing and carry bag variations, it is view-invariant and consequently not appropriate for a cross-view study.

Approach

Our approach uses a spatial-temporal attention mechanism to learn gait information from a sequence of silhouette images. Humans usually focus on distinctive features in a person's movements and distinguishing characteristics to recognise one another. In other words, their attention is diverted to specific regions in a scene to find salient features. A profound challenge in computer vision is detecting the salient regions of an image [216]. Numerous works in the literature focus on directing the network's attention with saliency maps [216, 217, 218]. It has also been used in the past for action recognition problems like [219] which uses selective focus on RGB videos. Since we established the ResNet (Figure 3.9) learning ability from silhouettes in chapter 3, we

use it in our architecture for spatial map feature extraction from the silhouette images in each sequence.

We then use a spatial attention mechanism to focus on the important areas of the spatial feature maps. After that, We use a Long Short Term Memory (LSTM) setup to learn temporal gait features. Since the output of the LSTM at each time has a different impact on the performance, we need to assign different weights to each output to control their contribution using a temporal attention mechanism.

Figure 4.2 shows a high-level overview of our approach in this chapter which uses an LSTM (section 2.1.4) and attentional interfacing (Section 2.1.5) for gait recognition. To summarise, the ResNet extracts spatial feature maps passing them into a spatial attention mechanism that pays attention to the salient regions of each image. Then to extract the temporal motion features, an LSTM is used to allow the network to learn when to remember and forget relevant motion information. Since different timesteps of the LSTM have different effects on the network, a temporal attention mechanism is employed to reduce redundancy by giving more weight to the discriminative gait features.

Our method is evaluated on CASIA-B [200], used in the previous chapter and OULP from the OU-ISIR Gait Database, Large Population Dataset [220] with 4007 subjects to get more view variations. The OULP dataset allows us to determine statistically significant performance differences between currently proposed gait features. We achieved great results on view-invariant and view-variant gait sequences, which are further improved in the next chapter of this thesis.

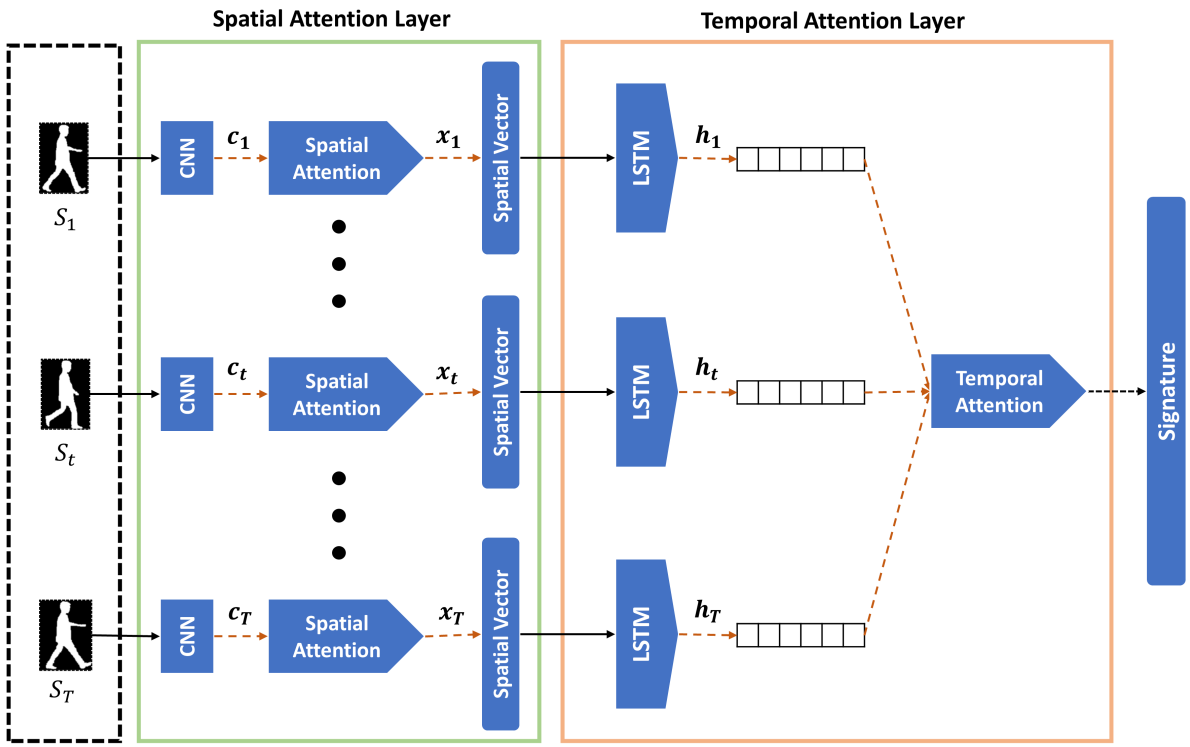


Figure 4.2: Overview of our spatial-temporal attention approach

4.2 Spatial Attention Layer

To extract the spatial features, we use the raw silhouette images from each sequence directly injected into our ResNet in figure 3.9. Contrary to popular methods such as [103] or [137] which use GEI as the input of their networks and lose an immense amount of temporal information in the process, our proposed approach accepts a random number of gait silhouette images as the input of the ResNet. The previous chapter shows that ResNet with four residual blocks and one convolutional layer shows excellent results for extracting spatial features from silhouette modality. So given $S = \{s_1, ..., s_t, ..., s_T\}$ as a set of silhouette images where s_t is the silhouette image at time t, the feature maps extracted by the network can be described as below where $c_t \in \Re^{N \times W \times H}$ for each s_t. N represents the number of feature maps with a width of W

and height of H.

$$C = \{c_1, ..., c_t, ..., c_T\} \tag{4.1}$$

Furthermore, the feature maps extracted from each image are fed into the spatial attention component to generate raw identity feature vectors with semantic features. Most of the existing CNN based gait recognition methods extract features indiscriminately using GEI. These methods have shown promising results for gait person Re-Id despite numerous inter and intraclass variations. However, they lose a large amount of temporal information [2, 221, 104]. Some of these works [222], [223], [224] and [221] use generative models or generative adversarial networks to perform view transformation and reduce the variations. On the other hand, discriminative models look for particular gait features under any condition (for example, viewpoint) and compare these features without deviation.

All the methods mentioned above suffer from the same problem: the loss of discriminative features. Due to the weight sharing strategy in CNNs, the spatial heterogeneity of gait energy images might be overlooked, resulting in the loss of discriminative data. This problem is expected since a person's body parts vary in movement and shape in a gait energy image. To extract the discriminative features, we use an attention mechanism that takes the feature maps from the network's output and shifts the focus onto the feature map regions related to a person's identity and finally creates a spatial feature vector. The attention mechanism solves the encoder-decoder bottleneck problem. The encoder-decoders are designed using RNN and are used in neural Machine translation and sequence to sequence (Seq2Seq) predictions, so they seem like a viable option for our Re-Id problem. In a seq2seq architecture, the encoder summarises the input into a single vector of fixed length. The drawback is that the system forgets the

earlier parts of the sequence as the sequence becomes longer. The attention mechanism solves this problem. The inner workings of our spatial attention mechanism are shown in Figure 4.3. As it can be seen, this is a dual attention network inspired by [225] in which the upper block contains three 1×1 convolutional layers with ReLU activation and a softmax layer, and the lower block is a residual attention system with a bottleneck architecture as explained in 3.2.

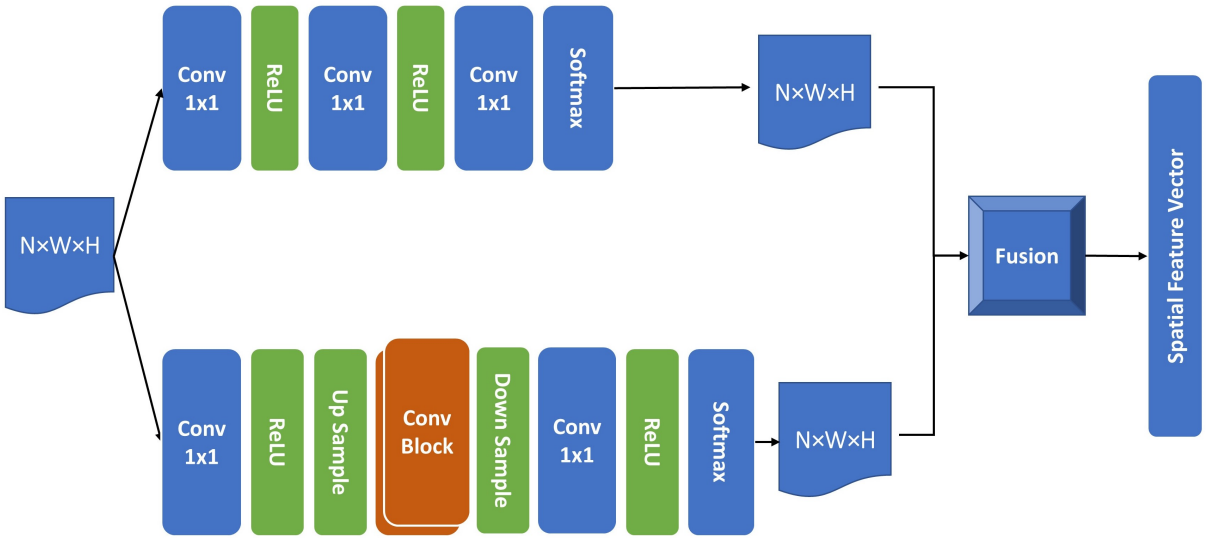


Figure 4.3: our dual spatial attention mechanism

The upper part of the network consists of three convolutional layers and two Relu layers with a softmax layer at the end. It focuses on small area semantic regions due to its 1×1 convolutional layers. A 1×1 convolution works like a mini neural network for each pixel on the feature maps [226] which gradually reduces their number without changing the size of its receptive field.

The lower part of the network has two convolutional layers, two residual blocks, up-sampling, down-sampling and a softmax layer. It focuses on large areas with semantic features since its receptive field increases after each conv block. It is crucial to note that a fully connected block has not been used in our spatial attention mechanism because we do not intend to lose any spatial information by flattening the feature map

or changing the size of the receptive field. Finally a max fusion operation is used to combine the extracted semantic features and generate a spatial feature vector of $X = \{x_1, ..., x_t, ..., x_T\}$ for the silhouette set of $S = \{s_1, ..., s_t, ..., s_T\}$. The feature vector X is created by calculating the below equation for all N elements of x_t.

$$x_t = \sum_{i=1}^{W} \sum_{j=1}^{H} W_{t,i,j}.c_{t,i,j} \tag{4.2}$$

In the above equation, W_t is the spatial attention weight matrix of c_t at time t and is of the size $W \times H$, which is produced by going through our attention block.

4.3 Temporal Attention Layer

The temporal Attention Layer receives the spatial feature vectors from the spatial Attention Layer and feeds them into its Long-Short Term Memory (LSTM) blocks to extract temporal gait features and model periodic motion. We use an LSTM setup similar to the one described in [227] and [228] to model the gait cycle of each sequence S. LSTM (generally described in 2.1.4) is a type of RNN architecture that can map input sequences to outputs with variable lengths. This quality makes them the ideal candidate to model complex temporal dynamics in a sequence of frames. A traditional RNN can also learn complex temporal dynamics by getting frame sequences as input, mapping them to hidden states and finally to the outputs. However, it is very challenging to train a traditional RNN for long term dynamics. Due to the numerous layers of a recurrent neural network, long sequences face the problem of vanishing or exploding gradients [162]. This problem accrues when propagating gradients with different time steps through the layers of the RNN. LSTM employs memory units that learn when to

forget and update the hidden states to solve this problem. For a gait cycle, repeated movements can be forgotten by the LSTM, and new movement patterns can be committed to the memory for extracting discriminative gait motion features.

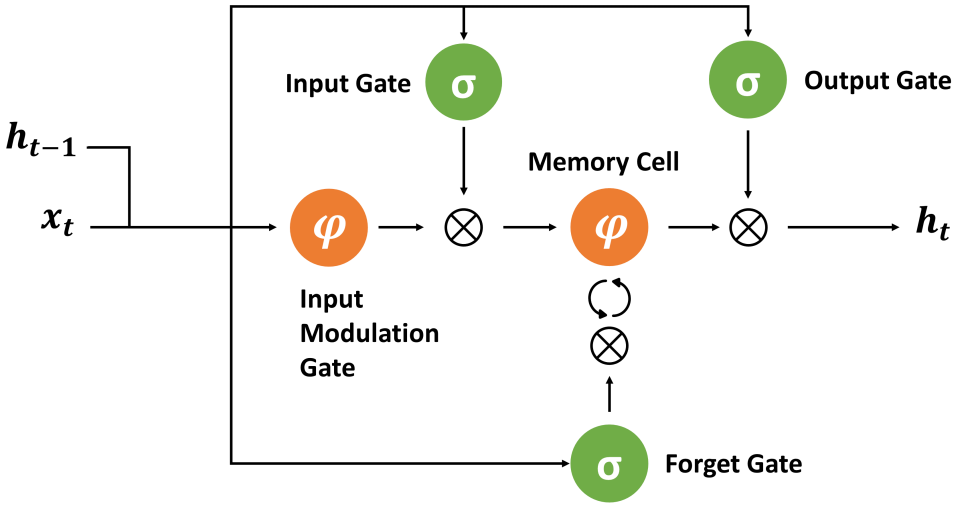


Figure 4.4: Overview of LSTM used in our temporal stream [10]

As can be seen from figure 4.4, the LSTM, in this case, is composed of a memory cell $c_t \in \mathbb{R}^N$, an input gate $i_t \in \mathbb{R}^N$, an output gait $o_t \in \mathbb{R}^N$ and a forget gate $f_t \in \mathbb{R}^N$ which can read and write to the memory cell and an input modulation gate which is denoted by $g_t \in \mathbb{R}^N$. The σ represents the sigmoid and φ represents the tanh non-linearity functions between $(0, 1)$ and $(-1, 1)$ respectively. Therefore, the sigmoid gates control the reading and writing of the memory cell. At each given time t the LSTM updates the hidden unit h_t using three variables as input. These variables are the current input x_t, the previous hidden state of all the hidden units h_{t-1} and the previous state of the memory cell c_{t-1}. The updated equations at each time step are listed below. The memory cell is the sum of the previous memory cell modulated by the forget gate and the input modulation gate, a function of the current input and the previous hidden state. The input gate modulates this function. Because both the input gate and the forget gate are sigmoid in the range of $(0, 1)$, they are used as

modulators that help LSTM decide to forget or consider the current input. The output gate modulates the memory cell at the time of transfer to the hidden state.

$$i_t = \sigma(W_{xi}x_t + w_{hi}h_{t-1} + b_i) \tag{4.3}$$

$$f_t = \sigma(W_{xf}x_t + w_{hf}h_{t-1} + b_f) \tag{4.4}$$

$$o_t = \sigma(W_{xo}x_t + w_{ho}h_{t-1} + b_o) \tag{4.5}$$

$$g_t = \varphi(W_{xc}x_t + w_{hc}h_{t-1} + b_c) \tag{4.6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{4.7}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{4.8}$$

$$h_t = o_t \odot \varphi(c_t) \tag{4.9}$$

With the memory cells' help, complex temporal dynamics such as gait temporal information will be learned during a frame sequence. The hidden states are computed for each frame in consecutive order from h_1 to h_T to create $H = \{h_1, ..., h_t, ..., h_T\}$ which is a representation of gait cycle information in a sequence. Given $h_t \in \mathbb{R}^N$ at time step t, then the overall output of the LSTM for each spatial input vector is a matrix of $N \times T$. We keep all the hidden states from the previous timesteps, and each hidden state composes a matrix of information. Because this method cannot forget the earlier information in the sequence, it will provide us with the exact opposite problem: "redundant information". Therefore, The LSTM output is biased towards the latest time steps. We do not need all the information (The full matrix), and we need to keep only the relevant temporal information at appropriate time steps over the entire sequence.

Therefore, our temporal attention block decreases the temporal features by assigning different weights to each output at each time step.

4.3.1 Mechanism for temporal attention

There are several applications for temporal attention in the literature including captioning images and videos [229], [230, 209], [231], video segmentation [232] and activity recognition [233], [234]. In all of these works, the fundamental idea is that the amount of valuable information provided by different frames in a sequence is not constant. Most frames provide contextual information, and only some contain discriminative information. For example, in gait recognition over a sequence representing a gait cycle, the frames that show clear stance and swing phases (Figure 2.2) should be more important than the rest. Based on this observation, our temporal attention block automatically pays different attention levels to different frames in the sequence.

To calculate the attention weight of a particular vector from the LSTM, we organise all of the T LSTM outputs into a $T \times N$ matrix M. Each row in the input matrix $M = \mathbb{R}^{T \times N}$ is the output of the LSTM at timestep t. M is then fed into our temporal attention mechanism. The temporal attention performs a $1 \times N$ convolution to create a $T \times 1$ Vector, which is the global distribution of the LSTM's output at time step t. Then two fully connected operations and a softmax are performed to create a $N \times 1$ score vector. A Relu operation is performed after each fully connected layer. The Relu activation is chosen as a result of the experiment. The first fully connected layer with a Relu operation will generate a vector of size $\frac{T}{4} \times 1$. Then the second fully connected layer and the Relu create a $T \times 1$ vector. This method has more non-linearity and can handle complicated correlations between the LSTM outputs at each timestep t.

Finally, the softmax layer generates the attentional score vector. In order to encode the temporal movements, we summarise all LSTM time steps using $\Sigma_{t=1}^{T} w_t.h_t$ on $H = \{h_1, ..., h_t, ..., h_T\}$ and pay attention to the most discriminative features. The w_t predicts the weight value of the image at each timestep. Ultimately, the $N \times 1$ score vector for the input sequence of silhouette images, S is generated. This vector is the generated spatial-temporal gait signature shown in figure 4.2. We use this feature extractor to extract the probe and gallery set motion features while testing.

4.4 The Experiment

To obtain the results of our approach, we use two publicly available datasets, CASIA-B and OULP. Since the number of categories is vast and the amount of training data is insufficient for each identity class, we needed to find a way to train our network independent of class information and find similarities between our extracted gait signatures. For this purpose, we adopted a Siamese network architecture (following similar work in [40]). As explained before, we can train two identical subnetworks with the same parameters and shared weights in Siamese architecture. We use the same network in figure 4.2 as the subnetworks and map the gait signatures to low dimensional space. This way, a similarity metric can be used. If the similarity metric is small, the pairs are from the same category, and if it is large, they are from different categories.

4.4.1 Training the network

To train our network, two random silhouette sequences S_i and S_j are selected as positive or negative pairs. Positive pairs have the same category, and negative pairs

have a different category. To indicate which (S_i, S_j) pair is positive and which one is negative, a one or a zero is given to each pair as binary labels. one is for positive pairs, and 0 is for negative pairs. We take all the positive sequences (same as the number of classes) for the training data and select the same number of negative pairs.

Consecutive loss layer

To map the silhouette sequences to points in a low dimensional space, we use a contrastive loss layer which takes the output of our two identical subnetworks. If (S_i, S_j) are a pair of silhouette sequences where each sequence has been processed by our network in figure 4.2 and given sequence feature vectors, v_i and v_j, the siamese network training objective will be determined by the below formula which is the consecutive loss function for the positive and negative pairs.

$$Loss(v_i, v_j) = \begin{cases} \frac{1}{2}||v_i - v_j||^2 & i = j \ (positive\ pairs) \\ \frac{1}{2}[max(m - ||v_i - v_j||, 0)]^2 & i \neq j \ (negative\ pairs) \end{cases} \tag{4.10}$$

The distance between feature vectors is determined by $||v_i - v_j||$, which is the euclidean distance between the two vectors. When the identity is the same, $i = j$, the LOSS function encourages the features to be close together, and when the identity is different $i \neq j$, the LOSS function separates the points in space by a margin of m. So a lower euclidean distance during training indicates more similarity between two silhouette sequences S_i and S_j. So to reach better results, we need to minimise the LOSS. To compute the backpropagation gradients of the temporal attention, we can use the below formula for every individual output of our LSTM related to a S_i silhouette

sequence at timestep t.

$$\begin{cases} w_t = \frac{\partial M}{\partial h_{it}} = \frac{\partial v_i}{\partial h_{it}} \\ \\ h_{it} = \frac{\partial M}{\partial w_t} = \frac{\partial v_i}{\partial w_t} \end{cases} \tag{4.11}$$

The backpropagation for the LOSS function is also formulated as

$$\begin{cases} \frac{\partial Loss}{\partial h_{it}} = \frac{\partial Loss}{\partial V_i} \times \frac{\partial v_i}{\partial h_{it}} = \frac{\partial Loss}{\partial V_i} \times w_t \\ \\ \frac{\partial Loss}{\partial w_t} = \frac{\partial Loss}{\partial V_i} \times \frac{\partial v_i}{\partial w_t} = (\frac{\partial Loss}{\partial V_i})^T \bullet h_{it} \end{cases} \tag{4.12}$$

Training the Siamese Network

To summarise the training process, the Siamese architecture's two sub-networks (Figure 4.2) are trained simultaneously on positive and negative pairs of silhouette sequences fed into the network. Each sub-network creates a gait signature vector which is then combined using the contrastive loss layer. The contrastive loss layer computes the LOSS using equation 4.10 for positive and negative pairs. Then equation 4.12 is used to train the model. For training, we use the stochastic gradient descent as used in [40]. The mini-batches are chosen randomly. The Loss function is calculated for each mini-batch in the contrastive loss layer and then backwards-propagated to lower layers. [235] uses a step learning rate policy which we will utilise for each mini-batch. The learning rate used is 10^-5, weight decay is 5×10^-5, and the momentum is set to 0.9 when we run our training. We multiply the initial learning rate by 0.7 every 2×10^5 iterations to avoid overfitting. Another reason for overfitting could be the different data distributions for pairs with negative and positive labels. Hence, 60% of the neurons were randomly eliminated from the LSTM while training.

4.4.2 Datasets

CASIA-B [200], and OULP [220] Two of the most extensive publically available datasets were used to test our approach. The CASIA-B data set has been extensively described in the past chapters, so we only go through the OULP data set in this section. Consequently, we explain how these data sets were used to test our approach.

OULP Dataset

OU-ISIR Gait Database, Large Population Dataset (OULP) is one of the most extensive datasets for vision-based gait recognition. The dataset includes 4007 subjects (2135 males and 1872 females) ranging from 1 to 94 years old. Two silhouette sequences are provided for each of these subjects. One sequence is used for the gallery and the other for the Probe. All images in this data set are of the size 88×128. Each of the main subsets is divided into five subsets based on the observation angles, 55°, 65°, 75°, 85°, and including all four angles. All these characteristics will make this data set ideal for our purposes. Figure 4.5 shows a sample of one subject taken from various angles in the dataset.

CASIA-B Dataset

CASIA-B is an extensive Multiview gait database, which was created in January 2005. There are 124 subjects, and the gait data was captured from eleven camera views. Three variations, namely view angle, clothing and carrying condition changes, are separately considered. The 124 subjects were videoed for three to five seconds in the same location indoors with the resolution of 320×240. The video frame rate is around $25fps$, and subjects are either with no extra condition, carrying a bag or backpack or

Figure 4.5: Sample Silhouette sequences from multiple angels of OULP dataset

wearing a coat or jacket. Ten videos were captured from each view, six with normal conditions, two with a jacket or coat and two carrying a bag or backpack. It is essential to mention that we are using all the viewpoints included in this dataset when extracting the gait signature. This challenging dataset with eleven view variations from 0 to 180 degrees proves very helpful in showing the robustness of our framework for multi-view gait recognition.

Evaluation Technique

For evaluation, we use the same Rank system from [111] described in chapter 3. To recap, Rank 1 accuracy corresponds to the scenario where the top one identity is the correct one. Rank 5 is when the correct identity is in the top five results, and Rank 10 is when the correct identity is in the top 10 results. This evaluation form is more practical, particularly in real-world scenarios with uncontrolled environmental variations.

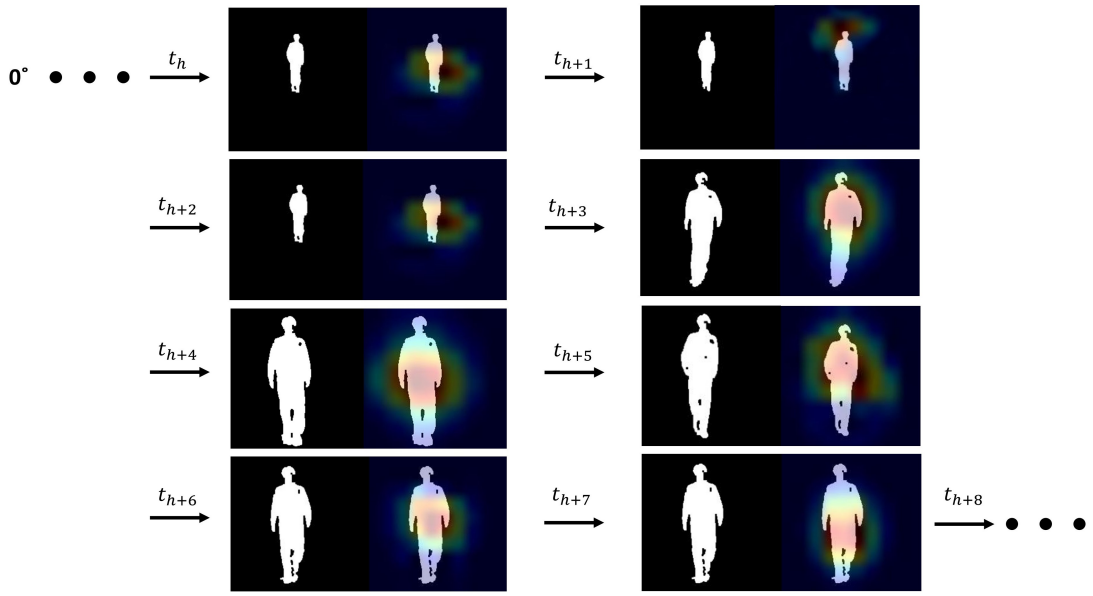


Figure 4.6: Example weight heat map visualisation for our attention mechanism proposed in this chapter for eight consecutive timesteps

4.4.3 Experiment on CASIA-B dataset

Experiment Design

Previous works such as [236, 237, 238, 239, 240, 241, 104] have experimented with CASIA-B dataset in the past. Accordingly, in order to be able to compare our method with these works, we set our experiment in a similar fashion. For the purpose of our experiment, the Data in the CASIA-B dataset is used in two different scenarios. For Scenario-1, a 20%, 80% division is used for the subjects. The training set consists of 19, and the testing set contains 105 randomly chosen subjects. For Scenario-2, a 70%, 30% split is used in which 75 subjects are for training, and 49 subjects are for testing. From the subjects selected for testing, the normal walk 1 to 4 are chosen as the validation set in both scenarios.

We chose four popular angles in the CASIA-B dataset for this experiment. $126°$, $0°$, $54°$ and $90°$ are chosen as probe angles in the previously mentioned studies, so we use the same angels to evaluate the performance of our network.

Table 4.1: Comparison between our method and the state of the art with 0°probe on Rank-1 Scenario-1 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[239]	-	45	34	20	8	5	5	14	18	25	40	21.4
[242]	-	85	47	26	25	28	25	27	37	68	95	46.30
Our Method	-	93	64	32.5	29.5	27.9	20.2	30.1	52.5	70.5	94.9	51.51

Experiment Results

Tables 4.1 to 4.7 show the results of our experiment for scenario-1 and scenario-2 on the CASIA-B dataset and compare them against the state of the art methods with four different angels as Probe. Some of the methods in the literature such as [48], [236], [237] only provided average results on certain angles. When analysing the tables, we need to keep in mind that scenario-1 provides a very small training set which usually results in a bad performance in deep learning-based methods. However, our method still shows better performance than some previous works. If we consider only the average result, which is the mean value of all the angles, our method outperforms all the others on Rank-1 for cross-view matching.

Table 4.2: Comparison between our method and the state of the art with 54°probe on Rank-1 Scenario-1 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[242]	24	65	97	-	95	63	53	48	34	23	22	52.40
[238]	28	67	98	-	97	86	76	69	58	31	20	63
Our Method	34.2	65.5	91.3	-	92	85.1	76	63.1	59	50.1	32.5	64.88

In scenario-2, the margin is even larger since we trained the network on a larger training set. In scenario-2 we reach the best result of 81.2% on 54° angle which exceeds [103] by 17.57%. So we can safely assume that our technique can successfully learn more accurate discriminative gait features and display a degree of generalisation

Table 4.3: Comparison between our method and the state of the art with 90°probe on Rank-1 Scenario-1 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[242]	18	24	41	66	96	-	95	68	41	21	13	48.30
[238]	17	26	54	84	98	-	98	84	50	25	14	55
Our Method	31.1	33	60.1	81.9	99	-	93.9	85.6	54.6	30	28	59.72

Table 4.4: Comparison between our method and the state of the art with 126°probe on Rank-1 Scenario-1 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[242]	25	29	35	49	60	78	98	-	98	75	22	56.90
[238]	18	33	58	68	78	83	98	-	95	63	27	62.10
Our Method	33	39.1	60.5	71	72	84.5	90.9	-	95	73	46	66.5

Table 4.5: Comparison between our method and the state of the art with 0°probe on Rank-1 Scenario-2 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[103]	-	87.10	58.6	39.52	28.23	33.87	31.45	37.90	46.77	62.10	67.74	49.27
Our Method	-	99	85	67	43	35	40	55	73	87	91.5	67.55

Table 4.6: Comparison between our method and the state of the art with 54°probe on Rank-1 Scenario-2 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[103]	50	63.71	83.87	-	89.52	82.26	72.58	65.32	57.26	43.55	28.23	63.63
Our Method	65	82	99	-	98	83	90.1	85	81	69.9	59	81.2

Table 4.7: Comparison between our method and the state of the art with 90°probe on Rank-1 Scenario-2 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[103]	32.26	35.48	52.42	70.16	95.16	-	95.16	80.65	56.45	33.87	29.03	58.06
Our Method	34	45	63	78.6	94.3	-	96.8	80.9	67	50.1	40	64.97

for cross-view gait recognition.

Figure 4.6 shows a heat-map visualisation of the weights for our attention mechanism on a single silhouette sequence at 0°. As seen, the attention mechanism focuses more and more on the relevant areas of the image at each timestep.

Table 4.8: Comparison between our method and the state of the art with 126°probe on Rank-1 Scenario-2 on CASIA-B dataset

Method	0	18	36	54	72	90	108	126	144	162	180	AVG
[103]	33.06	45.16	60.48	72.58	84.68	86.29	93.55	-	94.35	59.68	35.48	66.53
Our Method	58.9	60.2	73	83	85.9	80.5	94.9	-	93	79	50.6	75.9

4.4.4 Experiment on OULP dataset

Experiment Design

Previous methods such as [220, 100], [2, 57] have experimented with OULP dataset in the past, so in order to be able to compare our method with these works, we set our experiment in a similar fashion. For the purpose of our experiment, the Data in the OULP data set is used in two different scenarios. The first scenario (Scenario-1) uses all subject silhouette sequences of the gallery set for the training set. For the validation set, all the gallery sequences are used, and for the test set, all the probe silhouette sequences are chosen. This division is made for us by the dataset authors, making experimenting a lot more convenient. In the second scenario (Scenario-2), we choose a random $60\%, 20\%, 20\%$ division for training, validation and testing, respectively. We split each view into the gallery sequence and the probe sequence for source and query sets for testing. To compare against other works, we set the experiment as performed in [220]. First, the performance is compared for each observation angle using $55°$, $65°$, $75°$, and $85°$ viewpoints, since the gait feature property is dependent on the observation angle.

Experiment Results

The results of this comparison are presented in Table 4.9 and Table 4.10 for both Rank-5 and Rank-1 in scenario-1 and scenario-2, respectively. These results are collected

Table 4.9: Comparison between our method and the state of the art on Rank-1 (Left) and Rank-5 (Right) for Scenario-1 on OULP dataset

	Rank-1 (%)					Rank-5 (%)				
Methods	**55°**	**65°**	**75°**	**85°**	**AVG**	**A-55**	**A-65**	**A-75**	**A-85**	**AVG**
[220] GEI-NN	84.70	86.63	86.91	85.72	85.99	92.39	92.84	92.87	93.01	92.76
[220] FDF-NN	83.89	85.49	86.59	85.90	85.47	91.53	92.81	92.88	92.83	92.51
[57]	-	-	-	87.70	87.70	-	-	-	94.70	94.70
[100] CNN	73.96	76.71	77.87	78.82	76.84	86.64	88.67	89.39	90.09	88.70
[100] SiaNet	90.12	91.14	91.18	90.43	90.72	94.98	95.90	95.99	95.97	95.69
Our Method	98	98.3	98.1	98.3	98.175	99	99.8	99.6	99.8	99.55

Table 4.10: Comparison between our method and the state of the art on Rank-1 (Left) and Rank-5 (Right) for Scenario-2 on OULP dataset

	Rank-1 (%)					Rank-5 (%)				
Methods	**55°**	**65°**	**75°**	**85°**	**AVG**	**A-55**	**A-65**	**A-75**	**A-85**	**AVG**
[2]	98.80	98.90	98.90	98.90	98.88	-	-	-	-	-
Our Method	99	99.1	99.1	99.1	99.075	99.9	99.9	99.9	99.9	99.9

for view-invariant recognition, so naturally, they are higher for all different methods. However, considering the Average results, our method still outperforms SiaNet by 7.5% in scenario-1 Rank-1 and by 4% in Scenario-1 Rank-5. This improvement is due to the use of raw silhouette sequences in our method. Most other methods employ GEI for feature extraction, which will lose much useful temporal information when applied to gait recognition. This method can be successful but only on a single gate sequence representing a complete gait cycle. In scenario-2, the results are better by 0.2%, which is not very significant but shows that further improvements on our method are possible. The MT method used in [2] is very successful in this case because there are no view variations while doing their experiment.

Table 4.11: cross-view comparison between our method and the state of the art on Rank-1 Scenario-1 on OULP dataset

Methods	**Probe Angle**	**55°**	**65°**	**75°**	**85°**	**Mean**
[100]	55°	-	65.76	32.92	19.48	39.39
[3] PDVS	55°	-	76.20	61.45	45.50	61.05
[3] AVTM	55°	-	77.72	64.54	42.69	61.65
Our Method	55°	-	96.4	94	86.3	92.23
[100]	65°	72.58	-	78.54	51.83	67.65
[3] PDVS	65°	75.99	-	77.09	65.48	72.85
[3] AVTM	65°	75.63	-	76.36	62.76	71.58
Our Method	65°	95	-	96.6	93	94.86
[100]	75°	39.13	78.30	-	81.22	66.22
[3] PDVS	75°	60.25	76.20	-	76.52	70.99
[3] AVTM	75°	59.88	74.90	-	76.31	70.36
Our Method	75°	93.6	95.8	-	92	93.8
[100]	85°	19.45	44.93	71.39	-	45.26
[3] PDVS	85°	40.48	60.62	73.12	-	58.07
[3] AVTM	85°	40.17	61.87	74.32	-	58.79
Our Method	85°	87.2	93.7	94.5	-	91.8

Table 4.12: cross-view comparison between our method and the state of the art on Rank-1 Scenario-2 on OULP dataset

Methods	**Probe Angle**	**55°**	**65°**	**75°**	**85°**	**Mean**
[2]	55°	-	98.30	96.00	80.50	91.60
Our Method	55°	-	96.8	94.05	91.9	94.25
[2]	65°	96.30	-	97.30	83.30	92.30
Our Method	65°	97.3	-	97	96.8	97.03
[2]	75°	94.20	97.80	-	85.10	92.40
Our Method	75°	94.7	97.83	-	97.1	96.54
[2]	85°	90	96	98.40	-	94.80
Our Method	85°	85.5	95.98	97.3	-	92.93

Further Evaluation

To further evaluate our method for use in the real world, we consider that in most practical applications, a subject might be recorded with various viewpoints from one camera view to another. Even in a single surveillance video clip, the subject can still turn, bend or change trajectory while walking. Since the OULP dataset provides such variations, it is reasonable to undertake this challenging task. Other studies such as [2, 100] and [3] tried the cross-view gait recognition for their methods, but since they use GEI in most cases, their results are significantly underwhelming. From these methods, only [100] uses a siamese architecture, but they fail to capture most of the temporal information. Both methods in [3] only use 1912 subjects, but our method uses 3836 subjects in all cases. We compared our approach in Tables 4.11 and 4.12 for Scenario-1 and Scenario-2, respectively. As shown from the results in Scenario-1 (Table 4.11), we overwhelmingly overtake the other methods by differences as significant as 22% on

average. In the challenging scenario-2 (Table 4.12), the [2] method is better than ours by 1.87% on probe angle $85°$, but in other angles, we take over by a 5% difference on average.

4.5 Chapter Summary

This chapter proposed an approach to reduce the inter and intraclass variations faced in gait feature extraction, focusing on view variations. The proposed feature extraction architecture generates feature vectors by focusing on salient discriminative areas of a silhouette sequence using a spatial and a separate temporal attention mechanism. The spatial vectors from the spatial attention layer are fed into an LSTM architecture which models the periodic motion in each gait silhouette sequence. By encoding the motion features using LSTM, we can keep the aspect ratio of the feature maps and hold on to the temporal motion features for longer. Finally, the outputs of the LSTM at each time step are organised in a feature matrix for use in the temporal attention mechanism, which gives a different weight to each time step of the LSTM output. This way, it keeps the more relevant timesteps and discards the none specific and redundant outputs. Finally, the most appropriate time steps generate the gait signature.

We use sequences of silhouette images instead of the popular GEI as the input for this method for the same reason. The temporal information is averaged upon GEI's creation, and much of the discriminative temporal information is lost. We then compared our method for intra-view and inter-view variations using CASIA-B and OULP datasets, which in some cases showed superiority to the previous strategies in the literature. Scenarios with different data divisions were considered for each one of the datasets. Furthermore, Since the number of classes was large and the amount of train-

ing data were limited for each identity class, we used a siamese network architecture to train our network independent of class information.

In the next chapter, we adjusted our approach, which proved very effective in improving the performance and reducing the effect of viewpoint variations upon experimentation.

Chapter 5

A robust approach to reduce the effects of viewpoint variations in gait Person Re-Id

5.1 Introduction

This chapter improves the previous approach for cross-view gait recognition by making intelligent changes to our method. In the previous chapter, we employed the Siamese architecture based on our approach to minimise the Loss function in equation 4.10. However, Siamese architecture only adopts distance metrics to guarantee positive pairs close enough and negative pairs far away, at least by a margin, without considering the previous category information of gait images.

The Loss function uses the distance between feature vectors by employing euclidean distance, $||v_i - v_j||$, between gait signatures created by our feature extraction method presented in Figure 4.1. When the identity is the same,$i = j$, the Loss func-

tion encourages the features to be close together, and when the identity is different $i \neq j$, the Loss function separates the points in space by a margin of m. This operation happens without paying attention to the particular identities before training.

A method has to be adopted which solves the problem mentioned above and make our framework even more robust to irregular gait recognition hence acquiring better accuracy in real-world scenarios. If the category identity information is not included in the training process, valuable data is lost in translation. Moreover, with the contribution of the viewpoint to inter-class confusion, in real-world surveillance scenarios, the accuracy of our system might vary drastically with changes in viewpoint, illumination and irregularities in gait. As explained before, other works in the literature have not considered using the identity information along with trained data in gait human Re-ID, but methods such as The Null Foley-Sammon Transform method (NFST) have been used in the past to solve problems such as lack of enough training data.

In this chapter, by following the same examples, we try to map our spatial and temporal features into the null space by using the learned Null Foley-Sammon transform (LNFST) method to improve the accuracy of our model. We project the silhouette sequences with the same identity to a single point in a learned discriminative null space. We achieve better accuracy in cross-view gait recognition since the distance between multiple views of the same subject is calculated in the null space. Therefore producing a robust approach to improve the accuracy of cross-view gait recognition and reduce the effect of viewpoint variations. An ablation study in section 5.3 shows that this robust approach is not limited to our attention mechanism and can be applied in other spatial-temporal feature extraction scenarios.

5.1.1 Null Foley-Sammon Transform method (NFST)

The Null Foley-Sammon Transform method (NFST) was first introduced in [243] for the face recognition to solve the small sample size issue. Later [57] proposed a Kernelised version of the same method for person Re-Id, which showed better results than state of the art. This method tried to map multiple features into a discriminative null space. In [136] the authors introduced a similar method for learning efficient spatial-temporal Gait features with Deep Learning. This method proposes a siamese neural network based on gait energy images (GEI), extracting discriminative spatial gait features. They also exploit the deep 3-dimensional convolutional networks, C3D, to extract the temporal gait features. Finally, both spatial and temporal extracted features are embedded into the null space.

5.1.2 learned Null Foley-Sammon transform (LNFST)

The learned Null Foley-Sammon transform (LNFST) is not the same as Foley-Sammon transform (FST) or linear discriminant analysis (LDA) [244] presented in 1975. Although they were both used in the past to solve the small sample size problem, the purpose of FST is to create a projection matrix in which each column is an optimal discriminant direction that could maximise the fisher discriminant criterion. For example, if $W \in \mathbb{R}^{d \times m}$ represents the projection matrix and each column is represented by w the Fisher discriminant criterion is calculated as the below equation where S_b is the intra-class, and S_w is the inter-class scatter matrix.

$$\Im(W) = \frac{w^T S_b w}{w^T S_w w} \tag{5.1}$$

To optimise this equation, the below eigenproblem should be solved. If S_w is non-singular, all eigenvectors $\{w^1, w^2, ..., w^{C-1}\}$ corresponding to the largest eigenvectors of $S_b S_w^{-1}$ must be computed and used as the columns of the projection matrix. This way, the projection matrix W can represent the data as a discriminative subspace with $C-1$ columns.

$$S_b w = \lambda S_w W \tag{5.2}$$

Problem with Foley-Sammon Transform (FST)

The problem with the FST presents itself in cases with a small sample size where $d > N$, S_w is singular, and we have to use solutions such as adding a new term to S_W for regularisation or reducing d to prevent numerical problems. Hence, to create a robust method of cross-view gait recognition, we will use the NFST method to learn a discriminative subspace. This subspace contains individual identity class points for C identity classes. So we need to learn the optimal projection matrix W. If each column of the projection matrix is denoted by w the below conditions should be satisfied:

$$\begin{cases} w^T S_w w = 0 & (Inter - class\ scatter) \\ w^T S_b w > 0 & (Intra - class\ scatter) \end{cases} \tag{5.3}$$

this method provides the best separability on the silhouette sequences (points in null space) regarding the Fisher discriminant criterion. Since NFST is a linear model [48] and the person Re-Id is naturally a non-linear problem due to the non-linearity of a person's gait, for training on silhouette extracted features, we need to use a kernel function. To kernelise the discriminative null space [48] uses a kernel-based metric

learning method for person Re-Id. We use the kernel function k for the two v_i and v_j feature vectors as presented below:

$$k(v_i, v_j) = \{\Phi(v_i), \Phi(v_j)\} \tag{5.4}$$

Functions $\Phi(v_i), \Phi(v_j)$ map the feature vectors v_i and v_j into a spaces with higher dimensionality. First, a feature matrix F is created on the training data using the method illustrated in 4.1. The computed feature matrix is then turned into a kernel matrix K using the below equation:

$$K = \Phi(F)^T \Phi(F) \tag{5.5}$$

Learning the projection matrix

For learning the projection matrix of the discriminative null space $W \in \mathbb{R}^{d \times m}$, first the inter-class scatter matrix S_w and the intra-class scatter matrix S_t will go through the process of kernelisation using the below equations in which I is an identity matrix, L represents a block diagonal matrix, and M is a $N \times N$ matrix with entries equal to $\frac{1}{N}$. The blocks in this matrix are the same size as the data points, which for a class $c = \{1, ..., C\}$ is denoted by N_c.

$$\begin{cases} K_w = K(I - L)(I - L)^T K \\ K_t = K(I - M)(I - M)^T K \end{cases} \tag{5.6}$$

Moreover, to kernelise the Equation 5.3 the S_W now needs to be replaced with the K_w and the orthanormal basis of K_t needs to be computed and replace w. We use kernel PCA method to compute the orthanormal basis. So if V_w is the matrix containing $v_i^c - \mu^c$

vectors. The V_t is another matrix with $v_i - \mu$ and $\mu = \frac{1}{N}\Sigma_{i=1}^{N} v_i$. So we then have,

$$\begin{cases} S_w = \frac{1}{N} V_w V_w^T \\ \\ S_t = \frac{1}{N} V_t V_t^T \end{cases} \tag{5.7}$$

Now, if $\tilde{K}$ is a centred matrix, its eigendecomposition is written as below where E contains $N-1$ eigenvalues and V contains the corresponding eigenvectors matrix.

$$K_t = VEV^T \tag{5.8}$$

So the equation 5.3 could be re written as,

$$\begin{cases} H = ((I - M)\tilde{V})^T K (I - L) \\ \\ HH^T \beta = 0 \end{cases} \tag{5.9}$$

And the $C-1$ Null projection directions (NPDs) as:

$$w^(i) = ((I - M)\tilde{V})^T \beta^(i); \qquad i = \{1, ..., C-1\} \tag{5.10}$$

To summarise, the projection matrix W is learned by using the kernelised version of the labelled training set. Then test set is projected onto the subspace. Next, the cosine distance between the gallery and probe is calculated to compare the gallery and the probe in the null space. Here we could also use the euclidean distance, but a conscious choice has been made to calculate the cosine similarity since the magnitude of vectors does not matter in our problem; only the similarity matters. So all the gait signatures generated for a single identity by the method in Figure 4.1 are projected in the discriminative null space as a single point that minimises the intra-class distance

and maximises the within-class distance concurrently. This process is shown in figure 5.1. The input silhouette sequences belong to two random subjects with diverse view angles for each subject.

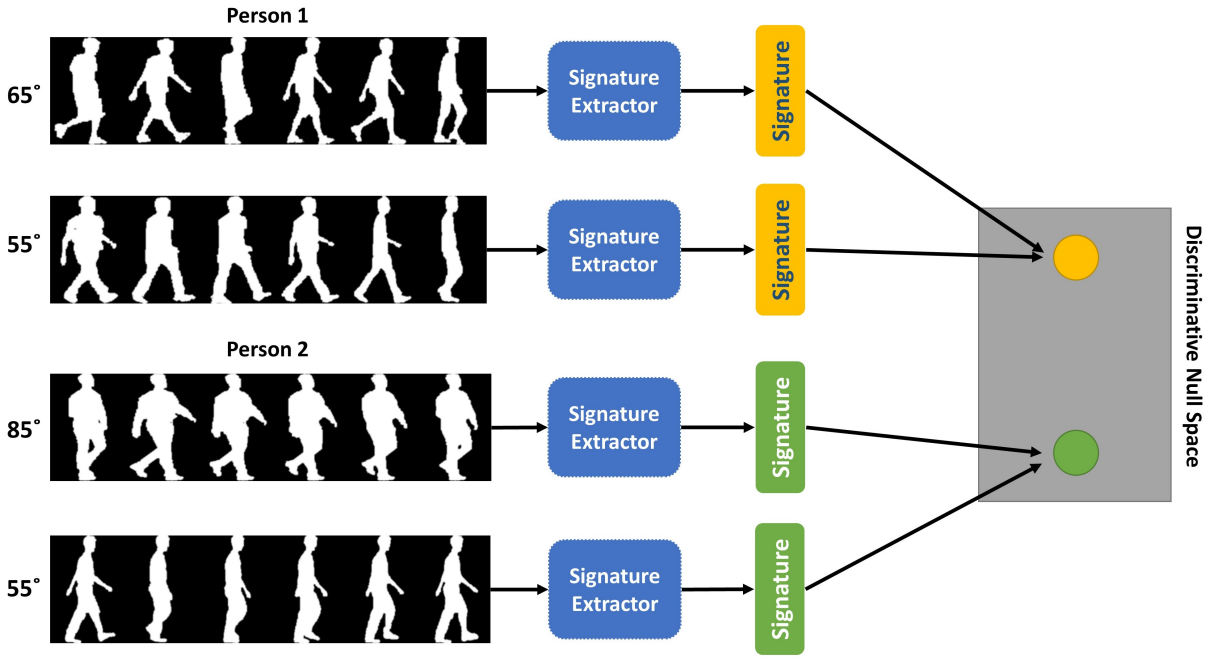


Figure 5.1: Mapping gait signature vector to a discriminative null space with consideration to the categories. The signature extractor is using the same approach as in figure 4.1

5.2 The Cross view recognition Experiment

In this section, to test our theory, we repeat the cross-view gait recognition experiment presented in the last chapter on both our benchmark datasets, CASIA-B and OULP and compare the results. Other than that, we conduct a comparative study testing the effect of our temporal attention layer and spacial attention layer on gait recognition. To do so, we evaluate our dual spacial attention mechanism and also evaluate the performance of our temporal attention mechanism by replacing it first with mean pooling, which considers all of the LSTM's outputs to create the gait signature and then max-pooling method, which considers the most prominent activation of all the LSTM's

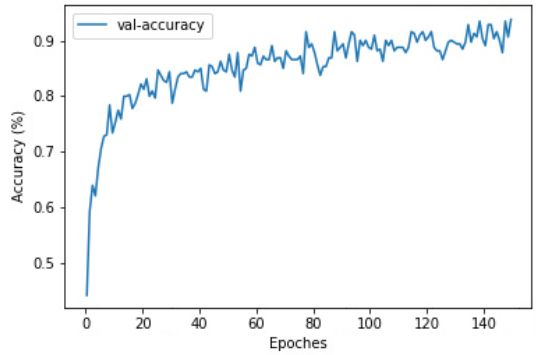
outputs as the gait signature. In the last chapter, we ran our experiment independently of the class information using Siamese architecture. This approach was taken due to the limited training data for each class. However, this chapter will take the class information under consideration using the previously mentioned method.

5.2.1 Training Procedure

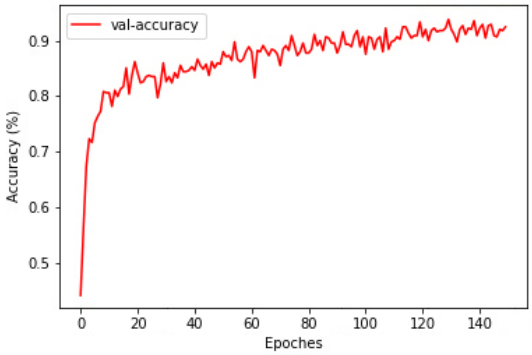
This time we resize all the input silhouette images to 176×256 to match the aspect ratio of our silhouette images on both datasets. Consequently, the spatial feature extractor will generate feature maps of size 6×8. We keep the length of the input silhouette sequences at 32 for CASIA-B, but for the OULP dataset, we changed the length of the input sequences by changing the frame rate to $\tilde{T}$ in each category. Since each sequence in OULP includes multiple views, skipping $\tilde{T}$ images allows us to create sequences with an abnormal gait cycle. using an abnormal gait cycle makes the task more challenging for our framework, but it evaluates the model's robustness for practical applications.

Modifications to the LSTM

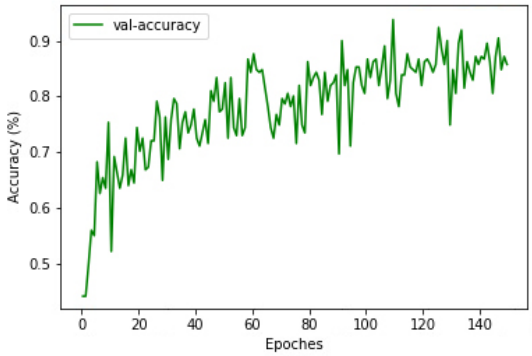
The LSTM was initially utilised with only one hidden layer, but we experimented with increasing numbers of hidden units to evaluate the performance of our gait signature generation with a more significant number of outputs from LSTM. This increase in the hidden units resulted in better validation accuracy every time. Figure 5.2 shows the increase in validation accuracy for four different hidden unit sizes on LSTM. The plots show that the maximum validation accuracy of 95.1% overfits after 2048 units and crashes to 83.7%.



(a) Range of 1024



(b) Range of 2048



(c) Range of 4096

Figure 5.2: Variations in validation accuracy by changing the number of hidden units in our LSTM after 150 Epochs. As you can see by increasing the numbers the accuracy increases between (a) and (b) but after 2048 hidden units the improvement stops. (a) maximum validation accuracy of 95.1% (b) maximum validation accuracy of 93.2% (c) over-fitting and crashing to 83.7%

Modifications to the LOSS function

We also set the margin m for the LOSS function in equation 4.10 to 1, which describes the minimum distance between two identity classes. In CASIA-B, due to the smaller number of identity classes, the number of hidden units for the LTSM component is set to 500. Next, we investigate the effect of abnormal gait cycles on the performance of our architecture in Figure 4.1. Table 5.1 compares the accuracy of our model while skipping $\tilde{T}$ images on Rank-1 and Rank-5. By analysing this table, we can see that the change in frame rate does not severely affect our model's gait recognition performance on OULP, which gives us the reason to conclude that this method is robust for abnormal

Table 5.1: The effect of skipping $\tilde{T}$ images on average of Rank-1 and Rank-5

	Rank-1 (%)					Rank-5 (%)				
Frame Rate ($\tilde{T}$)	0	1	2	3	4	0	1	2	3	4
16	96.3	97.4	96.8	96.1	95.9	97.5	97.9	97.1	97.1	96.8
32	98.01	97.01	96.10	-	-	98.9	98.3	98.3	-	-

gait cycles and can be applied to real-life scenarios. Although the results are not severely changed, we use the original frame rate setting for the rest of our experiment.

5.2.2 Experimental Results on OULP

For the OULP Scenario-1, we use all subject's silhouette sequences of the gallery set for the training. For validation in the test phase, the source utilises all the gallery sequences of all subjects, and the query uses all the probe silhouette sequences. For the second scenario's training set, we choose a random $60\%, 20\%, 20\%$ division for training, validation and testing, respectively. For the test set, probe images of each subject are taken as query and galley images for each one of the considered views are taken as the source.

The cross-view results for OULP on Rank-1 are presented in Tables 5.2 and 5.3 for scenario-1 and Scenario-2, respectively. The results in the table are compared against the results in Table 4.11 and 4.12 with the original approach. We can distinctly observe that using a discriminative null space improves the results by 3% to 4% on average.

5.2.3 Experimental Results on CASIA-B

To repeat our experiment on CASIA-B, we consider the same two scenarios from the last chapter. For Scenario-1, a $20\%, 80\%$ division is used for the subjects. The train-

ing set consists of 19, and the testing set contains 105 randomly chosen subjects. For Scenario-2, a $70\%, 30\%$ split is used in which 75 subjects are for training, and 49 subjects are for testing. The normal walk 1 to 4 are chosen as the validation set in both scenarios from the subjects selected for testing. The gait abnormality was not applied on the CASIA-B dataset due to the small sample size for each subject. The Rank-1 results for Scenario-1 and Scenario-2 are presented in Tables 5.4 to 5.7. When Considering average results on each probe view (0°, 54°, 90°, 126°), a performance improvement by a small margin of $2\% - 4\%$ is observed. This refinement is due to the use of the discriminative null space.

In individual cases where the gallery and probe angels are close together, we can see considerable improvements with intimate performances, which means many spatial and temporal features are shared between different view angles. However, the average performance margins on both datasets are not very large even though they exceed the previous work on the same two datasets. After the improvements, the best average performance on both datasets belongs to scenario-2 since, in both datasets, more training data is used in the division.

Our improved method's best average Rank-1 result for all the probe viewpoints on scenario-2 reaches 97.07% on OULP and 74.47% on CASIA-B, respectively. Moreover, on the more challenging Scenario-1, it reaches 96.2% for OULP and 63.02% for CASIA-B. These results show great promise on scenario-2, but they conclude that our approach is still not strong enough for practical applications. Therefore, we conduct an element removal (Ablation) study in the next section to better understand and improve our model.

Table 5.2: Cross view comparison between our Improved method and the state of the art on Rank-1 Scenario-1 on OULP dataset

Scenario - 1						
Methods	**Probe Angle**	**55°**	**65°**	**75°**	**85°**	**AVG**
[100]	55°	-	65.76	32.92	19.48	39.39
[3] PDVS	55°	-	76.20	61.45	45.50	61.05
[3] AVTM	55°	-	77.72	64.54	42.69	61.65
Original Method	55°	**-**	**96.4**	**94**	**86.3**	**92.23**
Improved Method	55°	**-**	**98.9**	**96.4**	**91.2**	**95.5**
[100]	65°	72.58	-	78.54	51.83	67.65
[3] PDVS	65°	75.99	-	77.09	65.48	72.85
[3] AVTM	65°	75.63	-	76.36	62.76	71.58
Original Method	65°	**95**	**-**	**96.6**	**93**	**94.86**
Improved Method	65°	**97.6**	**-**	**98.6**	**95.1**	**97.1**
[100]	75°	39.13	78.30	-	81.22	66.22
[3] PDVS	75°	60.25	76.20	-	76.52	70.99
[3] AVTM	75°	59.88	74.90	-	76.31	70.36
Original Method	75°	**93.6**	**95.8**	**-**	**92**	**93.8**
Improved Method	75°	**96.3**	**97.5**		**97.1**	**96.9**
[100]	85°	19.45	44.93	71.39	-	45.26
[3] PDVS	85°	40.48	60.62	73.12	-	58.07
[3] AVTM	85°	40.17	61.87	74.32	-	58.79
Original Method	85°	**87.2**	**93.7**	**94.5**	**-**	**91.8**
Improved Method	85°	**93.1**	**95.5**	**97.6**	**-**	**95.4**

Table 5.3: cross-view comparison between our improved method and the state of the art on Rank-1 Scenario-2 on OULP dataset

Scenario - 2						
Methods	**Probe Angle**	**55°**	**65°**	**75°**	**85°**	**AVG**
[2]	55°	-	98.30	96.00	80.50	91.60
Our Method	**55°**	**-**	**96.8**	**94.05**	**91.9**	**94.25**
Improved Method	**55°**	**-**	**98.9**	**96.5**	**94.5**	**96.6**
[2]	65°	96.30	-	97.30	83.30	92.30
Our Method	**65°**	**97.3**	**-**	**97**	**96.8**	**97.03**
Improved Method	**65°**	**98.9**	**-**	**97.8**	**97.9**	**98.2**
[2]	75°	94.20	97.80	-	85.10	92.40
Our Method	**75°**	**94.7**	**97.83**	**-**	**97.1**	**96.54**
Improved Method	**75°**	**95.9**	**99.1**	**-**	**98.3**	**97.7**
[2]	85°	90	96	98.40	-	94.80
Our Method	**85°**	**85.5**	**95.98**	**97.3**	**-**	**92.93**
Improved Method	**85°**	**92.1**	**97.9**	**97.5**	**-**	**95.8**

Table 5.4: Comparison between our method, our improved method and the state of the art with 0°probe on Rank-1 Scenario-1 and Scenario-2 on CASIA-B dataset

Scenario-1 *0°												
Methods	0	18	36	54	72	90	108	126	144	162	180	AVG
[239]	-	45	34	20	8	5	5	14	18	25	40	21.4
[242]	-	85	47	26	25	28	25	27	37	68	95	46.30
[237]	-	-	-	-	-	-	-	-	-	-	-	23.34
Our Method	**-**	**93**	**64**	**32.5**	**29.5**	**27.9**	**20.2**	**30.1**	**52.5**	**70.5**	**94.9**	**51.51**
Improved Method	**-**	**93.9**	**64.5**	**35.9**	**31.5**	**28.6**	**30.2**	**36.2**	**55**	**74.2**	**95**	**54.5**
Scenario-2 *0°												
[103]	-	87.10	58.6	39.52	28.23	33.87	31.45	37.90	46.77	62.10	67.74	49.27
Our Method	**-**	**99**	**85**	**67**	**43**	**35**	**40**	**55**	**73**	**87**	**91.5**	**67.55**
Improved Method	**-**	**99.8**	**85.9**	**68**	**50.1**	**37.3**	**42.05**	**56.7**	**74.9**	**87.6**	**93.9**	**69.6**

Table 5.5: Comparison between our method, our improved method and the state of the art with 54°probe on Rank-1 Scenario-1 and Scenario-2 on CASIA-B dataset

Scenario-1 *54°												
Methods	0	18	36	54	72	90	108	126	144	162	180	AVG
[242]	24	65	97	-	95	63	53	48	34	23	22	52.40
[238]	28	67	98	-	97	86	76	69	58	31	20	63
[237]	-	-	-	-	-	-	-	-	-	-	-	30.37
[239]	-	-	-	-	-	-	-	-	-	-	-	33.50
Our Method	**34.2**	**65.5**	**91.3**	**-**	**92**	**85.1**	**76**	**63.1**	**59**	**50.1**	**32.5**	**64.88**
Improved Method	**36.1**	**70.3**	**93.9**	**-**	**95.9**	**85.9**	**78.3**	**70.2**	**65.9**	**50.9**	**36.9**	**68.43**
Scenario-2 *54°												
[103]	50	63.71	83.87	-	89.52	82.26	72.58	65.32	57.26	43.55	28.23	63.63
Our Method	**65**	**82**	**99**	**-**	**98**	**83**	**90.1**	**85**	**81**	**69.9**	**59**	**81.2**
Improved Method	**68**	**82.9**	**99.8**	**-**	**98.5**	**85.2**	**92.2**	**91**	**83**	**71.9**	**60.9**	**83.34**

Table 5.6: Comparison between our method, our improved method and the state of the art with 90°probe on Rank-1 Scenario-1 and Scenario-2 on CASIA-B dataset

Scenario-1 *90°												
Methods	0	18	36	54	72	90	108	126	144	162	180	AVG
[242]	18	24	41	66	96	-	95	68	41	21	13	48.30
[238]	17	26	54	84	98	-	98	84	50	25	14	55
[237]	-	-	-	-	-	-	-	-	-	-	-	30.87
[239]	-	-	-	-	-	-	-	-	-	-	-	34.80
Our Method	**31.1**	**33**	**60.1**	**81.9**	**99**	**-**	**93.9**	**85.6**	**54.6**	**30**	**28**	**59.72**
Improved Method	**33.2**	**33.9**	**62**	**82.8**	**99.6**	**-**	**95.1**	**86.3**	**55.1**	**32.1**	**28.03**	**60.8**
Scenario-2 *90°												
[103]	32.26	35.48	52.42	70.16	95.16	-	95.16	80.65	56.45	33.87	29.03	58.06
[48]	-	-	-	-	-	-	-	-	-	-	-	60.40
[236]	-	-	-	-	-	-	-	-	-	-	-	63.70
Our Method	**34**	**45**	**63**	**78.6**	**94.3**	**-**	**96.8**	**80.9**	**67**	**50.1**	**40**	**64.97**
Improved Method	**34.9**	**46.9**	**65.7**	**80**	**96.4**	**-**	**99.1**	**83.9**	**68.1**	**50.9**	**42**	**66.79**

Table 5.7: Comparison between our method, our improved method and the state of the art with 126°probe on Rank-1 Scenario-1 and Scenario-2 on CASIA-B dataset

Scenario-1 *126°												
Methods	0	18	36	54	72	90	108	126	144	162	180	AVG
[242]	25	29	35	49	60	78	98	-	98	75	22	56.90
[238]	18	33	58	68	78	83	98	-	95	63	27	62.10
[237]	-	-	-	-	-	-	-	-	-	-	-	34.40
[239]	-	-	-	-	-	-	-	-	-	-	-	37.10
Our Method	**33**	**39.1**	**60.5**	**71**	**72**	**84.5**	**90.9**	**-**	**95**	**73**	**46**	**66.5**
Improved Method	**33.9**	**41.6**	**62.3**	**71.9**	**76**	**87**	**93.1**	**-**	**96.1**	**75.1**	**46.7**	**68.37**
Scenario-2 *126°												
[103]	33.06	45.16	60.48	72.58	84.68	86.29	93.55	-	94.35	59.68	35.48	66.53
[48]	-	-	-	-	-	-	-	-	-	-	-	65
[236]	-	-	-	-	-	-	-	-	-	-	-	74.80
Our Method	**58.9**	**60.2**	**73**	**83**	**85.9**	**80.5**	**94.9**	**-**	**93**	**79**	**50.6**	**75.9**
Improved Method	**59.5**	**60.9**	**75.3**	**88.6**	**85.9**	**84.8**	**95.7**	**-**	**95.1**	**79.7**	**56**	**78.15**

5.3 The element Removal experiment

The Spatial Attention Layer

In this section, we evaluate the effect of each individual spatial-temporal element in our architecture. Our approach presented in figure 4.1 uses a spatial-temporal attention mechanism to learn gait information from a sequence of silhouette images. As mentioned in chapter 4, the Spatial attention mechanism shown in figure 4.2 consists of two parts. The upper part is a typical CNN network that pays attention to the small-scale semantic regions, and the lower part utilises the bottleneck architecture, which results in a larger receptive field to pay attention to bigger semantic regions. These two elements are fused to create a spatial feature vector for our temporal layer. In the rest of the thesis, to better present the results, we refer to the upper part and lower part of our spatial attention as ”UPSA” and ”LPSA”, respectively.

Figure 5.3: Sample cross matching sequence pairs of the same subject from our approach from different view angles

The Temporal Attention Layer

The temporal Attention Layer receives spatial feature vectors from the spatial Attention Layer and feeds them into its LSTM blocks to model periodic motion. The modelled motion vector is then fed to a temporal attention element, hereafter referred to as "TAC", which decreases the motion vectors by calculating an attention weight for each particular vector from the LSTM. It organises all of the T LSTM outputs into a $T \times N$ matrix M where each row is the output of the LSTM at timestep t feeds it into our temporal attention mechanism. To evaluate the effect of TAC, we replaced it with a simple pooling operation. Both average pooling and max pooling were considered for this experiment. In max-pooling, outputs of the LSTM are considered, and the one with the maximum activation value is chosen as the gait signature. It will give us a small degree of temporal attention but Not so elaborate as the TAC. However, in average pooling, no temporal attention is considered since all the outputs of the LSTM are equally significant by nature.

We conduct our experiment on the CASIA-B Scenario-2 since it provides us with a

ht

Table 5.8: Rank-1 performance comparison of our method using mixture of elements on CASIA-B dataset scenario-2. UPSA: Upper spatial attention, LPSA: Lower Spacial Attention, TAC: Temporal Attention element, AVP: Average pooling, MAXP: Max pooling

Methods	Probe	0	18	36	54	72	90	108	126	144	162	180	AVG
UPSA-AVP	0	-	90.8	75	57	40.3	30	30.1	46.1	72.1	84.1	89.2	61.47
UPSA-MAXP		-	89.1	74.1	57	40	29.1	28.9	42	70.5	83.6	88	60.23
UPSA-TAC		-	91.5	75.9	60.4	46	35	37.1	48	72	85	90.9	64.18
LPSA-TAC		-	90	83.9	61.9	48.4	32	41.8	55.2	73.1	86.1	90	**66.24**
LPSA-UPSA-TAC		-	99.8	85.9	68	50.1	37.3	42.05	56.7	74.9	87.6	93.9	**69.6**
UPSA-AVP	54	53.9	71.6	97.8	-	91.6	80.1	81.9	80.1	78	63.9	45.6	74.45
UPSA-MAXP		54	73	96.7	-	90	80.2	82.1	83.6	77.2	64	50.1	75.09
UPSA-TAC		61	78.6	98	-	95.9	80.9	82.8	83	78.9	64.8	50.6	77.45
LPSA-TAC		60	75.9	98	-	92.8	82	88.6	90	81.9	69.7	49.3	**78.82**
LPSA-UPSA-TAC		68	82.9	99.8	-	98.5	85.2	92.2	91	83	71.9	60.9	**83.34**
UPSA-AVP	90	30.9	37.9	57.7	78	92.5	-	95	75.1	57.9	42.1	24.7	59.18
UPSA-MAXP		26	26.9	58	76.6	92	-	95	76.5	56.1	39.9	24	57.1
UPSA-TAC		31.8	45	60	78.5	94	-	96.5	81.6	59.6	43.5	29.1	61.96
LPSA-TAC		30	40.4	63.6	79	92.6	-	96.6	80.5	59	38	25.6	**60.53**
LPSA-UPSA-TAC		34.9	46.9	65.7	80	96.4	-	99.1	83.9	68.1	50.9	42	**66.79**
UPSA-AVP	126	53	45.9	70.7	85	81	78.8	92.1	-	92.3	67.8	53.1	73.92
UPSA-MAXP		46	51.3	71.5	86.5	84.9	77.1	94	-	92.3	70.1	40.3	71.4
UPSA-TAC		56.9	57.2	72	88	84.9	79.9	93.9	-	92.8	72.1	53.8	73.92
LPSA-TAC		53	54	74.2	84	79.4	83.8	93.9	-	93.5	76.3	47.1	**73.92**
LPSA-UPSA-TAC		59.5	60.9	75.3	88.6	85.9	84.8	95.7	-	95.1	79.7	56	**78.15**

better division and more training data. It is chosen to benefit from max and average pooling, which will obviously have less than optimal results using Scenario-1. Figure 5.3 shows a pair of sequences from the same subject at different angles in the CASIA-B dataset.

We chose 0° and 108° angles to show the apparent dissimilarity in spatial features. Looking at the figure, we can intuitively deduce that temporal cues play an essential part in matching these two sequences and using UPSA or LPSA along with a max or average pooling will not produce surpassing results. So, to evaluate the effects of UPSA on gait recognition performance, we first trained our model using only

UPSA as spatial attention and average pooling instead of temporal attention. Next, we conducted the training on UPSA and max-pooling, and finally, the experiment was repeated using UPSA and TAC. The effects of LPSA were evaluated the same way using the average pooling, max pooling and TAC instead of the temporal attention mechanism. Finally, we train the model with all the original elements.

5.3.1 Results of the cross-view experiment

Results of our cross-view experiment for $0°, 54°, 90°$ and $126°$ view angles as probe and all the other view angles as the gallery are presented in table 5.8. These results firmly support our theory about the importance of attention to temporal cues for $0°$ as probe and $108°$ as gallery matching mentioned above. Besides, our improved method provides better results on these angles by almost $\%12$, a significant margin. In closer view, angles to $0°$ such as $18°$ and $180°$ better appearance features are extracted, which along with the temporal element achieves 99.8%, 93.9% accuracy, respectively.

Moreover, average pooling and the UPSA produce an average result of 67.2% which exceeds the max pooling and UPSA average at 66% by 1.2%. Adding the temporal element at any viewpoint increases the average results by 3% to the height of 69.3%. Using the LPSA with the temporal element increases the average result by 69.9%, but none of these can compete with our improved model at 74.47% Rank-1 accuracy for cross-view Person Re-Id. Our improved method will overtake the best results by 4.5%, an excellent margin for cross-view gait recognition on the challenging CASIA-B dataset. Because the use of temporal element significantly increases the performance in all the cases, a conclusion could be made that TAC is robust for use in other architectures for paying attention to the most crucial motion information. Moreover, using

the temporal element with LPSA produces slightly better results than UPSA, and since the lower part of our spatial attention mechanism has a bigger receptive field due to its bottleneck architecture, it means that the large scale semantic regions are getting more attention from TAC.

5.4 Chapter Summary

This chapter improved our spatial-temporal attention network to create a robust approach to reduce the effects of viewpoint variations in gait Person Re-Id. Essentially we projected our gait signatures belonging to the same identity class to a single point in a learned discriminative null space to improve the accuracy of our model. We combined NFST with the Siamese architecture to reduce the effect of view angles on the cross-matching and achieve outstanding performance on OULP and CASIA-B datasets. To test our theory, in this section, we repeat the cross-view gait recognition experiment presented in the last chapter on both datasets and compare the results.

We also experimented with our LSTM block sizes to test the robustness of our improved approach against abnormal gait cycles by changing the frame rate and skipping images from the input gait silhouette sequences. Moreover, we conducted an element removal study on our gait signature extractor by removing and replacing the elements of our spatial and temporal attention mechanism to test the effect of each element on the Rank-1 gait recognition.

In the next chapter, we present the conclusion to this thesis and discuss what could be done next to reduce the effects of cross-view variation on Gait Person Re-Id.

Chapter 6

Conclusion and future work

This chapter summarises the work of the thesis and gives directions for future studies.

6.1 Conclusion

There are many different kinds of techniques to tackle the Gait Person re-identification problem. This thesis focused on the most effective methods for gait person re-id extraction to minimise inter-class and intra-class variations. This thesis mainly focused on view variations and searched for ways to improve and automate feature extraction and identity cross-matching. Multiple frameworks with a focus on convolutional neural networks were proposed and tested extensively throughout the thesis. Our attempts started with a comparative study on the effects of different modality inputs for feature extraction using popular convolutional neural networks. Several data modalities were considered to generate a single gait signature from input sequences in a video clip for gait recognition. Grayscale images, silhouette sequences, and optical flow maps were used as inputs for 2D and 3D convolutional neural networks and ResNet architectures. Finally, a new two-stream spatial-temporal architecture containing a spatial and

a temporal stream was proposed and thoroughly tested against appearance, clothing and viewpoint variations. This approach tried to preserve the temporal motion cues for as long as possible for use in gait signature extraction by manipulating low-level data blocks to achieve state of the art. Our low-level spatial-temporal technique performs exceptionally on low-resolution images since it used multiple modality fusion, but with limited training data, it will become degraded. This limitation was due to the nature of convolutional neural networks, which need an abundance of training data. The ResNet architecture proposed in this thesis solved this problem, but we concluded that the computational cost of the method is very high for practical applications. The use of multiple fusion techniques has also been tried out, which led us to conclude that fusion of more modalities will improve the performance of deep learning algorithms on the gait recognition task. This method produced outstanding results on the TUM-GAID and CASIA-B dataset, but clothing, carry and view variations degraded the recognition accuracy. View variations had the worst effect on our spatial-temporal technique. The experimental results gave us great insight into techniques such as the incremental technique, which decreased the number of parameters significantly while training, to improve the accuracy of deep learning models.

Since our proposed ResNet produced such excellent results on the Silhouette modality, we used it to extract spatial features from silhouette sequences for our following recommended approach. This modern method focused on reducing the effects of viewpoint variations in silhouette sequences by concentrating on salient discriminative areas of human gait. Our second spatial-temporal gait signature extractor used two attentional layers. Unlike the previous method, we did not try to preserve temporal information by manipulating CNN. Instead, we just used our ResNet to extract spatial

features and feed them to our two-stream spatial attention subnetwork, which focused on the essential salient areas of each frame and produced feature vectors fed to a long short-term memory sub-network. The LSTM, which is part of the temporal layer, was used to encode the periodic motion for each timestep of the silhouette sequence. Finally, our proposed temporal attention mechanism eliminated the redundant timesteps to create our spatial-temporal gait signature. This method was then improved by mapping the gait signatures to a null space to make cross-matching easier for our Siamese architecture. Our second approach was tested extensively against viewpoint variations and abnormal gait cycles, which shows excellent robustness for real-life applications in which a complete gait cycle or camera angles are not guaranteed. Both our attention mechanisms were tested separately for cross-view variation. The results revealed that using our temporal attention element significantly improves the gait recognition performance, exhibiting great promise for future work.

6.2 Future work

For future work, we intend to focus on spatial and temporal attention and test our method on a more challenging cross-view dataset to introduce more abnormality to the gait cycle and view variations. Also, we intend to test our attentional approach against other variations such as occlusion clothing changes, carry bags and walking speed. We believe that this robust approach will produce great results despite these challenges.

As the second line of study, we will focus on improving the elements of our attentional approach. Since each component of the architecture is independent of the others, they could be replaced relatively easy to test the effect of other state of the art

attention mechanisms.

As the third line of study, we intend to combine our two approaches by using other modalities such as optical flow maps as the input for our spatial-temporal attention mechanism.

Finally, since one of the most critical limitations of both our approaches was the lack of enough training data, we will look into Generative Adversarial Networks (GANs) for image generation task as a method to support our attention mechanism.

Bibliography

[1] Tino Stöckel, Robert Jacksteit, Martin Behrens, Ralf Skripitz, Rainer Bader, and Anett Mau-Moeller. The mental representation of the human gait in young and older adults. *Frontiers in psychology*, 6:943, 2015.

[2] Jinguang Han and Bir Bhanu. Individual recognition using gait energy image. *IEEE transactions on pattern analysis and machine intelligence*, 28(2):316–322, 2005.

[3] Toby HW Lam, King Hong Cheung, and James NK Liu. Gait flow image: A silhouette-based gait representation for human identification. *Pattern recognition*, 44(4):973–987, 2011.

[4] Junxi Feng, Xiaohai He, Qizhi Teng, Chao Ren, Honggang Chen, and Yang Li. Reconstruction of porous media from extremely limited information using conditional generative adversarial networks. *Physical Review E*, 100(3):033308, 2019.

[5] *course notes: idempotent productions*. Accessed: 2020-09-30.

[6] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.

[7] *What is the VGG neural network?* Accessed: 2020-11-25.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Afshine Amidi. *Recurrent Neural Networks cheatsheet*, Accessed: 2020-11-25.

[10] Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

[11] Mohammad Ali Saghafi, Aini Hussain, Halimah Badioze Zaman, and Mohamad Hanif Md Saad. Review of person re-identification techniques. *IET Computer Vision*, 8(6):455–474, 2014.

[12] Apurva Bedagkar-Gala and Shishir K Shah. A survey of approaches and trends in person re-identification. *Image and vision computing*, 32(4):270–286, 2014.

[13] Bahram Lavi, Ihsan Ullah, Mehdi Fatan, and Anderson Rocha. Survey on reliable deep learning-based person re-identification models: Are we there yet? *arXiv preprint arXiv:2005.00355*, 2020.

[14] Yann LeCun, Koray Kavukcuoglu, and Clément Farabet. Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE international symposium on circuits and systems*, pages 253–256. IEEE, 2010.

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[17] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[18] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009.

[19] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[20] Jianan Li, Xiaodan Liang, ShengMei Shen, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. Scale-aware fast r-cnn for pedestrian detection. *IEEE transactions on Multimedia*, 20(4):985–996, 2017.

[21] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[23] Afshin Dehghan, Shayan Modiri Assari, and Mubarak Shah. Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object

tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4091–4099, 2015.

[24] Jeany Son, Mooyeol Baek, Minsu Cho, and Bohyung Han. Multi-object tracking with quadruplet convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5620–5629, 2017.

[25] Ji Zhu, Hua Yang, Nian Liu, Minyoung Kim, Wenjun Zhang, and Ming-Hsuan Yang. Online multi-object tracking with dual matching attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 366–382, 2018.

[26] Jiarui Xu, Yue Cao, Zheng Zhang, and Han Hu. Spatial-temporal relation networks for multi-object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3988–3998, 2019.

[27] Quan Zhang, Haijie Cheng, Jianhuang Lai, and Xiaohua Xie. Dhml: Deep heterogeneous metric learning for vis-nir person re-identification. In *Chinese Conference on Biometric Recognition*, pages 455–465. Springer, 2019.

[28] Massimo Martini, Marina Paolanti, and Emanuele Frontoni. Open-world person re-identification with rgbd camera in top-view configuration for retail applications. *IEEE Access*, 8:67756–67765, 2020.

[29] Daniele Liciotti, Marina Paolanti, Emanuele Frontoni, Adriano Mancini, and Primo Zingaretti. Person re-identification dataset with rgb-d camera in a top-view configuration. In *Video Analytics. Face and Facial Expression Recognition and Audience Measurement*, pages 1–11. Springer, 2016.

[30] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[31] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):1–14, 2017.

[32] Liming Zhao, Xi Li, Yueting Zhuang, and Jingdong Wang. Deeply-learned part-aligned representations for person re-identification. In *Proceedings of the IEEE international conference on computer vision*, pages 3219–3228, 2017.

[33] Martin Hirzer, Csaba Beleznai, Peter M Roth, and Horst Bischof. Person re-identification by descriptive and discriminative classification. In *Scandinavian conference on Image analysis*, pages 91–102. Springer, 2011.

[34] Martin Koestinger, Martin Hirzer, Paul Wohlhart, Peter M Roth, and Horst Bischof. Large scale metric learning from equivalence constraints. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2288–2295. IEEE, 2012.

[35] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li. Person re-identification by local maximal occurrence representation and metric learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2197–2206, 2015.

[36] Yifan Sun, Liang Zheng, Yali Li, Yi Yang, Qi Tian, and Shengjin Wang. Learning part-based convolutional features for person re-identification. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[37] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294, 2018.

[38] Dapeng Chen, Hongsheng Li, Tong Xiao, Shuai Yi, and Xiaogang Wang. Video person re-identification with competitive snippet-similarity aggregation and co-attentive snippet embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1169–1178, 2018.

[39] Taiqing Wang, Shaogang Gong, Xiatian Zhu, and Shengjin Wang. Person re-identification by discriminative selection in video ranking. *IEEE transactions on pattern analysis and machine intelligence*, 38(12):2501–2514, 2016.

[40] Niall McLaughlin, Jesus Martinez Del Rincon, and Paul Miller. Recurrent convolutional network for video-based person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1325–1334, 2016.

[41] Xiaoke Zhu, Xiao-Yuan Jing, Xinge You, Xinyu Zhang, and Taiping Zhang. Video-based person re-identification by simultaneously learning intra-video and inter-video distance metrics. *IEEE Transactions on Image Processing*, 27(11):5683–5695, 2018.

[42] Liang Zheng, Zhi Bie, Yifan Sun, Jingdong Wang, Chi Su, Shengjin Wang, and Qi Tian. Mars: A video benchmark for large-scale person re-identification. In *European Conference on Computer Vision*, pages 868–884. Springer, 2016.

[43] Kan Liu, Bingpeng Ma, Wei Zhang, and Rui Huang. A spatio-temporal appearance representation for viceo-based pedestrian re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3810–3818, 2015.

[44] Jinjie You, Ancong Wu, Xiang Li, and Wei-Shi Zheng. Top-push video-based person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1345–1353, 2016.

[45] Arulkumar Subramaniam, Athira Nambiar, and Anurag Mittal. Co-segmentation inspired attention networks for video-based person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 562–572, 2019.

[46] Martin Hirzer, Peter M Roth, Martin Köstinger, and Horst Bischof. Relaxed pairwise learned metric for person re-identification. In *European conference on computer vision*, pages 780–793. Springer, 2012.

[47] Shangxuan Wu, Ying-Cong Chen, Xiang Li, An-Cong Wu, Jin-Jie You, and Wei-Shi Zheng. An enhanced deep feature representation for person re-identification. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–8. IEEE, 2016.

[48] Fei Xiong, Mengran Gou, Octavia Camps, and Mario Sznaier. Person re-identification using kernel-based metric learning methods. In *European conference on computer vision*, pages 1–16. Springer, 2014.

[49] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Towards open-world person re-identification by one-shot group-based verification. *IEEE transactions on pattern analysis and machine intelligence*, 38(3):591–606, 2015.

[50] Douglas Gray and Hai Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European conference on computer vision*, pages 262–275. Springer, 2008.

[51] Cheng-Hao Kuo, Sameh Khamis, and Vinay Shet. Person re-identification using semantic color names and rankboost. In *2013 IEEE workshop on applications of computer vision (WACV)*, pages 281–287. IEEE, 2013.

[52] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *Proceedings of the iEEE conference on computer vision and pattern recognition*, pages 1335–1344, 2016.

[53] Tong Xiao, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1249–1258, 2016.

[54] Sateesh Pedagadi, James Orwell, Sergio Velastin, and Boghos Boghossian. Local fisher discriminant analysis for pedestrian re-identification. In *Proceedings of*

the IEEE conference on computer vision and pattern recognition, pages 3318–3325, 2013.

[55] Masashi Sugiyama. Local fisher discriminant analysis for supervised dimensionality reduction. In *Proceedings of the 23rd international conference on Machine learning*, pages 905–912, 2006.

[56] Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(2), 2009.

[57] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a discriminative null space for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1239–1248, 2016.

[58] Elly Cosgrove. One billion surveillance cameras will be watching around the world in 2021, a new study says, Dec 2019.

[59] BBC.co.uk. Cctv: Too many cameras useless, warns surveillance watchdog tony porter, Jan 2015.

[60] Dung Nghi Truong Cong, Catherine Achard, Louahdi Khoudour, and Lounis Douadi. Video sequences association for people re-identification across multiple non-overlapping cameras. In *International Conference on Image Analysis and Processing*, pages 179–189. Springer, 2009.

[61] Radhouane Guermazi, Mohamed Hammami, and Abdelmajid Ben Hamadou. Violent web images classification based on mpeg7 color descriptors. In *2009*

IEEE International Conference on Systems, Man and Cybernetics, pages 3106–3111. IEEE, 2009.

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

[63] Michael J Lyons, Julien Budynek, and Shigeru Akamatsu. Automatic classification of single facial images. *IEEE transactions on pattern analysis and machine intelligence*, 21(12):1357–1362, 1999.

[64] Laurence T Maloney and Brian A Wandell. Color constancy: a method for recovering surface spectral reflectance. *JOSA A*, 3(1):29–33, 1986.

[65] Xuelin Qian, Wenxuan Wang, Li Zhang, Fangrui Zhu, Yanwei Fu, Tao Xiang, Yu-Gang Jiang, and Xiangyang Xue. Long-term cloth-changing person re-identification. *arXiv preprint arXiv:2005.12633*, 2020.

[66] Martin Hirzer, Csaba Beleznai, Peter M. Roth, and Horst Bischof. Person Re-Identification by Descriptive and Discriminative Classification. In *Proc. Scandinavian Conference on Image Analysis (SCIA)*, 2011.

[67] Minxian Li, Xiatian Zhu, and Shaogang Gong. Unsupervised person re-identification by deep learning tracklet association. In *Proceedings of the European conference on computer vision (ECCV)*, pages 737–753, 2018.

[68] Zhedong Zheng, Liang Zheng, and Yi Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3754–3762, 2017.

[69] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[70] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 79–88, 2018.

[71] Chengqiu Dai, Cheng Peng, and Min Chen. Selective transfer cycle gan for unsupervised person re-identification. *Multimedia Tools and Applications*, pages 1–17, 2020.

[72] Xuelin Qian, Yanwei Fu, Tao Xiang, Wenxuan Wang, Jie Qiu, Yang Wu, Yu-Gang Jiang, and Xiangyang Xue. Pose-normalized image generation for person re-identification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 650–667, 2018.

[73] Hehe Fan, Liang Zheng, Chenggang Yan, and Yi Yang. Unsupervised person re-identification: Clustering and fine-tuning. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(4):1–18, 2018.

[74] Yutian Lin, Xuanyi Dong, Liang Zheng, Yan Yan, and Yi Yang. A bottom-up clustering approach to unsupervised person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8738–8745, 2019.

[75] Yuhang Ding, Hehe Fan, Mingliang Xu, and Yi Yang. Adaptive exploration for unsupervised person re-identification. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(1):1–19, 2020.

[76] Federico Pala, Riccardo Satta, Giorgio Fumera, and Fabio Roli. Multimodal person reidentification using rgb-d cameras. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(4):788–799, 2015.

[77] Vladimir V Kniaz, Vladimir A Knyaz, Jirí Hladuvka, Walter G Kropatsch, and Vladimir Mizginov. Thermalgan: Multimodal color-to-thermal image translation for person re-identification in multispectral dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[78] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Correction to: Neural architecture search. In *Automated Machine Learning*, pages C1–C1. Springer, 2019.

[79] Kaiyang Zhou, Xiatian Zhu, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Learning generalisable omni-scale representations for person re-identification. *arXiv preprint arXiv:1910.06827*, 2019.

[80] Arun Ross, Jidnya Shah, and Anil K Jain. From template to image: Reconstructing fingerprints from minutiae points. *IEEE transactions on pattern analysis and machine intelligence*, 29(4):544–560, 2007.

[81] Antitza Dantcheva, Jean-Luc Dugelay, and Petros Elia. Soft biometrics systems: Reliability and asymptotic bounds. In *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–6. IEEE, 2010.

[82] Mark S Nixon, Tieniu Tan, and Rama Chellappa. *Human identification based on gait*, volume 4. Springer Science & Business Media, 2010.

[83] Charmaine Cordina. Face blindness. *MMSA*, 2020.

[84] Sarah Bate and Rachel J Bennetts. The rehabilitation of face recognition impairments: a critical review and future directions. *Frontiers in Human Neuroscience*, 8:491, 2014.

[85] Joseph M DeGutis, Christopher Chiu, Mallory E Grosso, and Sarah Cohan. Face processing improvements in prosopagnosia: successes and failures over the last 50 years. *Frontiers in human neuroscience*, 8:561, 2014.

[86] Lazzaro di Biase, Alessandro Di Santo, Maria Letizia Caminiti, Alfredo De Liso, Syed Ahmar Shah, Lorenzo Ricci, and Vincenzo Di Lazzaro. Gait analysis in parkinson's disease: An overview of the most accurate markers for diagnosis and symptoms monitoring. *Sensors*, 20(12):3529, 2020.

[87] Hank White and Samuel Augsburger. Gait evaluation for patients with cerebral palsy. *Orthopedic Care of Patients with Cerebral Palsy*, pages 51–76, 2020.

[88] Mengxuan Li, Shanshan Tian, Linlin Sun, and Xi Chen. Gait analysis for post-stroke hemiparetic patient by multi-features fusion method. *Sensors*, 19(7):1737, 2019.

[89] Michael W Whittle. Clinical gait analysis: A review. *Human movement science*, 15(3):369–387, 1996.

[90] Wiebren Zijlstra and At L Hof. Assessment of spatio-temporal gait parameters from trunk accelerations during human walking. *Gait & posture*, 18(2):1–10, 2003.

[91] Aditi Roy, Shamik Sural, and Jayanta Mukherjee. A hierarchical method combining gait and phase of motion with spatiotemporal model for person re-identification. *Pattern Recognition Letters*, 33(14):1891–1901, 2012.

[92] Gunawan Ariyanto and Mark S Nixon. Model-based 3d gait biometrics. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–7. IEEE, 2011.

[93] Michela Goffredo, Imed Bouchrika, John N Carter, and Mark S Nixon. Self-calibrating view-invariant gait biometrics. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(4):997–1008, 2009.

[94] Stephen Lombardi, Ko Nishino, Yasushi Makihara, and Yasushi Yagi. Two-point gait: Decoupling gait from body shape. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1041–1048, 2013.

[95] Haiping Lu, Konstantinos N Plataniotis, and Anastasios N Venetsanopoulos. A layered deformable model for gait analysis. In *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 249–254. IEEE, 2006.

[96] Nirattaya Khamsemanan, Cholwich Nattee, and Nitchan Jianwattanapaisarn. Human identification from freestyle walks using posture-based gait feature. *IEEE Transactions on Information Forensics and Security*, 13(1):119–128, 2017.

[97] Imed Bouchrika, John N Carter, and Mark S Nixon. Towards automated visual surveillance using gait for identity recognition and tracking across multiple non-intersecting cameras. *Multimedia Tools and Applications*, 75(2):1201–1221, 2016.

[98] Michal Balazia and Konstantinos N Plataniotis. Human gait recognition from motion capture data in signature poses. *IET Biometrics*, 6(2):129–137, 2017.

[99] Xianye Ben, Peng Zhang, Weixiao Meng, Rui Yan, Mingqiang Yang, Wenhe Liu, and Hui Zhang. On the distance metric learning between cross-domain gaits. *Neurocomputing*, 208:153–164, 2016.

[100] Cheng Zhang, Wu Liu, Huadong Ma, and Huiyuan Fu. Siamese neural network based gait recognition for human identification. In *2016 ieee international conference on acoustics, speech and signal processing (ICASSP)*, pages 2832–2836. IEEE, 2016.

[101] Huadong Ma and Wu Liu. A progressive search paradigm for the internet of things. *IEEE MultiMedia*, 25(1):76–86, 2017.

[102] Manuel Montero-Odasso, Marcelo Schapira, Enrique R Soriano, Miguel Varela, Roberto Kaplan, Luis A Camera, and L Marcelo Mayorga. Gait velocity as a single predictor of adverse events in healthy seniors aged 75 years and older. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 60(10):1304–1309, 2005.

[103] Zifeng Wu, Yongzhen Huang, Liang Wang, Xiaogang Wang, and Tieniu Tan. 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, 2016.

[104] Shiqi Yu, Haifeng Chen, Qing Wang, Linlin Shen, and Yongzhen Huang. Invariant feature extraction for gait recognition using only one uniform model. *Neurocomputing*, 239:81–93, 2017.

[105] Wilhelm Braune and Otto Fischer. *Der Gang des Menschen: I. Theil: Versuche am unbelasteten und belasteten Menschen*, volume 1. BS Hirzel, 1895.

[106] Eadweard Muybridge. *Animal Locomotion: An Electro-photographic Investigation of Consecutives Phases of Animal Movement: Prospectus and Catalogue of Plates... Males (nude)]. I.* 1969.

[107] FRANÇOIS DAGOGNET. L'animal selon condillac, introd. a eb de condillac, traité des animaux, j. *Vrin, Paris*, pages 7–131, 1987.

[108] Michal Balazia and Petr Sojka. Gait recognition from motion capture data. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(1s):1–18, 2018.

[109] Henryk Josiński, Agnieszka Michalczuk, Daniel Kostrzewa, Adam Świtoński, and Konrad Wojciechowski. Heuristic method of feature selection for person re-identification based on gait motion capture data. In *Asian Conference on Intelligent Information and Database Systems*, pages 585–594. Springer, 2014.

[110] Athira M Nambiar, Alexandre Bernardino, Jacinto C Nascimento, and Ana LN Fred. Towards view-point invariant person re-identification via fusion of anthro-

pometric and gait features from kinect measurements. In *VISIGRAPP (5: VISAPP)*, pages 108–119, 2017.

[111] Martin Hofmann, Jürgen Geiger, Sebastian Bachmann, Björn Schuller, and Gerhard Rigoll. 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, 2014.

[112] Apurva Bedagkar-Gala and Shishir K Shah. Gait-assisted person re-identification in wide area surveillance. In *Asian Conference on Computer Vision*, pages 633–649. Springer, 2014.

[113] Lan Wei, Yonghong Tian, Yaowei Wang, and Tiejun Huang. Swiss-system based cascade ranking for gait-based person re-identification. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[114] Zheng Liu, Zhaoxiang Zhang, Qiang Wu, and Yunhong Wang. Enhancing person re-identification by integrating gait biometric. *Neurocomputing*, 168:1144–1156, 2015.

[115] Yumi Iwashita, Ryosuke Baba, Koichi Ogawara, and Ryo Kurazume. Person identification from spatio-temporal 3d gait. In *2010 International Conference on Emerging Security Technologies*, pages 30–35. IEEE, 2010.

[116] Athira Nambiar, Alexandre Bernardino, Jacinto C Nascimento, and Ana Fred. Context-aware person re-identification in the wild via fusion of gait and anthropometric features. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 973–980. IEEE, 2017.

[117] Vijay John, Gwenn Englebienne, and Ben Krose. Person re-identification using height-based gait in colour depth camera. In *2013 IEEE International Conference on Image Processing*, pages 3345–3349. IEEE, 2013.

[118] Athira Nambiar, Jacinto C Nascimento, Alexandre Bernardino, and José Santos-Victor. Person re-identification in frontal gait sequences via histogram of optic flow energy image. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 250–262. Springer, 2016.

[119] Pratik Chattopadhyay, Shamik Sural, and Jayanta Mukherjee. Information fusion from multiple cameras for gait-based re-identification and recognition. *IET Image Processing*, 9(11):969–976, 2015.

[120] Ryo Kawai, Yasushi Makihara, Chunsheng Hua, Haruyuki Iwama, and Yasushi Yagi. 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, 2012.

[121] Taiqing Wang, Shaogang Gong, Xiatian Zhu, and Shengjin Wang. Person re-identification by video ranking. In *European conference on computer vision*, pages 688–703. Springer, 2014.

[122] Michal Balazia and Petr Sojka. You are how you walk: Uncooperative mocap gait identification for video surveillance with incomplete and noisy data. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 208–215. IEEE, 2017.

[123] Athira M Nambiar, Alexandre Bernardino, and Jacinto C Nascimento. Cross-context analysis for long-term view-point invariant person re-identification via

soft-biometrics using depth sensor. In *VISIGRAPP (5: VISAPP)*, pages 105–113, 2018.

[124] M Burnfield. Gait analysis: normal and pathological function. *Journal of Sports Science and Medicine*, 9(2):353, 2010.

[125] Moshe Gabel, Ran Gilad-Bachrach, Erin Renshaw, and Assaf Schuster. Full body gait analysis with kinect. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1964–1967. IEEE, 2012.

[126] Aaron F. Bobick and James W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on pattern analysis and machine intelligence*, 23(3):257–267, 2001.

[127] Francisco M Castro, Manuel J Marín-Jimenez, and Rafael Medina-Carnicer. Pyramidal fisher motion for multiview gait recognition. In *2014 22nd International Conference on Pattern Recognition*, pages 1692–1697. IEEE, 2014.

[128] Jin Tang, Jian Luo, Tardi Tjahjadi, and Fan Guo. Robust arbitrary-view gait recognition based on 3d partial similarity matching. *IEEE Transactions on Image Processing*, 26(1):7–22, 2016.

[129] Tee Connie, Michael Kah Ong Goh, and Andrew Beng Jin Teoh. A grassmannian approach to address view change problem in gait recognition. *IEEE transactions on cybernetics*, 47(6):1395–1408, 2016.

[130] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer*

Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1, pages 539–546. IEEE, 2005.

[131] Chen Shen, Zhongming Jin, Yiru Zhao, Zhihang Fu, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. Deep siamese network with multi-level similarity perception for person re-identification. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1942–1950, 2017.

[132] Lin Wu, Yang Wang, Junbin Gao, and Xue Li. Where-and-when to look: Deep siamese attention networks for video-based person re-identification. *IEEE Transactions on Multimedia*, 21(6):1412–1424, 2018.

[133] Meng Zheng, Srikrishna Karanam, Ziyan Wu, and Richard J Radke. Re-identification with consistent attentive siamese networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5735–5744, 2019.

[134] Jane Bromley, James W Bentz, Léon Bottou, Isabelle Guyon, Yann LeCun, Cliff Moore, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993.

[135] Tao Lu, Qiang Zhou, Wenhua Fang, and Yanduo Zhang. Discriminative metric learning for face verification using enhanced siamese neural network. *Multimedia Tools and Applications*, pages 1–18, 2020.

[136] Wu Liu, Cheng Zhang, Huadong Ma, and Shuangqun Li. Learning efficient spatial-temporal gait features with deep learning for human identification. *Neuroinformatics*, 16(3-4):457–471, 2018.

[137] Shuangqun Li, Wu Liu, Huadong Ma, and Shaopeng Zhu. Beyond view transformation: Cycle-consistent global and partial perception gan for view-invariant gait recognition. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2018.

[138] Cassandra Carley, Ergys Ristani, and Carlo Tomasi. Person re-identification from gait using an autocorrelation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[139] Yiwei He, Junping Zhang, Hongming Shan, and Liang Wang. Multi-task gans for view-specific feature learning in gait recognition. *IEEE Transactions on Information Forensics and Security*, 14(1):102–113, 2018.

[140] Xiuhui Wang, ShiLing Feng, and Wei Qi Yan. Human gait recognition based on self-adaptive hidden markov model. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2019.

[141] Youjiang Xu, Yahong Han, Richang Hong, and Qi Tian. Sequential video vlad: Training the aggregation locally and temporally. *IEEE Transactions on Image Processing*, 27(10):4933–4944, 2018.

[142] Shichao Zhao, Yanbin Liu, Yahong Han, Richang Hong, Qinghua Hu, and Qi Tian. Pooling the convolutional layers in deep convnets for video action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(8):1839–1849, 2017.

[143] Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. Weakly supervised action localization by sparse temporal pooling network. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition, pages 6752–6761, 2018.

[144] Hanqing Chao, Yiwei He, Junping Zhang, and Jianfeng Feng. 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, 2019.

[145] Shuangqun Li, Wu Liu, and Huadong Ma. Attentive spatial–temporal summary networks for feature learning in irregular gait recognition. *IEEE Transactions on Multimedia*, 21(9):2361–2375, 2019.

[146] BingZhang Hu, Yan Gao, Yu Guan, Yang Long, Nicholas Lane, and Thomas Ploetz. Robust cross-view gait identification with evidence: A discriminant gait gan (diggan) approach on 10000 people. *arXiv preprint arXiv:1811.10493*, 2018.

[147] Yanyun Wang, Chunfeng Song, Yan Huang, Zhenyu Wang, and Liang Wang. Learning view invariant gait features with two-stream gan. *Neurocomputing*, 339:245–254, 2019.

[148] Ziyuan Zhang, Luan Tran, Feng Liu, and Xiaoming Liu. On learning disentangled representations for gait recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

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

[150] Christopher JC Burges. *Dimension reduction: A guided tour*. Now Publishers Inc, 2010.

[151] Tamar Avraham and Michael Lindenbaum. Learning appearance transfer for person re-identification. In *Person Re-Identification*, pages 231–246. Springer, 2014.

[152] Ryan W Thomas, Daniel H Friend, Luiz A Dasilva, and Allen B Mackenzie. Cognitive networks: adaptation and learning to achieve end-to-end performance objectives. *IEEE Communications magazine*, 44(12):51–57, 2006.

[153] Jason Weston, Frédéric Ratle, Hossein Mobahi, and Ronan Collobert. Deep learning via semi-supervised embedding. In *Neural networks: Tricks of the trade*, pages 639–655. Springer, 2012.

[154] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[155] Itamar Arel, Derek C Rose, and Thomas P Karnowski. Deep machine learning-a new frontier in artificial intelligence research [research frontier]. *IEEE computational intelligence magazine*, 5(4):13–18, 2010.

[156] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106, 1962.

[157] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[158] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[159] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[160] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[161] JS Borer, C Hochreiter, and S Rosen. Right ventricular function in severe non-ischaemic mitral insufficiency. *European heart journal*, 12(suppl_B):22–25, 1991.

[162] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[163] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Santiago López-Tapia, and Nicolás Pérez de la Blanca. Evaluation of cnn architectures for gait recognition based on optical flow maps. In *2017 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5. IEEE, 2017.

[164] Francisco M Castro, Manuel J Marín-Jiménez, and Nicolás Guil. Multimodal features fusion for gait, gender and shoes recognition. *Machine Vision and Applications*, 27(8):1213–1228, 2016.

[165] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *CVPR 2011*, pages 3169–3176. IEEE, 2011.

[166] Mihir Jain, Hervé Jégou, and Patrick Bouthemy. Better exploiting motion for better action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2555–2562, 2013.

[167] Martin Storath and Andreas Weinmann. Fast median filtering for phase or orientation data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):639–652, 2017.

[168] Florent Perronnin and Diane Larlus. Fisher vectors meet neural networks: A hybrid classification architecture. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3743–3752, 2015.

[169] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer, 2010.

[170] Tommi S Jaakkola, David Haussler, et al. Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems*, pages 487–493, 1999.

[171] Francisco Manuel Castro, Manuel J Marín-Jiménez, Nicolás Guil, and Nicolás Pérez De La Blanca. Automatic learning of gait signatures for people identification. In *International Work-Conference on Artificial Neural Networks*, pages 257–270. Springer, 2017.

[172] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[173] Bowen Zhang, Limin Wang, Zhe Wang, Yu Qiao, and Hanli Wang. Real-time action recognition with enhanced motion vector cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2718–2726, 2016.

[174] AJ Piergiovanni and Michael S Ryoo. Representation flow for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9945–9953, 2019.

[175] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003.

[176] Berthold KP Horn and Brian G Schunck. Determining optical flow. In *Techniques and Applications of Image Understanding*, volume 281, pages 319–331. International Society for Optics and Photonics, 1981.

[177] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981.

[178] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *European conference on computer vision*, pages 25–36. Springer, 2004.

[179] Nils Papenberg, Andrés Bruhn, Thomas Brox, Stephan Didas, and Joachim Weickert. Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision*, 67(2):141–158, 2006.

[180] Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE transactions on pattern analysis and machine intelligence*, 33(3):500–513, 2010.

[181] Tak-Wai Hui and Ronald Chung. Determining motion directly from normal flows upon the use of a spherical eye platform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2267–2274, 2013.

[182] Tak-Wai Hui and Ronald Chung. Determining shape and motion from monocular camera: A direct approach using normal flows. *Pattern recognition*, 48(2):422–437, 2015.

[183] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015.

[184] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017.

[185] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018.

[186] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional

baseline). In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 480–496, 2018.

[187] Lei Qi, Lei Wang, Jing Huo, Yinghuan Shi, and Yang Gao. Greyreid: A two-stream deep framework with rgb-grey information for person re-identification. *arXiv preprint arXiv:1908.05142*, 2019.

[188] Wei Zeng, Cong Wang, and Feifei Yang. Silhouette-based gait recognition via deterministic learning. *Pattern recognition*, 47(11):3568–3584, 2014.

[189] Liang Wang, Tieniu Tan, Weiming Hu, and Huazhong Ning. Automatic gait recognition based on statistical shape analysis. *IEEE transactions on image processing*, 12(9):1120–1131, 2003.

[190] Milene Arantes and Adilson Gonzaga. Human gait recognition using extraction and fusion of global motion features. *Multimedia tools and applications*, 55(3):655–675, 2011.

[191] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012.

[192] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[193] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceed-*

ings of the IEEE international conference on computer vision, pages 4489–4497, 2015.

[194] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[195] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[196] Andrea Vedaldi and Karel Lenc. Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692, 2015.

[197] Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*, 2014.

[198] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

[199] Shuai Zheng, Junge Zhang, Kaiqi Huang, Ran He, and Tieniu Tan. Robust view transformation model for gait recognition. In *2011 18th IEEE International Conference on Image Processing*, pages 2073–2076. IEEE, 2011.

[200] Shiqi Yu, Daoliang Tan, and Tieniu Tan. A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 4, pages 441–444. IEEE, 2006.

[201] Brian D Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.

[202] Stephanie Glen. *Covariate Definition in Statistics*, Accessed: 2018-11-28.

[203] Francisco M Castro, Manuel J Marín-Jiménez, N Guil Mata, and Rafael Muñoz-Salinas. Fisher motion descriptor for multiview gait recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 31(01):1756002, 2017.

[204] Manuel J Marín-Jiménez, Francisco M Castro, Nicolás Guil, F de La Torre, and R Medina-Carnicer. Deep multi-task learning for gait-based biometrics. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 106–110. IEEE, 2017.

[205] Tenika Whytock, Alexander Belyaev, and Neil M Robertson. Dynamic distance-based shape features for gait recognition. *Journal of Mathematical Imaging and Vision*, 50(3):314–326, 2014.

[206] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[207] Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. *arXiv preprint arXiv:1412.7449*, 2014.

[208] William Chan, Navdeep Jaitly, Quoc V Le, and Oriol Vinyals. Listen, attend and spell. *arXiv preprint arXiv:1508.01211*, 2015.

[209] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.

[210] Qingji Guan, Yaping Huang, Zhun Zhong, Zhedong Zheng, Liang Zheng, and Yi Yang. Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. *arXiv preprint arXiv:1801.09927*, 2018.

[211] Li Yao, Eric Poblenz, Dmitry Dagunts, Ben Covington, Devon Bernard, and Kevin Lyman. Learning to diagnose from scratch by exploiting dependencies among labels. *arXiv preprint arXiv:1710.10501*, 2017.

[212] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[213] Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*, 2016.

[214] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-alone self-attention in vision models. *arXiv preprint arXiv:1906.05909*, 2019.

[215] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[216] Stas Goferman, Lihi Zelnik-Manor, and Ayellet Tal. Context-aware saliency detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(10):1915–1926, 2011.

[217] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

[218] Loris Bazzani, Hugo Larochelle, and Lorenzo Torresani. Recurrent mixture density network for spatiotemporal visual attention. *arXiv preprint arXiv:1603.08199*, 2016.

[219] Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. Action recognition using visual attention. *arXiv preprint arXiv:1511.04119*, 2015.

[220] Haruyuki Iwama, Mayu Okumura, Yasushi Makihara, and Yasushi Yagi. 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, 2012.

[221] Shiqi Yu, Haifeng Chen, Edel B Garcia Reyes, and Norman Poh. Gaitgan: Invariant gait feature extraction using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 30–37, 2017.

[222] Amit Kale, AK Roy Chowdhury, and Rama Chellappa. Towards a view invariant gait recognition algorithm. In *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, 2003.*, pages 143–150. IEEE, 2003.

[223] Worapan Kusakunniran, Qiang Wu, Jian Zhang, and Hongdong Li. Support vector regression for multi-view gait recognition based on local motion feature selection. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 974–981. IEEE, 2010.

[224] Yasushi Makihara, Atsuyuki Suzuki, Daigo Muramatsu, Xiang Li, and Yasushi Yagi. Joint intensity and spatial metric learning for robust gait recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5705–5715, 2017.

[225] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[226] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[227] Wojciech Zaremba and Ilya Sutskever. Learning to execute. *arXiv preprint arXiv:1410.4615*, 2014.

[228] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, pages 1764–1772, 2014.

[229] Ziwei Yang, Yahong Han, and Zheng Wang. Catching the temporal regions-of-interest for video captioning. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 146–153, 2017.

[230] Jun Xu, Ting Yao, Yongdong Zhang, and Tao Mei. Learning multimodal attention lstm networks for video captioning. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 537–545, 2017.

[231] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen. Video captioning with attention-based lstm and semantic consistency. *IEEE Transactions on Multimedia*, 19(9):2045–2055, 2017.

[232] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning deep spatio-temporal dependence for semantic video segmentation. *IEEE Transactions on Multimedia*, 20(4):939–949, 2017.

[233] Alejandro Hernandez Ruiz, Lorenzo Porzi, Samuel Rota Bulò, and Francesc Moreno-Noguer. 3d cnns on distance matrices for human action recognition. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1087–1095, 2017.

[234] Wanru Xu, Zhenjiang Miao, Xiao-Ping Zhang, and Yi Tian. A hierarchical spatio-temporal model for human activity recognition. *IEEE Transactions on Multimedia*, 19(7):1494–1509, 2017.

[235] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of*

the IEEE conference on computer vision and pattern recognition, pages 2625–2634, 2015.

[236] Maodi Hu, Yunhong Wang, Zhaoxiang Zhang, James J Little, and Di Huang. View-invariant discriminative projection for multi-view gait-based human identification. *IEEE Transactions on Information Forensics and Security*, 8(12):2034–2045, 2013.

[237] Worapan Kusakunniran, Qiang Wu, Jian Zhang, Hongdong Li, and Liang Wang. Recognizing gaits across views through correlated motion co-clustering. *IEEE Transactions on Image Processing*, 23(2):696–709, 2013.

[238] Xianglei Xing, Kejun Wang, Tao Yan, and Zhuowen Lv. Complete canonical correlation analysis with application to multi-view gait recognition. *Pattern Recognition*, 50:107–117, 2016.

[239] Chandrashekhar Padole and Hugo Proença. An aperiodic feature representation for gait recognition in cross-view scenarios for unconstrained biometrics. *Pattern Analysis and Applications*, 20(1):73–86, 2017.

[240] Zhaoxiang Zhang, Jiaxin Chen, Qiang Wu, and Ling Shao. Gii representation-based cross-view gait recognition by discriminative projection with list-wise constraints. *IEEE transactions on cybernetics*, 48(10):2935–2947, 2017.

[241] Huimin Wu, Jian Weng, Xin Chen, and Wei Lu. Feedback weight convolutional neural network for gait recognition. *Journal of visual communication and image representation*, 55:424–432, 2018.

[242] Sabesan Sivapalan, Daniel Chen, Simon Denman, Sridha Sridharan, and Clinton Fookes. Histogram of weighted local directions for gait recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 125–130, 2013.

[243] Yue-Fei Guo, Lide Wu, Hong Lu, Zhe Feng, and Xiangyang Xue. Null foley–sammon transform. *Pattern recognition*, 39(11):2248–2251, 2006.

[244] Donald H. Foley and John W Sammon. An optimal set of discriminant vectors. *IEEE Transactions on computers*, 100(3):281–289, 1975.