Fall 2019

Person Identification With Convolutional Neural Networks

Kang Zheng

Recommended Citation

Zheng, K.(2019). *Person Identification With Convolutional Neural Networks.* (Doctoral dissertation). Retrieved from https://scholarcommons.sc.edu/etd/5601

Person Identification with Convolutional Neural Networks

by

Kang Zheng

Bachelor of Engineering
Harbin Institute of Technology, 2012

Submitted in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy in

Computer Science and Engineering

College of Engineering and Computing

University of South Carolina

2019

Accepted by:

Song Wang, Major Professor

Michael N. Huhns, Committee Member

Yan Tong, Committee Member

Lannan Luo, Committee Member

Xiaofeng Wang, Committee Member

Cheryl L. Addy, Vice Provost and Dean of the Graduate School

Acknowledgments

I want to gratefully thank my advisor, Prof. Song Wang, for the tremendous help and support he has provided for me as I am pursuing my Ph.D. He is insightful when discussing about my research. He is patient when guiding me through difficulties. He is strict when pushing me for higher standard. Not only is he a great advisor for research, but he is also a great mentor for life.

I would like to thank committee members for my doctoral dissertation, Prof. Michael Huhns, Prof. Yan Tong, Prof. Lannan Luo, and Prof. Xiaofeng Wang, for their help and constructive suggestions. It is my great honor to have them as my committee members.

I would like to thank my colleagues: Yu Cao, Jarrell Waggoner, Dhaval Salvi, Youjie Zhou, Ping Liu, Xiaochuan Fan, Yuewei Lin, Shizhong Han, Zibo Meng, Jun Zhou, Dazhou Guo, Hongkai Yu, Yang Mi, Hao Guo, Haozhou Yu, Yuhang Lu, Jun Chen, Jing Wang, James O'Reilly, Ahmed Shehab Khan, Jie Cai, Zhiyuan Li, and other fellow lab-mates. They have been providing positive and encouraging atmosphere. We have shared not only hard work but also laughter.

Finally, I want to thank my parents and my girlfriend for their love and support during the past few years. They always have faith in me, for the easy times and for the difficult times.

Abstract

Person identification aims at matching persons across images or videos captured by different cameras, without requiring the presence of persons' faces. It is an important problem in computer vision community and has many important real-world applications, such as person search, security surveillance, and no-checkout stores. However, this problem is very challenging due to various factors, such as illumination variation, view changes, human pose deformation, and occlusion. Traditional approaches generally focus on hand-crafting features and/or learning distance metrics for matching to tackle these challenges. With Convolutional Neural Networks (CNNs), feature extraction and metric learning can be combined in a unified framework.

In this work, we study two important sub-problems of person identification: cross-view person identification and visible-thermal person re-identification. Cross-view person identification aims to match persons from *temporally synchronized* videos taken by wearable cameras. Visible-thermal person re-identification aims to match persons between images taken by visible cameras under normal illumination condition and thermal cameras under poor illumination condition such as during night time.

For cross-view person identification, we focus on addressing the challenge of view changes between cameras. Since the videos are taken by wearable cameras, the underlying 3D motion pattern of the same person should be consistent and thus can be used for effective matching. In light of this, we propose to extract view-invariant motion features to match persons. Specifically, we propose a CNN-based triplet network to learn view-invariant features by establishing correspondences between 3D human MoCap data and the projected 2D optical flow data. After training, the triplet net-

work is used to extract view-invariant features from 2D optical flows of videos for matching persons. We collect three datasets for evaluation. The experimental results demonstrate the effectiveness of this method.

For visible-thermal person re-identification, we focus on the challenge of domain discrepancy between visible images and thermal images. We propose to address this issue at a class level with a CNN-based two-stream network. Specifically, our idea is to learn a center for features of each person in each domain (visible and thermal domains), using a new relaxed center loss. Instead of imposing constraints between pairs of samples, we enforce the centers of the same person in visible and thermal domains to be close, and the centers of different persons to be distant. We also enforce the feature vector from the center of one person to another in visible feature space to be similar to that in thermal feature space. Using this network, we can learn domain-independent features for visible-thermal person re-identification. Experiments on two public datasets demonstrate the effectiveness of this method.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recognizing persons is one of the most fundamental tasks humans need to perform on a daily basis. For person recognition, we rely mostly on a person's face. However, a person's appearance such as clothes, accessories, and hair style can also be used for person recognition. Besides, walking style, or gait, is also an important cue. In current person recognition systems, face images are still the first choice for recognizing a person's identity. However, face images are not always available, which makes face recognition not applicable in some scenarios. Person identification aims at identifying a person's identity from a given image or video of the person, without requiring the presence of the person's face. In this research, we focus on person identification, which matches person across images or videos.

In a general setting, we are given a person image or video as the query and our goal is to find the same person in a gallery (database) of person images or videos. Since there are many person identities in the gallery, we usually rank the person images or videos according to their similarities to the query person image or video. The higher the ground-truth person image or video ranks, the better. Figure 1.1 shows an example result of person identification. The ground-truth person images in the gallery set are highlighted in green box. The gallery images are ranked in descending order from left to right by the similarity to query image.

For a practical person identification system, three modules will be needed: person detection, person tracking (for video-based person identification), and person retrieval [153]. As shown in Fig. 1.2, persons are first detected with a human detection model and then associated across frames to obtain the tracking result. We follow the common protocol and assume that both person detection and person tracking are accomplished, to focus on the person retrieval module as shown in Fig. 1.2(c). Specifically, as shown in Fig. 1.3, the pipeline of person identification can be divided into two stages: feature extraction and distance computation. Features are first extracted as a representation for each image or video in both query and gallery sets. Distance

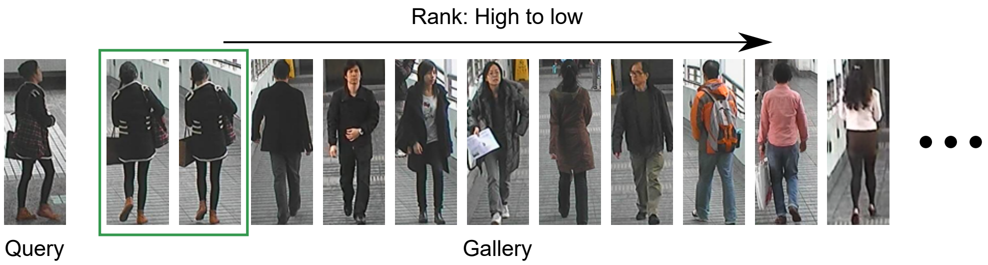


Figure 1.1 An example result of person identification. The gallery images are ranked in descending order from left to right by the similarity to query image.

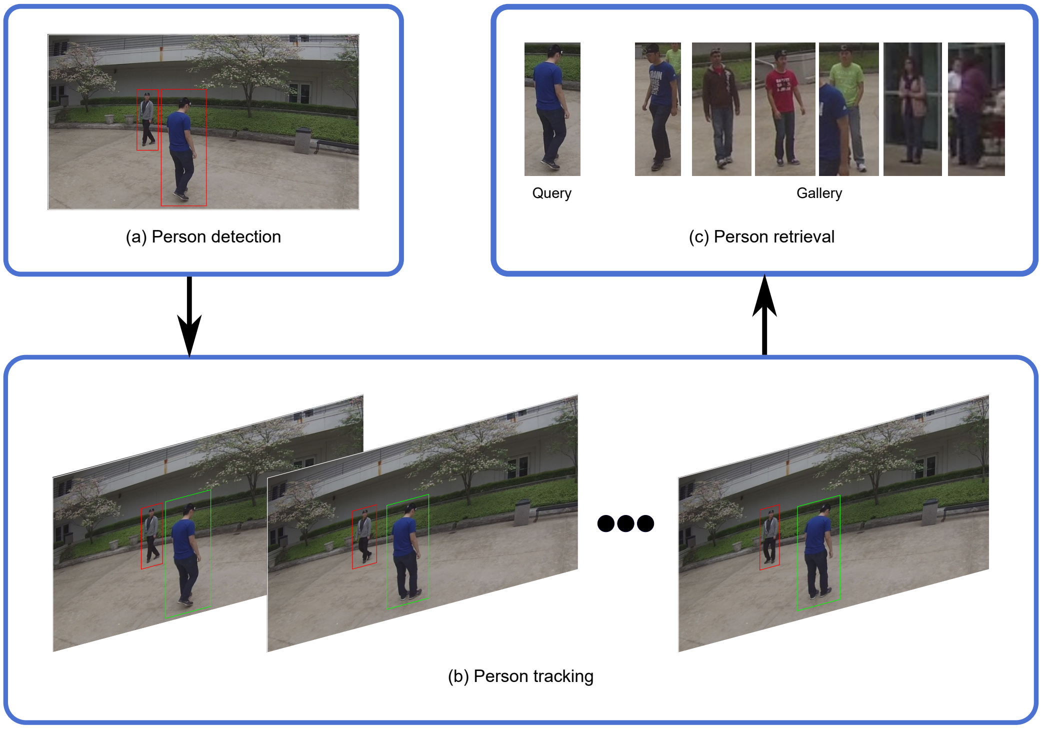


Figure 1.2 Three modules for a practical person identification system: person detection, person tracking, and person retrieval.

metric computes the distance or similarity between each image or video in the query set to each image or video in the gallery set. Both feature extraction and distance metric can be hand-designed or learned with deep networks.

Person identification is a very important problem in computer vision community.

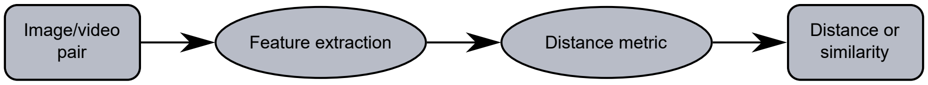


Figure 1.3 An illustration of the pipeline for person identification.

First, it can be applied to various real-world vision-based scenarios. For example, we can apply person identification to search for a criminal in escape given an image or video of the criminal. Specifically, we can match the given image or video to those captured by surveillance cameras in different places such as gas stations, toll stations, airports, and grocery stores. In this case, visual comparisons by human is labor-intensive, inefficient and prone to error. Thus, automatic search with person identification is much desirable. Similarly, person identification can be applied for searching for missing person in a large theme park. In an event scene with crowded people, police officers can use wearable cameras to capture videos for surveillance, in which the same persons can be identified with person identification for detecting abnormal activity. With the fast growing internet-of-things (IoT) devices, person identification can be embedded in video-based security surveillance systems. Person identification could also be applied to identify the same persons with shopped items for no-checkout stores such as Amazon Go for the convenience of shopping experience. Second, the study of person identification can improve our understanding of key factors in human and computer vision systems, and may further benefit the refinement of person identification system and other automatic vision-based systems. An accurate, robust and efficient person identification system requires effective features for matching. This is a very challenging task since there are many different variations in person images captured.

1.1 Challenges

Although studied for over a decade, person identification remains a very challenging problem. Even with today's big data and state-of-the-art deep learning techniques, this problem is far from being solved. The complexity of this problem mostly comes from many different variations when capturing persons' images or videos. Such variations could make not only the same person look dissimilar, but also different persons similar. We will describe these challenges in detail in the following.

Illumination variation

The change of illumination can dramatically change a person's appearance even though he/she is captured in the same place wearing the same clothes. One famous example is the dress shown in Fig. 1.4. The dress is blue and black as shown in Fig. 1.4(a). However, with a different illumination as in Fig. 1.4(b), the dress could be perceived as white and gold by some people, if not all. For person identification, appearance features, especially colors, are important for matching. Different illumination can lead to different appearances of the same person, thus making person identification more difficult.

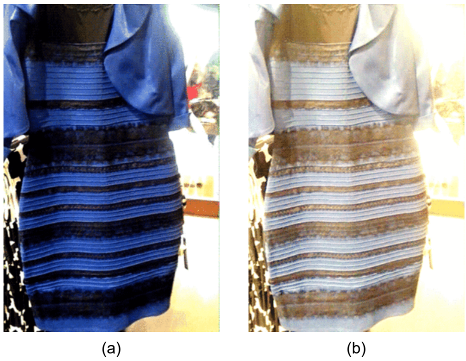


Figure 1.4 An illustration of appearance change caused by illumination variation.

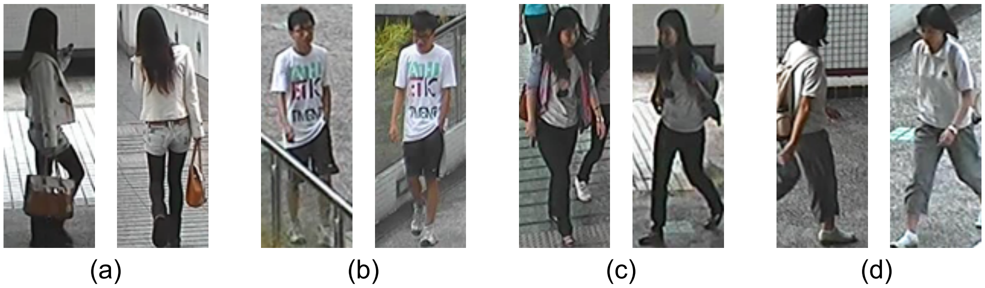


Figure 1.5 An illustration of view angle variations: each column is a pair of images for the same person.

View angle variation

The view angle from which a person is captured also affects the appearances. Figure 1.5 shows four pair of person images captured from different view angles in CUHK02 dataset [69]. For each pair of person images, we can see that the appearance varies largely as the view angle changes. Because each image of a person is the projection from 3D space to 2D space, different view angles result in different 2D projections and very different appearances. Therefore, the view angle variation is also harmful to the accuracy of person identification.

Scale variation

The distance from a person to the camera determines how large this person will appearance in the image captured. The larger the distance is, the smaller the person appears. Figure 1.6 shows three pairs of person images with different scales in SYSU-MM01 dataset [124]. Because of the scale variation, the features extracted usually have different dimensions and cannot be matched directly. One common approach is to resize person images to a fixed size such that features of all images will have the same dimension. However, this approach will lose information for images larger than the fixed size.

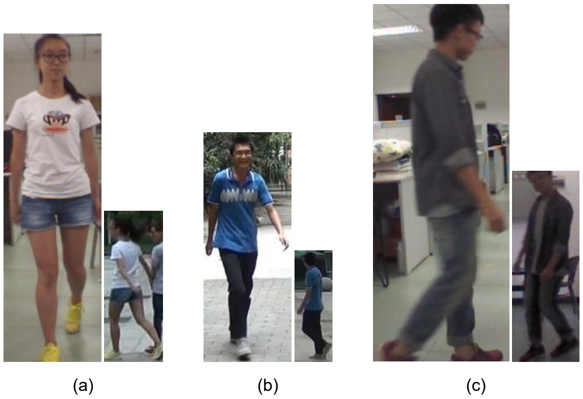

(a) (b) (c)

Figure 1.6 An illustration of scale variations: each column is a pair of images for the same person.

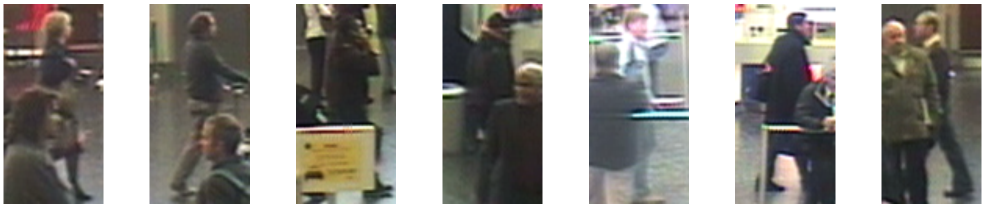


Figure 1.7 Example images with occlusions.

Partial occlusion

Person identification is often applied to visual surveillance systems, where the cameras are installed in tall buildings or poles with crowded people in the camera view. This can cause the problem of severe occlusion in the images captured. Figure 1.7 shows some example images with occlusions from iLIDS-VID dataset [120]. These occlusions will affect the features extracted and deteriorate the person identification accuracy.

Low image quality

Another challenge in person identification is the low image quality in the data, such as the images from iLIDS-VID dataset [120] shown in Fig. 1.7. Although high resolution cameras are adopted everywhere, person identification datasets are usually for

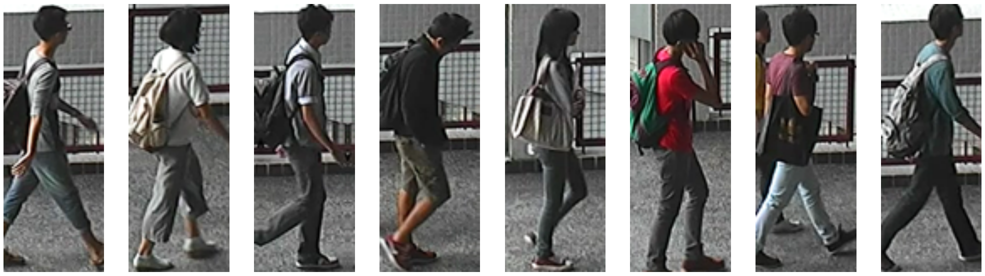


Figure 1.8 Example images with human pose variation.

security surveillance and thus captured at low resolution to save storage. Such low image quality will produce low quality features, which will lower the performance of person identification.

Human pose variation

Humans can have a large variety of poses for different situations, such as standing, sitting, walking, and riding bike. Such variations cause the large variation of person images. For person identification, walking is the most common situation, which eases the problem to some extent. However, walking also has different phases with different poses. Besides, a person can carry a bag or other accessories while walking, thus resulting in more pose variations. Figure 1.8 shows some different poses from CUHK02 dataset [69]. Moreover, the human poses are projected from 3D physical space into 2D image space, which leads to the loss of information and results in some ambiguity.

Change of clothes, accessories, and hair styles

Currently, person identification assumes that the same person wears the same clothes. Thus, most existing approaches for person identification rely on appearance features for matching persons across images or videos. However, in practice, people may change their clothes, accessories, hair styles, and other appearances from time to time. In this situation, these approaches will be much less effective and we need to resort to more robust features for matching such as biometric features: gait, height, body limb shape, and so on.

Faulty detection

Natural images are usually captured with more than just one person included. Therefore, manual annotation or automatic detection are needed to obtain the bounding box for each person, so that better features are extracted for person identification. Previously, manual annotation is adopted as the scale of person identification datasets is small. With larger and larger datasets being collected, automatic detection is used more often than manual annotation as it is more efficient. Ideally, the bounding boxes are tightly detected around the person. However, even state-of-the-art human detection models are not good enough, which can produce undesirable or even wrong predictions of bounding boxes. Figure 1.9 shows some faulty detections from SYSU-MM01 dataset. Some images only contain part of a person, while others have too much background or even no person inside at all. The bad detection further introduces misalignment of persons in both the image space and feature space.

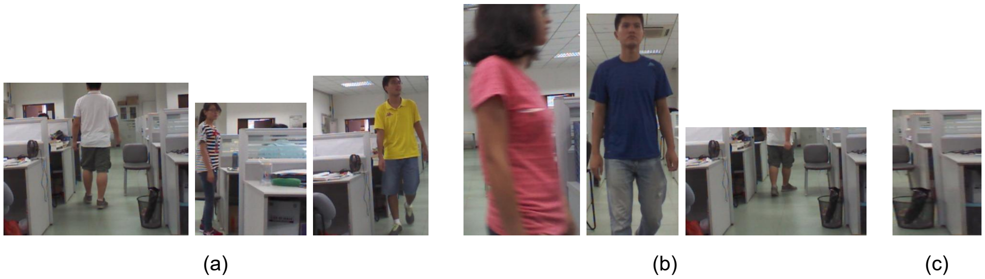


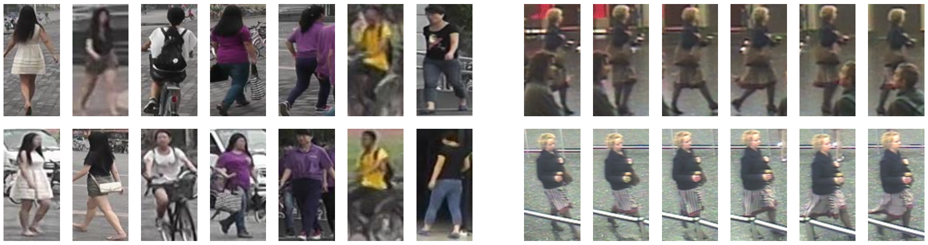
Figure 1.9 Faulty detections with (a) too many background, (b) only part of human body, and (c) no human at all.

1.2 Taxonomy of Person Identification

With more and more attention drawn to person identification, several important sub-problems have been raised by researchers. These problems can be divided into two categories: person re-identification and cross-view person identification. These sub-problems are mainly divided by the scenarios of applications. In this section, we will describe these sub-problems of person identification in more detail.

1.2.1 Person Re-Identification

Person re-identification aims at matching persons between images or videos captured at different times and different places without view overlap. It is the most widely studied problem so far. It can be categorized into image-based and video-based depending on the format of input data. Figure 1.10 shows some pairs of person images and videos for image-based and video-based person re-identification, respectively. Since the images or videos to be matched are captured at different times and different places, the difficulty in matching can be elevated by variations such as illumination, background, viewpoint, human pose, and scale. Besides, we cannot take advantage of video temporal synchronization as they are not captured at the same time. Previously, various hand-crafted features [25, 84, 32, 85, 6, 2, 9, 56, 107] and metric learning approaches [105, 101, 154, 72, 44, 122, 57, 73, 119] are derived to tackle this problem. Recently, convolutional neural networks (CNNs) are widely applied for person re-identification. State-of-the-art performances are achieved with CNN-based approaches [114, 112, 13, 67, 145, 111, 146, 131, 156, 86, 87, 144, 121, 71, 157], some of which have surpassed human-level. However, current person re-identification solutions still suffer from the common problem of overfitting because of the insufficient dataset scale, which more researchers are focusing on recently [108, 27, 100, 121, 157].



(a) Image-based person re-identification.

(b) Video-based person re-identification.

Figure 1.10 Examples of (a) image-based person re-identification from Market-1501 dataset [152] and (b) video-based person re-identification from iLID-VIDS dataset [120]. Each column corresponds to the same person.

Partial Person Re-Identification

In [155], Zheng et al. introduced a new problem named *partial person re-identification*, which focuses on the issue of occlusion in person re-identification. In this new problem setting, the goal is to find the same person captured with full body appearance given only a partial query image. In practice, a person may not be captured with full-body image by camera due commonly happening occlusions. Occlusions can result from self-occlusion, crowded people, or static obstacles such as trees or poles. It is also possible that a person may be intentionally hiding which causes occlusion [155]. For example, a criminal in escape is very likely to intentionally hide from surveillance cameras to avoid being caught. Figure 1.11 shows example images for partial person re-identification. In this problem setting, the occlusion always exists and can happen to any part of a human body. Therefore, it is important to correctly detect the occlusion to extract robust features for partial person re-identification. To address this issue, several approaches are developed recently [54, 39, 24, 104, 113, 40, 49, 82].

Person Search

As mentioned before, a practical person identification system requires person detection, person tracking, and person retrieval. Because in real-world applications, only

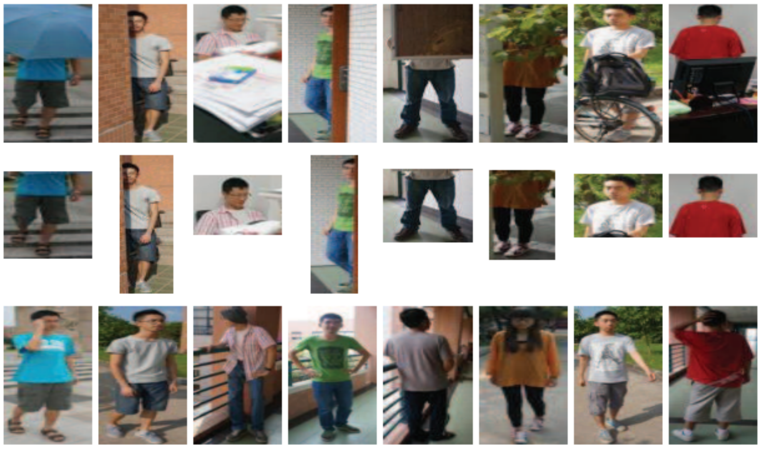


Figure 1.11 Examples of partial person re-identification in [155]. First row: original partial person images, second row: input for partial person re-identification annotated by operators, third row: corresponding full-body images.

whole-scene images are provided. Each instance of persons in the whole-scene images need to be detected and tracked first before they can be used in person retrieval for identification. Recently, more attention are drawn to unify the person detection and person retrieval process, namely person search [132, 129, 78, 42, 77, 11, 127, 63, 47, 8, 134, 95]. In person search, the person tracking process is neglected since it only applies to videos and videos are not always available. Specifically, person search first detects individual persons from whole-scene images, and then matches each individual person to a query person, as shown in Fig. 1.12. These two processes are unified in the sense that features extracted from the whole-scene image are utilized again for person retrieval. Thus, the overall inference procedure could be significantly accelerated for better efficiency. However, it also poses a new question: are the features extracted from whole-scene images optimal for person retrieval, which is bounded to only a local area? As for now, this remains an open question, since whole-scene image features are extracted from the whole image. The features contain both information of the person and information of background, which may or may not be useful. If

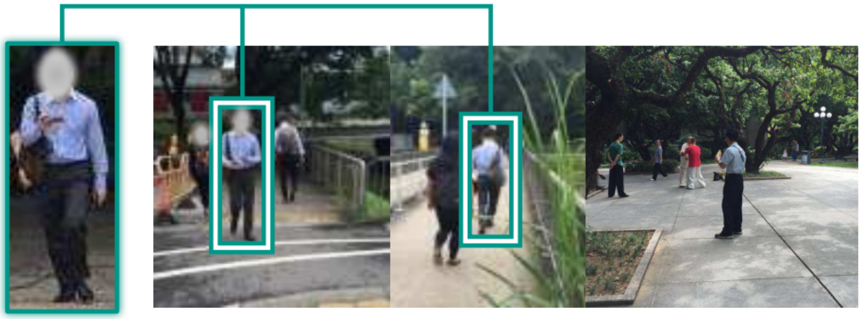


Figure 1.12 An illustration of person search in [129].

not, how should we utilize the whole-scene image features to better search the person of interest?

Visible-Thermal Person Re-Identification

Visible-thermal person re-identification aims at matching persons between visible (RGB) images and thermal (infrared) images. This problem is rarely studied [124, 137, 138, 17] yet it can be applied to substantially improve security surveillance under poor illumination conditions, such as night time. As shown in Fig. 1.13, thermal images lack color information due to the poor illumination condition. As a result, besides the common challenges seen in regular person re-identification, there exists semantic domain discrepancy between visible images and thermal images. Existing efforts [124, 137, 138, 17] try to reduce such domain discrepancy by enforcing the feature representations of both visible images and thermal images into a common feature space.

1.2.2 Cross-View Person Identification

In [151], we are the first to propose a new important sub-problem of person identification: cross-view person identification (CVPI), which matches persons between

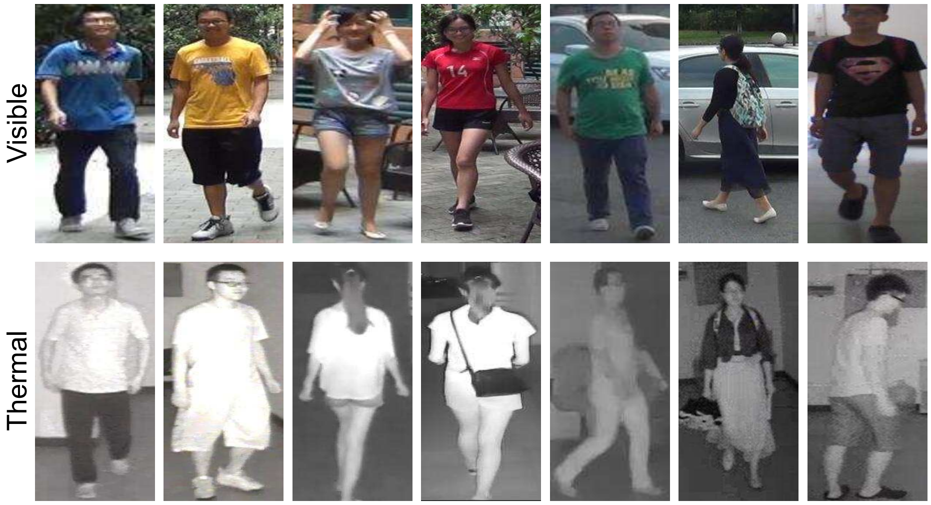


Figure 1.13 Visible-thermal person re-identification. Each column contains a visible image and a thermal image from the same person.

temporally synchronized videos taken by wearable cameras. This problem is important since it can facilitate the understanding of event scenes captured by multiple wearable cameras such as Google Glass and GoPro. For example, in a protest scene, police officers on site can use Google Glasses for surveillance and their captured videos from different view angles can provide complementary information for recognizing abnormal activities of people. In order to perform multi-view activity recognition, we need to first identify the same person of common interest from multiple videos, which is the goal of CVPI.

For CVPI, the videos taken by multiple wearable cameras are *temporally synchronized*, i.e., these videos are aligned in a way that the corresponding frames in all these videos are taken at the same time. This can be achieved by synchronizing clocks in these cameras. We show an example of a person captured by two cameras from two different views synchronously in Fig. 1.14, where the two videos are a matching or positive pair. If two temporally synchronized videos capture two different persons, they are a non-matching or negative pair.

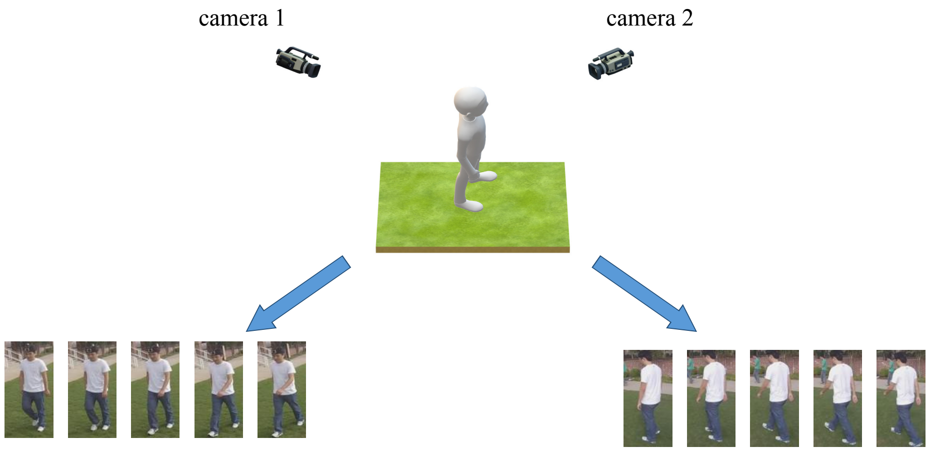


Figure 1.14 An example of two temporally synchronized videos capturing the same person from two different view angles.

Table 1.1 Differences between CVPI and person reID.

	CVPI	Person reID
Video	Temporally synchronized	Not synchronized
Application	The same scene	Different scenes
Camera	Wearable camera	Static camera
View angle	Horizontal view	Tilted view

While the proposed CVPI and the long studied person re-identification (reID) both aim to match persons, these two problems have many differences, as shown in Table 1.1. The most important difference is that the pair of videos are temporally synchronized in CVPI, but not in person reID. CVPI aims to match the same person captured at the same time and scene, while person reID aims to match the same person captured at different times and scenes. The videos in CVPI are usually taken by wearable cameras such as GoPro or Google Glass, while the videos in person reID are usually taken by static cameras mounted on buildings or poles. Besides, videos in CVPI are usually captured by wearable cameras at a normal human height, while videos in person reID are usually captured from a tilted view by static cameras mounted on buildings or poles. CVPI and person reID have different challenges:

person reID could not take advantage of video temporal synchronization and CVPI is expected to match person with much larger view differences.

1.3 Proposed Research

1.3.1 Cross-View Person Identification

In this research, we first study the problem of cross-view person identification. We propose to utilize human motion consistency for CVPI, since the same person must have the same underlying movement in temporally synchronized videos. To represent human motions, we use optical flow as following motion-based approaches. However, optical flow is the motion projected from 3D space to 2D space, and thus view-variant. To address this problem, we propose to learn a model to extract view-invariant features from optical flows, by training with optical flow data together with 3D human motion capture (Mocap) data. Specifically, we propose a novel approach to synthesizing optical flows from a 3D Mocap database. We train a Triplet Network (TN) consisting of three sub-networks: two for the synthesized optical flow sequences from different views and one for 3D Mocap sequences. The sub-networks for the optical flow sequences are further fine-tuned on real-world optical flows and then used to extract view-invariant features for CVPI. To evaluate this method, we collected three datasets with pairs of synchronized videos taken by wearable cameras. Experimental results show that, training the proposed network on the synthesized optical flow sequences together with 3D Mocap sequences can achieve better performance than only training sub-networks on optical flow sequences. The proposed method performs comparably with state-of-the-art methods, using only the motion information. Further combination of the proposed method with an appearance-based method achieves new state-of-the-art performance.

1.3.2 Visible-Thermal Person Re-Identification

We also study the problem of visible-thermal person re-identification (VT-reID), which is rarely studied despite its great importance in person identification. VT-reID aims to match persons across images taken by visible cameras under normal illumination condition and thermal cameras under poor illumination condition such as during night time. As thermal images lack color information due to the difference of illumination condition and imaging process, one major challenge of VT-reID is the semantic domain discrepancy between visible images and thermal images. In this research, we propose to learn domain-independent features for visible-thermal person re-identification with class-level constraints to alleviate this problem, in which each person is treated as a class. The intuition is to reduce the influence of individual samples that are too hard to identify or even wrongly-labeled, by weighing all samples within each class. Specifically, we train a two-stream CNN network to extract features from visible and thermal images separately. For each person, we learn one center from features in visible domain and one center from features in thermal domain, with a new relaxed version of center loss [123]. Then, we apply the pull and push constraints to the centers of the same or different persons in visible and thermal domains. More specifically, we enforce centers of the same person in visible domain and thermal domain to be close to each other, and centers of different persons to be distant from each other. Furthermore, for two different persons, the inter-class difference between their centers in the visible domain should be similar to that in the thermal domain. We formulate these class-level constraints as class-level supervision to train the two-stream CNN network, which is later used to extract features from visible and thermal images separately for person re-identification. Experiments on two public datasets demonstrate the effectiveness of this method.

1.4 Structure of the Dissertation

This dissertation is organized as follows. In Chapter 2, we introduce some essential background knowledge for this research. In Chapter 3, we review previous works related to this research. Chapter 4 and 5 primarily details the proposed research. Finally, we conclude this dissertation in Chapter 6.

Chapter 2

Background

In this chapter, we briefly review the essential background knowledge for this research. Specifically, we will introduce Convolutional Neural Networks (CNNs), Long Short-Term Memory networks (LSTM), and some metric learning approaches. CNNs have been widely applied to person identification and achieved significant performances in various person identification datasets. LSTM is typically used to process sequential input data, such as audio, video and texts. Another indispensable element of person identification is metric learning. The metric learning in deep neural networks are usually called *deep metric learning*, which guides the network to learn effective feature embeddings. We will introduce these important concepts in the following sections.

2.1 Convolutional Neural Network

Convolutional Neural Networks (CNNs) are a member of the neural network family and similary to a regular neural network. A regular neural network contains a sequence of layers, each consisting of a group of neurons. Figure 2.1(a) shows a 3-layer neural network. Each layer of neurons receive some input, perform a dot product and optionally add a non-linearity in the end. The output of one layer is fed to the next layer. Each neuron in one layer is connected to all the neurons in the preceding layer, which is usually called *fully-connected*. The input layer receives input from raw input data, and the output layer (last layer) outputs the predictions. Each layer bears a set of weights/parameters for the dot-product operation. The parameters are to be learned from data, by minimizing different loss functions depending on the tasks. Loss functions, such as cross-entropy loss or mean square error (MSE) loss, are commonly used to train the network. To process image data as input, we usually reshape the image pixels into a long vector and feed it to the regular neural network. However, regular neural networks do not scale well to larger images because of the full connectivity between neurons of neighboring layers. As image sizes increases,

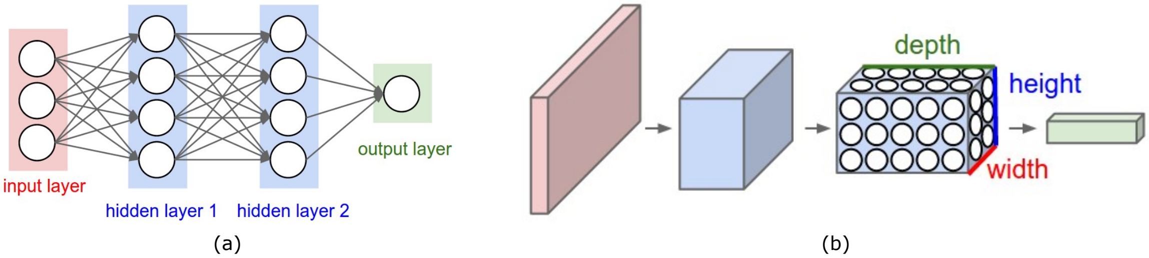


Figure 2.1 (a) An ordinary 3-layer neural network. (b) A convolutional neural network. (Source: http://cs231n.github.io/convolutional-networks/)

the number of parameters will increase exponentially as there are many layers in a regular neural network. This will not only increase the computation but also lead to overfitting problem.

Similar to regular neural network, a CNN also contains many layers, with many neurons in each layer. However, the neurons in CNN layers are different from the neurons in ordinary NNs. Specifically, the neurons in CNN are arranged in three dimensions: width, height, and depth, as shown in Fig. 2.1(b). Another difference between CNN and regular neural network is the connectivity between neurons of neighboring layers. In CNN, a neuron in one layer is only connected to a part of neurons in the preceding layer, and the parameters are shared between neurons. Such a design has two advantages: 1) it reduces the computation and the number of learnable parameters significantly; 2) compared to neurons in regular neural network, it is more natural because neurons should perceive different regions in an image in the same manner. There are three major types of layers in CNNs: convolutional layer, pooling layer, and fully-connected layer. Figure. 2.2 shows the architecture of AlexNet [58].

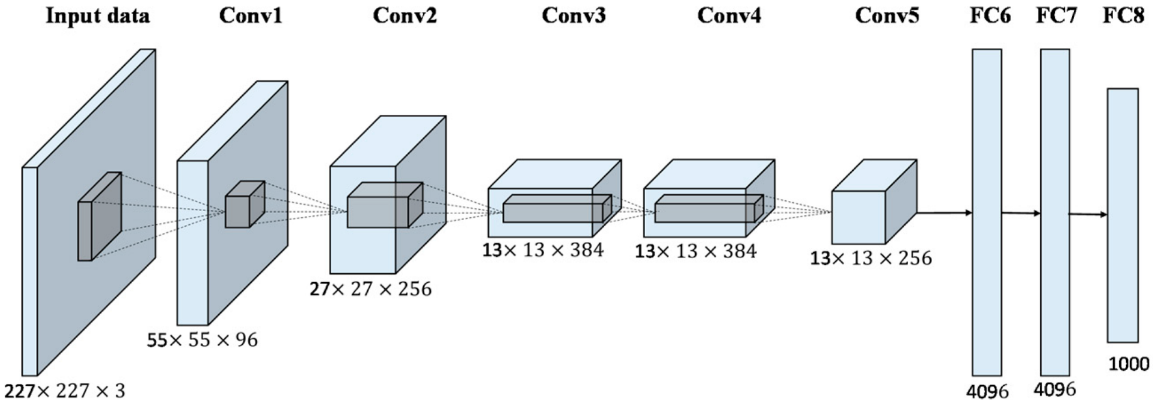


Figure 2.2 An illustration of the AlexNet architecture.

Convolutional Layer

Convolutional layer is essential to CNN, which changes the way how regular neural network operates on input. Taking AlexNet as an example, each layer contains $c_i \times h_i \times w_i$ neurons, where c_i, h_i and w_i are the depth, height, and width of i-th layer (l layers in total). Usually, the height and width in each layer are the same for CNN architectures. We will use k_i to represent both height and width from now on. In each layer, a neuron is only connected to a local region of neurons in the preceding layer, as shown in Fig. 2.2. Each layer has a set of filters, also known as filters, which determines how each neuron connect with the preceding layer. For example, the first convolutional layer in AlexNet has 96 kernels of size 7×7. Thus, each neuron connects to 7 × 7 neurons in the input layer, which are basically image pixels. The kernels slide over the whole image and convolve with each region to produce the feature maps. These feature maps are later fed to the second convolutional layer. Padding and striding are often used in the convolution operation. The spatial size of output feature map is calculated as following:

$$k_i = \frac{k_{i-1} - m + p}{s} + 1, 1 \leq i \leq l \tag{2.1}$$

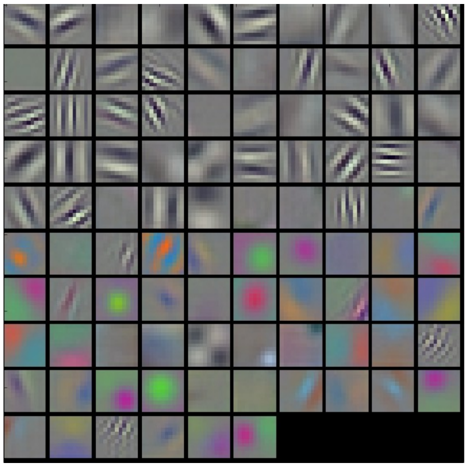


Figure 2.3 Visualization of filters in the first convolutional layer of AlexNet.

where m, p, and s are the kernel size, padding size and striding size, respectively. For each kernel, there is another bias parameter added to the convolution output. The bias parameter is also learnable.

Figure 2.3 shows the visualizations in the first convolutional layer of AlexNet. Based on the visualization, we can see that the filters can detect edges, colors, and other patterns with different orientations. Later study in ZF-Net [142] shows that low layers in CNN extracts low-level features such as edges while high layers extract high-level features such as parts and regions.

After the convolution operation, a nonlinear mapping function is applied to add nonlinearity between input and output of each layer. Such function is usually called *activation function*. The most commonly used nonlinear function is Rectified Linear Unit(ReLU) [58]. It can speed up the convergence of CNN compared to previously used sigmoid function. ReLU function is simply defined as: $f(x) = \max(0, x)$. Later, more advanced activation functions such as ELU [16], Leaky ReLU [89], and PReLU [37] are proposed.

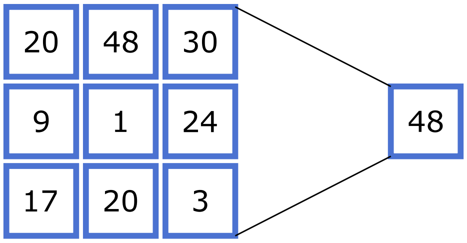


Figure 2.4 An example of pooling operation for a 3 × 3 kernel.

Pooling Layer

Another key component in CNN is the pooling layer. The function of pooling layer is to reduce the spatial size of feature maps. Specifically, the pooling layer scan through the whole feature map region by region, and pools a number for each region to produce the output feature map. Similar to convolutional layer, there are two parameters that determines how the pooling operates: kernel size and stride. The kernel size determines the size of each region, while the stride determines the space between two neighboring regions. Note that these two parameters are not learnable, but predefined. Pooling strategies include max-pooling and average-pooling. Max-pooling produces one number for each region, as shown in Fig. 2.4. In AlexNet [58], kernel size is 3 × 3 and stride is 2, which is also the common choice of other CNN architectures.

Despite such a simple operation, pooling layer has many advantages. First, the computation time is reduced as the spatial sizes are reduced. The computation time reduction is significant considering the number of layers in CNNs can go up to tens, hundreds and even over a thousand. Pooling layer also increases nonlinearity as the pooling operation is nonlinear. The reason we can perform pooling is that neighboring pixels in images are usually similar and not always informative. We can use pooling to reduce some redundancy. Pooling layer also enlarges the receptive field [83], which is essential for CNN to gain more context information for prediction. Finally, pooling layer can improve the translation-invariance of CNN, which is necessary for tasks caring more about global context, such as classification.

Fully-Connected Layer

For a fully-connected (FC) layer, each neuron is connected to all neurons in the preceding layer. FC layers are already used in regular neural networks. Because of the connection pattern, FC layers can obtain information from the global context. As a result, they are usually appended at the end of a network to make predictions. For example, AlexNet has three consecutive FC layers at the end. However, the huge number of parameters in FC layers make them the computation bottleneck of CNN. In view of this, recent CNN architectures, such as ResNet [38], use global average pooling to obtain global context information and only keep one FC layer.

2.2 Long Short-Term Memory Network

Recurrent Neural Network (RNN) is a network with recurrent connections, as shown in Fig. 2.5(a). RNN can be considered as a sequence of regular neural networks by unrolling it, whose parameters are shared. In the equivalent sequential neural networks, the output of neurons in the preceding network is fed to the next network, as shown in Fig. 2.5(b). With the recurrent connection, RNN can propagate information through time. Therefore, RNN can process sequence data, such as audio, video, and text.

Long Short-Term Memory (LSTM) network [45] improves RNN to solve the long-term dependency problem. Specifically, the key component of LSTM is the cell state c_t indexed by time step t, which memorizes previous information. It also has input gate i_t, output gate o_t and forget gate f_t, as shown in Fig. 2.6. Specifically, the

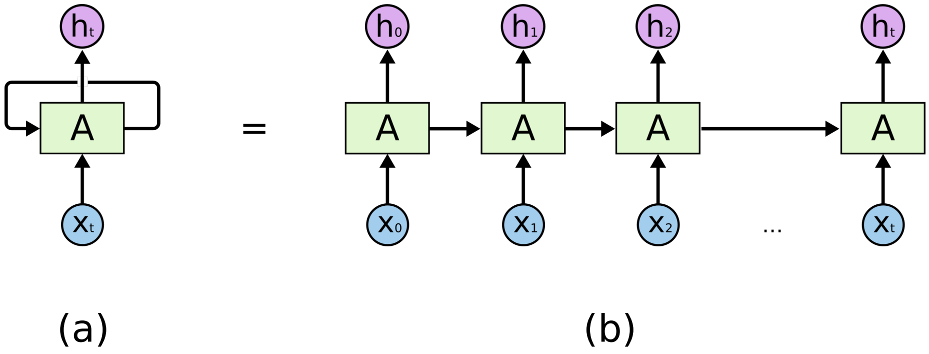


Figure 2.5 An example Recurrent Neural network (RNN) with one input and one output. (a) The original RNN with the recurrent connection. (b) The equivalent unrolled RNN. (Source: https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

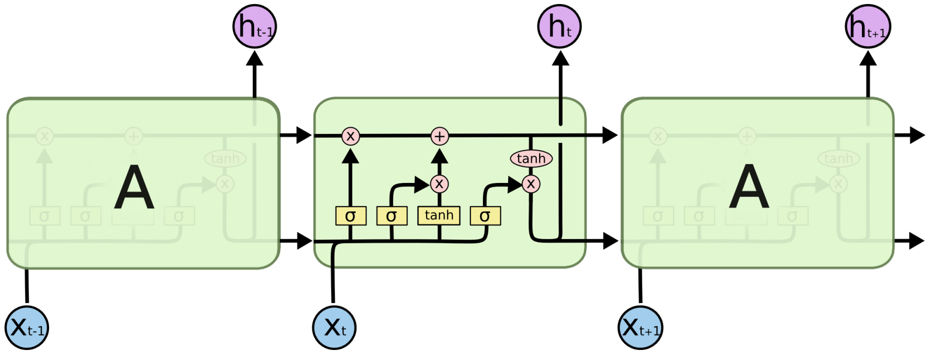


Figure 2.6 An unrolled Long Short-Term Memory (LSTM) network. (Source: https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

network states is updated as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \tag{2.2}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \tag{2.3}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \tag{2.4}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \tag{2.5}$$

$$h_t = o_t \cdot tanh(c_t), \tag{2.6}$$

where σ is the sigmoid function and $\cdot$ denotes element-wise multiplication. $\mathbf{W} =$

$\{W_{xi}, W_{xf}, W_{xo}, W_{xc}, W_{hc}\}$ are the weight parameters and $\mathbf{b} = \{b_i, b_f, b_o, b_c\}$ are the bias parameters. The parameters are learned using back-propagation through time (BPTT). We use the hidden state h_t as the features at each time step, and all the hidden states $\mathbf{h} = \{h_1, h_2, \ldots, h_t, \ldots\}$ as sequence-level features.

2.3 Deep Metric Learning

A metric, or distance function, is a function that defines a distance between each pair of samples. With good metrics, samples of the same class are close to each other, while samples of different classes are far away from each other. Formally, let us define a distance metric as a mapping $D : X \times X \to \mathbb{R}^+$ over a vector space X. A metric have to satisfy the following conditions:

$$D(x_i, x_j) \geq 0 \tag{2.7}$$

$$D(x_i, x_j) = D(x_j, x_i) \tag{2.8}$$

$$D(x_i, x_j) \leq D(x_i, x_k) + D(x_k, x_j) \tag{2.9}$$

$$D(x_i, x_j) = 0 \Leftrightarrow x_i = x_j. \tag{2.10}$$

Note that a metric is called a pseudo-metric is the fourth condition is not satisfied.

Mahalanobis Distance

In general, a metric can be defined by computing the Euclidean distances after applying a linear or non-linear transformation L such that $x \to L(x)$. Formally, we define the metric as:

$$D_L(x_i, x_j) = ||L(x_i - x_j)||_2^2. \tag{2.11}$$

We can expand the above equation into the following formulation:

$$D_L(x_i, x_j) = (x_i - x_j)^T L^T L(x_i - x_j). \tag{2.12}$$

Further, we can express the squared distance with the square matrix $M = L^T L$, which is guaranteed to be positive semi-definite. With the term M, we can denote the squared distance as:

$$D_M(x_i, x_j) = (x_i - x_j)^T M(x_i - x_j), \tag{2.13}$$

which is a pseudo-metric and referred to as Mahalanobis distance. Prior to deep learning, Mahalanobis distance and kernel-based metrics are often used to character the linear or nonlinear relations among samples.

Deep Metric Learning

In the context of deep learning, deep neural networks are used to represent complex nonlinear transformations that are effective in representing the structure of data. To learn the metrics with deep neural networks, various kinds of loss functions were proposed, such as contrastive loss [14], triplet loss [106], and quadruplet loss [12].

Contrastive loss [14] is used to separate samples of different classes by a fixed margin and pull samples of the same class as close as possible. Formally, it is defined as:

$$L_{contrastive} = yd^2 + (1 - y)\max(m - d, 0)^2, y = 0, 1, \tag{2.14}$$

where $d = ||f(x_i) - f(x_j)||_2$ is the L_2 distance between features of sample x_i and x_j. $y = 1$ indicates x_i and x_j are from the same class, $y = 0$ indicate they are from different classes. m is the fixed margin.

Triplet loss [106] relaxes the contrastive loss by comparing a triplet of samples: anchor, positive and negative. The positive sample comes from the same class as the anchor, while the negative sample comes from a different class. As shown in Fig. 2.7, the goal of triplet loss is to learn the ranking between anchor-positive pair and anchor-negative pair. Ideally, the anchor-positive distance should be smaller than

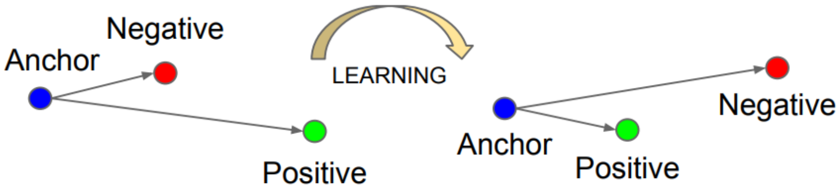


Figure 2.7 The triplet loss [106] optimizes the ranking between anchor-positive pair and anchor-negative pair, such that the anchor-positive distance is smaller than the anchor-negative distance.

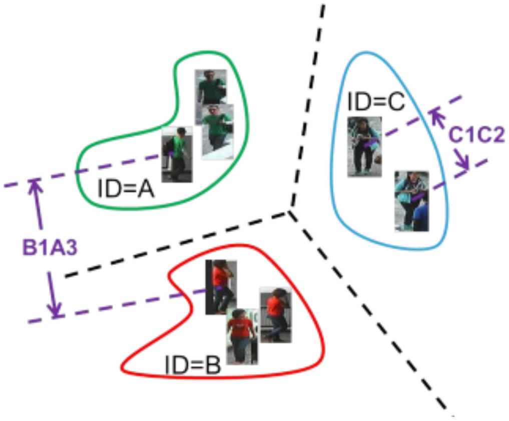


Figure 2.8 Quadruplet loss [12] also considers the relation between anchor-negative pair and a different anchor-positive pair.

the anchor-negative distance. Therefore, the triplet loss is defined as:

$$L_{triplet} = \max(d_i^{(ap)} - d_i^{(an)} + \alpha, 0), \tag{2.15}$$

where $d_i^{(ap)} = ||f(x_i^a) - f(x_i^p)||_2$ and $d_i^{(an)} = ||f(x_i^a) - f(x_i^n)||_2$ are the distances of anchor-positive pair and anchor-negative pair, respectively. α is a margin that separates anchor-positive pair and anchor-negative pair.

Quadruplet loss [12] further improves based on triplet loss, by considering the relation between anchor-negative pair and a different anchor-positive pair. Specifically, it adds another loss term:

$$L_{quadruplet} = \max(d_j^{(ap)} - d_i^{(an)} + \alpha, 0), \tag{2.16}$$

where $d_j^{(ap)} = ||f(x_j^a) - f(x_j^p)||_2$.

Chapter 3

Literature Review

3.1 Person Re-Identification

Person re-identification is widely studied and closely related to cross-view person identification. Different from cross-view person identification, which matches persons across temporally synchronized videos taken by wearable cameras, person re-identification aims to associate persons captured by cameras with non-overlapping views at different times and places. Person re-identification is also related to visible-thermal person re-identification, with the distinction that no image modality difference exists in person re-identification.

Traditional approaches on person re-identification can be divided into two categories: hand-crafted feature extraction [25, 3, 2, 84, 135, 72, 32, 148, 147, 65, 3, 120, 79] and distance metric learning [101, 154, 44, 120, 105, 130, 72, 143, 10].

3.1.1 Hand-crafted Features.

Hand-crafted image features include Scale-Invariant Feature Transform (SIFT) [81], Histogram of Oriented Gradients (HOG) [18], Local Binary Pattern (LBP) [99], Speeded up robust features (SURF) [5], color histograms and so on. These features could be extracted from images using RGB, HSV, YUV, or Lab color space. For person re-identification, various features have been proposed, such as Symmetry-Driven Accumulation of Local Features (SDALF) [25], BiCov [84], HPE signature [6], Haar-based and DCD-based signature [2], Local Descriptors encoded by Fisher Vector (LDFV) [85] and Local Maximally Occurrence (LOMO) [72]. Gray et al. [32] design view-invariant ensemble of localized features (ELF). Karanam et al. [53] propose to learn viewpoint invariant dictionaries from training data to extract image features. Zhao et al. [148, 147] proposed to combine salience with appearance-based features. In [65, 3], attribute-centric and part-based feature representations are proposed to learn adaptive weighted features for each individual to account for the variations across subjects. For video-based person re-identification, spatio-temporal features include Histogram of optical Flow (HoF) [9], HoGHoF [64], HOG3D [56] and 3D-SIFT [107]. Wang et al. [120] use HOG3D to represent fragments of videos. In [79], a new spatio-temporal representation based on the walking cycle extraction is used.

3.1.2 Distance Metric Learning.

Distance metric learning methods include Mahalanobis distance [105], Support Vector Ranking (SVR) [101], Probabilistic Relative Distance Comparison (PRDC) [154], Cross-view Quadratic Discriminant Analysis (XQDA) [72], Relaxed Pairwise Learned Metric (RPLM) [44], Large Margin Nearest Neighbor (LMNN) [122], KISSME [57], MLAPG [73], and Semi-Coupled Dictionary Learning (SCDL) [119], etc. Zheng et al. [154] propose Probabilistic Relative Distance Comparison (PRDC) algorithm by incorporating ranking problem into a probabilistic framework. Mahanabolis or kernel-based metric learning are used in [44, 105, 130]. Liao et al. [72] propose Cross-view Quadratic Discriminant Analysis (XQDA) algorithm which combines metric learning and subspace learning. Zhang et al. [143] propose to match people in a discriminative null space of the training data, where images of the same person collapse into a single point. In [10], spatial constraints are taken into consideration for similarity learning.

3.1.3 Deep Learning Approaches.

With the thriving of Convolutional Neural Networks (CNNs) [66, 58], approaches based on CNNs have achieved great success in person re-identification [70, 149, 1, 128, 13, 93, 118, 125, 133, 15, 43, 12], which can combine feature extraction and distance metric learning in a unified framework. In CNN-based frameworks, network architecture and loss functions guide the feature learning process together, where the loss functions play the role of metric learning. Therefore, to obtain more effective features learned by deep networks, both new architectures and new loss functions are proposed. Since person identification is a problem similar to both image and instance retrieval [153], classification loss such as cross-entropy loss and verification losses such as contrastive loss [14], triplet loss [106] are often adopted for learning features.

3.1.4 Learning global features.

Li et al. [70] propose a Filter Pairing Neural Network (FPNN) to handle misalignment, photometric and geometric transforms, occlusions and background clutter in person re-identification. Ahmed et al. [1] use an improved CNN architecture to extract local features from a pair of images and train the network with binary verification loss. In [150, 15], Siamese network trained with contrastive loss is used for video-based person re-identification. Hermans et al. [43] propose a variant of triplet loss which renders the hard sample mining unnecessary and outperforms the original triplet loss. Chen et al. [12] further propose quadruplet network which uses an additional negative sample beyond the original triplet to train the network better. Xiao et al. [128] present domain-guided pooling to learn deep features from data of multiple domains. Wang et al. [118] jointly learn single-image representation and cross-image representation to exploit their connection for better person re-identification.

3.1.5 Learning local features.

However, the above approaches are learning features in a holistic manner and may missing important local details for effective re-identification. Recently, part-based and attribute-based methods have been proposed to extract effective local features [114, 112, 13, 67, 145, 111, 146, 131]. Varior et al. [114] propose gated Siamese network to emphasize fine local patterns in mid-level features. Su et al. [112] learn mid-level human attributes with a semi-supervised learning framework for robust re-identification across different datasets. In [13, 67], features from global full-body and local body parts are jointly learned and combined for person re-identification. SpindleNet [145] extracts features from local parts in a coarse-to-fine manner and fuse them hierarchically to obtain better representation. Su et al. [111] use human pose as cues when learning features from local parts. Zhao et al. [146] aligns deep features based on body parts for person re-identification. Xu et al. [131] present a joint spatial and temporal attention pooling network to extract features dependent on the input pair for video-based person re-identification.

3.2 Cross-View Person Identification

We first proposed cross-view person identification (CVPI) in temporally synchronized videos taken by wearable cameras in [151]. This problem is related to several exiting branches of researches, which we will describe in detail in the following sections.

3.2.1 Learning spatio-temporal features.

Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network [45] are commonly used to learn features from sequence input such as videos, speeches and sentences. Because RNN and LSTM networks have recurrent connections, which can propagate information along temporal sequence. Recently, RNN and LSTM networks have been applied to video-based person re-identification [94, 35, 115, 133]. McLaughlin et al. [94] combines Convolutional Neural Network (CNN) with RNN to extract spatial features from image frames and temporal features from the whole sequence. A Siamese architecture is used to learn the distance metric. Haque et al. [35] apply Recurrent Attention Model (RAM) to person re-identification for depth-based videos. Varior et al. [115] divide the image into horizontal stripes and feed features of these stripes as a sequence to LSTM network. The output of LSTM is then used for person re-identification. Similarly, Yan et al. [133] use local color histogram and LBP features as input to LSTM network. In this research, we also employ CNN and LSTM networks to extract features for CVPI, but using optical flows as input and also incorporating 3D human skeleton data as a stream of the network.

3.2.2 Data augmentation.

More recently, some researchers focus their attention on producing more data [156, 86, 87] with Generative Adversarial Nets [31]. Such kind of approach serves similarly to data augmentation, except that they produce data with more variations which can be used to train models with more generalization ability. In this research, we develop a triplet network for CVPI. Our proposed triplet network learns to extract view-invariant features from videos with the supervision of two contrastive losses.

3.2.3 Multi-view Learning

Another line of work related to the CVPI problem is the multi-view learning for cross-view person re-identification. However, as described previously, person re-identification differs from CVPI in many aspects. Most importantly, the videos are temporally synchronized in CVPI, but not in person re-identification. The videos in CVPI are taken by wearable cameras such as GoPro, while the videos in person re-identification are usually taken by static cameras. Thus, the view angle of videos in CVPI may constantly change as the camera wearer can freely move, which brings additional challenges.

Existing methods of multi-view learning for person re-identification aims to reduce the variation by learning a dictionary or shared feature space. Specifically, Li et al. [68] propose cross-view dictionary learning model to improve the discriminative and robust representations for person re-identification, with different representation levels, including image-level, horizontal part-level, and patch-level. Yu et al. [140] propose asymmetric metric learning via view-specific projection for each view based on asymmetric clustering on cross-view person images. Person images are projected into a shared feature space with less view-specific biases. In [21], a unified learning framework for multi-view learning and domain adaptation is formulated, which can be applied to cross-view person re-identification problem. The proposed method shares similar spirit with cross-view dictionary learning or asymmetric metric learning in reducing the feature difference caused by view angle difference. However, different from them, we approach the problem by introducing 3D human Mocap data to learn a model for view-invariant feature extraction.

3.2.4 Gait Recognition

Gait recognition also aims to match person with the gait features, which are considered as a kind of behavior biometric feature that is unique for each person. Chrono-gait images [117], gait flow images [62], and gait energy images (GEI) [34] are the common gait representations for recognition/verification. GEI is shown to be the most stable and effective features [51], despite GEI is simply defined as averaged human silhouette along all frames in a video. Both CVPI and gait recognition uses motion-based features for matching. However, different from CVPI, the video pairs for gait recognition are not temporally synchronized.

Cross-view gait recognition approaches include the use of hand-crafted view-invariant features [29, 61] and learning view-invariant projections [92, 60], which normalize gait features from one view to another for matching. Recently, CNNs are used for gait recognition [109, 126]. [109] consider gait recognition as a classification problem and employ a shallow CNN model for feature extraction. Wu et al. [126] propose three different schemes for gait recognition: matching local features at the bottom layer, matching mid-level features at the top layer and matching global features at the top layer. By learning from paired input GEI images, the method [126] learns view-invariant projections for gait recognition. Different from gait recognition methods, which use GEI images with high-level semantics, we train the deep network using optical flows without high-level semantics. We also use 3D human Mocap data to improve the view-invariance by associating optical flows to 3D view-invariant data.

3.2.5 Image/Video Generation

Recently, generative models, especially generative adversarial nets (GANs) [31], have been widely used for generation of natural images or videos, such as Deep Convolutional GAN [103], Stacked GAN [48], Plug and Play Generative Network [96]. GANs are further applied to generate realistic person images [86, 88, 102, 156]. However, the images generated by these approaches cannot be used to learn motion features for CVPI, since they are not videos. For synthetic video generation, Vondrick et al. [116] present a generative model that learns from large amount of unlabeled videos to generate scene dynamics based on static images. Nevertheless, this approach cannot generate person videos with associated 3D human motion capture data, which the proposed triplet network needs to learn view-invariant motion features. In this research, we propose a novel approach to generate sequences of optical flows from 3D human motion capture data with different camera parameters, which are later fed to the proposed triplet network. Different from GANs, the proposed approach to synthesizing optical flows is non-parametric and requires no training process.

3.3 Visible-Thermal Person Re-Identification

Person re-identification is also closely related to visible-thermal person re-identification. However, existing approaches on person re-identification are not suitable for visible-thermal person re-identification, since the networks are used to extract features from visible images only and can not be used directly to handle the domain discrepancy between visible and thermal images. In this research, we propose to learn domain-independent features with class-level supervision. Specifically, we learn a center for each person in each domain (visible and thermal). We pose constraints on the centers, which serves as class-level supervisions.

3.3.1 Visible-Thermal Person Re-Identification

Compared to person re-identification, visible-thermal person re-identification attempts to match persons between images or videos taken by visible cameras and thermal cameras. Existing approaches mainly focus on dealing with the domain discrepancy between visible and thermal images by adopting CNN networks to learn domain-independent features [137, 138, 124]. Wu et al. [124] first propose a deep zero-padding network which can extract features from both visible and thermal images. Ye et al. [137] propose a two-stream CNN network for extracting features in shared feature space from visible and thermal images. A further hierarchical cross-modality metric learning method is applied to enhance the discriminability of the learned features. In [138], a similar two-stream CNN network is trained with identity loss and triplet loss to extract domain-independent features from visible and thermal images, without an additional metric learning step. However, all these approaches are minimizing the domain discrepancy at the sample level, where sample pairs or triplets are used for training. In this research, we propose to learn domain-independent features with class-level supervision, in which class centers instead of samples of each class are constrained.

3.3.2 Domain Adaptation

Domain adaptation is also related to visible-thermal person re-identification, since both problems deal with data from different domains. Some of the existing work utilizes label information to bridge the gap between source and target domains. Daumé et al. [19] propose to train classifiers for features mapped from data of both source and target domain. Kulis et al. [59] propose to learn asymmetric non-linear transformation that maps points from one domain to another using supervised data from both domains. Yao et al. [136] introduce semi-supervised domain adaptation with subspace learning, which explores invariant low-dimensional structures to correct the mismatch between domains and use unlabeled data in target domain to exploit the underlying intrinsic information. Donahue et al. [22] propose to adapt classifiers trained on source domain for target domain, by imposing smoothness constraints on the classifier scores over unlabeled data.

Other existing work on domain adaptation seeks to align or map the subspaces of source and target domains in a unsupervised way. Gong et al. [30] propose a kernel-based method to exploit the intrinsic low-dimensional structures by modeling domain shifts with an infinite number of subspaces, which characterize changes in geometric and statistical properties from source domain to target domain. Lin et al. [75] propose to find the intrinsic low-dimensional subspaces shared by source and target domains. Fernando et al. [26] propose to learn a mapping function which aligns subspaces with target ones for domain adaptation. Ni et al. [98] propose a subspace interpolation method through dictionary learning to link source and target domains. However, these methods are usually adapting the source domain to the target domain, whereas we seek to find a shared feature representation from visible and thermal images for person re-identification.

Recently, unsupervised approaches are proposed for style transfer [158, 80, 139, 110, 55, 159], which can applied to domain adaptation at pixel level.

3.3.3 Class-Level Supervision

Wen et al. [123] are the first to propose center loss to reduce intra-class variation, by minimizing the distance between a sample and its corresponding class center. However, we believe this constraint is too strict and impractical because some difference between samples and class centers should be allowed. Therefore, we propose to relax this constraint by penalizing only samples whose distances to the corresponding class centers are larger than a predefined threshold. Cai et al. [7] introduce inter-class distance between centers of different classes and try to maximize it to improve feature discriminability. He et al. [41] propose triplet-center loss which uses incorporates class centers in the triplet loss. However, these methods are dedicated for classification problems in single domain and cannot be applied to visible-thermal person re-identification directly.

Yu et al. [141] have also observed that both hard and easy samples are useful in learning good features for person re-identification. They propose to weigh all samples within each mini-batch, instead of using only hard samples, to train the network. However, their method only addresses the problem in visible domain, and it weighs samples within a mini-batch instead of a class. In contrast, we use all the samples within each class in each domain to learn a center and impose constraints on the class centers.

Chapter 4

Cross-View Person Identification

4.1 Motivation

Cross-view person identification (CVPI) aims to match persons between temporally synchronized videos taken by wearable cameras. In [151], Zheng et al. propose to estimate the underlying 3D human poses in each video and use them for CVPI. However, this method suffers from inaccurate pose estimation. In this work, we propose to address CVPI by utilizing human motion consistency. For a pair of temporally synchronized videos that capture the same person, the underlying 3D motion must be consistent, i.e., identical and synchronized. Specifically, we extract optical flows to represent the human motion in each video. However, simply examining the similarity of optical flows is not reliable for CVPI because of two issues: 1) optical flows around human contains both the desired human motion and undesired camera motion; 2) optical flows computed from a matching pair of videos can be significantly different due to view angle difference between the two videos. The first issue can be addressed by camera-motion compensation algorithms, such as [74]. In this work, we mainly focus on addressing issue 2) for better CVPI.

Our basic idea is to learn view-invariant motion features by introducing 3D human skeleton data into the training process. Specifically, we propose to learn common features shared by the optical flow sequences and the underlying 3D human skeleton sequence. Since 3D human skeletons are independent of view change, the learned features of optical flow sequences are view-invariant. We propose a Triplet Network (TN) which consists of two flow-stream sub-networks and one skeleton-stream sub-network. After training the proposed TN, the two flow-stream sub-networks can be used to extract view-invariant features from two optical-flow sequences for CVPI.

4.2 Proposed Method

4.2.1 Overview

We propose to utilize the motion consistency based on optical flows in videos for CVPI, since the same person's motion in synchronized videos are consistent, which means that the same body parts should be moving the same way (upward, downward, etc). Specifically, we propose a Triplet Network (TN) to learn view-invariant features from videos' optical flows for CVPI, as shown in Fig. 4.1. The network is built on Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. It consists of two flow-stream sub-networks and one skeleton-stream sub-network. We feed two sequences of optical flows (from different views) and one sequence of 3D human skeletons to the three sub-networks respectively. The flow-stream sub-networks consist of identical CNN and LSTM networks with shared parameters. The skeleton-stream sub-network only contains LSTM networks. A fully-connected layer is added to each of the three streams at the end, which outputs the feature embedding for each stream.

For training, we use contrastive loss between the pair of flow-stream features, as well as between flow-stream features and skeleton-stream features. We assume that the features of optical flows and the features of 3D human skeletons should be similar if they originate from the same person at the same time. Conversely, they should be dissimilar if they are from different persons or the same person at different times. We make the same assumption for two sequences of optical flows from different camera views. By simultaneously minimizing both contrastive losses, we can improve the view-invariance in the optical-flow-based features. Since no existing dataset contains optical flow data and corresponding synchronized 3D human skeleton data, we synthesize optical flows from 3D human skeleton sequences in CMU Mocap

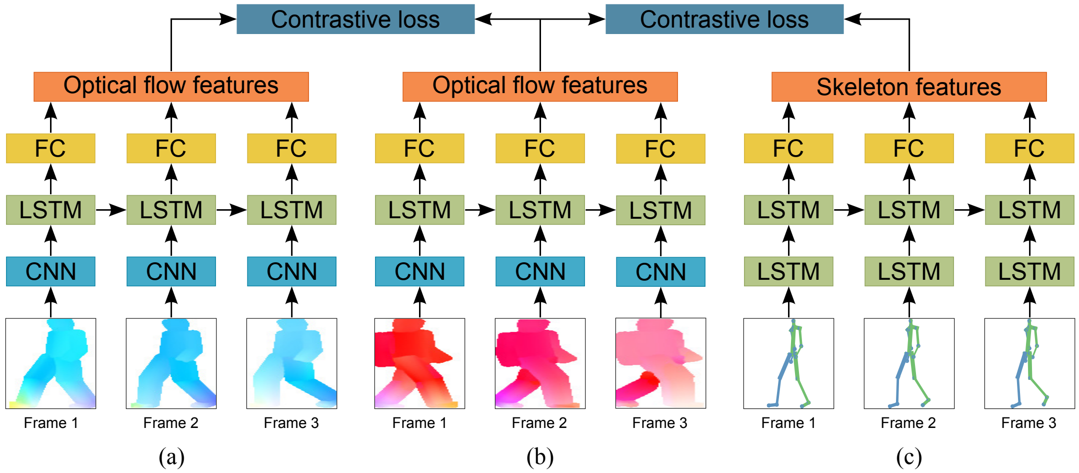


Figure 4.1 An illustration of the proposed Triplet Network. We input two sequences of synthetic optical flows from different camera views to the flow streams (a) and (b) respectively and the underlying 3D human skeleton sequence (from CMU Mocap database) to the skeleton stream (c). Contrastive loss between the pair of flow stream features, as well as contrastive loss between the flow stream features and the skeleton stream features, are used to train the network. We use both contrastive losses for all the frames in each flow sequence pair and each flow-skeleton sequence pair. For clarity, we only show three frames of optical flows and human skeletons.

database[1]. To synthesize optical flows from different view angles, we project the same 3D human skeleton sequence with different camera parameters. The flow-stream subnetworks are further fine-tuned on the training data of person identification. We elaborate on the details of the proposed method in the following sections.

4.2.2 Synthesizing Optical Flows from Mocap Data

To obtain optical flows with corresponding 3D human skeletons, we propose a novel approach to synthesizing optical flow sequences from 3D human skeleton sequences in CMU Mocap database. We first generate synthetic dense trajectories with the method proposed in [33]. Specifically, human body parts between each pair of joints are

[1] https://mocap.cs.cmu.edu

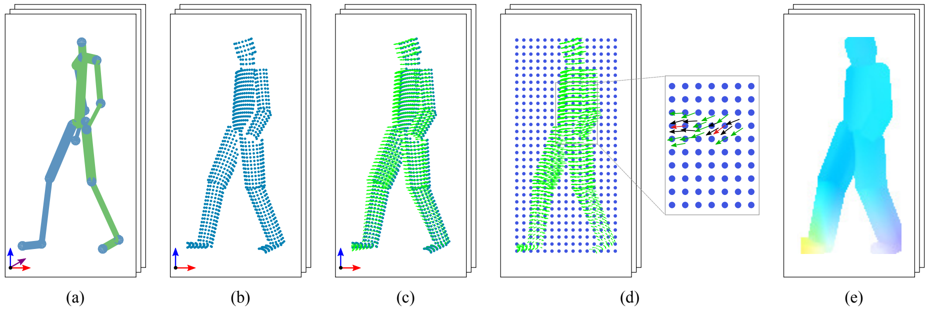


Figure 4.2 An illustration of the process for synthesizing optical flows from 3D human skeleton data. We first take the 3D human skeletons (a) and approximate the human body surface with cylinders. The human body surface is projected to 2D space as shown in (b). Then, we densely sample on the surface to generate dense trajectories between two frames as shown in (c), where dark blue points are sampled body surface points and green arrows are displacement vectors. For each pixel, we use k nearest trajectories to interpolate the flow vector as shown in (d), which generates the resulting flow image (e).

approximated by cylinders and projected to 2D space, as shown in Fig. 4.2(ab). Then, human body surfaces are densely sampled over time to obtain the dense trajectories, as shown in Fig. 4.2(c). Each trajectory consists of L frames of displacement vectors representing the motion of a pixel over $L + 1$ frames. Each displacement vector can be viewed as a flow vector. Therefore, to synthesize optical flows from dense trajectories, we only need $L = 1$ frame of dense trajectories since optical flow is the displacement of each pixel between two neighboring frames. To simulate optical flows viewed from different viewpoints, we synthesize dense trajectories with various camera parameters, using orthographic projection. Specifically, we set the polar angle to $\theta = \pi/2$ assuming that the person's videos are taken by cameras at a similar height. The azimuthal angle ϕ is set to different values: $\phi = \{0, \pi/3, 2\pi/3, \pi, 4\pi/3, 5\pi/3\}$. To make optical flows similar to real-world video optical flows, we need to convert the dense trajectories from the world coordinate system to the image coordinate system. The spatial position of each trajectory will be converted to the image coordinates, and each trajectory will be re-scaled as the flow vector for this pixel, based on the ratio of world coordinate system to image coordinate system. We use the center of the human bounding box as the center of the synthesized optical flow image. However, the location of a trajectory may be on a sub-pixel, leaving many pixels without flow vectors, as shown in Fig. 4.2(d). Therefore, we need to interpolate each pixel in the synthesized flow image. For each pixel, we use its k nearest trajectories to linearly interpolate the flow vector, based on the distances between the k nearest trajectories and this pixel. In the experiments, we set $k = 4$ for synthesizing optical flows. We show two examples of the interpolated flow vectors in Fig. 4.2(d), where the nearest trajectories are shown in black and the synthesized flow vectors are shown in red. For pixels whose nearest trajectory is more than 5 pixels away, we simply set its flow vector as zero-vector. Figure 4.2(e) shows an example of the resulting optical flow image, which is visualized using the flow-color encoding in [4].

4.2.3 Sequence-Level Feature Extraction

Feature Extraction from Optical Flows. As shown in Fig. 4.1, we use the flow-stream sub-networks to extract features from a pair of optical flow sequences and obtain sequence-level features. The parameters of these two flow-stream sub-networks are shared. We use the same network architecture as in Long-term Recurrent Convolutional Networks (LRCN) [23]. The CNN network consists of five convolutional layers, followed by max-pooling layers and dropout layers. One layer of LSTM is followed by a dropout layer to avoid overfitting. We add a fully-connected layer after the LSTM layer to obtain the feature embedding for each optical flow sequence.

Following the preprocessing in LRCN [23], we also formulate the optical flows as flow images and feed them to the flow-stream sub-networks. Specifically, we use the horizontal and vertical part of optical flow as the first two channels of the flow image. The magnitude of optical flow is used as the third channel. The flow images are resized to 227×227 pixels before they are fed to CNN network. Each frame of an input optical flow sequence is fed to the CNN first. The CNN features are then used as input to LSTM, which extracts features for each frame. Through the cell state in LSTM, information in early time steps can be propagated to later time steps. With the final full-connected layer, we obtain the frame-level feature embeddings for an optical flow sequence. All the frame-level features of the sequence are aggregated as the sequence-level features.

Feature Extraction from 3D Human Skeletons. In this paper, we improve the sequence-level feature extraction from optical flows by introducing an additional modality of 3D human skeleton data, which are just the ones used for synthesizing the optical flows in the other two streams. Each sequence of 3D human skeleton is represented by $T \times J \times 3$ coordinates, where T is the number of frames in this sequence and J is the number of human joints. Since a person's action/motion is independent of its spatial position, 3D human skeleton locations are normalized to a person-centric coordinate system. More specifically, we set the hip joint as the origin and rotate the coordinates so that the person is facing along positive $\mathbf{x}$-axis in the first frame of the sequence. The coordinates of the joints are normalized into the range of $[0, 1]$ to remove subject variance. To extract sequence-level features from 3D human skeleton sequences, we use two layers of LSTM as shown in Fig. 4.1. This two-layer LSTM is also referred to as eLSTM following [91], which stands for encoder LSTM. The output of eLSTM is also embedded through a fully-connected layer. Finally, sequence-level features of 3D human skeletons are obtained by aggregating each frame's features.

4.2.4 Network Training

We initialize the parameters of flow-stream sub-networks from LRCN [23] model. The parameters of skeleton-stream sub-network are randomly initialized using the Xavier initialization method [28]. To construct a training sample for the proposed network, we take two optical flow sequences synthesized from the same Mocap sequence as a positive pair and two synthesized from different Mocap sequences as a negative pair, for the flow-stream sub-networks. Each input pair of optical flow sequences are synthesized from two different views, randomly selected from the pre-specified six views. Similarly, we feed an optical flow sequence and its original 3D Mocap skeleton sequence as a positive pair to the second flow-stream sub-network and skeleton-stream sub-network. An optical flow sequence and a different 3D Mocap skeleton sequence, which is not used to generate the optical flow sequence, are used as a negative pair. Because 3D skeleton are independent from the view settings, enforcing the features from 2D optical flows to be similar to the features from 3D Mocap skeletons, which the 2D optical flows are synthesized from, will improve the view-invariance of optical flow features. During the training process, we simultaneously minimize the contrastive loss between the pair of flow stream features, as well as between the flow stream features and the skeleton stream features. The input optical flow sequences to the flow-stream sub-networks are denoted as $\mathbf{v}^{(a)}$ and $\mathbf{v}^{(b)}$ respectively. The input 3D human skeleton sequence is denoted by $\mathbf{P}$. The flow stream feature extraction is represented as a function $F_{\mathbf{v}}(\cdot)$, and the skeleton stream feature extraction is represented as a function $F_{\mathbf{P}}(\cdot)$. The contrastive losses are defined as follows:

$$L_{\mathbf{vv}} = y_1 d_{\mathbf{vv}}^2 + (1 - y_1) max(m_1 - d_{\mathbf{vv}}, 0)^2, \tag{4.1}$$

$$d_{\mathbf{vv}} = \left\| F_{\mathbf{v}}(\mathbf{v}^{(a)}) - F_{\mathbf{v}}(\mathbf{v}^{(b)}) \right\|_2, \tag{4.2}$$

$$L_{\mathbf{vP}} = y_2 d_{\mathbf{vP}}^2 + (1 - y_2) max(m_2 - d_{\mathbf{vP}}, 0)^2, \tag{4.3}$$

$$d_{\mathbf{vP}} = \left\| F_{\mathbf{v}}(\mathbf{v}^{(b)}) - F_{\mathbf{P}}(\mathbf{P}) \right\|_2. \tag{4.4}$$

Here, $d_{\mathbf{vv}}$ and $d_{\mathbf{vP}}$ are Euclidean distances. For positive pairs of optical flow sequences $y_1 = 1$, the features $F_{\mathbf{v}}(\mathbf{v}^{(a)})$ and $F_{\mathbf{v}}(\mathbf{v}^{(b)})$ are encouraged to be similar, while the features are encouraged to be separated by the margin m_1 for negative pairs $y_1 = 0$. Similarly, $F_{\mathbf{v}}(\mathbf{v}^{(b)})$ and $F_{\mathbf{P}}(\mathbf{P})$ are similar for $y_2 = 1$ and separated by a margin m_2 for $y_2 = 0$. Suppose we have a particular optical flow sequence $\mathbf{v}_i^{(b)}$ synthesized from $\mathbf{P}_i$, where i is the sample index. Positive pairs of optical flows are obtained by synthesizing optical flows from $\mathbf{P}_i$ with different camera parameters. Negative pairs are obtained by randomly selecting the optical flows from other 3D skeleton sequences regardless of camera parameters. Similarly, we use the skeleton sequence $\mathbf{P}_i$ and optical flow sequence $\mathbf{v}_i^{(b)}$ to form a positive pair. We randomly select from human skeleton sequences in the Mocap database other than $\mathbf{P}_i$ to form a negative pair with $\mathbf{v}_i^{(b)}$. We do not use contrastive loss between $F_{\mathbf{v}}(\mathbf{v}^{(a)})$ and $F_{\mathbf{P}}(\mathbf{P})$ because it is implied in the two considered contrastive losses: the one between $F_{\mathbf{v}}(\mathbf{v}^{(b)})$ and $F_{\mathbf{P}}(\mathbf{P})$ and the one between $F_{\mathbf{v}}(\mathbf{v}^{(a)})$ and $F_{\mathbf{v}}(\mathbf{v}^{(b)})$.

Due to the difference between synthetic optical flows and optical flows of real-world videos, we further fine-tune the network on the training data of each CVPI video dataset. More specifically, we remove the skeleton stream from the network since there is no 3D human skeleton information for the training data of videos in CVPI dataset. We then sample positive pairs and negative pairs of optical flow sequences similar to sampling synthetic optical flow sequences. The fine-tuning is accomplished by minimizing contrastive loss between outputs of these two flow-stream sub-networks. For videos in CVPI datasets, the camera motion introduced by the movement of camera wearers can lead to some errors in optical flow computation. We use a simple motion compensation (MC) technique as in [74] to address this issue: we compute the average optical flow outside the human bounding box for each frame as the camera motion, which is then subtracted from the optical flow inside the human bounding box.

4.2.5 Cross-View Person Identification

After training the proposed network, we use the flow-stream sub-network to extract sequence-level features from optical flows in videos for person identification. The features of each video are represented by an $n \times p$ dimensional vector, where n is

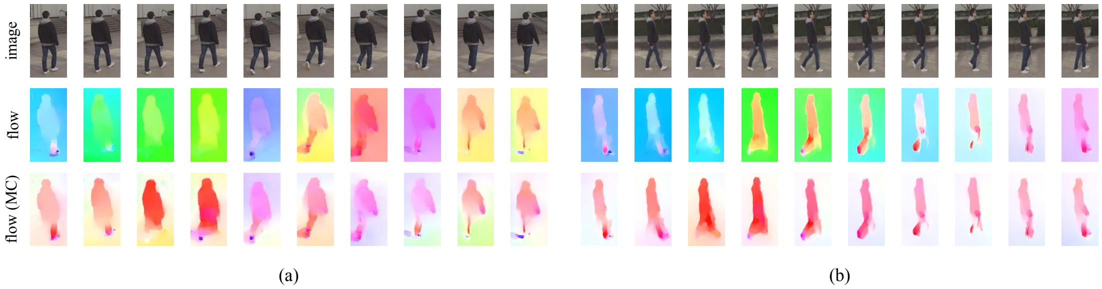


Figure 4.3 An example of our newly collected SYN dataset. From top to bottom are the images, the optical flows and the optical flows after motion compensation. (a) and (b) represent the images/optical flows from two cameras, respectively.

the number of frames in the video and p is the number of outputs in the final fully-connected layer. Features of the gallery videos are extracted and stored beforehand. For each probe video, we extract its features and compute the similarities between this video to all videos in the gallery set. We use the inverse of the Euclidean distance to measure similarity between videos in a pair. Suppose $F_i^{(a)}$ and $F_j^{(b)}$ represent the features of a video from camera a and a video from camera b respectively, the similarity score between them is defined as follows:

$$S_{i,j} = \frac{1}{\left\| F_i^{(a)} - F_j^{(b)} \right\|_2}. \tag{4.5}$$

The similarity score is further normalized to the range [0,1] as follows:

$$Score_{i,j}^{T} = \frac{S_{i,j}}{\max_{i,j} S_{i,j}} \tag{4.6}$$

4.3 Experiments

In this section we evaluate the proposed method on two datasets in [151], SEQ 1 and SEQ 2, as well as our newly collected dataset, SYN. We compare the proposed method with three state-of-the-art methods and analyze the effectiveness of the proposed method.

SEQ 1 and SEQ 2 contain 114 and 88 pairs of synchronized videos from two different cameras views respectively. The videos are taken by GoPro cameras which are mounted on the wearers' heads. The videos are taken in a football field where multiple pedestrians are present. There are totally 6 subjects walking around and recorded by the cameras. Each subject walks for 4 to 26 times and each video has 120 frames. The same person's videos taken at different times are considered to be non-matching pairs, since their movements are not synchronized and consistent with each other. In some videos, the subject is occluded by other pedestrians. All subjects in SEQ 1 and SEQ 2 are wearing white T-shirts and blue jeans.

We also collect a new dataset, which contains 208 pairs of synchronized videos from two camera views. We refer to this dataset as SYN. Compared to SEQ 1 and SEQ 2, SYN has more video pairs which can provide more reliable evaluation. Also, SYN dataset contains less camera motion which can facilitate the analysis of view-invariant feature learning with less impact by camera motion. The videos are captured by GoPro cameras in an outdoor environment near a building. There are totally 14 subjects, each of whom walks for 14 to 15 times. Each video has 120 frames. All subjects in this dataset are wearing dark jackets. In this dataset, there are no other pedestrians crossing through in each video. Each person in each video is cropped manually as a detection to obtain fairly tight bounding boxes for experiments. Figure 4.3 shows an example pair of synchronized videos of the same person captured by two cameras, together with optical flows visualized as color images using the flow-color encoding in [76]. The optical flows after motion compensation using the method in [74] is shown in the bottom of Fig. 4.3, denoted as “flow (MC)". We can see that the original optical flows are mixed with undesired camera motion in the background, while the motion-compensated optical flows have much less background motion, which will improve the motion-based feature extraction.

For evaluation, we follow the generally adopted protocol and split each dataset into two subsets of equal size, i.e., one for training and one for testing. We use Cumulative Matching Characteristics (CMC) as the metric for evaluation. Videos from one camera are used as probe set and videos from the other camera are used as gallery set. For each probe video, we compute the similarity score of the true matching video and find its rank in all videos of gallery set. To obtain more stable results, we repeat the process over 10 random dataset splits and report the average performance.

Implementation Details. We use Caffe [52] to implement the proposed Triplet Network (TN). Specifically, we fine-tuned the flow-stream sub-networks whose parameters are initialized from LRCN [23] model. The LRCN model is trained on optical flow sequences of UCF-101 action recognition dataset. The detailed network architecture and configuration of the flow-stream sub-network are shown in Fig. 4.4. We empirically set the last fully-connected layer to 512 units, which outputs the feature embedding for each video sequence. For skeleton-stream sub-network, we use two LSTM layers to extract the features, where the first layer contains 1,024 units and the second layer contains 512 units. This LSTM is also followed by a fully-connected layer of 512 units. We use 16 time steps in LSTM layers for both flow streams and skeleton stream, which corresponds to 16 frames. To avoid gradient vanishing and gradient explosion problems in Back-Propagation Through Time (BPTT), we clip the gradients to 15 if they are larger than this value. Both the margins of the two contrastive losses are set to 1.

For training the network, we synthesize 9,654 optical flow sequences from CMU Mocap database. These optical flow sequences are synthesized with 6 sets of different camera parameters as described in Section 4.2.3. Each frame of 3D human skeleton is described by $18 \times 3 = 54$ coordinates, where 18 is the number of joints used. In this work, we use equal length of optical flow sequences and 3D human skeleton

conv1	**conv2**	**conv3**	**conv4**	**conv5**	**fc6**	**lstm7**	**fc8**
7x7x96	5x5x384	3x3x512	3x3x512	3x3x384	4096	256	512
stride 2	stride 2	stride 1	stride 1	stride 1	dropout	dropout	
norm.	norm.			norm.			
pool 3x3	pool 3x3			pool 3x3			

Figure 4.4 The architecture of the flow-stream sub-network.

sequences as input for the proposed network. Specifically, we use 112 frames of 3D human skeletons and optical flows in the form of triplets. The 3D human skeleton sequences in the CMU Mocap database have various length ranging from 2 frames to over 5,000 frames. We segment long sequences into equal-length 112-frame sequences and discard sequences shorter than 112 frames.

Figure 4.5 shows an example of optical flows synthesized with different camera camera parameters ($\phi = \{0, \pi/3, 2\pi/3, \pi, 4\pi/3, 5\pi/3\}$) from the 3D human skeleton sequence. Obviously, the synthesized optical flows are not as real as optical flows extracted from real-world videos. The background is clean with no camera motion, and the shape of foreground motion is fairly rough because of the cylinder approximation for human body parts. Therefore, we need to further fine-tune the flow-stream sub-networks on real-world optical flow data.

In the experiments, we train the network on an NVIDIA GTX 1070 GPU. The network is trained on Mocap synthetic optical flow and skeleton dataset for 2 epochs, which takes about one day. We only train for 2 epochs because many Mocap sequences are similar and the loss can converge after 2 epochs. We further fine-tune the flow-stream sub-network on training data of each video dataset for 400 epochs. This takes about 4 hours for a dataset of 200 image sequences. For testing, feature extraction and similarity computation between each pair of videos costs less than one second.

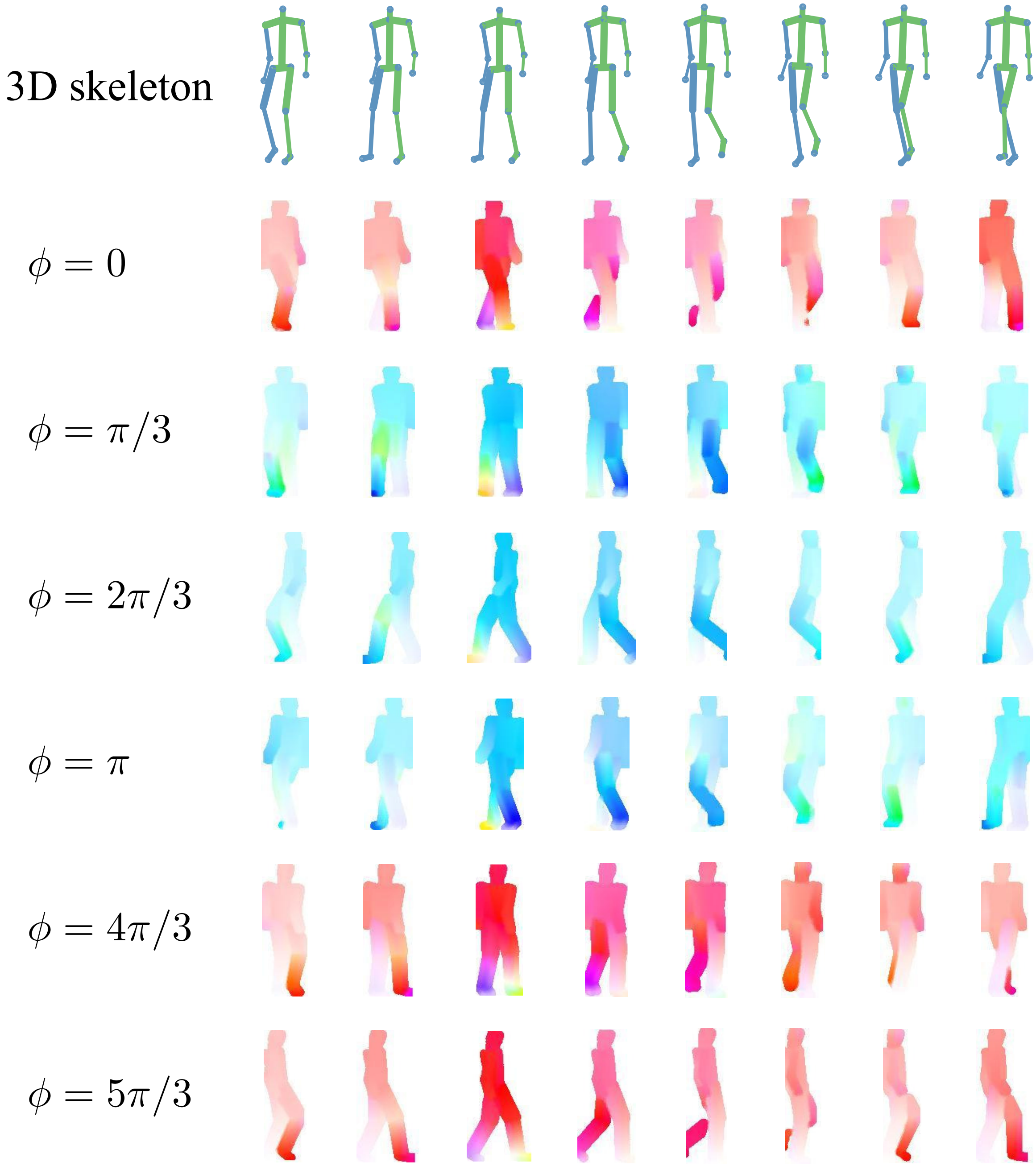


Figure 4.5 Example of optical flows synthesized from the same 3D MoCap skeleton with different camera parameters.

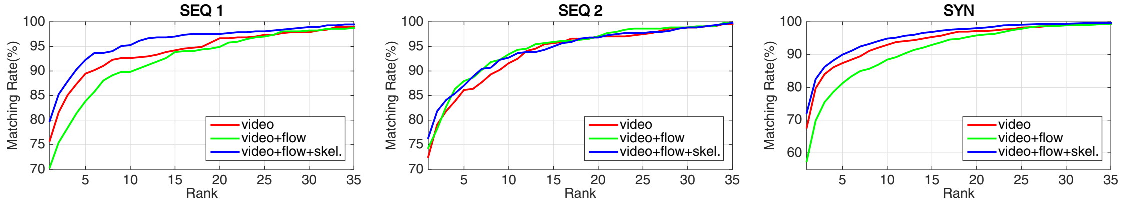


Figure 4.6 Comparison of the resulting CMC using different training data.

4.3.1 Effectiveness of Synthetic Data for Training

As mentioned above, the network training consists of 1) using synthesized optical flow-data and their 3D skeleton data to train the network, and 2) using real training videos to fine-tune the flow-stream sub-networks. To evaluate the effectiveness of using synthetic optical flow dataset and 3D human skeleton dataset in training, we compare three variants of the proposed method: "*video+flow+skel.*" which runs both 1) and 2); "*video+flow*" which only uses synthesized optical flow data (without using 3D skeleton data) to training the flow-stream sub-networks, followed by running 2); and "*video*" which runs only 2). The evaluation is conducted on all three datasets. The results are shown in Fig. 4.6. We can see that training with synthetic optical flows and 3D human skeleton first can improve the performance on all the datasets. However, using only synthetic optical flow dataset for the first-step training does not improve the performance. This is mainly caused by the difference between synthetic optical flow and real-world video optical flow. Overall, adding 3D human skeleton can benefit the model and improve the matching rates. This proves the effectiveness of using 3D human skeleton data for CVPI.

To better understand the effect of skeleton data during training, we visualize in Fig. 4.7 the feature maps of the same person under different cameras using the proposed network with and without incorporating skeleton data for training. Specifically, we extract the features from the last layer of the proposed network for the

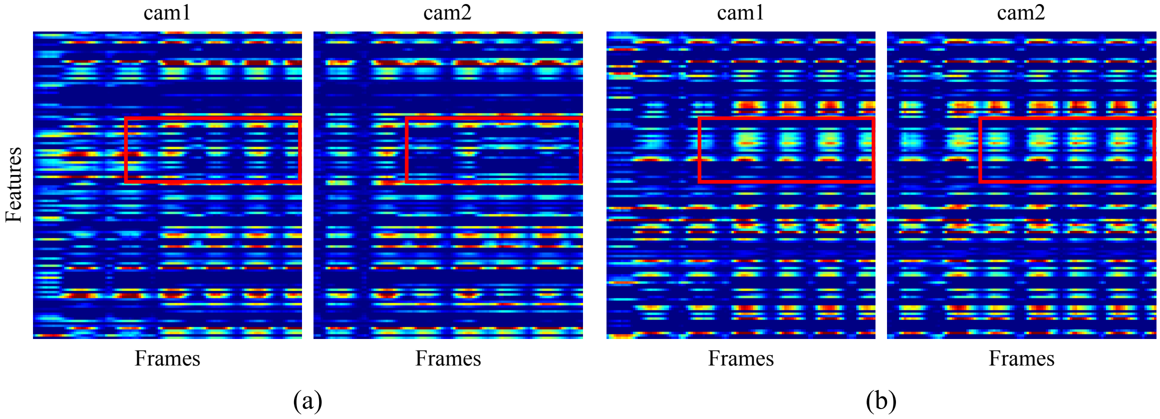


Figure 4.7 Visualization of the learned features of the same person under different cameras: (a) without using 3D skeleton data in training, (b) incorporating 3D skeleton data in training.

same person. In this figure, the horizontal axis represents the frame number and the vertical axis represents the dimension of features. We can see that the highlighted areas of the two feature maps in Fig. 4.7(b) are more similar than that of the two feature maps in Fig. 4.7(a). This indicates that we can extract features with better view invariance by incorporating the 3D skeleton data in training. We also show an example of the features of different person under different cameras using the proposed network with and without incorporating skeleton data for training in Fig. 4.8. Features from the last layer of the proposed network is visualized the same way as above. The areas highlighted in the red box of the two features maps in Fig. 4.8(a) are similar to each other, while the areas of the same part in Fig. 4.8(b) are not. By incorporating 3D skeleton data in training, the network can extract better features that can discriminate different persons.

4.3.2 Effectiveness of Motion Compensation

To alleviate the adverse effect caused by the motion of wearable cameras, we subtract the average optical flow outside the human bounding box from the optical flow

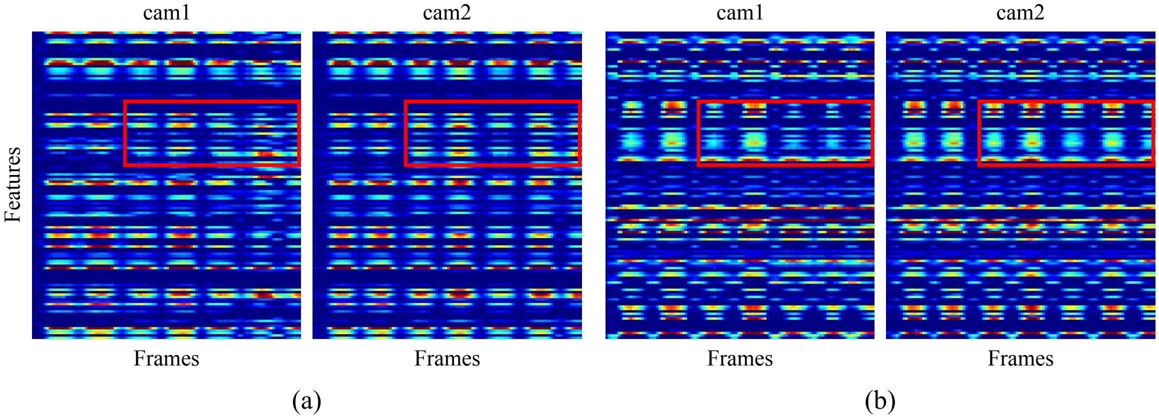


Figure 4.8 Visualization of the learned features of different persons under different cameras: (a) without using 3D skeleton data in training, (b) incorporating 3D skeleton data in training.

inside human bounding box, as in [74]. This technique is simple yet effective. To prove this, we compare the performance of the models fine-tuned on training data with and without motion compensation when computing optical flows. As shown in Table 4.1, the matching rates at all ranks for all three datasets are improved with motion compensation. The rank-1 matching rate is improved by a large margin of more than 7%, which means there is a significant amount of camera motion in this dataset and the motion compensation method applied is effective.

Table 4.1 Comparison of matching rates (%) of the proposed method with and without motion compensation (MC).

Rank	1	5	10	20
SEQ 1 w/o MC	72.28	90.88	94.04	**98.25**
SEQ 1 w MC	**79.82**	**92.28**	**95.26**	97.54
SEQ 2 w/o MC	75.68	86.82	92.05	**97.05**
SEQ 2 w MC	**76.36**	**87.05**	**92.73**	96.82
SYN w/o MC	71.06	88.17	92.69	96.63
SYN w MC	**72.21**	**90.00**	**94.90**	**98.08**

4.3.3 Effect of Video Length

We evaluate how the length of videos affects the final matching rates. Specifically, for each dataset, we evaluate the rank-1 matching rates given K frames in each video. We set K from 10 to 100 with the step size of 10. Since each video contains more than 100 frames, we randomly select K consecutive frames in each video pair for similarity calculation. Specifically, for a video with K frames, we extract K sets of frame-level features using the flow-stream sub-network, consisting of CNN and LSTM networks. The frame-level features of these K frames are aggregated as the sequence-level features, which are used to compute the similarity to another video for cross-view person identification. Note that, by extracting frame-level features and then aggregating over all the frames, the proposed network does not need to be retrained when video length changes. The results are shown in Fig. 4.9. We can see that, using more frames from the video improves the matching rates. Notice that the matching rates of SYN dataset are lower than those of the other two datasets, especially when very few number of synchronized frames are available. This is because SYN dataset contains more subjects and it is more difficult to identify the same person from a larger set of subjects. We can also see that the matching rates do not increase much after the number of frames reaches 90.

Table 4.2 Comparison of the proposed method with state-of-the-art methods on SEQ 1, SEQ 2 and SYN dataset in terms of Rank CMC (%).

Dataset	SEQ 1				SEQ 2				SYN			
CMC Rank	1	5	10	20	1	5	10	20	1	5	10	20
DVR [120]	16.14	50.53	66.84	82.83	11.14	34.09	53.64	77.05	12.69	41.83	59.04	75.87
3DHPE [151]	16.14	50.70	67.02	81.93	17.95	51.82	71.14	89.55	8.65	35.67	50.48	64.52
RFA [133]	68.42	96.84	98.25	**99.30**	69.77	**96.36**	98.41	**99.32**	56.83	92.40	97.02	98.85
GEINet [109]	10.18	32.11	50.53	69.30	4.55	22.95	38.86	59.55	15.29	36.44	48.75	63.17
LB [126]	22.11	60.35	73.51	85.96	19.55	45.68	63.64	80.91	8.37	28.56	40.87	58.27
Ours	79.82	92.28	95.26	97.54	76.36	87.05	92.73	96.82	72.21	90.00	94.90	98.08
Ours+RFA λ:2	79.82	97.19	**98.42**	**99.30**	79.77	95.91	**98.64**	**99.32**	70.67	96.73	98.56	99.71
Ours+RFA λ:1	85.09	97.02	98.25	**99.30**	**82.05**	95.68	97.73	**99.32**	76.92	97.31	**99.33**	**100**
Ours+RFA λ:0.5	**87.02**	**97.37**	97.89	98.95	**82.05**	94.32	96.59	**99.32**	**82.12**	**98.37**	**99.33**	**100**

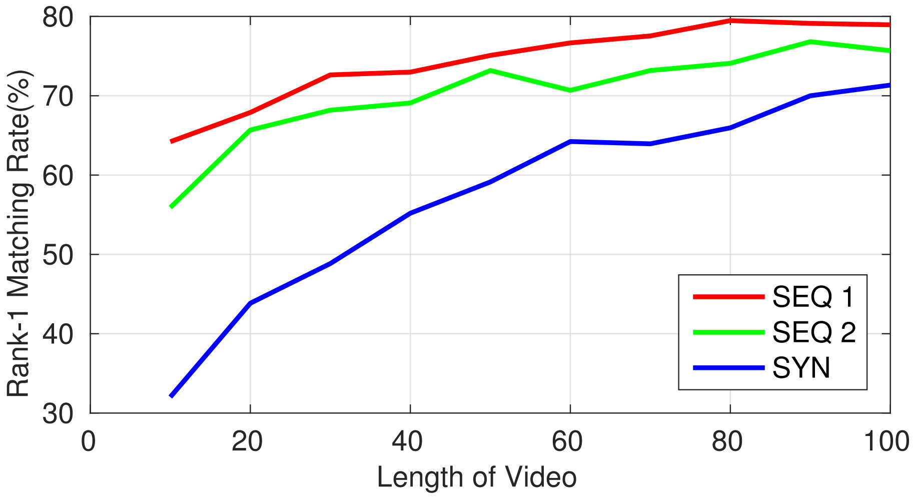


Figure 4.9 Rank-1 matching rates (%) on videos of different lengths.

4.3.4 Comparison to the State of the Art

We compare the proposed method with several state-of-the-art methods, including 3D pose estimation for person identification (3DHPE) [151], Discriminative Video Ranking (DVR) [120] and Recurrent Feature Aggregation (RFA) [133]. We also compare with two state-of-the-art gait recognition methods: GEINet [109] and matching Local features at Bottom layer (LB) [126]. For fair comparison, all the results are obtained by training and testing using the same dataset split. Since 3DHPE method in [151] is unsupervised, we only use the test data to evaluate this method. RFA is an appearance-based method, which takes color and Local Binary Patterns (LBP) features as input. We resize each frame to the size of 128 × 64 pixels and extract the color and LBP features to feed to RFA. GEINet and LB are gait-based methods, which require Gait Energy Image (GEI) [34] as the input. GEI is the averaged silhouette of a walking person. We also resize each frame to the size of 128 × 64 pixels and use the state-of-the-art segmentation method Mask R-CNN [36] to extract the human silhouette at each frame of each video. The results are shown in Table 4.2.

We can see that the gait recognition methods perform much worse than the proposed method. This is because the gait recognition methods, GEINet and LB, average human silhouettes along a video sequence with good alignment. For videos captured by static cameras in existing gait recognition datasets, human silhouettes can be easily extracted by using background subtraction methods, since there is no camera motion and the background is clean. However, due to the fact that the videos in the CVPI datasets are captured by wearable cameras, we can not extract human silhouettes using a simple background subtraction method. The problem may get even worse when occlusion occurs in some frames. Actually, we found that even state-of-the-art human segmentation methods such as Mask R-CNN can not extract human silhouette well. The proposed method has much higher CMC performance than 3DHPE and DVR. This is because we fine-tuned the proposed model from LRCN [23] model, which has been trained on a large amount of data.

We further combine the proposed TN's output similarity scores with the output scores of RFA as follows:

$$Score = Score^T + \lambda Score^R, \tag{4.7}$$

where $Score$ is the combined similarity, $Score^T$ is the normalized similarity score computed by the proposed TN method, as defined in Eq. (4.6), and $Score^R$ is the normalized similarity score computed by the comparison RFA method. By setting $\lambda = \{2, 1, 0.5\}$, the results are shown in the bottom three rows of Table 4.2. Clearly, combining RFA's output with TN's output can improve the CMC performance significantly and the weight $\lambda = 0.5$ leads to the best performance. Specifically, the combination achieves about 7%, 6%, and 10% improvement of rank-1 matching rate on SEQ 1, SEQ 2, and SYN datasets, respectively. This shows that appearance features and motion features well complement each other in the proposed CVPI task.

We further conduct an experiment to fuse their output similarity scores with more different values of λ. More specifically, we fuse the output similarity scores of RFA and TN on the SYN dataset with $\lambda \in [0.1, 10]$. The resulting matching rates are shown in Fig. 4.10, where the λ values are shown in log scale. It is obvious that $\lambda < 1$ leads to higher matching rates, which indicates the relatively more important role of the motion features when combined with the appearance features for CVPI. Specifically, the rank-1 matching rate reaches the maximum value of 83.46% when $\lambda = 0.3$.

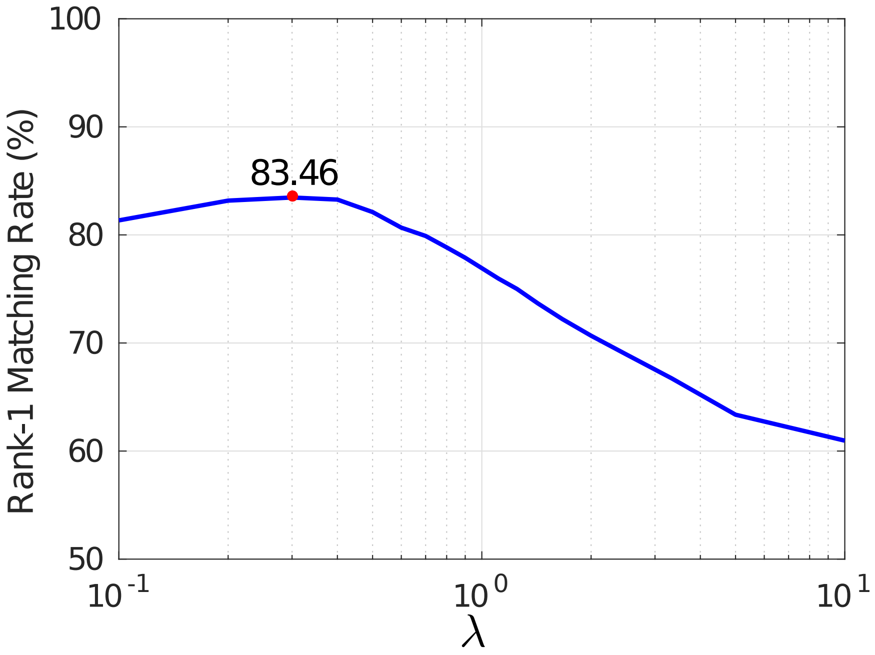


Figure 4.10 Rank-1 matching rates (%) using the combined similarity scores with different weights. The best rank-1 matching rate 83.46% is achieved when $\lambda = 0.3$.

As a purely appearance-based method, distinguishing two walking sequences of the same subject taken at different times will be difficult for RFA, and thus comparing it with the proposed method on such sequences is not completely fair. We conduct an additional experiment for a more fair comparison. For each subject, all the walking sequences are labeled with the same identity. As SEQ 1, SEQ 2 and SYN datasets have 6, 6 and 14 subjects, respectively, comparing on these small number of subjects may not be very informative. We merge the SEQ 1, SEQ 2 and SYN datasets into one dataset, with 26 subjects in total. We randomly divide the datasets into training and testing subsets equally by subjects, 13 subjects for training and 13 subjects for testing. To obtain more stable results, we repeat the split for 10 times and compute the average performance. More specifically, for each of the 10 splits during testing, we select one walking sequence for each subject to form a testing set of 13 paired sequences for evaluation, where each identity only appears in one video pair. As the subject may walk 4 to 26 times, we use the maximum times a subject walks to repeat this selection process. Finally, we obtained 284 such testing sets for evaluation. We also combine the proposed method by fusing the similarity scores as in Eq. (4.7) with $\lambda = \{2, 1, 0.5\}$. We also compare with GEINet [109] and LB [126]. The results are shown in Table 4.3. We can see that the two gait-based methods perform poorly. This is because the GEI images are averaged human silhouettes (masks), which are not accurately segmented due to camera motion. The proposed method has slight lower matching rates than RFA (less than 3% in rank-1 CMC). Nevertheless, combining the proposed method and RFA achieves better results, which indicates that the proposed motion features well complement the widely used appearance features. Specifically, $\lambda = 1.0$ achieves the rank-1 matching rate of 65.82%.

Table 4.3 Comparison of the RFA with the proposed method on the merged dataset in terms of Rank CMC (%).

Rank	1	5	10	13
GEINet [109]	31.18	73.48	90.25	100
LB [126]	26.98	74.38	93.93	100
RFA	56.50	96.40	**100**	**100**
Proposed	53.95	92.36	99.57	**100**
Proposed+RFA ($\lambda = 2$)	65.11	**96.83**	**100**	**100**
Proposed+RFA ($\lambda = 1$)	**65.82**	96.32	**100**	**100**
Proposed+RFA ($\lambda = 0.5$)	65.63	95.67	**100**	**100**

4.3.5 Effect of Synchronization Error

In this section, we study the influence of synchronization error on the resulting matching rates in terms of CMC. Specifically, for each pair of synchronized videos, we shift

Table 4.4 Effect of synchronization error on the rank-1 matching rates in terms of Rank CMC(%). The rank-1 matching rates without synchronization error ($\Delta t = 0$) for SEQ 1, SEQ 2 and SYN are 79.82%, 76.36% and 72.21%, respectively.

Δt	1	2	3	4	5
SEQ 1	78.16	72.64	64.21	55.88	49.39
SEQ 2	75.91	75.57	74.66	73.30	72.05
SYN	68.08	56.11	39.23	27.07	18.95
Δt	6	7	8	9	10
SEQ 1	43.42	39.04	36.67	33.60	31.93
SEQ 2	70.11	68.64	66.48	63.75	60.69
SYN	13.75	10.39	8.22	7.21	7.02

either one in the pair by $\Delta t = 1, 2, \ldots, 10$ frames. The extracted features based on the shifted video pairs are then used to compute the pair-wise similarity scores to obtain the rank-1 matching rates for all three datasets. The results are shown in Table 4.4. Note that, the rank-1 matching rate of SEQ 2 is always the highest, while SYN is always the lowest. This is simply because SYN dataset has the largest number of subjects in the gallery set while SEQ 2 dataset has the smallest. We can see from Table 4.4 that, the rank-1 matching rates drop as the synchronization error increases. When the synchronization error reaches 10 frames, the rank-1 matching rates are 31.93%, 60.69% and 7.02% for SEQ 1, SEQ 2 and SYN datasets, respectively. It indicates that the synchronization error is harmful to the performance.

4.3.6 Qualitative Results

In this section we discuss several correct and incorrect matching examples, as shown in Fig. 4.11. Examples in three columns are from SEQ 1, SEQ 2 and SYN datasets respectively. Correct matching examples are shown in the bottom two rows, while incorrect ones are shown in the top two rows. Probe sequences are in the first and third rows. For the correct matching examples, all pairs of persons have consistent movement. For the failure case of SEQ 1 example, these two persons are walking very consistently though they are not the same person. It is also easy to confuse motion

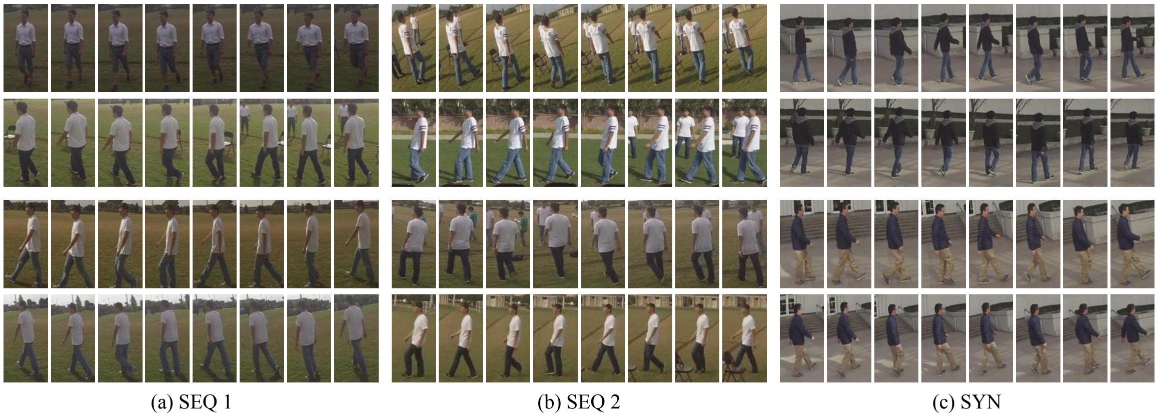


Figure 4.11 Matching examples. (a), (b), (c) are from SEQ 1, SEQ 2 and SYN datasets respectively. Top two rows show the failure matching examples, while bottom two rows correspond to successful matching examples.

of the left leg with the right leg because of the ambiguity in projecting 3D human skeleton onto 2D image planes. As shown in the failure case of SEQ 2 example, these two persons have very similar motions. While the person in probe video moves his left leg, the person in the matched gallery video moves his right leg. The incorrect matching example from SYN dataset can be caused by the variance of features in the true matching gallery video. Camera motion is another factor that can corrupt the proposed method, which relies on motion information.

4.3.7 Cross-Dataset Testing

We also investigate the proposed method's generality by exploring the cross-dataset performance. Specifically, we first train the proposed TN network on synthetic Mocap optical flow and skeleton dataset, using LRCN model for initialization. Then, we fine-tune this model on SEQ 1 dataset and test on SEQ 2 dataset. To better understand the performance of the proposed method, we compare to TN model without Mocap data for training and RFA [133]. The results are shown in Table 4.5.

Clearly, the cross-dataset performance is worse than the within-dataset perfor-

Table 4.5 Cross-dataset performance in terms of Rank CMC(%).

Rank	1	5	10	20
TN w Mocap	**11.36**	**25.00**	**38.64**	**63.64**
TN w/o Mocap	4.55	11.36	27.27	50.00
RFA [133]	5.00	14.77	32.50	61.14

mance due to the dataset bias. Specifically, SEQ 1 and SEQ 2 datasets are captured at the same scene, in a football field. However, camera wearers are different and may move differently, which results in different camera motion patterns in these two datasets. Another reason is that camera wearer may be standing close to or far away from the subject. Then, the person will have different sizes in the video. Also, camera wearers can have different heights and may show different angles when viewing a subject, as shown in Fig. 4.12. Training the proposed triplet network flow-streams on one dataset may not generalize very well on the other dataset. As the synthetic optical flows are synthesized without considering these differences, training the proposed network with 3D Mocap data and synthetic optical flows could not well address the generalization problem between these two real datasets. Nevertheless, we can see that the proposed method improves the matching rates with additional 3D Mocap data for training. This proves the proposed method with 3D Mocap data can improve view-invariant feature learning and have better generality.

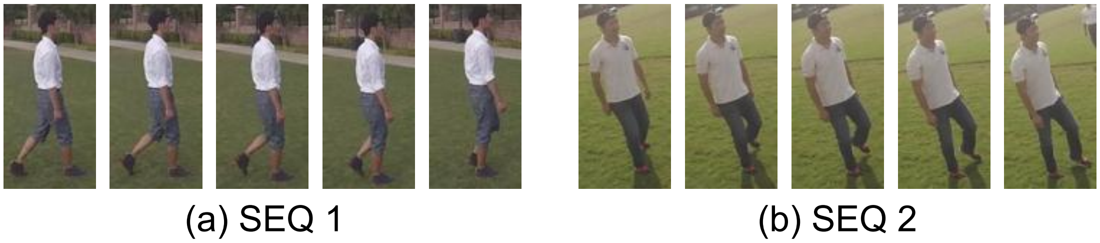


Figure 4.12 An example of videos from SEQ 1 and SEQ 2 datasets.

4.4 Chapter Summary

In this chapter, we studied the Cross-View Person Identification (CVPI) problem by identifying the same person in temporally synchronized videos taken by different wearable cameras. We proposed a Triplet Network (TN) for CVPI using only motion information. We also proposed to synthesize optical flow dataset from CMU Mocap database for training the network, where the underlying 3D human skeleton data are used as a third stream of the proposed network, which we found that can help learn more view-invariant features for person identification. Experiments on three datasets showed that, using only motion information, the proposed method can achieve comparable results with the state-of-the-art methods. Further combination of the proposed method with an appearance-based method achieves the new state-of-the-art performance. From the experimental results, we can also conclude that motion features and appearance features are complementary to each other for CVPI.

Chapter 5

Visible-Thermal Person Re-Identification

5.1 Motivation

Visible-thermal person re-identification (VT-reID) aims to match persons from images taken by visible cameras under normal illumination condition and thermal cameras under poor illumination condition such as during night time. Compared to person re-identification between visible images, VT-reID is even more challenging because of the lack of information in thermal images. For visible images, colors can provide rich information for identification, which are not available in thermal images. Therefore, different people under thermal cameras are more difficult to distinguish.

VT-reID aims to match persons across images taken by visible cameras under normal illumination condition and thermal cameras under poor illumination condition such as during night time. As thermal images lack color information due to the difference of illumination condition and imaging process, one major challenge of VT-reID is the semantic domain discrepancy between visible images and thermal images. Previous efforts [124, 137, 138, 17] seek to address this problem by learning domain-independent features using CNNs. Specifically, two-stream networks [137, 138], one-stream network using zero-padding [124] are trained with identity loss, contrastive loss [14] and/or triplet loss [106]. These approaches generally minimize the domain discrepancy by constraining features of the same person to be close to each other and features of different persons to be distant from each other, across visible and thermal domains. However, these constraints are applied at the sample level, where features of each sample pair or triplet of images in the training set are used. With hard sample mining, this can over-emphasize certain samples, which can mislead the network training. For example, if a pair of images of the same person are too different, their features will be mined more often than others and forced to be similar. Because of this, the convergence of the network can be disrupted. Network training will get even worse if the training set contains wrongly labeled samples.

In this research, we propose to learn domain-independent features for visible-thermal person re-identifcation with class-level constraints to alleviate this problem, in which each person is treated as a class. The intuition is to reduce the influence of individual samples that are too hard to identify or even wrongly-labeled, by weighing all samples within each class. Specifically, we train a two-stream CNN network to extract features from visible and thermal images separately. For each person, we learn one center from features in visible domain and one center from features in thermal domain, with a new relaxed version of center loss [123]. Then, we apply the pull and push constraints to the centers of the same or different persons in visible and thermal domains. More specifically, we enforce centers of the same person in visible domain and thermal domain to be close to each other, and centers of different persons to be distant from each other. Furthermore, for two different persons, the inter-class difference between their centers in the visible domain should be similar to that in the thermal domain. We formulate these class-level constraints as class-level supervision to train the two-stream CNN network, which is later used to extract features from visible and thermal images separately for person re-identification.

5.2 Proposed Method

5.2.1 Overview

We use a two-stream Convolutional Neural Network (CNN) to extract features from visible and thermal images separately. As shown in Fig. 5.1, the network takes a mini-batch of visible images and thermal images of corresponding persons as input. The output features from visible and thermal images are used to learn centers in visible and thermal domains for each person in the mini-batch, using a relaxed version of center loss [123], termed Relaxed Center Loss. As shown in Fig. 5.2, for two different persons, we use two centers to represent each one of them in the visible and thermal domains, respectively. We then impose constraints on the four centers as class-level supervision to learn domain-independent features. More specifically, we pull the centers of the same person in visible and thermal domains close to each other, and push the centers of different persons away from each other. These two constraints are formulated as loss functions and named intra-class loss and inter-class loss, respectively. Furthermore, for the two different persons as shown in Fig. 5.2, the inter-class difference between their centers in the visible domain should be similar to that in the thermal domain. We formulate this constraint as another loss and name it center-vector loss. The intra-class loss, inter-class loss and center-vector loss, as a group, are called Class-Level Domain Adaptation (CLDA) losses considering the similar nature to domain adaptation. The two-stream CNN network is trained using identity loss, relaxed center loss, and CLDA losses, as shown in Fig. 5.1. More details are described in the following sections.

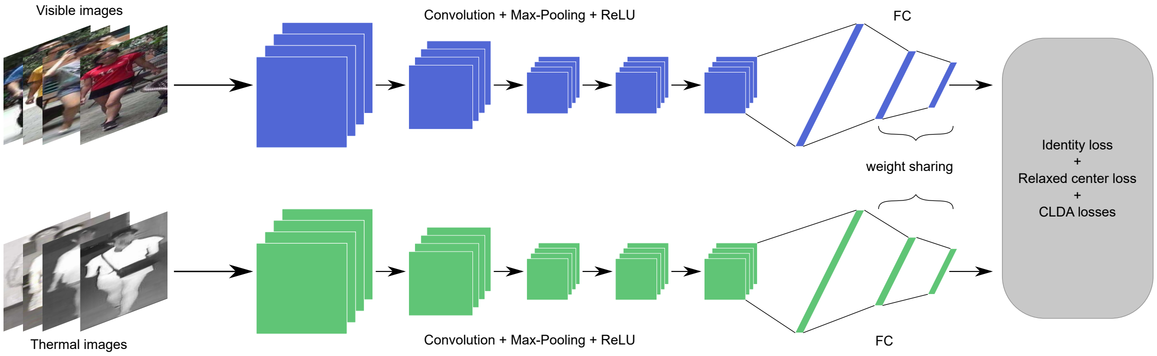


Figure 5.1 An illustration of the proposed method using two-stream Convolutional Neural Network. The network takes a mini-batch of visible and thermal images of corresponding persons as input. The extracted features are then used to learn centers in visible and thermal domains for the persons involved in the mini-batch. Identity loss, relaxed center loss and class-level domain adaptation (CLDA) losses are used to train the two-stream CNN network.

5.2.2 Two-Stream Convolutional Neural Network

To extract features from visible and thermal images, we need two different Convolutional Neural Networks with different parameters, since visible and thermal images are from different modalities. In this paper, we use two-stream CNN network, in which

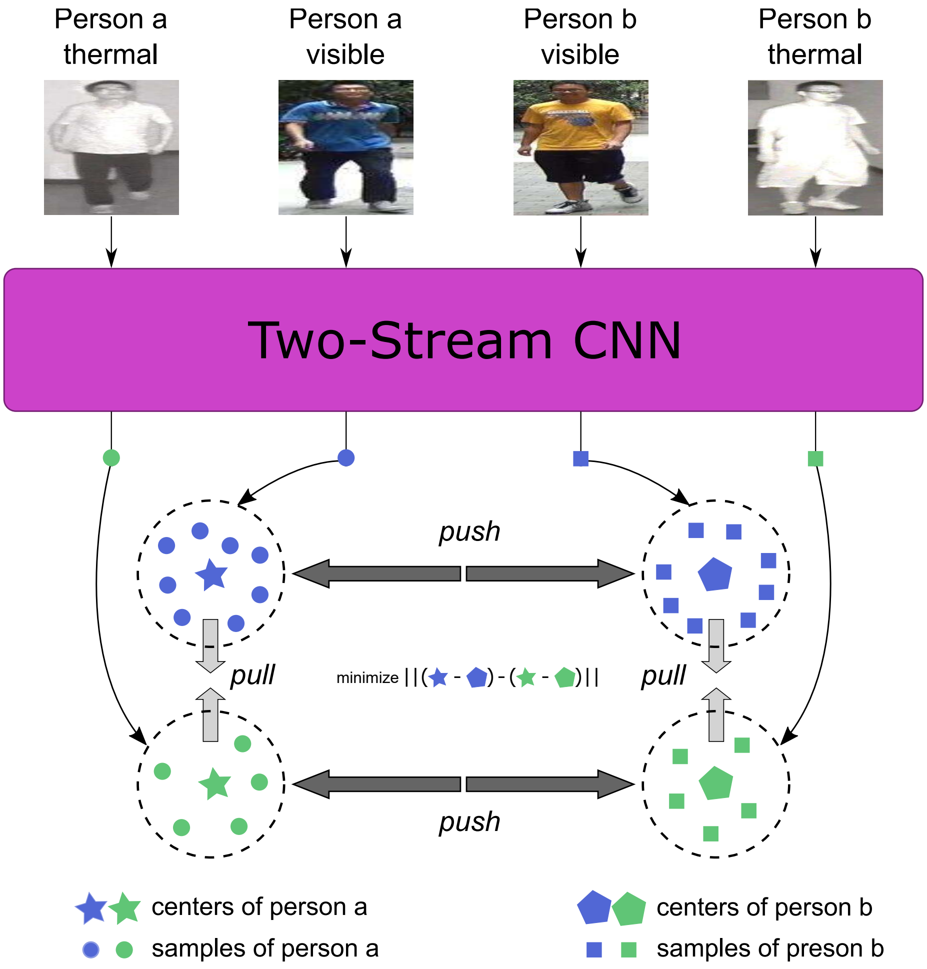


Figure 5.2 An illustration of the proposed class-level domain adaptation (CLDA) losses. For each person in the mini-batch, we update the centers of this person in both visible and thermal domains, using the relaxed center loss. Then, we compute the CLDA losses for each quadruplet, consisting of intra-class loss, inter-class loss and center-vector loss.

the two streams are used for visible and thermal images, respectively. Both streams use AlexNet [58] as the backbone architecture, with exactly the same configuration. Specifically, as shown in Fig. 5.1, each stream contains all five convolutional layers and the first fully-connected layer from AlexNet, followed by a batch normalization layer [50]. An additional fully-connected layer, which is shared by both streams, is used to embed the features extracted from both visible and thermal images into a shared feature space. The embedded features are further L2-normalized. For classification, a shared fully-connected layer is used to predict the class probabilities. We use cross-entropy loss as the identity loss for classification. Due to the limited training data, we initialize the parameters of each stream with the AlexNet model pretrained on the ImageNet dataset [20].

5.2.3 Relaxed Center Loss

Center loss [123] is first proposed to learn centers of each class to reduce intra-class variation. For features of the i-th training sample $\mathbf{x}_i$ with label y_i, the center loss is defined as:

$$\mathcal{L}_c = \frac{1}{2}\|\mathbf{x}_i - \mathbf{c}_{y_i}\|_2^2, \tag{5.1}$$

where $\mathbf{x}_i \in \mathbb{R}^d$ is the extracted deep features, $\mathbf{c}_{y_i} \in \mathbb{R}^d$ is the center of class y_i and d is the dimension of features. However, center loss enforces to pull all samples of a class to the center, which collapses the whole class into a single point in the feature space. This constraint is too strict and impractical. Therefore, we propose Relaxed Center Loss by only penalizing samples whose distances to the corresponding centers are larger than a predefined threshold. Specifically, the relaxed center loss is defined as:

$$\mathcal{L}_{rc} = \max(\frac{1}{2}\|\mathbf{x}_i - \mathbf{c}_{y_i}\|_2^2 - r, 0), \tag{5.2}$$

where r is the predefined threshold. Ideally, the center of each class is computed as the average of the features of all training samples within this class. However, it is impractical and inefficient because of limited memory. Therefore, we update the center of class k iteratively with features in mini-batches by [123]:

$$\mathbf{c}_k^{t+1} = \mathbf{c}_k^t - \alpha\Delta\mathbf{c}_k^t, \tag{5.3}$$

$$\Delta\mathbf{c}_k^t = \frac{\sum_{i=1}^{M}\delta(y_i = k)\cdot(\mathbf{c}_k - \mathbf{x}_i)}{1 + \sum_{i=1}^{M}\delta(y_i = k)}, \tag{5.4}$$

where $\delta(condition) = 1$ if the *condition* is satisfied and $\delta(condition) = 0$ if not. t represents the t-th training iteration and M is the batch size. α is the learning rate of centers, which is set to $[0, 1]$ to avoid large perturbations. The gradient of $\mathcal{L}_{rc}$ with respect to $\mathbf{x}_i$ is computed as:

$$\frac{\partial \mathcal{L}_{rc}}{\partial \mathbf{x}_i} = \begin{cases} \mathbf{x}_i - \mathbf{c}_{y_i} & \text{if } \frac{1}{2}\|\mathbf{x}_i - \mathbf{c}_{y_i}\|_2^2 > r \\ 0, & \text{otherwise.} \end{cases} \tag{5.5}$$

In this paper, we use the relaxed center loss to learn centers for each person in both visible and thermal domains.

5.2.4 Class-Level Domain Adaptation Losses

Suppose we have N different persons in the training set. For a person with label $k \in \{1, 2, \ldots, N\}$, the centers in visible and thermal domains are denoted as $\mathbf{c}_{V,k}$ and $\mathbf{c}_{T,k}$, where V and T stand for visible and thermal domains, respectively. Similarly, the centers in visible and thermal domains for person l $(l \neq k)$ are denoted as $\mathbf{c}_{V,l}$ and $\mathbf{c}_{T,l}$, respectively. We define the intra-class loss as:

$$\mathcal{L}_{intra} = \frac{1}{2}\sum_{k=1}^{N} \|\mathbf{c}_{V,k} - \mathbf{c}_{T,k}\|_2^2, \tag{5.6}$$

, and the inter-class loss is defined as:

$$\begin{aligned} \mathcal{L}_{inter} = &\frac{1}{2}\sum_{k=1}^{N}\sum_{l=1,l\neq k}^{N} \max(m - \|\mathbf{c}_{V,k} - \mathbf{c}_{V,l}\|_2^2, 0) \\ &+\frac{1}{2}\sum_{k=1}^{N}\sum_{l=1,l\neq k}^{N} \max(m - \|\mathbf{c}_{T,k} - \mathbf{c}_{T,l}\|_2^2, 0), \end{aligned} \tag{5.7}$$

where m is the margin which separates the centers of different persons. We define the center-vector loss as:

$$\mathcal{L}_{cv} = \frac{1}{2}\sum_{k=1}^{N}\sum_{l=1,l\neq k}^{N} \|(\mathbf{c}_{V,k} - \mathbf{c}_{V,l}) - (\mathbf{c}_{T,k} - \mathbf{c}_{T,l})\|_2^2. \tag{5.8}$$

However, as mentioned earlier, the centers can not be computed from all the training samples at once. Therefore, we update centers for each person in visible and thermal domains iteratively. As shown in Fig. 5.1, the input for the two-stream network is a mini-batch of visible and thermal images of corresponding persons. Features from visible and thermal images are extracted by the visible and thermal streams, respectively. The centers of each person in visible and thermal domains are updated as in Eq. (5.3) and (5.4) using the features in the mini-batch. More specifically, the centers are updated as follows:

$$\mathbf{c}^{t+1}_{D,y_{i'}} = \mathbf{c}^{t}_{D,y_{i'}} - \alpha\Delta\mathbf{c}^{t}_{D,y_{i'}}, \tag{5.9}$$

where $D \in \{V, T\}$ is the domain indicator, $i' \in \{i, j\}$, and $\Delta\mathbf{c}^{t}_{D,y_{i'}}$ is the update for the center of sample i' in domain D computed from the whole mini-batch at iteration t.

The proposed CLDA losses are also computed based on mini-batches. As shown in Fig. 5.2, we form a quadruplet, consisting of two images of the two different persons. We denote their corresponding deep features as $\mathbf{x}_{V,i}$, $\mathbf{x}_{V,j}$, $\mathbf{x}_{T,i}$ and $\mathbf{x}_{T,j}$, and their labels as y_i and y_j ($y_i \neq y_j$). The proposed CLDA losses are then computed based on the centers and features from the quadruplets in the mini-batch. More specifically, the total loss is summed from the losses for each quadruplet in the mini-batch. Therefore, we discuss the proposed CLDA losses for each quadruplet. By substituting the center updates in Eq. (5.9), the CLDA losses for each quadruplet at iteration t are defined as follows:

$$\begin{aligned}\mathcal{L}_{intra} = \sum_{i' \in \{i,j\}} \frac{1}{2} \|(\mathbf{c}^{t}_{V,y_{i'}} - \alpha\Delta\mathbf{c}^{t}_{V,y_{i'}}) \\ -(\mathbf{c}^{t}_{T,y_{i'}} - \alpha\Delta\mathbf{c}^{t}_{T,y_{i'}})\|_2^2,\end{aligned} \tag{5.10}$$

$$\begin{aligned}\mathcal{L}_{inter} = \frac{1}{2}\max(m - \|\mathbf{d}_V\|_2^2, 0) \\ +\frac{1}{2}\max(m - \|\mathbf{d}_T\|_2^2, 0),\end{aligned} \tag{5.11}$$

$$\mathcal{L}_{cv} = \frac{1}{2}\|\mathbf{d}_{cv}\|_2^2, \tag{5.12}$$

where

$$\begin{aligned}
\mathbf{d}_V &= (\mathbf{c}^t_{V,y_i} - \alpha\Delta\mathbf{c}^t_{V,y_i}) - (\mathbf{c}^t_{V,y_j} - \alpha\Delta\mathbf{c}^t_{V,y_j}), \\
\mathbf{d}_T &= (\mathbf{c}^t_{T,y_i} - \alpha\Delta\mathbf{c}^t_{T,y_i}) - (\mathbf{c}^t_{T,y_j} - \alpha\Delta\mathbf{c}^t_{T,y_j}), \\
\mathbf{d}_{cv} &= \mathbf{d}_V - \mathbf{d}_T.
\end{aligned} \tag{5.13}$$

Using the chain rule for derivatives, we can derive the gradients from the proposed CLDA losses for the features of each quadruplet as follows:

$$\frac{\partial \mathcal{L}_{intra}}{\partial \mathbf{x}_{D,i'}} = (-1)^{\delta(D=T)} \cdot \frac{\alpha(\mathbf{c}^t_{V,y_{i'}} - \mathbf{c}^t_{T,y_{i'}} - \alpha\Delta\mathbf{c}^t_{V,y_{i'}} + \alpha\Delta\mathbf{c}^t_{T,y_{i'}})}{1 + \sum_{n=1}^{M} \delta(y_n = y_{i'})}, \tag{5.14}$$

$$\frac{\partial \mathcal{L}_{inter}}{\partial \mathbf{x}_{D,i'}} = \begin{cases} \frac{(-1)^{\delta(D=V)}\alpha \cdot \mathbf{d}_D}{1+\sum_{n=1}^{M}\delta(y_n=y_{i'})} & \text{if } m - \|\mathbf{d}_D\|_2^2 > 0, \\ 0 & \text{otherwise,} \end{cases} \tag{5.15}$$

$$\frac{\partial \mathcal{L}_{cv}}{\partial \mathbf{x}_{D,i'}} = \frac{(-1)^{\delta(D=T)\cdot\delta(i'=j)}\alpha \cdot \mathbf{d}_{cv}}{1 + \sum_{n=1}^{M} \delta(y_n = y_i)}, \tag{5.16}$$

where $D \in \{V, T\}$ and $i' \in \{i, j\}$.

For training, we combine identity loss, relaxed center loss and the proposed CLDA losses as follows:

$$\mathcal{L}_{total} = \mathcal{L}_s + \lambda_1\mathcal{L}_{rc} + \lambda_2\mathcal{L}_{intra} + \lambda_3\mathcal{L}_{inter} + \lambda_4\mathcal{L}_{cv} \tag{5.17}$$

where $\mathcal{L}_s$ is the cross-entropy identity loss and λ_1, λ_2, λ_3, λ_4 are the weights for balancing these losses.

5.3 Experiments

5.3.1 Experimental Settings

Datasets and evaluation metrics We evaluate the proposed method for visible-thermal person re-identification on two public datasets: SYSU-MM01 dataset [124] and RegDB dataset [97].

SYSU-MM01 dataset is collected from 491 persons with 6 cameras, including four visible cameras and two thermal cameras. Each person is captured by at least two cameras. In total, there are 22,258 visible images and 11,909 thermal images. This dataset is more challenging since some cameras are placed in indoor environment and others in outdoor environment. This dataset has a fixed split with 296 identities for training, 99 for validation and 96 for testing. We use images from both training and validation sets to train the two-stream CNN network as in [138]. For testing, visible images are used as the gallery set and thermal images are used as the probe set. We evaluate the proposed method using the *all-search* mode under single-shot setting since it is the most challenging case as mentioned in [124].

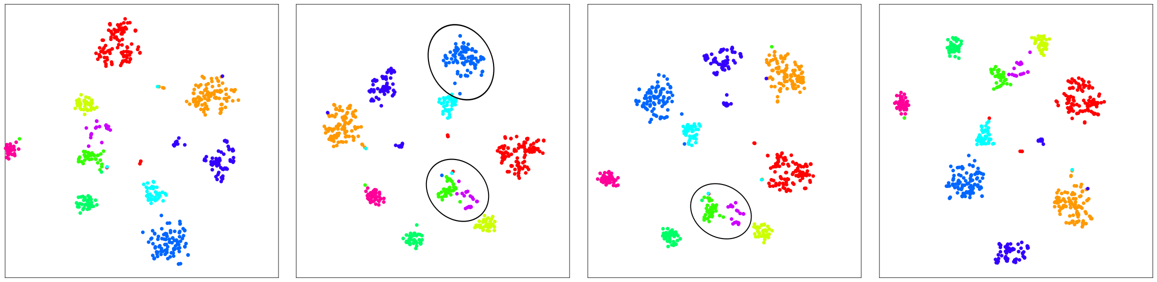
RegDB dataset is collected from 412 persons from different views with respect to each person's body, such as front, back and side views. For each person, 10 visible images and 10 thermal images are captured with a dual-camera system including a visible camera and a thermal camera. We follow the evaluation protocol in [137] and randomly split the dataset into two subsets of equal size, one for training and one for testing. For testing, we mainly use visible images as the gallery set and thermal images as the probe set. To obtain more stable results, we repeat this process over 10 random dataset splits and report the average results.

To evaluate the proposed method and comparison methods, we use Cumulative Matching Characteristics (CMC) and mean Average Precision (mAP) as the metrics. mAP is adopted because there might be multiple matching images in the gallery set for a probe image [152].

Implementation details In this paper, we use AlexNet as the backbone for the two-stream CNN network, both streams of which share the same network architecture. Specifically, each stream contain 5 convolution layers, 3 max-pooling layers, 3 fully-connected (FC) layer and 1 batch normalization layer, with the configuration

Table 5.1 Effects of the proposed CLDA losses, evaluated on both SYSU-MM01 and RegDB datasets in terms of CMC (%) and mAP (%).

Datasets			SYSU-MM01				RegDB			
$\mathcal{L}_{intra}$	$\mathcal{L}_{inter}$	$\mathcal{L}_{cv}$	R=1	R=10	R=20	mAP	R=1	R=10	R=20	mAP
✓			19.48	63.89	80.85	22.87	28.57	50.23	61.36	28.57
	✓		18.32	62.61	79.59	22.09	24.58	46.64	57.64	26.25
	✓	✓	20.03	64.59	81.55	23.37	28.34	51.23	62.58	29.51
✓		✓	19.74	64.21	81.69	23.03	**28.73**	**52.08**	**63.66**	**29.86**
✓	✓		19.34	63.41	80.70	22.67	27.54	49.68	61.38	28.61
✓	✓	✓	**20.34**	**64.98**	**82.11**	**23.47**	28.49	51.90	63.23	29.83



Proposed

Proposed w/o intra-class loss

Proposed w/o inter-class loss

Proposed w/o center-vector loss

Figure 5.3 t-SNE embedding of features of visible images extracted models trained with different losses.

in the following order: 1) conv11x11, stride=4, feature maps=96, 2) maxpool3x3, stride=2, 3) conv5x5, stride=1, feature maps=256, 4) maxpool3x3, stride=2, 5) conv3x3, stride=1, feature maps=384, 6) conv3x3, stride=1, feature maps=384, 7) conv3x3, stride=1, feature maps=256, 8) fc4096, 9) batch-norm, 10) fc1024, 11) fc and 12) softmax. The number of neurons in the final FC layer is set as the number of identities, which is 395 for SYSU-MM01 dataset and 206 for RegDB dataset. The parameters of layers 1) to 9) of the two streams are not shared, which are used to extract features from visible and thermal images separately. The parameters of FC layers 10) and 11) are shared by the two streams, which are used to embed visible and thermal features into a shared feature space and predict the class probabilities, respectively. The features in FC layer 10) are L2-normalized before being used to update centers and compute relaxed center loss as well as the proposed class-level domain adaptation losses. The softmax layer is used to compute identity loss. After the two-stream CNN network is trained, we extract features from visible and thermal images from the output of FC layer 10) separately. The features of visible and thermal images are then used to compute pairwise Euclidean distances for evaluation.

We use PyTorch[1] to implement and train the two-stream CNN network. The parameters of layer 1) to layer 8) are initialized from the AlexNet model pretrained on ImageNet dataset, while the parameters of other layers are randomly initialized. The margin m for the inter-class loss is set to 0.9 and the threshold r for relaxed center loss is set to 0.1. In our experiments, we empirically set $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 1$ for Eq. (5.17) to combine the losses for training the network. Since we use AlexNet as the backbone for the two-stream CNN network, the input image has to be resized to 227×227 pixels. During training, we resize each image (visible or thermal) to 256×256 pixels and randomly crop to 227×227 pixels for data augmentation. The batch size is set to 64. The two-stream CNN network is first trained for 20,000 iterations and 2,000 iterations on SYSU-MM01 dataset and RegDB dataset, respectively. Then, the network is further fine-tuned for 25,000 iterations and 3,000 iterations on SYSU-MM01 dataset and RegDB dataset, respectively. The learning rate is set to 0.01 initially and decayed polynomially until zero.

5.3.2 Effectiveness of Class-Level Domain Adaptation Losses

We first evaluate the effectiveness of the proposed method, which trains the two-stream CNN network using Class-Level Domain Adaptation (CLDA) losses together with identity loss and relaxed center loss. We analyze the effect of each term in the CLDA losses. Specifically, we remove one or two terms at a time as a variant of the proposed method to train the two-stream CNN network. The experiment results on both SYSU-MM01 and RegDB datasets are reported in Table 5.1. On SYSU-

[1] https://pytorch.org

MM01 dataset, the proposed method obtains the best performance with all three loss terms included for training: intra-class loss, inter-class loss and center-vector loss. Removing intra-class loss or inter-class loss leads to a slight drop, while removing center-vector loss leads to a larger drop. This is because the center-vector loss poses similar constraint to intra-class loss while incorporating inter-class center-vectors. Removing intra-class loss and center-vector loss at the same time results in the worst performance, since inter-class loss alone cannot introduce constraints to bridge the gap between visible and thermal features. Inter-class loss can only pushes features of different persons away, while intra-class loss is more dedicated to extracting domain-independent features by enforcing visible and thermal features to be similar for the same persons. We can see that all three losses are useful. On RegDB dataset, the best result is achieved by the model trained without inter-class loss. This may be because images of the same person in RegDB dataset have very clean background and share similar poses, and the inter-class loss is not of much help.

To better understand the effects of these loss terms, we also visualize the features learned by the proposed method with all CLDA losses or with one term removed at a time for training. Specifically, we choose 10 identities in the test set in the SYSU-MM01 dataset and extract the features from visible images of these identities. The features are embedded into 2D space using t-SNE embedding [90], as shown in Fig. 5.3. We can see that, without intra-class or inter-class loss, the features learned are less distinctive. The use of center-vector loss does not result in obvious difference in the embedded features. This may be because the difference cannot be reflected in the lower dimension which the features are embedded into. Nevertheless, the quantitative results in Table. 5.1 prove the usefulness of the center-vector loss.

5.3.3 Effectiveness of Relaxed Center Loss

We demonstrate the effectiveness of the proposed relaxed center loss, by comparing the proposed method using center loss [123] or relaxed center loss on both SYSU-MM01 and RegDB datasets. Specifically, we replace the relaxed center loss with the original center loss and train for 25,000 iterations and 3,000 iterations on SYSU-MM01 and RegDB datasets, respectively. The comparison results are shown in Table 5.2. Obviously, the proposed method performs better with the relaxed center loss than center loss. As mentioned before, center loss enforces to pull features of all samples in a class to the center, which tries to collapse the whole class into a single point in the feature space. Such a constraint is too strict and impractical. During our experiments, we notice that the center loss is much higher than both intra-class loss and inter-class loss as they converge. This means the network is more focused on minimizing the center loss, which is unnecessary for learning domain-independent features. Relaxed center loss, however, converges to a smaller value than both intra-class and inter-class losses, which makes the network focus on learning domain-independent features from visible and thermal images. Consequently, the use of relaxed center loss leads to better matching results in terms of both CMC and mAP.

Table 5.2 Comparison of the proposed method using center loss and relaxed center loss on SYSU-MM01 dataset in terms of CMC (%) and mAP (%).

SYSU-MM01	R=1	R=5	R=10	R=20	mAP
Ours w. $\mathcal{L}_c$	19.70	47.36	63.75	80.69	22.99
Ours w. $\mathcal{L}_{rc}$	20.34	48.59	64.98	82.11	23.47
RegDB	R=1	R=5	R=10	R=20	mAP
Ours w. $\mathcal{L}_c$	27.22	39.89	48.40	60.07	28.93
Ours w. $\mathcal{L}_{rc}$	28.49	42.72	51.90	63.23	29.83

5.3.4 Comparison and Combination of Class-Level Supervision and Sample-Level Supervision

We compare the proposed method, using class-level supervision, with BDTR [138], which trains the network with identity loss and triplet loss at the sample level. We further incorporate the sample-level triplet loss into the proposed method, as a combination of class-level supervision and sample-level supervision. The triplet loss is incorporated into the proposed method with a weight λ_t as follows:

$$\mathcal{L}'_{total} = \mathcal{L}_{total} + \lambda_t \mathcal{L}_t, \tag{5.18}$$

where $\mathcal{L}_t$ is the triplet loss used in [138].

We evaluate the proposed method, BDTR, and their combination using different values of λ_t on both SYSU-MM01 and RegDB datasets. For fair comparison, we use PyTorch to re-implement the BDTR method following the steps in [138]. The results are shown in Table 5.3. Compared to the original BDTR, the re-implemented BDTR performs worse on RegDB dataset but better on SYSU-MM01 dataset. This is because the original BDTR is implemented using TensorFlow package[2]. Besides, the parameters of the two-stream CNN network is initialized from a model pretrained on ImageNet dataset on the PyTorch platform[3], which could also result in the difference in performance. From now on, we use the re-implemented BDTR as the reference for further analysis in this and later experiments for fair comparison.

Compared to BDTR, the proposed method achieves higher CMC matching rates and mAP. This demonstrates that, the proposed class-level domain adaptation losses are more effective for learning domain-independent features than sample-level triplet loss for visible-thermal person re-identification. The incorporation of sample-level triplet loss into the proposed method can result in better CMC matching rates and

[2] https://www.tensorflow.org

[3] https://github.com/jiecaoyu/pytorch_imagenet

mAP's. Specifically, $\lambda_t = 0.1$ achieves the best performance on SYSU-MM01 dataset and $\lambda_t = 0.5$ achieves the best mAP on RegDB dataset. This proves that both class-level and sample-level supervisions are useful for learning domain-independent features and that they are complementary to each other.

Table 5.3 Comparison of the proposed method, BDTR and their combination on SYSU-MM01 and RegDB datasets in terms of CMC (%) and mAP (%).

SYSU-MM01	R=1	R=10	R=20	mAP
BDTR [138]	17.01	55.43	71.96	19.66
BDTR (re-implemented)	19.53	64.74	81.05	23.13
Ours	20.34	64.98	82.11	23.47
Ours+triplet $\lambda_t = 1.0$	20.34	**65.51**	82.25	23.80
Ours+triplet $\lambda_t = 0.5$	20.20	65.25	82.11	23.81
Ours+triplet $\lambda_t = 0.1$	**20.70**	65.27	**82.38**	**23.86**
RegDB	R=1	R=10	R=20	mAP
BDTR [138]	33.47	58.42	67.52	31.83
BDTR (re-implemented)	26.90	49.00	59.73	28.48
Ours	28.49	51.90	**63.23**	29.83
Ours+triplet $\lambda_t = 1.0$	28.85	50.91	62.03	30.04
Ours+triplet $\lambda_t = 0.5$	28.72	51.45	62.47	**30.04**
Ours+triplet $\lambda_t = 0.1$	**28.85**	**51.96**	62.64	29.92

Table 5.4 Comparison of the proposed method with several state-of-the-art methods on SYSU-MM01 dataset and RegDB dataset in terms of CMC (%) and mAP (%).

Datasets	SYSU-MM01				RegDB			
Method	R=1	R=10	R=20	mAP	R=1	R=10	R=20	mAP
Zero-Padding [124]	14.80	54.12	71.33	15.95	17.75	34.21	44.35	18.90
TONE [137]	12.52	50.72	68.60	14.42	16.87	34.03	44.10	14.92
TONE+XQDA	14.01	52.78	69.06	15.97	21.94	45.05	55.73	21.80
TONE+MLAPG	12.43	50.64	68.72	14.61	17.82	40.29	49.73	18.03
TONE+SCDL	6.58	35.62	56.32	10.32	8.06	22.09	28.89	10.03
TONE+rCDL	7.02	37.31	57.64	10.46	9.47	22.96	29.42	10.26
TONE+HCML [137]	14.32	53.16	69.17	16.16	24.44	47.53	56.78	20.80
BDTR	19.53	64.74	81.05	23.13	26.90	49.00	59.73	28.48
Ours	20.34	64.98	82.11	23.47	28.49	51.90	**63.23**	29.83
Ours+triplet $\lambda_t = 0.1$	**20.70**	**65.27**	**82.38**	**23.86**	**28.85**	**51.96**	62.64	**29.92**

5.3.5 Comparison with State of the Art

In this section, we compare the performances of the proposed method with several state-of-the-art methods on both SYSU-MM01 and RegDB datasets: Zero-Padding [124], TONE+HCML [137] and BDTR [138]. Zero-Padding is a one-stream network, which can be used to extract features from both visible and thermal images. TONE+HCML is a two-stage framework, which implements feature extraction and metric learning separately. We also report some results from [138], which combine TONE with other metric learning methods, including XQDA [72], MLAPG [73], SCDL [119], rCDL [46]. BDTR also trains a two-stream CNN network, as in the proposed method, with sample-level triplet loss and identity loss. We also incorporate the triplet loss into the proposed method with $\lambda_t = 0.1$ for Eq. (5.18).

Compared to Zero-Padding, TONE, TONE+HCML or TONE with other metric learning methods, the proposed method achieves much better performance. We can see that the proposed method achieves better performance than BDTR on both SYSU-MM01 and RegDB datasets. We also notice that the proposed method outperforms the BDTR method by a slightly larger margin on RegDB dataset than on SYSU-MM01 dataset. This can be explained by the difference between the natures of BDTR and the proposed method. BDTR uses triplet loss, which is a sample-level supervision and could over-emphasize the importance of each sample for small-scale datasets such as RegDB. The proposed class-level supervision can alleviate this effect. However, for a larger datasets such as SYSU-MM01, the importance of each sample is automatically reduced and the improvement of class-level supervision is reduced accordingly. The combination of both class-level supervision and sample-level supervision can further improve the results. Specifically, the combination achieves 1.37% and 1.95% improvement of rank-1 matching rate for SYSU-MM01 dataset and RegDB dataset, respectively. This demonstrates that the class-level supervision and sample-level supervision are both useful for learning domain-independent features for

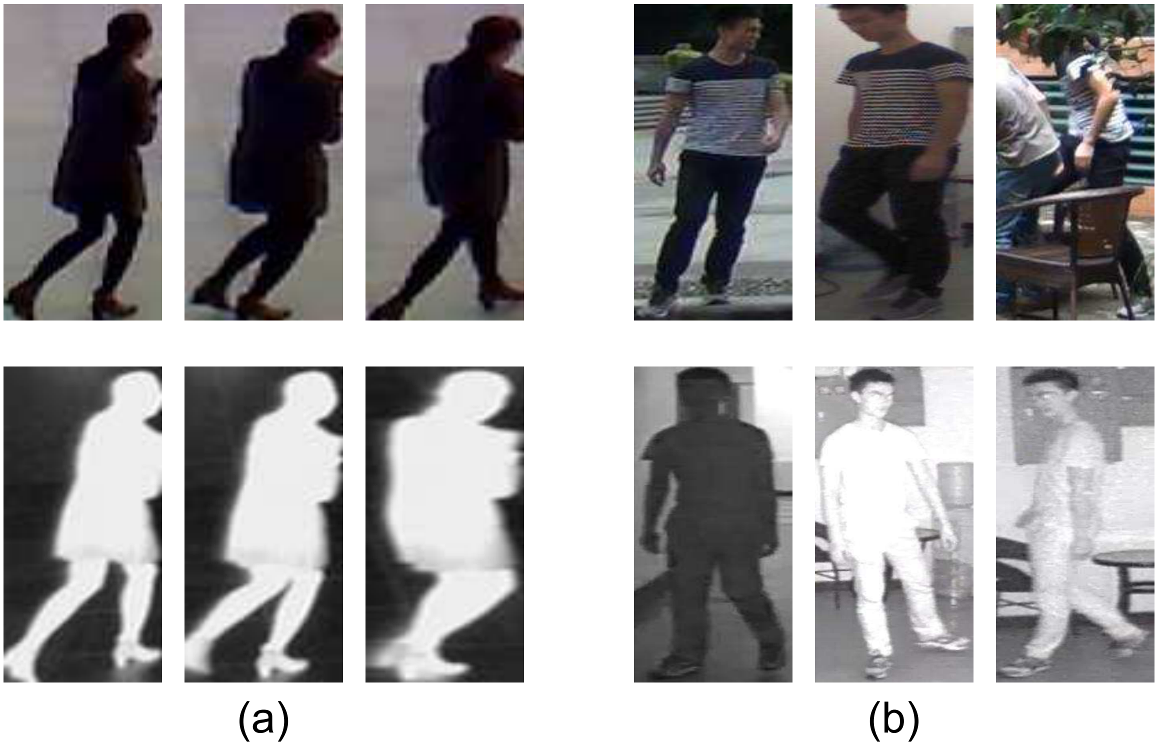


Figure 5.4 Sample visible and thermal images of the same person in (a) RegDB dataset and (b) SYSU-MM01 dataset.

visible-thermal person re-identification.

We also observe the proposed method and all comparison methods perform much better on RegDB dataset than SYSU-MM01 dataset. This is mainly because there are more identities in SYSU-MM01 dataset than in RegDB dataset and it is more difficult to identify the true match from a larger gallery set. Besides, compared to SYSU-MM01 dataset, visible and thermal images of the same identities from RegDB dataset have very clean background and similar poses as shown in Fig. 5.4, which makes the re-identification easier. Furthermore, the persons in images of RegDB dataset are more well-aligned than those in SYSU-MM01 dataset. In some extreme occasions, images from SYSU-MM01 dataset do not even have any person at all. All these challenges lead to the lower matching performance on SYSU-MM01 dataset than on RegDB dataset.

5.3.6 Different Query Settings

We also evaluate the performance of different query settings on RegDB dataset as in [137]. Specifically, we use visible images as the gallery set under the “Visible to Thermal" setting and thermal images as the gallery set under the “Thermal to Visible" setting. As shown in Table 5.5, we compare the performances of the proposed method, BDTR and their combination using $\lambda_t = 0.1$. Under both settings, the proposed method achieves rank-1 matching rate of around 28% and mAP of around 29%. Also, the proposed method outperforms BDTR in both cases. Combining BDTR with the proposed method results in better performance. As discussed before, the class-level supervision and the sample-level supervision are both useful for learning domain-independent features for visible-thermal person re-identification.

Table 5.5 Evaluation of the proposed method, BDTR and their combination in different query settings on RegDB dataset in terms of CMC (%) and mAP (%).

Setting	Visible to Thermal			
Method	R=1	R=10	R=20	mAP
BDTR	26.90	49.00	59.73	28.48
Ours	28.49	51.90	63.23	29.83
Ours+BDTR $\lambda_t = 0.1$	28.85	51.96	62.64	29.92
Setting	Thermal to Visible			
Method	R=1	R=10	R=20	mAP
BDTR	26.02	47.57	60.15	28.09
Ours	27.76	51.09	63.43	29.61
Ours+BDTR $\lambda_t = 0.1$	27.82	51.20	63.27	29.67

5.4 Chapter Summary

In this chapter, we studied the visible-thermal person re-identification problem. We propose class-level supervision to train a CNN-based two-stream network. Specifically, we learn a center for each person in each domain (visible and thermal) with a new relaxed center loss. Then, we apply the pull and push constraints to the centers of the same or different persons in visible and thermal domains. More specifically, we enforce centers of the same person in visible domain and thermal domain to be close to each other, and centers of different persons to be distant from each other. Furthermore, for two different persons, the inter-class difference between their centers in the visible domain should be similar to that in the thermal domain. We formulate these class-level constraints as class-level supervision to train the two-stream CNN network, which is later used to extract features from visible and thermal images separately for person re-identification. Experiments on two public datasets demonstrate the effectiveness of the proposed method.

Chapter 6

Conclusion

In this research, we studied two important sub-problems of person identification, cross-view person identification and visible-thermal person re-identification. Cross-view person identification matches persons across temporally synchronized videos captured by wearable cameras such as GoPro and Google Glass. It can be applied to scenarios that need person match across wearable-camera videos. We are the first to propose this problem to facilitate the understanding of event scenes, in which fixed camera network are not sufficient and wearable cameras are needed. Visible-thermal person re-identification aims to match persons across visible images and thermal images, which can enhance the security surveillance under poor illuminations such as night time.

For cross-view person identification, we proposed to utilize motion information to match persons across temporally synchronized videos. Specifically, we proposed a new triplet network to extract view-invariant features, by associating 2D optical flows with 3D human skeletons. We further proposed a new method for synthesizing 2D optical flows from 3D human skeleton sequence, to train the proposed triplet network. The network was then fine-tuned on real video datasets.

We collected three datasets to evaluate the proposed method. Experimental results demonstrated the effectiveness of the proposed method, especially the incorporation of 3D human skeleton data. With the incorporation of 3D human skeleton data for network training, the view invariance of the motion features can be improved and thus better person identification is achieved. We further combined the proposed method, which only uses motion information, with an appearance-based person identification method. The combination of appearance information and motion information can achieve better performance. We noticed that the performance of the proposed method is lower if there exists temporal synchronization error between two videos. Therefore, more robustness against the temporal synchronization error is needed. For example, we can use temporal pooling to reduce the influence of temporal synchronization error, in the same spirit that spatial pooling can combat against spatial translation.

For visible-thermal person re-identification, we proposed to use class-level supervision to learn domain-independent features. Specifically, we proposed a new relaxed center loss to learn centers for each person: one for visible domain and one for thermal domain. We then enforced centers of the same person to be close to each other and centers of different persons to be distant from each other across visible and thermal domains. Furthermore, for two different persons, the inter-class difference between their centers in the visible domain should be similar to that in the thermal domain. We formulated these class-level constraints as loss functions to train a two-stream convolutional neural network, each stream of which extracts features from visible or thermal images separately.

We evaluated the proposed method on two datasets. Experimental results demonstrated that the class-level supervision outperforms the sample-level supervision. The combination of class-level and sample-level supervisions can achieve better performance. The proposed method takes the whole image as input and output the features, which can be considered as global feature extraction. We can also put more attention to features from local body parts with different weights to improve the performance.

6.1 Future Work

In the first work, the incorporation of 3D human skeleton data improves the view-invariance of motion features for person identification. This provides us a new path when addressing some challenges, which is to use an additional modality of data to regularize the feature learning process. We can apply this kind of method to other research topics such as action recognition, human pose estimation and so on. Also, data generation by synthesis can be very useful, which is also seen in other researches such as semantic segmentation by learning from video games. However, synthesizing realistic data is difficult, since we need to consider many factors such as illumination, background, and non-rigid deformation. Therefore, we can consider using conditional generative adversarial networks to implicitly incorporate these factors into synthesis, to generate more realistic data.

Another issue in cross-view person identification and many other person identification sub-problems, came to sight is the performance of cross-dataset testing. If a model is trained on one dataset, the performance on another dataset is significantly lower. This problem can be explained from two perspectives. First, we can consider it as an insufficient data problem, which leads to overfitting of the network. A straightforward solution to this is to collect larger datasets with more variations. Second, this problem can be considered as the dataset domain difference problem. To address this problem, we need domain adaptation approaches that are robust to some domain variations.

In the second work, the proposed method addresses the domain discrepancy problem in a feature level. We can also address the domain discrepancy problem by directly generating images in a unified manner. Specifically, we can use conditional generative adversarial network to generate visible image from thermal image and thermal image from visible image. Thus, for either visible image or thermal image, we can obtain a unified representation in image level which reduces the domain discrepancy. For better visible-thermal person re-identification, we can also exploit human body part information more deeply. Because different body parts can contribute differently to the identification of a person. For example, we can use state-of-the-art human parsing algorithms to divide each person image into several semantic regions and extract features based on these regions.

Bibliography

[1] Ejaz Ahmed, Michael Jones, and Tim K Marks, *An improved deep learning architecture for person re-identification*, CVPR, 2015.

[2] Slawomir Bak, Etienne Corvee, Francois Bremond, and Monique Thonnat, *Person re-identification using Haar-based and DCD-based signature*, AVSS, 2010.

[3] ———, *Person re-identification using spatial covariance regions of human body parts*, AVSS, 2010.

[4] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski, *A database and evaluation methodology for optical flow*, International Journal of Computer Vision (2011).

[5] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, *Surf: Speeded up robust features*, ECCV, 2006.

[6] Loris Bazzani, Marco Cristani, Alessandro Perina, Michela Farenzena, and Vittorio Murino, *Multiple-shot person re-identification by hpe signature*, ICPR, 2010.

[7] Jie Cai, Zibo Meng, Ahmed Shehab Khan, Zhiyuan Li, James O'Reilly, and Yan Tong, *Island loss for learning discriminative features in facial expression recognition*, FG, 2018.

[8] Xiaojun Chang, Po-Yao Huang, Yi-Dong Shen, Xiaodan Liang, Yi Yang, and Alexander G Hauptmann, *Rcaa: Relational context-aware agents for person search*, ECCV, 2018.

[9] Rizwan Chaudhry, Avinash Ravichandran, Gregory Hager, and René Vidal, *Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions*, CVPR, 2009.

[10] Dapeng Chen, Zejian Yuan, Badong Chen, and Nanning Zheng, *Similarity learning with spatial constraints for person re-identification*, CVPR, 2016.

[11] Di Chen, Shanshan Zhang, Wanli Ouyang, Jian Yang, and Ying Tai, *Person search via a mask-guided two-stream cnn model*, ECCV, 2018.

[12] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang, *Beyond triplet loss: a deep quadruplet network for person re-identification*, CVPR, 2017.

[13] 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*, CVPR, 2016.

[14] Sumit Chopra, Raia Hadsell, and Yann LeCun, *Learning a similarity metric discriminatively, with application to face verification*, CVPR, 2005.

[15] Dahjung Chung, Khalid Tahboub, and Edward J Delp, *A two stream siamese convolutional neural network for person re-identification*, CVPR, 2017.

[16] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter, *Fast and accurate deep network learning by exponential linear units (elus)*, arXiv preprint arXiv:1511.07289 (2015).

[17] Pingyang Dai, Rongrong Ji, Haibin Wang, Qiong Wu, and Yuyu Huang, *Cross-modality person re-identification with generative adversarial training.*, IJCAI, 2018.

[18] Navneet Dalal and Bill Triggs, *Histograms of oriented gradients for human detection*, CVPR, 2005.

[19] Hal Daumé III, *Frustratingly easy domain adaptation*, arXiv preprint arXiv:0907.1815 (2009).

[20] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, *ImageNet: A large-scale hierarchical image database*, CVPR, 2009.

[21] Zhengming Ding, Ming Shao, and Yun Fu, *Robust multi-view representation: A unified perspective from multi-view learning to domain adaption.*, IJCAI, 2018.

[22] Jeff Donahue, Judy Hoffman, Erik Rodner, Kate Saenko, and Trevor Darrell, *Semi-supervised domain adaptation with instance constraints*, CVPR, 2013.

[23] Jeffrey Donahue, Lisa Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell, *Long-term recurrent convolutional networks for visual recognition and description*, CVPR, 2015.

[24] Xing Fan, Hao Luo, Xuan Zhang, Lingxiao He, Chi Zhang, and Wei Jiang, *Scpnet: Spatial-channel parallelism network for joint holistic and partial person re-identification*, ACCV, 2018.

[25] Michela Farenzena, Loris Bazzani, Alessandro Perina, Vittorio Murino, and Marco Cristani, *Person re-identification by symmetry-driven accumulation of local features*, CVPR, 2010.

[26] Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars, *Unsupervised visual domain adaptation using subspace alignment*, ICCV, 2013.

[27] Mengyue Geng, Yaowei Wang, Tao Xiang, and Yonghong Tian, *Deep transfer learning for person re-identification*, arXiv preprint arXiv:1611.05244 (2016).

[28] Xavier Glorot and Yoshua Bengio, *Understanding the difficulty of training deep feedforward neural networks*, AISTATS, 2010.

[29] 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) (2009).

[30] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman, *Geodesic flow kernel for unsupervised domain adaptation*, CVPR, 2012.

[31] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, *Generative adversarial nets*, NIPS, 2014.

[32] Douglas Gray and Hai Tao, *Viewpoint invariant pedestrian recognition with an ensemble of localized features*, ECCV, 2008.

[33] Ankur Gupta, Julieta Martinez, James J. Little, and Robert J. Woodham, *3D pose from motion for cross-view action recognition via non-linear circulant temporal encoding*, CVPR, 2014.

[34] Ju Han and Bir Bhanu, *Individual recognition using gait energy image*, IEEE Transactions on Pattern Analysis and Machine Intelligence (2005).

[35] Albert Haque, Alexandre Alahi, and Li Fei-Fei, *Recurrent attention models for depth-based person identification*, CVPR, 2016.

[36] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, *Mask R-CNN*, ICCV, 2017.

[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, *Delving deep into rectifiers: Surpassing human-level performance on imagenet classification*, ICCV, 2015.

[38] ———, *Deep residual learning for image recognition*, CVPR, 2016.

[39] Lingxiao He, Jian Liang, Haiqing Li, and Zhenan Sun, *Deep spatial feature reconstruction for partial person re-identification: Alignment-free approach*, CVPR, 2018.

[40] Lingxiao He, Zhenan Sun, Yuhao Zhu, and Yunbo Wang, *Recognizing partial biometric patterns*, arXiv preprint arXiv:1810.07399 (2018).

[41] Xinwei He, Yang Zhou, Zhichao Zhou, Song Bai, and Xiang Bai, *Triplet-center loss for multi-view 3d object retrieval*, CVPR, 2018.

[42] Zhenwei He and Lei Zhang, *End-to-end detection and re-identification integrated net for person search*, ACCV, 2018.

[43] Alexander Hermans, Lucas Beyer, and Bastian Leibe, *In defense of the triplet loss for person re-identification*, arXiv preprint arXiv:1703.07737 (2017).

[44] Martin Hirzer, Peter M Roth, Martin Köstinger, and Horst Bischof, *Relaxed pairwise learned metric for person re-identification*, ECCV, 2012.

[45] Sepp Hochreiter and Jürgen Schmidhuber, *Long short-term memory*, Neural Computation (1997).

[46] De-An Huang and Yu-Chiang Frank Wang, *Coupled dictionary and feature space learning with applications to cross-domain image synthesis and recognition*, ICCV, 2013.

[47] Qingqiu Huang, Wentao Liu, and Dahua Lin, *Person search in videos with one portrait through visual and temporal links*, ECCV, 2018.

[48] Xun Huang, Yixuan Li, Omid Poursaeed, John E Hopcroft, and Serge J Belongie, *Stacked generative adversarial networks.*, CVPR, 2017.

[49] Sara Iodice and Krystian Mikolajczyk, *Partial person re-identification with alignment and hallucination*, ACCV, 2018.

[50] Sergey Ioffe and Christian Szegedy, *Batch normalization: Accelerating deep network training by reducing internal covariate shift*, ICML, 2015.

[51] 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 (2012).

[52] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, *Caffe: Convolutional architecture for fast feature embedding*, ACMMM, 2014.

[53] Srikrishna Karanam, Yang Li, and Richard J Radke, *Person re-identification with discriminatively trained viewpoint invariant dictionaries*, ICCV, 2015.

[54] Junyeong Kim and Chang D Yoo, *Deep partial person re-identification via attention model*, ICIP, 2017.

[55] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim, *Learning to discover cross-domain relations with generative adversarial networks*, arXiv preprint arXiv:1703.05192 (2017).

[56] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid, *A spatio-temporal descriptor based on 3d-gradients*, BMVC, 2008.

[57] Martin Koestinger, Martin Hirzer, Paul Wohlhart, Peter M Roth, and Horst Bischof, *Large scale metric learning from equivalence constraints*, CVPR, 2012.

[58] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, *Imagenet classification with deep convolutional neural networks*, NIPS, 2012.

[59] Brian Kulis, Kate Saenko, and Trevor Darrell, *What you saw is not what you get: Domain adaptation using asymmetric kernel transforms*, CVPR, 2011.

[60] 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 (2013).

[61] Worapan Kusakunniran, Qiang Wu, Jian Zhang, Yi Ma, and Hongdong Li, *A new view-invariant feature for cross-view gait recognition*, IEEE Transactions on Information Forensics and Security (2013).

[62] Toby HW Lam, King Hong Cheung, and James NK Liu, *Gait flow image: A silhouette-based gait representation for human identification*, Pattern Recognition (2011).

[63] Xu Lan, Xiatian Zhu, and Shaogang Gong, *Person search by multi-scale matching*, ECCV, 2018.

[64] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld, *Learning realistic human actions from movies*, CVPR, 2008.

[65] Ryan Layne, Timothy M Hospedales, Shaogang Gong, and Q Mary, *Person re-identification by attributes.*, BMVC, 2012.

[66] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE (1998).

[67] Dangwei Li, Xiaotang Chen, Zhang Zhang, and Kaiqi Huang, *Learning deep context-aware features over body and latent parts for person re-identification*, CVPR, 2017.

[68] Sheng Li, Ming Shao, and Yun Fu, *Person re-identification by cross-view multi-level dictionary learning*, IEEE Transactions on Pattern Analysis and Machine Intelligence (2018).

[69] Wei Li and Xiaogang Wang, *Locally aligned feature transforms across views*, CVPR, 2013.

[70] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang, *DeepReID: Deep filter pairing neural network for person re-identification*, CVPR, 2014.

[71] Wei Li, Xiatian Zhu, and Shaogang Gong, *Harmonious attention network for person re-identification*, CVPR, 2018.

[72] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li, *Person re-identification by local maximal occurrence representation and metric learning*, CVPR, 2015.

[73] Shengcai Liao and Stan Z Li, *Efficient psd constrained asymmetric metric learning for person re-identification*, ICCV, 2015.

[74] Yuewei Lin, Kareem Abdelfatah, Youjie Zhou, Xiaochuan Fan, Hongkai Yu, Hui Qian, and Song Wang, *Co-interest person detection from multiple wearable camera videos*, ICCV, 2015.

[75] Yuewei Lin, Jing Chen, Yu Cao, Youjie Zhou, Lingfeng Zhang, Yuan Yan Tang, and Song Wang, *Cross-domain recognition by identifying joint subspaces of source domain and target domain*, IEEE Transactions on Cybernetics (2017).

[76] Ce Liu et al., *Beyond pixels: exploring new representations and applications for motion analysis*, Ph.D. thesis, Massachusetts Institute of Technology, 2009.

[77] Hao Liu, Jiashi Feng, Zequn Jie, Karlekar Jayashree, Bo Zhao, Meibin Qi, Jianguo Jiang, and Shuicheng Yan, *Neural person search machines*, ICCV, 2017.

[78] Hong Liu, Wei Shi, Weipeng Huang, and Qiao Guan, *A discriminatively learned feature embedding based on multi-loss fusion for person search*, ICASSP, 2018.

[79] Kan Liu, Bingpeng Ma, Wei Zhang, and Rui Huang, *A spatio-temporal appearance representation for video-based pedestrian re-identification*, ICCV, 2015.

[80] Ming-Yu Liu, Thomas Breuel, and Jan Kautz, *Unsupervised image-to-image translation networks*, NIPS, 2017.

[81] David G Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision (2004).

[82] Hao Luo, Xing Fan, Chi Zhang, and Wei Jiang, *Stnreid: Deep convolutional networks with pairwise spatial transformer networks for partial person re-identification*, arXiv preprint arXiv:1903.07072 (2019).

[83] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel, *Understanding the effective receptive field in deep convolutional neural networks*, NIPS, 2016.

[84] Bingpeng Ma, Yu Su, and Frédéric Jurie, *BiCov: A novel image representation for person re-identification and face verification*, BMVC, 2012.

[85] ———, *Local descriptors encoded by fisher vectors for person re-identification*, ECCV, 2012.

[86] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool, *Pose guided person image generation*, NIPS, 2017.

[87] Liqian Ma, Qianru Sun, Stamatios Georgoulis, Luc Van Gool, Bernt Schiele, and Mario Fritz, *Disentangled person image generation*, arXiv preprint arXiv:1712.02621 (2017).

[88] ———, *Disentangled person image generation*, CVPR, 2018.

[89] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng, *Rectifier nonlinearities improve neural network acoustic models*, ICML, 2013.

[90] Laurens van der Maaten and Geoffrey Hinton, *Visualizing data using t-sne*, Journal of Machine Learning Research (2008).

[91] Behrooz Mahasseni and Sinisa Todorovic, *Regularizing long short term memory with 3D human-skeleton sequences for action recognition*, CVPR, 2016.

[92] Yasushi Makihara, Ryusuke Sagawa, Yasuhiro Mukaigawa, Tomio Echigo, and Yasushi Yagi, *Gait recognition using a view transformation model in the frequency domain*, ECCV, 2006.

[93] Niall McLaughlin, Jesus Martinez del Rincon, and Paul Miller, *Recurrent convolutional network for video-based person re-identification*, CVPR, 2016.

[94] ———, *Recurrent convolutional network for video-based person re-identification*, CVPR, 2016.

[95] Bharti Munjal, Sikandar Amin, Federico Tombari, and Fabio Galasso, *Query-guided end-to-end person search*, arXiv preprint arXiv:1905.01203 (2019).

[96] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski, *Plug & play generative networks: Conditional iterative generation of images in latent space.*, CVPR, 2017.

[97] Dat Tien Nguyen, Hyung Gil Hong, Ki Wan Kim, and Kang Ryoung Park, *Person recognition system based on a combination of body images from visible light and thermal cameras*, Sensors (2017).

[98] Jie Ni, Qiang Qiu, and Rama Chellappa, *Subspace interpolation via dictionary learning for unsupervised domain adaptation*, CVPR, 2013.

[99] Timo Ojala, Matti Pietikainen, and Topi Maenpaa, *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns*, IEEE Transactions on Pattern Analysis and Machine Intelligence (2002).

[100] Peixi Peng, Tao Xiang, Yaowei Wang, Massimiliano Pontil, Shaogang Gong, Tiejun Huang, and Yonghong Tian, *Unsupervised cross-dataset transfer learning for person re-identification*, CVPR, 2016.

[101] Bryan Prosser, Wei-Shi Zheng, Shaogang Gong, Tao Xiang, and Q Mary, *Person re-identification by support vector ranking.*, BMVC, 2010.

[102] Xuelin Qian, Yanwei Fu, Wenxuan Wang, Tao Xiang, Yang Wu, Yu-Gang Jiang, and Xiangyang Xue, *Pose-normalized image generation for person re-identification*, arXiv preprint arXiv:1712.02225 (2017).

[103] Alec Radford, Luke Metz, and Soumith Chintala, *Unsupervised representation learning with deep convolutional generative adversarial networks*, arXiv preprint arXiv:1511.06434 (2015).

[104] Min Ren, Lingxiao He, Haiqing Li, Yunfan Liu, Zhenan Sun, and Tieniu Tan, *Robust partial person re-identification based on similarity-guided sparse representation*, CCBR, 2017.

[105] Peter M Roth, Martin Hirzer, Martin Koestinger, Csaba Beleznai, and Horst Bischof, *Mahalanobis distance learning for person re-identification*, Person Re-Identification, 2014.

[106] Florian Schroff, Dmitry Kalenichenko, and James Philbin, *Facenet: A unified embedding for face recognition and clustering*, CVPR, 2015.

[107] Paul Scovanner, Saad Ali, and Mubarak Shah, *A 3-dimensional sift descriptor and its application to action recognition*, ACMMM, 2007.

[108] Zhiyuan Shi, Timothy M Hospedales, and Tao Xiang, *Transferring a semantic representation for person re-identification and search*, CVPR, 2015.

[109] Kohei Shiraga, Yasushi Makihara, Daigo Muramatsu, Tomio Echigo, and Yasushi Yagi, *Geinet: View-invariant gait recognition using a convolutional neural network*, ICB, 2016.

[110] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb, *Learning from simulated and unsupervised images through adversarial training*, CVPR, 2017.

[111] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian, *Pose-driven deep convolutional model for person re-identification*, ICCV, 2017.

[112] Chi Su, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian, *Deep attributes driven multi-camera person re-identification*, ECCV, 2016.

[113] Yifan Sun, Qin Xu, Yali Li, Chi Zhang, Yikang Li, Shengjin Wang, and Jian Sun, *Perceive where to focus: Learning visibility-aware part-level features for partial person re-identification*, CVPR, 2019.

[114] Rahul Rama Varior, Mrinal Haloi, and Gang Wang, *Gated siamese convolutional neural network architecture for human re-identification*, ECCV, 2016.

[115] Rahul Rama Varior, Bing Shuai, Jiwen Lu, Dong Xu, and Gang Wang, *A Siamese long short-term memory architecture for human re-identification*, ECCV, 2016.

[116] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba, *Generating videos with scene dynamics*, NIPS, 2016.

[117] Chen Wang, Junping Zhang, Jian Pu, Xiaoru Yuan, and Liang Wang, *Chrono-gait image: A novel temporal template for gait recognition*, ECCV, 2010.

[118] Faqiang Wang, Wangmeng Zuo, Liang Lin, David Zhang, and Lei Zhang, *Joint learning of single-image and cross-image representations for person re-identification*, CVPR, 2016.

[119] Shenlong Wang, Lei Zhang, Yan Liang, and Quan Pan, *Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis*, CVPR, 2012.

[120] Taiqing Wang, Shaogang Gong, Xiatian Zhu, and Shengjin Wang, *Person re-identification by video ranking*, ECCV, 2014.

[121] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian, *Person transfer gan to bridge domain gap for person re-identification*, CVPR, 2018.

[122] Kilian Q Weinberger, John Blitzer, and Lawrence K Saul, *Distance metric learning for large margin nearest neighbor classification*, NIPS, 2006.

[123] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, *A discriminative feature learning approach for deep face recognition*, ECCV, 2016.

[124] Ancong Wu, Wei-Shi Zheng, Hong-Xing Yu, Shaogang Gong, and Jianhuang Lai, *Rgb-infrared cross-modality person re-identification*, CVPR, 2017.

[125] 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*, WACV, 2016.

[126] 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 (2016).

[127] Jimin Xiao, Yanchun Xie, Tammam Tillo, Kaizhu Huang, Yunchao Wei, and Jiashi Feng, *Ian: the individual aggregation network for person search*, Pattern Recognition (2019).

[128] Tong Xiao, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang, *Learning deep feature representations with domain guided dropout for person re-identification*, CVPR, 2016.

[129] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang, *Joint detection and identification feature learning for person search*, CVPR, 2017.

[130] Fei Xiong, Mengran Gou, Octavia Camps, and Mario Sznaier, *Person re-identification using kernel-based metric learning methods*, ECCV, 2014.

[131] Shuangjie Xu, Yu Cheng, Kang Gu, Yang Yang, Shiyu Chang, and Pan Zhou, *Jointly attentive spatial-temporal pooling networks for video-based person re-identification*, arXiv preprint arXiv:1708.02286 (2017).

[132] Yuanlu Xu, Bingpeng Ma, Rui Huang, and Liang Lin, *Person search in a scene by jointly modeling people commonness and person uniqueness*, ACMMM, 2014.

[133] Yichao Yan, Bingbing Ni, Zhichao Song, Chao Ma, Yan Yan, and Xiaokang Yang, *Person re-identification via recurrent feature aggregation*, ECCV, 2016.

[134] Yichao Yan, Qiang Zhang, Bingbing Ni, Wendong Zhang, Minghao Xu, and Xiaokang Yang, *Learning context graph for person search*, CVPR, 2019.

[135] Yang Yang, Jimei Yang, Junjie Yan, Shengcai Liao, Dong Yi, and Stan Z Li, *Salient color names for person re-identification*, ECCV, 2014.

[136] Ting Yao, Yingwei Pan, Chong-Wah Ngo, Houqiang Li, and Tao Mei, *Semi-supervised domain adaptation with subspace learning for visual recognition*, CVPR, 2015.

[137] Mang Ye, Xiangyuan Lan, Jiawei Li, and Pong C. Yuen, *Hierarchical discriminative learning for visible thermal person re-identification*, AAAI, 2018.

[138] Mang Ye, Zheng Wang, Xiangyuan Lan, and Pong C. Yuen, *Visible thermal person re-identification via dual-constrained top-ranking*, IJCAI, 2018.

[139] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong, *Dualgan: Unsupervised dual learning for image-to-image translation*, arXiv preprint (2017).

[140] Hong-Xing Yu, Ancong Wu, and Wei-Shi Zheng, *Cross-view asymmetric metric learning for unsupervised person re-identification*, ICCV, 2017.

[141] Rui Yu, Zhiyong Dou, Song Bai, Zhaoxiang Zhang, Yongchao Xu, and Xiang Bai, *Hard-aware point-to-set deep metric for person re-identification*, ECCV, 2018.

[142] Matthew D Zeiler and Rob Fergus, *Visualizing and understanding convolutional networks*, ECCV, 2014.

[143] Li Zhang, Tao Xiang, and Shaogang Gong, *Learning a discriminative null space for person re-identification*, CVPR, 2016.

[144] Xuan Zhang, Hao Luo, Xing Fan, Weilai Xiang, Yixiao Sun, Qiqi Xiao, Wei Jiang, Chi Zhang, and Jian Sun, *Alignedreid: Surpassing human-level performance in person re-identification*, arXiv preprint arXiv:1711.08184 (2017).

[145] Haiyu Zhao, Maoqing Tian, Shuyang Sun, Jing Shao, Junjie Yan, Shuai Yi, Xiaogang Wang, and Xiaoou Tang, *Spindle net: Person re-identification with human body region guided feature decomposition and fusion*, CVPR, 2017.

[146] Liming Zhao, Xi Li, Jingdong Wang, and Yueting Zhuang, *Deeply-learned part-aligned representations for person re-identification*, arXiv preprint arXiv:1707.07256 (2017).

[147] Rui Zhao, Wanli Ouyang, and Xiaogang Wang, *Person re-identification by salience matching*, ICCV, 2013.

[148] ———, *Unsupervised salience learning for person re-identification*, CVPR, 2013.

[149] ———, *Learning mid-level filters for person re-identification*, CVPR, 2014.

[150] Kang Zheng, Xiaochuan Fan, Yuewei Lin, Hao Guo, Hongkai Yu, Dazhou Guo, and Song Wang, *Learning view-invariant features for person identification in temporally synchronized videos taken by wearable cameras*, ICCV, 2017.

[151] Kang Zheng, Hao Guo, Xiaochuan Fan, Hongkai Yu, and Song Wang, *Identifying same persons from temporally synchronized videos taken by multiple wearable cameras*, CVPR Workshop, 2016.

[152] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian, *Scalable person re-identification: A benchmark*, ICCV, 2015.

[153] Liang Zheng, Yi Yang, and Alexander G Hauptmann, *Person re-identification: Past, present and future*, arXiv preprint arXiv:1610.02984 (2016).

[154] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang, *Person re-identification by probabilistic relative distance comparison*, CVPR, 2011.

[155] Wei-Shi Zheng, Xiang Li, Tao Xiang, Shengcai Liao, Jianhuang Lai, and Shaogang Gong, *Partial person re-identification*, ICCV, 2015.

[156] Zhedong Zheng, Liang Zheng, and Yi Yang, *Unlabeled samples generated by gan improve the person re-identification baseline in vitro*, ICCV, 2017.

[157] Zhun Zhong, Liang Zheng, Zhedong Zheng, Shaozi Li, and Yi Yang, *Camera style adaptation for person re-identification*, CVPR, 2018.

[158] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros, *Unpaired image-to-image translation using cycle-consistent adversarial networks*, ICCV, 2017.

[159] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman, *Toward multimodal image-to-image translation*, NIPS, 2017.