Universitá Degli Studi di Udine

A Distributed Video Surveillance System to Track Persons in Camera Networks

by

Niki Martinel

Supervisor: Prof. Christian Micheloni

Anno Accademico 2013/2014

Abstract

This thesis focuses on the topics of information visualization in a video surveillance system and on distributed person re-identification.

Visualizing the proper information in a concise and informative fashion is a very challenging tasks that is mainly driven by the situation, the task that has to be performed and last but not least, by the operator that is using the system. Designing a successful system that is able to support all of these requirements and constraints is the first goal of this thesis. Towards this end we design and develop four system prototypes and evaluate each of them by means of standard Human-Computer Interaction principles. We show that this approach leads to an advanced system that is capable to support the task of tracking a person moving through multiple cameras field-of-views using only a single display.

The advanced visualization system was built to support the task of tracking a person through multiple overlapping cameras. However, in a real scenario this is not always feasible and we have to deal with disjoint cameras, hence, the system may fail to track the same person moving across them. In light of this, we propose three different methods to tackle the person re-identification problem so as we can re-associate a person that moves out from one camera and then reappears in another one at a different time instant. The first method builds a discriminative signature for each person that is matched by using a robust distance measure. The second method studies the transformation of features across cameras, while the last one builds upon the idea that as features get transformed so is the distance between them. Finally, we consider the issues of a fully centralized camera-camera re-identification system and introduce a distributed re-identification framework. For each approach, experimental results on public benchmark datasets are given.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Christian Micheloni for his constant support and guidance throughout these years. He introduced me to the interesting and challenging computer vision problems that absorbed most of my professional life over the past few years. I'm also grateful to Prof. Claudio Piciarelli and Prof. Gian Luca Foresti for their support and guidance. I'm very grateful to Prof. A. K. Roy-Chowdhury that treated me as a special guest in his lab. He was a source of inspiration for many of the ideas that are in this thesis. I'm very grateful to many of my teachers at different institutions who instilled in me the quest for knowledge and the courage to probe the unknown. I would also like to thank many of my colleagues and friends for their help and advice, without which this thesis would not have seen the light. Words cannot express my gratitude and indebtedness to my parents, who have given up so much in life in order that I could reach this stage today. Last but not least, I would like to express my deepest gratitude to my girlfriend Elisa, for patiently bearing with me through the preparation ot this thesis and supporting me throughout. She will always be a source of inspiration for me.

Contents

List of Figures

List of Tables

1 Introduction

The first chapter briefly introduces the current state of Video Surveillance Systems (VSSs), including application fields, typical architectures and possible drawbacks. Then, the topics of displaying footages and the proper information in a Video Surveillance System, and distributed re-identification are discussed in more details. Within this context, the contributions and the organization of this thesis are defined.

1.1 An Introduction to Video Surveillance

Nowadays there is a growing interest in surveillance applications due to both the increasing demand of more safety and security in urban environments and the constantly decreasing price plummet of sensors and processors. These facts, together with the maturity reached by algorithms and techniques, are introducing novel automatic surveillance systems able to monitor the remote and often unattended environments (e.g., metro lines and railway platforms, highways, airport waiting rooms or taxiways, nuclear plants, public areas, etc.).

To achieve such objectives, advanced surveillance systems can use many different kinds of sensors that range from tactile/pressure sensors (e.g., border surveillance) to chemical sensors (e.g., industrial plant surveillance or counter terrorism activities) to audio and visual sensors. For monitoring wide areas, the most informative and versatile ones are the visual sensors. When visual sensors (i.e. cameras) are used to monitor the behavior of people, objects or processes for security purposes we are talking about video surveillance.

The term "video surveillance" is very generic and the community uses it to refer to several types of systems differing both in their architectures and in their objectives. From a structural point of view, VSSs can be classified on the basis of:

- the type of sensors (e.g. b/w or color cameras, infra-red or even range cameras[1]);

[1] a *range camera* is a camera where the imaging process is based on the distance of the observed objects, rather than on their chromaticity/lightness properties: the nearer the object, the brighter its image.

- the sensors' degrees of freedom (i.e. fixed, zoom, Pan-Tilt-Zoom or fully mobile cameras);
- the system topology (i.e. single camera, network of cameras, hierarchical systems, etc.);
- the way data is transmitted (i.e. broadcast, CCTV—closed-circuit television);
- the final destination of the acquired data (i.e. displayed on a monitor, saved on storage units, processed by a software, etc.).

Despite this variety of possible architectures, the most popular systems are still the CCTV-based ones: video sequences acquired by typically static cameras are directly transmitted to a limited set of monitors so that human operators can remotely observe the monitored scene.

Since a camera-based system can be used in many different contexts, VSSs can also be classified by their purpose. Popular examples are:

- traffic monitoring systems, where cameras are used to check the traffic conditions and detect anomalies like car accidents or traffic jam;
- safety systems, like the ones mounted on subways so as the operators can check if the people are clear of doors before closing them and start the train;
- statistical analysis systems, as those used in some shopping malls, where the video data is used to collect statistical information for marketing purposes, e.g. finding when and where the flow of customers is higher so as to optimize the products displacement;
- industrial systems, where the surveillance system is used to detect possible anomalies in the production chain, e.g. defective pieces.

Even if video surveillance has been used in many different fields, its main application is in security management and law enforcement. Since the last decades, VSSs have been used for crime prevention, direct surveillance and forensic activities. Many times, prevention is generally achieved by simply deploying a camera in the environment. This usually discourages potential offenders by letting them think that they may be observed or even recorded while committing a criminal action; the dissuading power of such a system has even led to some extreme cases, where fake surveillance cameras with no imaging capabilities at all are installed in public areas.

In the last years, the number of installed VSSs has had a huge growth. This is mainly due to an increased demand for security pushed by the terroristic acts that have recently struck both Middle East and western countries. Governments, institutions and even private entities aimed to increase public security by means of video surveillance: today, CCTV systems can be found almost everywhere, in banks, mall shops, train stations, airports, and more generally in any public crowded place. This growth has led to some exceptional cases, as in the United Kingdom: it is estimated that more than 500,000 CCTV cameras cover large portions of the city of

London, while in the entire country there are more than 4,200,000 cameras, one for every 12 people [101].

The quick growth of VSSs leads to several new practical problems. Traditional CCTV camera systems are becoming less suitable for modern applications since they mainly rely on the interaction with a human operator. Let us consider the surveillance of a large public area, e.g. an airport: there could be hundreds of cameras operating at the same time and transmitting multiple video streams to the CCTV system monitors. Considering the large number of monitors and knowing that the operator's attention quickly decreases through time, such a system would require a prohibitive amount of human resources in order to be fully operational. Since, in practice, human resources are limited, the result is an inefficient system where potential events of interest may be missed due to operators' faults.

The practical problems mentioned above have led to an increasing interest in computer-based automatic or semi-automatic surveillance systems. In a computer-based surveillance system, appropriate computer vision algorithms are applied to video data streams coming from cameras in order to automatize some of the tasks of a traditional system. The system could perform relatively simple tasks such as the detection of moving objects, up to complex activity analysis for the identification of anomalous behaviors. A computer-based surveillance system can outperform a traditional one and support the human operator activities by filtering the potentially interesting video sequences that need further investigation: this way, a human operator is required to directly inspect the video sequences only when really needed. This reduces the negative impacts of the decrease of attention.

1.2 Displaying the Proper Information in a Video Surveillance System

Computer-based video surveillance applications covers several processing tasks, starting from low-level image processing tasks up to more abstract, high-level tasks concerning the interpretation of what the image sequences represent. In literature, this high-level part has been referred to with several names: *event detection* [102, 134, 71], *activity recognition* [67, 16, 139] *behavior analysis* [143, 61], *scene understanding* [18]. Though such names have slightly different meanings, the main idea is always to give a semantic interpretation of the monitored scene. While the low-level processes deal with pixels, colors and light intensities, the higher-level ones do not consider the acquired images as a mere group of pixels, but as something with a meaning that must be extracted. At the beginning of the processing chain the system deals with raw data, and the main tasks could be the detection of edges or fast-changing image regions, while at the end of the process the system deals with entities, like cars or people, and their interactions with other entities or the surrounding environment. Despite such useful information can be extracted, surveillance systems generally involve multiple and different cameras that are monitoring the same environment, so, the support of human operators is something that is still required.

1.2.1 Why is it so Important to Display the Proper Information?

Presenting the useful information to the human operators is a challenging task as footages from many different cameras should be displayed at the same time. As shown in Figure 1.1, current systems usually display such video sequences through a large number of monitors. This creates the problem of requiring a prohibitive amount of human resources and brings to a quick decrease of the attention of the human operators through time. Not only that, most of the information that is usually displayed is useless. Displaying such information is not only overloading the operators capabilities, but it is also preventing them to catch the relevant events that may be worth to further investigate.

Therefore, trying to display the proper information in an easy and informative fashion is a very important task. However, due to the large amount of data that needs to be processed and the difference between all the possible tasks that operators are required to perform, it is still a challenge and it has only recently started to attract the attention of the community.

1.2.2 Issues in Current Video Surveillance Systems

Displaying the proper information through a VSS is an extremely challenging task. There are several problems that affect it and among them we can just mention a few.

- First, the information that should be displayed to the operators is task-dependent, that is, the system should present the operators only the information that is useful for the task they are currently performing. So, the information to be displayed is not generally well-defined nor unique.
- Second, even if the information to be displayed can be precisely defined, is not uncommon that different operators disagree on what the proper information is (with "proper" we mean relevant to the specific task).
- Third, different operators have different physical and psychological capabilities, so the tasks they can complete in a proper amount of time (this is very important in video surveillance as the decisions taken in fraction of seconds can save many lives) and in the correct way are different and thus require different solutions.

Analyzing in details the stated issues we may claim the following. In the first case the problem arises from the multitude of current surveillance tasks that human operators are required to perform. For instance, let's see what can be the proper information to be displayed for the same event happening under two different scenarios. Let's consider the problem of monitoring the entrance of a subway. During the ordinary operations an operator may want to have information about the identity of people entering in the subway, e.g. he/she may want to know if a suspicious person is getting in, or whether a person is bringing in the subway offending objects like weapons,etc.. However, after an accident/attack occurred, the same operator may

Figure 1.1: A typical surveillance station where many monitors are used to display footage coming from different cameras. The operators are generally required to simultaneously monitor all of them and to take the appropriate decisions in a limited amount of time.

want to obtain different information. For instance he/she may want to have information about which entry the offender is exiting or which subway access is the safest one so as people can easily leave the subway. These are only few examples of the possible variations of one single event. Nevertheless, though such information cannot be automatically extracted from footages or properly displayed, and the process requires a huge mental effort, a human being would generally complete the tasks.

The second problem is more related to each human being. For instance, let us keep on thinking about the previous example where a suspicious person has been identified. Suppose also that the system tells the operator that a weapon has been detected and the subject is carrying it into the subway. One operator may want the system to display the most probable direction that the subject will take so as the police can intervene and stop he/she as soon as possible. Another operator, instead, may also want the system to display the criminal record of the subject so as he/she can better understand who is the person they'll face. Still, another operator, may want the system to display the previous positions where the subject was detected, etc..

These many different needs make the task of visualizing the proper information even more challenging as the system may have a different behavior depending on which operator is monitoring the area.

Finally, the third issue is probably the most challenging one. We, as humans, have different cognitive capabilities, reaction times, and each of us has different skills. Thus, the system should display to the different operators the information in a proper way and in a sufficient amount of time such as they can take the appropriate decisions. Let for instance consider two operators one of which has a colorblind problem while the other one has not. If the system is not designed to address the problems coming from such difference between the operators' sight capabilities, in case of an anomalous/interesting event, one of the operators may fail in detecting it and thus he/she does not properly complete the monitoring task. Commonly proposed surveillance system do not generally consider such design issues and do not take the appropriate considerations when displaying the relevant information to operators having these kind of problems.

Even though these are mainly Human-Computer Interaction (HCI) design issues, to address them, low-level as well as high-level semantic information should be extracted from the footages using computer vision based algorithms. One of the most interesting problems, related to both the HCI stated issues and to the high-level semantic information that can be extracted from the footage, is the problem of tracking a person across multiple disjoint cameras, that is, the person re-identification problem. The problem is interesting from the HCI aspect the proper information should be displayed in an easy and intuitive fashion without overloading the operator cognitive capabilities, e.g. the system should not require the operator to follow the persons by merely requiring him/her to look at the different monitors through which footages are displayed. From the computer vision point of view, such task is even more challenging as the system should be able to reliably perform the data association problem that arise when the object is leaving one camera FoV to reappear in a different one at a different instant of time.

1.3 Re-Identification/Tracking Across Disjoint Cameras

The size of the monitored environment introduces many different problems, from the number of sensors to deploy, to their configuration, to the way they communicate and cooperate to achieve a global objective. As a matter of fact, as the dimension of the monitored environment grows, it quickly becomes hard to deploy a network of video sensors such that there are enough overlapping FoVs to cover every point of the monitored area. In this context, even though sensors are becoming cheaper, a full coverage of the area is still not affordable due to the amount of human supervision, privacy concerns, and maintenance costs involved [34]. In addition, monitoring every point of the environment implies high-computational costs and high-speed (i.e. bandwidth) networks.

These limitations yield to the development of video analytics systems that provide partial area coverage. This introduces the blind areas called "blind-gaps" that bring in new challenging problems as no information can be obtained from these areas. One of the most interesting problems is to re-identify people moving across different FoVs through "blind-gaps". This is known as the person re-identification problem, formally defined as the problem of associating a given person acquired by a camera to persons previously acquired by any other camera in the network at any location and at any time instant.

1.3.1 Why Re-identification?

Knowing whether a specific person is present in a given scene, at a given position and time is of paramount importance for surveillance tasks. By attacking such problem, the video data coming from disjoint cameras can be used to recognize the author of a crime or to reconstruct the sequence of events before it (see Figure 1.2 for an example).

1.3.2 Issues in Current Re-Identification Approaches

One of the biggest problems the re-identification community must face is the quality and the huge difference between the image frames coming from the disjoint cameras. The re-identification problem is addressed relying on data extracted by image processing modules, able to detect moving objects and classify them. Many state-of-the-art works on re-identification still make the assumption that the data coming from low-level modules is free from errors, but unfortunately this is not the case. Such an hypothesis could be useful when defining a system from a theoretical point of view, but in practical applications low-level errors could have negative impacts on the performances of high-level processes. Nowadays, probably no computer vision problem can still be considered totally solved, and in fact yet today the community produces works on the same topics that are being addressed since decades, such as object detection and tracking; the hypothesis of error-free data should thus be considered unrealistic. Not only that, assumed that the error-free data is available, the task of re-identifying persons moving across cameras is challenging due to the open issues of multi-camera video analysis such as changes of scale, illumination, viewing angle and pose. This is especially true in case of person re-identification due to the non-rigid shape of the human body.

State-of-the-art methods have tried to address such problems by designing robust features that aim to describe a target across different cameras or by finding the optimal distance measures between features. An alternative class of approaches has looked into the problem of finding linear and nonlinear transformation functions between appearance features extracted from image pairs. These functions are used to transform the features extracted from the candidate targets and to perform the re-identification in the transformed feature space. Despite this, target re-identification in a non-overlapping multi-camera scenario is still an open issue due to the unknown nature of the transformation of features between cameras.

(a) (b)

(c) (d)

Figure 1.2: Images acquired by 4 different and disjoint CCTV cameras before the Boston Marathon bombing attack on April 15, 2013. In (a) the authors are viewed from the front. In (b) the authors are viewed from the back. In (c) and (d) the authors are viewed from different side views.

While recent methods are achieving good re-identification performance, those have mainly focused on performing the re-identification between camera pairs and do not consider the re-identification from the network point of view. Thus, the related computational and networking costs involved in such process have not received attention by the re-identification community. However, this is a very interesting and challenging aspect as the process of re-identify a target generally involves high computational resources and very high dimensional image representations.

1.4 Distributed Computations

Due to the availability of modern low-cost sensors, wide area camera networks are gaining increasing importance in a wide range of applications like surveillance, disaster response, environmental monitoring, just to mention a few. Multiple sensors can cover a wider area, provide views from different angles and the fusion of all their measurements may lead to robust scene understanding. Among different information fusion approaches, distributed network of smart cameras [21, 122] are often chosen over centralized or hierarchical approaches due to their scalability to a large number of sensors, ease of installation and high tolerance to node failure. The development of such automated techniques for aggregating and interpreting information from multiple video streams in real-life scenarios is a challenging area of research [35].

1.4.1 Why not a Centralized Approach?

Visual sensor networks generally come with a relatively large amount of computational and storage resources, but these are spread, both spatially and topologically. Consequently, random access to a distant resource is very expensive in terms of required network bandwidth, especially in wireless multi-hop networks. While this issue can be trivially addressed by copying all data and letting each node processing it separately, this does not solve the polynomially-increasing communication burden.

The main characteristic of a fully-centralized architecture is the ability of the processing node to locally access any piece of stored information. At the current state of technology, a central processing node would be implemented as a single computer or tightly connected computing cluster with local memory (RAM) and locally attached storage. The processing power and storage capacity of such centralized server are assumed to be sufficient to process all data from all attached cameras simultaneously. Thus, in general an algorithm that exploits a centralized approach had to face the following constraints

- Bandwidth constraints
- Security issues
- Difficulties in analyzing a huge amount of data

However, in large scale camera networks the stated assumptions are unrealistic. With the growing network size, limits on network throughput and processing power of the central unit are reached sooner or later. As an alternative solution, processing in a very large camera network can be organized in a truly distributed manner using smart cameras, thus providing enough processing power and storage capacity even for complex tasks. Depending on the nature of a system, there may be additional reasons in favor of a decentralized implementation, such as the need for redundancy or resilience to sabotage. In a network of embedded smart cameras, while, from the system viewpoint, nodes are spread across the network, each sensor has local memory and storage, and it is capable of performing processing operations on board.

Consequently, in a distributed framework each of the cameras in the network act as autonomous agent that is able both to record and analyze the data locally. But the cost of data access depends on the physical and topological distances between network nodes, which is an inherent property of distributed systems in general. So, when performing processing operations in a network of embedded smart cameras, we can move the computational steps to the edge of the network and leave each node perform the required actions. Then each sensor exchanges only the proper information to its neighbors to finally reach a shared and global objective in a fully distributed fashion.

1.5 Contribution of the Thesis

The contribution of this thesis is two-fold. First, an advanced VSS is designed to display the proper task-dependent information to surveillance operators that are monitoring a wide area. The system is mainly designed to help operators tracking persons across camera views. This raised the need for a system capable of re-identify the subjects as they move through disjoint cameras FoVs. This leads to the second contribution of the thesis, that is, a distributed approach to address the challenges of the person re-identification problem. Three different approaches are investigated to show the performance and the positive and negative aspects of each of them. Specifically, this thesis makes the following original contributions.

Adaptive Human Interface for VSS

The main objective of the proposed VSS is the development of an effective and powerful information visualization technique that properly displays only the most relevant cameras and information contents to simplify the operators' tracking tasks. The key idea is to visualize only most probable streams, i.e. those that will be involved with the motion of the tracked persons. Towards this goal we propose a novel dynamic organization, activation and switching of the User Interface (UI) elements based on the output of video analytics algorithms.

The first objective is to distill the volumes of monitoring information into a human manageable quantity. This is achieved by introducing an hand-off task between different camera views so that a single person can be tracked across different FoVs. The proposed camera planning algorithm uses geographical clues and exploits the predicted trajectories to build an accurate camera activation plan. The camera activation plan together with the tracking data is used to provide only the relevant data to the novel UI.

The next and final objective is to present the filtered visual information to the operators such that they can take appropriate decisions in a limited amount of time. This is achieved by first exploiting the activation plan and tracking data such as only the proper streams are selected. Then, the selected video streams are displayed through a novel UI that allows the operators to focus only on a single view without requiring them to switch between monitors as well as UI elements. A map represen-

tation that exploits the *detail plus overview* technique is also introduced to make the task of inspecting the whole are less tough.

Distributed Re-Identification

In the adaptive human interface for VSS the main goal is to support surveillance operators in the task of tracking persons moving within the monitored environment. However, as a target exits a camera FoV it should be re-identified as it enters a different one, so as the task of tracking targets across camera FoV can be tackled. This is where the person re-identification problem comes into picture. To tackle the re-identification problem state-of-the-art methods use robust features that have invariant properties across cameras. However, the communication and processing resources needed to deal with such large amount of data that has to be shared across the whole network, make the problem intractable if a centralized approach is adopted. To address the re-identification challenges we investigate three main approaches and introduce a distributed framework.

In the first approach we address the re-identification by means of a discriminative signature based method. Given an image, we first detect the persons together with their body parts using camera specific learned models of those. Then, after finding the silhouette of a person we extract four local and global features to create a discriminative signature of a person. This is finally matched with other signatures from other cameras using a weighted combination between local and global feature distances. While this method is effective for images that have similar appearance, it is not capable of dealing with even simple color variations that occur between cameras.

To tackle this issue, in the second approach, we propose to study the nature of the transformation between appearance features extracted from different cameras. Towards this goal we first detect the salient body parts from the given person image, then we extract color and texture features from local dense patches. For each pair of features extracted from two images, we capture the transformation of those by exploiting the principles of the Dynamic Time Warping technique. We form the function space of all feasible and infeasible transformations between such features. Then, we reduce the dimensionality of such function space and learn the decision boundary that bests separates the set of feasible and infeasible transformation functions. Finally, we perform the re-identification by classifying the transformation as feasible (i.e. the person is re-identified) or infeasible (the person is not re-identified). This method strongly outperforms the previously proposed one when large color and illumination variations are present, however, due to the very high dimensionality of the function space it is computationally expensive.

To try to reduce the required high computational costs and introduce a tractable solution that can be later extended to perform the re-identification over the whole network, we build upon the idea that as the features get transformed across cameras so are the differences between them. So, we address the re-identification as follows. We first extract color, texture and shape features from the given images to capture most of the discriminative information, then for each pair of images coming from disjoint cameras we compute the distances between all such features. This results

in a small feature space where, as before, we can train a classifier and learn the parameters of the decision boundary that separates the set of positive pairs (the two images are from the same person) and the set of negative pairs (the two images are from different persons). While being simple yet effective, this method still considers the re-identification from the point of view of only two cameras in the network.

To move this point of view and consider the re-identification as a network process we finally propose a distributed re-identification framework. To achieve a distributed re-identification we first introduce a camera matching cost measure, then we use it in a derivation of the Distance Vector (DV) routing algorithm. This allows us to route the signature of a probe person to the nodes of the network in a priority fashion. Not only that, using an update rule similar to the one proposed by the original DV routing algorithm the network is capable of adapting through time. While we present such framework using the discriminative signature based method it can be easily extended to other ones.

1.6 Organization of the Thesis

The thesis is organized along the lines for the previous section. In Chapter 2 we give a brief literature review about current VSS and re-identification approaches. The advanced and user centered VSS is described in Chapter 3. Then in the next three chapters the problem of person re-identification is addressed in three different ways. In Chapter 4 features extracted from persons silhouettes are accumulated over multiple frames to form a discriminative person signature. The study of the nature of the transformation of features, applied to the problem of person re-identification, is described in Chapter 5. Then, in Chapter 6 two preliminary studies are conducted to explore the transformation of feature dissimilarities and to extend the proposed camera-camera re-identification approaches to the camera network point of view. Finally, conclusions and possible future directions are drawn.

2

Literature Review

In this chapter, a brief literature review of current video surveillance, with focus on works in the field of user interfaces for VSS, is given. Then we discuss state-of-the-art works for person re-identification. As current person re-identification methods can be grouped into two main different categories,i.e. biometric-based and appearance-based methods, this part first analyzes the re-identification methods along this categorization. Then, since appearance-based methods are the most widely used in camera networks, an even finer distinction is made between them. Discriminative signature based methods, metric learning based methods and feature transformation based methods are discussed. Finally, evaluation methodologies and commonly used person re-identification benchmark datasets are described.

2.1 User Interfaces for Video Surveillance Systems

The considerations expressed in the previous chapter justify the paramount importance that video surveillance has gained in the computer vision field during the last years (see for example the several special issues published on this topic [29, 44, 24, 2, 121, 116]). Countless works have been proposed both on the low-level part of the video processing chain (e.g. moving object detection, object tracking) and on the high-level part (e.g. video understanding, event detection). Despite this, only in few cases the research in this field has led to the development of a full surveillance system; this is mainly because of the inherent ambiguity in high-level tasks (e.g. how to uniquely define what an "interesting event" is?) and the difficulties in linking low- and high-level modules: high-level surveillance algorithms typically rely on good features extracted by the low-level modules, but this is often not the case.

The computer vision and video surveillance community have mainly focused on algorithms to extract valuable information from footages. Despite most of these algorithms are efficient and have high performance, the human part is still involved in the process of monitoring video streams from multiple cameras.

As pointed out in [124], the human ability to understand and interact with a large amount of data could be increased through visual analytic tools. A perceptual user

interface that allows users interaction by means of gestures was introduced in [64]. Common gestures are used to simplify the user-interaction with the video analytic system. Despite this less attention have been posed on UI elements. In [155] an attention-aware human-machine interface (HMI) to monitor human operators attention was proposed. The VSAM project described in [30] demonstrates that a single human operator can effectively monitor a significant area of interest. The proposed UI exploits the VSAM technology to automatically display graphical representations of individuals into the digital environment. The ADVISOR system [130] selects relevant outputs and displays the relevant video feeds to the operator using a novel HCI. In [144] a framework for video surveillance based on the context of the experiential environment for efficient and adaptive computations was proposed. An analysis procedure is used to select only the interesting data thus avoiding exhaustive analysis of the irrelevant data. In [50] a Dynamic Object Tracking Systems introduced a novel VSS user interface. The same authors extended it by inspecting activity patterns [52] and introducing geometric tools [51]. A new backbone system that was used to develop advanced monitoring techniques, integrating cameras installed around the monitored area and centralized information, was introduced in [106]. In [20] a two-tiered VSS that self-adapts to current user needs was proposed. Similarly, in [28], the Virtual Document Planner was introduced to reduce the visual clutter and to display only situation-tailored information.
Similar techniques were proposed in commercial products. The IBM Smart Surveillance system (S3) [129, 65] uses a web-based service interface to support video based behavioral analysis. The Smart Surveillance Engine (SSE) provides a plug and play framework for video analytics. A web-based UI is used to access to past events using metadata information and a SQL based query language. In [132] an integrated command and control solution designed to support security management is proposed. 3D site maps are displayed together with useful information to help contain and prevent dangerous events. Similarly, in [133] a graphical model of the monitored site allows users to select specific areas in order to display footages related to anomalous events. Finally, the Tag and Track system [66] allows users to select and track people across different camera views. The UI uses geographical cues and colors to display information and the tracking history of detected persons. In Table 2.1 the relevant properties -to be considered with respect to the HCI principles- of the state-of-the-art systems are shown.

Despite many of these works help improving end-users capabilities, they still require huge mental efforts to the human operators. In particular, three main open issues can be defined: i) each user is required to monitor a large amount of footages at the same time; ii) tasks like tracking across multiple cameras require manual interaction with the UI to select desired camera views; iii) the position and the colors of UIs elements are not chosen accordingly to HCI principles.

Table 2.1: Main properties of commercial and research systems.

System	Wide Areas	Crowded Scenarios	On-line	Retrieve data from repository	Multiple Object Tracking	Area Map	Multi-camera Visualization
VSAM [30]	✓		✓		✓	✓	
ADVISOR [130]		✓	✓	✓	✓	✓	
DOTS [50, 52, 51]			✓			✓	✓
IBM (S3) [65]	✓			✓			✓
Siemens Surveillance Vantage [132]	✓		✓	✓		✓	✓
Siemens SiteIQ [133]	✓		✓	✓		✓	
Ipsotek [66]	✓	✓	✓		✓	✓	✓

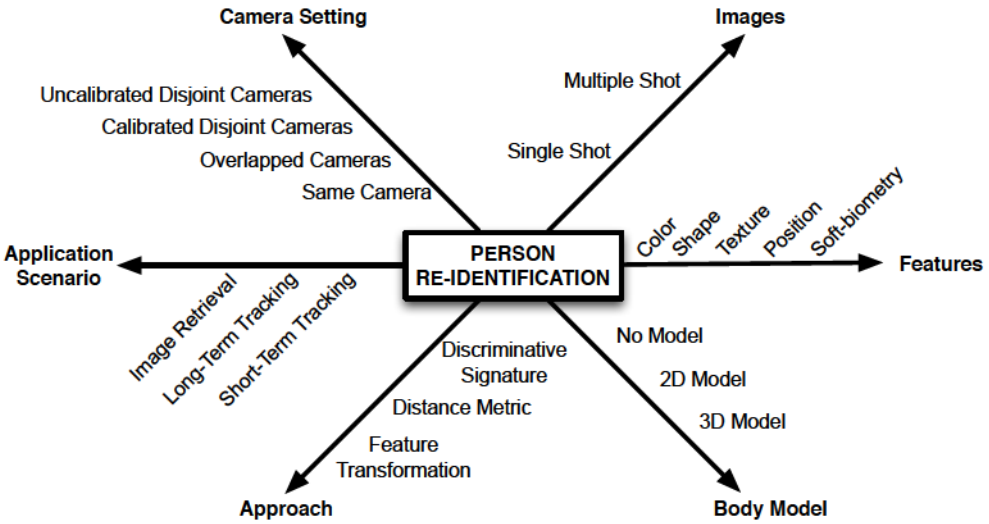


Figure 2.1: Multidimensional taxonomy for person re-identification algorithms.

2.2 Person Re-Identification

In the last few years the problem of re-identifying persons across multiple disjoint cameras has received increasing attention. Due to this increasing attention, the proposed approaches to the person re-identification problem used many different combination of appearance- or biometric-based features, body models,etc.. Such multidimensional taxonomy and categorization of the person re-identification algorithms is depicted in Figure 2.1. A deep analysis of all these possible categorization is out of the scope of this thesis, so, for a more comprehensive analysis the reader can refer to the recent survey given in [142]. According to this, in the rest of the section, an introduction to biometric-based methods is given, then appearance-based methods are analyzed with respect to their approach modality.

2.2.1 Biometrics-based Methods

Biometrics-based methods exploit passive biometric features to build persons signatures. Despite the efforts in the field, it is hard to extract biometrics-based features from an unconstrained environment and a precise camera configuration and sensor deployment are still required. It's a matter of fact that outdoor deployments are not able to handle the great variation of person poses to allow a reliable feature extraction for biometric re-identification. Not only that, due to the wide FoV of monitoring camera the low spatial resolution of images is not suitable to extract reliable features from either the face or the gait of a detected person. Because of this, biometrics-based methods are generally applied to indoor environments. Despite those issues, efforts have been pushed to perform the re-identification using biometric features.

Gait-based biometric features are the most used ones for re-identification purposes. In [145] a gait-based recognition algorithm based on spatial-temporal silhouette analysis has been proposed. A background subtraction algorithm and a correspondence procedure are used to extract the silhouettes, then, a Principal Component Analysis (PCA)-based eigenspace transformation is applied to time-varying distance signals. A supervised learning method is used to perform the re-identification in the eigenspace. In [68] a view independent person re-identification method is proposed. 3D models of targets are synthesized from images acquired from 16 different cameras and gait features are extracted from those images acquired from different viewpoints. In [56] the Gait Energy Image (GEI) spatio-temporal gait representation was introduced to characterize human walking properties for individual recognition by gait. Real sequences and synthetic sequences are used within a statistical approach for learning effective features from real and synthetic templates. In [17] a hybrid dynamical model of human motion was used, together with a classification algorithm, to recognize human gaits. Temporal statistics are extracted from the images, and used to infer a dynamical model that explicitly represents ground contact events. An algorithms is then used to estimate model parameters, and a distance measure between such models is used to reconize an individual gait. In [138] an approach for comparing two sequences of deforming shapes using both parametric models and nonparametric methods was proposed. Kendall's definition of shape is used for feature extraction. Then parametric models like the autoregressive model and autoregressive moving average model on the tangent space were used to capture the nature of human gait. A modification of the Dynamic time-warping algorithm was also used to consider the nature of the non-Euclidean space in which the shape deformations take place. In [149] the robust sparse coding (RSC) based classification was introduced and applied to the task of face recognition. The RSC seeks for the maximum likelihood estimation solution of the sparse coding problem using an efficient iteratively reweighted sparse coding algorithm. The learned dictionary is then used for face recognition.

Face-based biometric features were also used for re-identification purposes. In [42] the person re-identification is applied to detect persons in TV shows. The task is addressed using features extracted from the faces by considering temporal association of them across whole video tracks. Each detected face is divided into non-overlapping 8×8 pixel blocks. A dense vector of DCT coefficients is computed from applyind DCT to each block. All the dense DCT coefficient vectors are concatenated to build the feature vector. Similarly, in [10] features are extracted from faces detected using a cascade of boosted MCT feature histograms. DCT coefficients are exploited to train a SVM in order to build the model and classify new examples. Finally, in [31] color and texture-based soft biometric features extracted from hair and skin patches are proposed.

Table 2.2: Main contributions in the field of person re-identification

Authors	Year	Approach	Features	Temporal Information	Representation
Javed *et al.* [69]	2005	Feature Transformation	Color	Yes	Color appearance with color brightness transfer function (BTF)
Gilbert *et al.* [49]	2006	Feature Transformation	Color	Yes	Consensous-color conversion of munsell color space with color transformation matrix
Gheissari *et al.* [48]	2006	Discriminative Signature	Color and shape	Yes	Graph partition based representation
Hu *et al.* [63]	2006	Discriminative Signature	Geometry	Yes	Principal axis with segmentation
Wang *et al.* [146]	2007	Discriminative Signature	Color, gradients and shape	No	Co-occurrence spatial context
Chen *et al.* [25]		Feature Transformation	Color	Yes	Color appearance with temporal color brightness transform and spatial information
Prosser *et al.* [118]	2008	Feature Transformation	Color	Yes	Color appearance with temporal color brightness transform and spatial information

Continue on next page

Table 2.2 – Continued from previous page

Authors	Year	Approach	Features	Temporal Information	Representation
Javed *et al.* [70]	2008	Feature Transformation	Color	Yes	Color appearance with spatial temporal color brightness transform and spatial information
Gray and Tao[53]	2008	Discriminative Signature	Color, gradients and filters	No	Selected histogram features by Adaboost
Zheng *et al.* [153]	2009	Discriminative Signature	Color and gradients	No	Grouping as dynamic spatial context
Bak *et al.* [6]	2010	Discriminative Signature	Color	No	Covariance matrix between parts
Farenzena *et al.* [41]	2010	Discriminative Signature	Color and structure	No	Symmetry-based ensemble of local features
Prosser *et al.* [119]	2010	Metric Learning	Color, gradients, filters	No	Quantified histogram feature by RankSVM
Cheng *et al.* [26]	2011	Discriminative Signature	Color and structure	No	Pictorial structures modeling
Dikmen *et al.* [36]	2011	Metric Learning	Color	No	Large Margin Nearest Neighbor with Rejection on densely sampled color histogram features
Kostinger *et al.* [76]	2012	Metric Learning	Color and texture filters	No	KISS Metric Learning on densely sampled color and texture features

Continue on next page

Table 2.2 – Continued from previous page

Authors	Year	Approach	Features	Temporal Information	Representation
Datta *et al.* [32]	2012	Feature Transformation	Color and shape	No	Weighted brightness transfer function on color and shape histogram features
Kviatkovsky *et al.* [78]	2013	Discriminative Signature	Color	No	Capturing color shape distribution in the log-chromaticity color space using shape context descriptor
Zhao *et al.* [152]	2013	Discriminative Signature	Color and gradients	No	Bi-directional weighted matching on densely sampled color and SIFT features
Li *et al.* [82]	2013	Feature Transformation	Color, shape and texture filters	No	Metric learning on locally aligned color, shape and texture histogram features
Zheng *et al.* [154]	2013	Metric Learning	Color and texture filters	No	Relative distance comparison on color and texture features

2.2.2 Appearance-based Methods

Appearance-based methods exploit appearance features to build a person's specific signature by assuming that people do not change clothes within the "blind-gaps". Since the person re-identification problem can be defined as an association problem in a wide area camera network where the goal is to track persons across the "blind-

gaps", this is a reasonable assumption to rely on. Moreover, as shown in [46], clothes represent a meaningful feature that allows even humans to distinguish between individuals. Differently to the biometric approaches, appearance-based methods do not require a precise camera deployment and can be applied to both indoor and outdoor environments. Due to this, as shown in Table 2.2, appearance-based methods are the most widely used for person re-identification.

Existing appearance-based works predominantly focus on finding the best set of features [85], the most discriminative models [92] or the optimal similarity measure [154] that can be used to represent and match a target viewed in different cameras at different time instants [37]. Following the approach categorization shown in Figure 2.1, three main kind of approaches can be identified: i) discriminative signatures based methods, ii) metric learning based methods, iii) transformation learning based methods.

2.2.3 Discriminative Signatures Based Methods

Person representations by means of color, shape and texture features have been the most common choice for discriminative signature based methods. In [46] segmented clothing regions are used to extract color and textures histograms that together with face features build persons' signatures. Similarly, in [135] a color-position histogram descriptor is computed on image regions that can be clustered according to their colors. Persons are then re-identified using an algorithm based on spectral analysis and Support Vector Machines (SVM). A dense grid structure method over feature distribution has been exploited in [7] to compute the Mean Riemannian Covariance Grid (MRCG) descriptor. In [48] a region-based segmented image is used to extract spatio-temporal local feature from multiple consecutive frames of each person. The proposed signatures are built upon a decomposable triangulated graph that captures the spatial distribution of the local descriptions. In [146] a two-layer appearance method is proposed. A layer computes the Histogram of Oriented Gradients in the Log-RGB color space. The other captures the spatial relationships between appearance labels.

The ensemble of localized features (ELF) approach [53] addressed the viewpoint invariant pedestrian recognition issue using an AdaBoost framework to learn the most discriminating local features. In [5] the AdaBoost framework is applied to haar-like and dominant color features extracted from multiple frames. An unsupervised approach to learn the most discriminating feature extracted from different individuals has been proposed in [85] to determine the feature importance of an individual driven by person's appearance attributes. In [73], the Implicit Shape Model (ISM) and SIFT descriptors are exploited to generate view specific identity models of persons. These models have been used to convert signatures between different views. In [47] a 2D rigid part based color appearance model is used to localize and match individuals in 3D system computed by means of the structure-from-motion technique. The possible location and contextual cues are exploited through an Markov Random Field (MRF) framework to perform the re-identification. A high-dimensional signature formed of texture, gradient and color features is proposed in [128]. The re-identification is performed by projecting the high-dimensional signature into a low-dimensional dis-

criminant latent space by Partial Least Squares (PLS) reduction. In [84] pairwise dissimilarity measures between people representations has been adapted for a nearest neighbor classification method. In [100] environmental cues and appearance features extracted from a vertical stripe around the head location are used to propose a Landmark-Based Model (LBM). An association phase integrates information from appearance features and candidate positions to perform the re-identification. In [26] Pictorial Structures (PS) were used to recognize individuals by selectively focusing on the body parts. For single image re-identification, PS were used to localize the body parts, extract and match their descriptors. When multiple images of a single individual are available, PS were used to learn the appearance of an individual. The proposed method is based on the statistical learning of pixel attributes collected through spatio-temporal reasoning. In [13] a method to characterize the appearance of individuals exploiting body visual cues was proposed. A symmetry-driven appearance-based descriptor was proposed to encode three complementary visual characteristics of the human appearance: the overall chromatic content, the spatial arrangement of colors into stable regions, and the presence of recurrent local motifs with high entropy. Such features are extracted by following symmetry and asymmetry perceptual principles. A weighted similarity measure between feature descriptors is finally used to perform the re-identification. In [86] an unsupervised approach was used to find the best features for re-identification. Features extracted from different individuals were weighted adaptively by their salient and inherent appearance attributes. Such features together with their importance are then used with generic universal weights obtained using existing distance metric learning methods to re-identify targets across disjoint cameras. In [147], instead of exploring new features, they proposed to make a better use of multiple images. The proposed method builds a collaborative representation over all the gallery images (of known person individuals) to best approximate the query images (containing an unknown person) via affine combinations. Then, by enforcing the sparsity of the samples used for approximating the two nearest points, the relative importance of the gallery images belonging to different persons has the ability to reveal the identity of the querying person. In [8], a human signature that handles difference in illumination, pose and camera parameters was proposed using a novel model based on Mean Riemannian Covariance (MRC) patches extracted from tracks of a particular individual. A similarity measure using Riemannian manifold theory was also proposed to distinguish sets of patches belonging to a specific individual. In [78], particular aspects of the log-chromaticity color distribution structure structure of person appearance were shown to be invariants. Such intra-distribution structure had shown to be invariant under a wide range of imaging conditions. The proposed person signature is formed using the shape context descriptors to represent the intra-distribution structure. In [152], an unsupervised salience learning method was used to find distinctive features without requiring identity labels in the training procedure. Adjacency constrained patch matching was used to build dense correspondence between image pairs, which shows effective- ness in handling misalignment caused by large viewpoint and pose variations. Then human salience was learned in an unsupervised manner. To improve the performance of person re-identification, human salience was incorporated in patch matching to find reliable and discriminative

matched patches. In [88], a representation that relies on the combination of Biologically Inspired Features (BIF) and covariance descriptors was used to compute the similarity of the BIF features at neighboring scales. In [89] a person re-identification signature descriptor was proposed by exploiting Fisher Vectors. A simple vector of attributes consisting in the pixel coordinates, its intensity as well as the first and second-order derivatives was computed for each pixel of the image. These local descriptors were turned into Fisher Vectors before being pooled to produce a global representation of the image. The so-obtained Local Descriptors encoded by Fisher Vector (LDFV) were finally used to match pedestrian across camera views.

2.2.4 Metric Learning Based Methods

According to [36], in a metric learning framework a set of training data is used to learn an optimal non-Euclidean metric which minimizes the distance between features of pairs of true matches while maximizing the same between pairs of wrong matches. In [151] the re-identification problem is formulated as a local distance comparison problem by introducing an energy-based loss function that measures the similarity between appearance instances. In [76], important issues on scalability and the required degree of supervision of existing Mahalanobis metric learning methods were faced. Labels were used in form of equivalence constraints by introducing a strategy to learn a distance metric from equivalence constraints, based on a statistical inference perspective. This framework was then used to learn the optimal distance metric between image pairs of the same or different persons acquired by two disjoint cameras. In [60], a relaxation of the positivity constraint of the Mahalanobis metric was posed introduced to pose the re-identification problem as a local distance comparison problem. A an energy-based loss function was introduced to measure the similarity between appearance instances by calculating the distance between corresponding subsets (with the same semantic meaning) in feature space. The exploited loss function favors short distances, which indicate high similarity between different appearances of people, and penalizes large distances and overlaps between subsets, which reflect low similarity between different appearances. In this way, fast people re-identification was carried out. In [59] the re-identification problem was addressed by learning a Mahalanobis metric using pairs of labeled samples from different cameras. Building on the ideas of Large Margin Nearest Neighbor classification, a more efficient solution which additionally provides much better generalization properties was achieved. In [60] the same authors introduced a metric learning to find a suitable space for matching samples from different cameras. Improvements in terms of computational costs were achieved by relaxing the original hard constraints, thus getting a simpler problem that avoids iterative procedures. In [36] a metric learning framework was used to obtain a robust metric for large margin nearest neighbor classification with rejection (i.e., the classifier will return no matches if all neighbors are beyond a certain distance). In order to use the rejection option a cost function similar to the Large Margin Nearest Neighbor (LMNN) was introduced. In [83] different visual metrics were optimally learned for different candidate sets. Towards this objective a transfer learning framework was employed. Given a large training set, the training samples

are selected and re-weighted according to their visual similarities with the query sample and its candidate set. A weighted maximum margin metric is online learned and transferred from a generic metric to a candidate-set-specific metric. In [105] the Pairwise Constrained Component Analysis (PCCA) algorithm was introduced to learn distance metrics from sparse pairwise similarity/dissimilarity constraints in high dimensional input space. PCCA learns a projection into a low-dimensional space where the distance between pairs of data points respects the desired constraints. In [110], a metric learning approach for person re-identification was introduced. First, unsupervised PCA dimensionality reduction was performed under some constraints such that the redundancy in color-space representation was kept. Then, dimensionality was reduced using a Local Fisher Discriminant Analysis defined by a training set. In [3], re-identification is performed by measuring cosine similarity between the gallery and the probe descriptors which, in turn, are constructed by measuring similarity with the reference data in a Regularized Canonical Correlation Analysis (RCCA) subspace.

To best understand the mechanism of metric learning methods the interested readers are directed to two survey papers [148, 14] on this subject.

2.2.5 Transformation Learning Based Methods

One of the early works of finding the transformation of features between cameras was proposed in [115]. A BTF between the appearance features is computed by finding the optimal path in the feature correlation matrix. A similar approach is proposed in [70] where, to handle the appearance change of an object as it moves from one camera to another, the subspace computed for all BTFs is learned by using probabilistic principal component analysis. The subspace is then used for persons matching. An incremental learning framework to model linear color variations between cameras was proposed in [49]. Both [70] and [49] learned space-time probabilities of moving targets between cameras and used them as cues for association. However, transition time information may be unreliable if camera FoVs are significantly non-overlapping. In [117] a sparse color information preserving Cumulative BTF (CBTF) is learned from training examples collected from a pair of camera views. The bi-directional color mapping information from the training data significantly reduced false positives. In [83] the insight that different visual metrics should be optimally learned for different candidate sets is exploited using a transfer learning framework. A weighted maximum margin metric is learned and transferred from a generic metric to a candidate-set-specific metric. In [32] a Weighted Brightness Transfer Function (WBTF) that assigns unequal weights to observations based on how close they are to test observations is proposed. Closer the observation of the training set to the test one higher the weight. Lower the weight viceversa. In [131] an iterative method that model the effects of illumination changes over time is proposed to improve the accuracy of BTFs for re-identification. A fixed training stage for the Brightness Transfer Function can be used, because the intra-camera appearance of colors is rendered more constant. In [4] the re-identification problem is posed as a classification problem in the feature space formed of concatenated features of persons viewed in two different cameras.

2.2.6 Evaluation Methodology

In this section we give a brief introduction to the common performance evaluation methodologies used to validate person re-identification methods.

First of all, the re-identification community poses the re-identification problem by assuming that two sets of pedestrian images are available: the gallery set $\mathcal{G}$ (for which labels are known) and the probe set $\mathcal{P}$ (the set of pedestrians we want to re-identify). The goal is to assign the same label to the image of a person in the set $\mathcal{P}$, to the corresponding image of the same person in $\mathcal{G}$. The re-identification mechanism commonly depends on how the two sets are organized, that is, on how many images of a person are available. This gives rise to two matching philosophies: i) single-shot, when only one image of a person is present in each of the two sets; ii) multiple-shot, when both $\mathcal{G}$ and $\mathcal{P}$ contain multiple images of a person in each of the two sets.

To show the performance of a method, the literature suggests to report the performance in terms of recognition rate by the Cumulative Matching Characteristic (CMC) curve and the normalized Area Under Curve (nAUC) score for the CMC curve. The CMC curve shows the recognition performance as a function of the rank score and represents the expectation of finding the correct match in the top k matches. Ideally rank 1 should be assigned only to the signatures computed for the same person. On the other hand, nAUC is independent to the size of the dataset used for evaluation so it gives an overall score of how well a re-identification method performs. Such values are generally computed 10 or 100 different times using independent random splits, then, the average is taken as the representative value.

Many benchmark datasets are now available to evaluate the performance of a person re-identification method. More details about each dataset are reported in Table 2.3 and are discussed below.

ETHZ Dataset

The ETHZ dataset [40] contains video sequences of urban scenes captured from moving cameras. It contains a large number of different people in uncontrolled conditions. It has originally been proposed for pedestrian detection, but in [128] a modified version of the dataset was provided for the task of person re-identification. This version consists of person images extracted from three video sequences structured as follows: SEQ. #1 containing 83 persons (4,857 images), SEQ. #2 containing 35 persons (1,961 images), and SEQ. #3 containing 28 persons (1,762 images). Since the original video sequences are captured from moving cameras, images have a range of variations in human appearance and some even suffer from heavy occlusions. However, for the same reason the dataset does not provide a realistic scenario for person re-identification with multiple disjoint cameras. Despite this limitation it is commonly used for person re-identification.

VIPeR Dataset

The VIPeR dataset [54] is one of the most challenging datasets for person re-identification due to the changes in illumination and pose, and the low spatial resolution of images.

Table 2.3: Details and comparison of commonly used person re-idenentification benchmark datasets. For the CAVIAR4REID dataset, values in brackets are for persons appearing in both cameras. For ETHZ dataset values in brackets are for SEQ.#1, SEQ.#2 and SEQ.#3 respectively.

Dataset	People	Image info	Cams	Additional Info
ETHZ [128] SEQ.(#1,#2,#3)	(83, 35, 28)	Images: (4856, 1690, 1762) Avg. images per person per camera: (59, 48, 63) Size: 13×30 to 158×432	(1,1,1)	Scenario: outdoor Challenges: color changes, occlusions, sptial resolution http://homepages.dcc.ufmg.br/~william/
VIPeR [54]	632	Images: 1264 Avg. images per person per camera: 1 Size: 48×128	2	Scenario: outdoor Challenges: viewpoint variation, color changes http://vision.soe.ucsc.edu/node/178
CAVIAR4REID [26]	72 (50)	1220 (1000) Avg. images per person per camera: 10 (10) Size: 17×39 to 72×144	2	Scenario: indoor Challenges: viewpoint variation, color changes, spatial resolution www.lorisbazzani.info
i-LIDS [153]	119	Images: 476 Avg. images per person per camera: 2 Size: 21×53 to 176×326	2	Scnario:indoor Challenges: viewpoint variation, color changes, occlusions www.ilids.co.uk
WARD [92]	70	Images: 4786 Avg. images per person per camera: 69 Size: 15×36 to 70×189	3	Scenario: outdoor Challenges: viewpoint variations, spatial resolution, color changes http://users.dimi.uniud.it/~niki.martinel/
RAiD [99]	43	Images: 6060 Avg. images per person per camera: 47 Size: 64×128	3	Scenario: outdoor and indoor Challenges: Severe illumination and viewpoint variations, spatial resolution changes
3DPeS [9]	200	Images:1012 Avg. images per person per camera: 3 Size: 31×100 to 176×267	8	Scenario: outdoor Challenges: viewpoint variation, color changes www.openvisor.org
DANA36 [111]	15	Images: 23641 (3372) Avg. images per person per camera: 27 (6) Size: 14×66 to 526×1054	36	Scenario: outdoor/indoor Challenges: viewpoint variation, color changes and spatial resolution http://vision.fe.uni-lj.si/RESEARCH/dana36/

This dataset contains one image each from two cameras of 632 persons. While it comes with many different persons, only a single image per person per camera is available and only two cameras have acquired the scene. So, even it is widely used to evaluate the re-identification performance, this is not much representative of a real scenario in which many cameras and multiple frames of a same person are generally available.

CAVIAR4REID Dataset

The CAVIAR4REID dataset [26] contains images of pedestrians extracted from the CAVIAR repository. It is composed of 1220 images of 72 pedestrians out of which 50 are viewed by two disjoint cameras. It is more interesting than the ETHZ and the VIPeR, as more than a single image per person is acquired by two cameras. Other challenges in this dataset includes a broad change in the image resolution, with a minimum and maximum size of 17×39 and 72×144, respectively, severe pose variations, illumination changes and occlusions.

i-LIDS Dataset

The iLIDS Multiple-Camera Tracking Scenario (MCTS) repository is a dataset captured by a CCTV multi-camera network at an airport arrival hall at the rush hour. In [153], 479 images of 119 pedestrians were extracted from these videos to evaluate a context-based pedestrian re-identification method. The resulting images derive from non-overlapping cameras, under quite large illumination changes and subject to occlusions. This dataset is very challenging since often only the top part of the person is visible.

WARD Dataset

The WARD dataset [92] contains 4786 images of 70 different people acquired by three non-overlapping cameras in a real surveillance scenario. This dataset is of particular interest because it has a huge illumination variation apart from resolution and pose changes.

RAiD Dataset

The Re-identification Across indoor-outdoor Dataset (RAiD) dataset is a newly collected one [99] that comes with large illumination variations that are not present in most of the publicly available benchmark datasets. To make sure there is enough variation of appearance between cameras, subjects were asked to walk through 3 cameras of which 2 are outdoor and 1 is indoor. That results in a set of 6060 images of 43 persons walking through 1 indoor (denoted as camera 1) and 2 outdoor cameras (denoted as camera 16 and camera 22).

3DPeS Dataset

The 3DPeS dataset [9] contains different sequences of 200 people taken from a multi-camera distributed surveillance system. There are 8 cameras and each one is presented with different light conditions and calibration parameters, so the persons were detected multiple times with different viewpoints. Not only that, they were captured at different time instants during the course of different days, in clear light and in shadowy areas. This results in a challenging dataset with strong variation of light conditions.

DANA36 Dataset

The DANA36 dataset [111] consists of 23,641 images, depicting 15 persons and nine vehicles. The dataset was acquired from 36 stationary camera views using a variety of surveillance cameras with resolutions ranging from standard VGA to three megapixel. 27 cameras observed the persons and vehicles in an outdoor environment, while the remaining 9 observed the same persons indoors. Due to variety of camera locations, vantage points and resolutions, the dataset provides means to adjust the difficulty of the re-identification task in a controlled and documented manner.

3

Adaptive Human Interface for a Video Surveillance System

In this chapter, a novel adaptive human interface for a video surveillance system is introduced. The system is briefly introduced at the begin of the chapter, then, the modules that compose it are described in details. Finally, the user centered development process is described and experimental results are shown.

3.1 Introduction

Video Surveillance Systems (VSSs) have rapidly progressed in the past 10 years [33]. Even though the number of cameras installed for surveillance purposes is increasing, it has been shown [43] that large scale deployments are still not supporting the requests since both low-level and high-level computer vision tasks are not enough robust yet. Compared to the great amount of research done for the high level tasks [80, 103, 137], just a few researchers focused their attention on the usability of video analytics systems. Modern systems [55, 136, 155] still require operators' endeavor to monitor the vast amount of acquired data. As a result, the human attention and capabilities are overpowered. Only in the last few years, the research community has proposed new user interfaces (UI) to better assist end-users in their monitoring tasks [155, 20, 50, 93]. In particular, the new proposed methods for wide area analysis [125] highlight relevant areas and guide the user attention only on critical information while the development of UI for tracking tasks is almost not considered.

In current video analytics systems objects have to be followed through multiple cameras and surveillance operators have to switch between camera views and monitors as well. In many cases, to follow objects between camera views, video surveillance operators employ a single monitor which generally have quite small dimensions [123]. Commercial products usually propose VSSs that are equipped with huge wall screens and/or some remote smaller displays [140]. It is a matter of fact that such solutions still require a huge mental effort. For these reasons, VSSs must provide effective UIs such that relevant information are provided in a coherent and useful way.

The development of an effective and powerful information visualization technique is the goal of this work [97]. The idea is to properly visualize only the most important cameras and information contents to simplify the operators' tasks. The main novelty is the dynamic organization, activation and switching of the UI elements based on the output of video analytics algorithms. Rather than displaying all available camera views, only most probable streams, i.e. those that will be involved with the objects motion, are presented. To determine the most probable streams, the system must predict the objects trajectories and the cameras that best acquire such possible paths. So, to reach the goal, two main challenges should be addressed: i) to distill the volumes of monitoring information into a human manageable quantity; ii) to present the filtered visual information to end-users such that they can take appropriate decisions in a limited amount of time.

The first challenge is addressed by the Video Analytics Module (VAM) using an approach similar to [120]. The hand-off between different camera views is used to track a single object among different fields-of-view that are geographically adjacent. The proposed camera planning algorithm uses geographical clues and exploits the predicted trajectories to build an accurate camera activation plan. The camera activation plan together with the tracking data is used to provide only the most valuable data to the novel information visualization technique. In particular, the VAM cooperates with the UI reasoning algorithm to show only those views that can ease end-users tasks.

The Human-Computer Interface (HCI) addresses the second challenge. The new visualization algorithm exploits the VAM activation plan and tracking data to arrange UI elements accordingly to visual semantic information. In particular, camera views are arranged such that the operators have to focus only on relevant information. The proposed system uses the *overview plus detail* representation technique [27] to better display geographical clues.

The rest of the chapter is organized as follows. The main system contribution and its advantages with respect to current VSSs are given in section 3.2. A description of the system is given in section 3.3. Section 3.4 introduces the trajectory clustering algorithm and cluster trees. Details about the three main HCI components are given in section 3.5. Experimental results are shown in section 3.6. Finally, conclusions and future works are discussed in section 3.7.

3.2 Contributions and Advantages

Despite many of these works help improving end-users capabilities, they still require huge mental efforts to the human operators. In particular, the main open issues are the followings: i) each user is required to monitor a large amount of footages at the same time; ii) tasks like tracking across multiple cameras require manual interaction with the UI to select desired camera views; iii) the position and the colors of UIs elements are not chosen accordingly to Human-Computer Interaction principles. The proposed work deals with those issues introducing: i) a predictive and autonomous selection of camera views; ii) a dynamic activation, selection and organization of video

streams; iii) an information visualization technique that eases surveillance tasks. In Table 3.1,the features of the proposed system are compared with the most important related works.

The predictive and autonomous selection of camera views allows to distill the number of video streams that are showed to the operators. Then, the dynamic activation, selection and organization of video streams enables the operators to focus only on those footages that matter for the specific task they are performing. In particular, the task of tracking pedestrians across camera views is considered. The selected and activated video streams are finally displayed through a novel user interface that allows the operators to focus only on a single view without requiring them to switch between monitors as well as user interface elements. A map representation that exploits the detail plus overview technique is also introduced to make the task of inspecting the whole are less tough.

3.3 System description

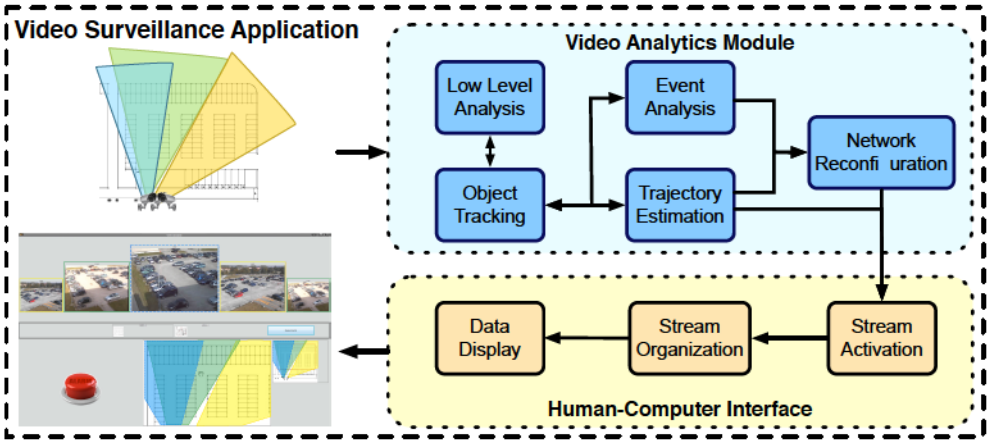


Figure 3.1: Proposed system. The Video Analytics Module fuses information about trajectory predictions and object tracking to reconfigure the network and select only relevant streams. The HCI module organizes and displays the selected streams through an advanced UI.

As shown in Figure 3.1, the architecture of the proposed VSS is organized in two main modules: i) the Video Analytics Module and ii) the Human-Computer Interface module.
The VAM module focuses on camera tasking operations. Video streams are analyzed to identify events of interest that have to be provided to human operators together with useful information. For such a purpose, the VAM detects and recognizes all the active (i.e. moving or temporary stationary) objects in the monitored environment. When an object is acquired, the tracking algorithm starts to track the object. Then,

Table 3.1: Comparison of the proposed method with commercial and research systems.

System	Wide Areas	Crowded Scenarios	On-line	Retrieve data from repository	Multiple Object Tracking	Area Map	Multi-camera Visualization	Predictive Camera Selection
Proposed	✓		✓	✓	✓	✓	✓	✓
VSAM [30]	✓		✓		✓	✓		
ADVISOR [130]		✓	✓	✓	✓	✓		
DOTS [50, 52, 51]			✓			✓	✓	
IBM (S3) [65]	✓			✓			✓	
Siemens Surveillance Vantage [132]	✓		✓	✓		✓	✓	
Siemens SiteIQ [133]	✓		✓	✓		✓		
Ipsotek [66]	✓	✓	✓		✓	✓	✓	

a high-level component correlates the objects activities along time and space through the different camera views. Such an analysis is used by the trajectory estimator [112] to predict the trajectories of the objects of interest. By using past information about activities and trajectories, this component is able to path-plan the movements of the objects of interest such that the camera network can be opportunely tasked or redirected in order to improve the analysis capabilities [104]. The reconfiguration component proposed in [113] is used to automatically reconfigure the PTZ cameras and improve the system performance. An activity density map is exploited to optimally cover the monitored area on the basis of the activity probability.

The estimated trajectories and the camera network configuration are input to the HCI module. The objective of the HCI module is to organize and display video streams to better support operators' tasks. The HCI module is composed by: i) the stream activation, ii) the stream organization and iii) the data display components. The stream activation component exploits VAM data to select and activate only relevant video streams. Given the estimated evolution of the environment (trajectories and involved cameras), it selects the streams that most probably will acquire objects activities. The stream organization sorts the selected camera views with respect to their estimated importance. In this way the volumes of monitoring information is distilled into a human manageable quantity. Finally, the data display component displays the organized streams on the UI together with useful information provided by the VAM. Human-Computer Interaction principles are exploited to guide the user attention only to critical information.

3.4 VAM module

The Video Analytics Module extracts information about the events observed in the monitored environment by detecting moving objects and processing their trajectories. Let assume the trajectory of each single moving object can be extracted by analyzing the video sequences, and do not consider the possibility of overcrowded scenarios. Moving objects are detected by means of a change detection algorithm and classified using a neural network, then the position of each object is filtered using a Kalman filter and a Camshift color tracker [45].

As new trajectories are acquired, the trajectory clustering algorithm proposed in [112] organizes the detected trajectory clusters in a probability-labeled tree. This allows to detect the clusters with higher probability of being matched, corresponding to the zones where it is more frequent to identify a moving object. This information is useful for event analysis tasks such as predicting object movements in the near future. The algorithm is here briefly summarized, for full details see [112].

3.4.1 Trajectory-cluster matching

A trajectory T_i is modeled by a list of vectors t_{ij}, each one representing the 2D spatial coordinates of object i at time j: $T_i = \{t_{i1} \dots t_{in}\}$ where $t_{ij} = (x_{ij}, y_{ij})$. The spatial coordinates can be computed directly on the image plane -even though in this work

coordinates are expressed in a world reference frame. This is achieved by projecting the image plane position of each object on a map of the monitored environment using a homographic projection. Clusters (groups of trajectories with similar spatial features) are represented in a similar way, with the addition of an approximation of the local variance σ_{ij}^2 of the cluster i at time j: $C_i = \{c_{i1} \dots c_{in}\}$ where $c_{ij} = (x_{ij}, y_{ij}, \sigma_{ij}^2)$.

In order to check if a trajectory matches a given cluster, a trajectory-to-cluster distance has been defined. Given a trajectory $T = \{t_1 \dots t_n\}$ and a cluster $C = \{c_1 \dots c_m\}$ the adopted distance is defined as

$$D(T, C) = \frac{1}{n} \sum_{i=1}^{n} d(t_i, C) \tag{3.1}$$

where

$$d(t_i, C) = \min_j \left(\frac{dist(t_i, c_j)}{\sqrt{\sigma_j^2}} \right) \quad j \in \{\lfloor (1-\delta)i \rfloor \dots \lceil (1+\delta)i \rceil\} \tag{3.2}$$

with $\delta < 1$ constant and $dist(t_i, c_j)$ the Euclidean distance between the trajectory point t_i and the cluster point c_j omitting the variance component. Using equation 3.1 the distance of a trajectory from a cluster is thus the mean of the normalized distances of each trajectory point t_i with the closest cluster point within a temporal window whose size, controlled by parameter the δ, increases through time. The variable-size temporal window allows matching also in case of limited temporal shifts between trajectories and matching clusters, avoiding at the same time matches with excessively large temporal distances.

Finally, when a trajectory matches a cluster, the cluster itself must be updated with the information of the newly matched trajectory. The updating equations implement a running average with exponential forgetting of the trajectory data:

$$\begin{aligned} x &= (1-\alpha)x + \alpha\hat{x} \\ y &= (1-\alpha)y + \alpha\hat{y} \\ \sigma^2 &= (1-\alpha)\sigma^2 + \alpha[dist(t_i, c_j)]^2 \end{aligned} \tag{3.3}$$

where $c_j = (x, y, \sigma^2)$ and $t_i = (\hat{x}, \hat{y})$ are the matching points as in eq. 3.2.

3.4.2 Cluster trees

The trajectory-to-cluster matching and updating equations described in the previous section cannot be directly applied in real-life scenarios as typically only partial matches can be detected (e.g. a trajectory starts close to a cluster and later leaves it). In order to model these behaviors the concept of *cluster trees* is applied as in [112]. A cluster tree is a tree where each node is a cluster representing a spatial portion of the environments shared by a set of sub-trajectories, and arcs represent connections between clusters. For example, the trajectories in Figure 3.2(a) all share the same initial region, modeled by cluster c1. When the trajectories diverge toward two different regions, these regions are represented by two new clusters, c2 and c3, and their

link with c1 is modeled in the tree structure shown in Figure 3.2(c). The tree data structure is preferred to a graph one, since the system is forced to model as a single cluster only shared prefixes (initial parts of trajectories) rather than suffixes; this is most useful for trajectory prediction and anomaly detection tasks.

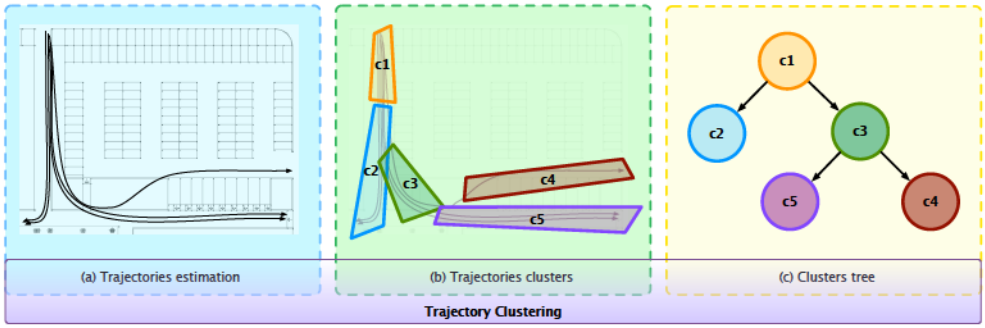


Figure 3.2: Cluster trees represent the structure of a set of trajectories with partial sharing.

In order to create and update the cluster trees, the following procedure is used:

1) when a new trajectory is detected, its distance from existing clusters is computed using eq. 3.1;

2) if a match is not found, new cluster including the trajectory is created;

3) if a match is found, the cluster is dynamically updated according to eq. 3.3;

4) if the trajectory leaves a cluster:

- a) the cluster is split in two parts in order to create a new branch if needed. The tree structure is updated accordingly;
- b) a match among the children of the just-left cluster is searched, then the algorithm is iterated from point 2).

Points 2) and 4) rely on the trajectory-to-cluster distance $D(T, C)$ defined in equation 3.1 to check if a trajectory is matching or leaving a cluster. The distance is normalized according to the local cluster variance, and thus it is directly linked to the probability of the trajectory to belong to the statistical model represented by the considered cluster. For example, if $D(T, C) < 2$, it means that on average the trajectory falls within the 2σ range from the cluster center (a range including the 95% of the trajectories represented by that statistical model).

The described procedure allows to dynamically create and update cluster trees such as the one shown in Figure 3.2(c). Arcs can be labeled with probabilities, computed by counting the number of trajectories matching each node. Specifically, if node C has n children nodes $c_1 \dots c_n$, the arc connecting C and c_i is labeled with probability $\frac{|c_i|}{\sum_{j=1}^{n} |c_j|}$ where $|c_i|$ is the number of trajectories matching cluster c_i.

In [112], labeled arcs are used for anomaly detection. The total probability of a fully developed trajectory is defined as the product of all the probabilities in the path from the first to the last node matched by the trajectory. Probabilities are used to predict the most probable future evolution of a partial trajectory. This feature is exploited in the proposed work to automatically select and organize the cameras that most probably will observe a given object.

3.5 HCI module

Information about sensors streams and objects trajectories extracted by the VAM module are used by the HCI module to tailor contents that have to be displayed to the end-users. Three innovative components are introduced by the HCI module: i) the stream activation, ii) the stream organization and ii) the data display.

3.5.1 Stream activation

The stream activation component connects information given by the VAM to the stream organization and data display components. It uses the information from the trajectory estimation and network reconfiguration components to select and activate only relevant streams. In particular, the estimated path correlated to the fields-of-view computed by the network reconfiguration component, allows to plan the hand-off and activate the cameras that will, most probably, cover the motion of the object. Such cameras are then included in a priority queue that is used to keep the visual focus on the selected object.

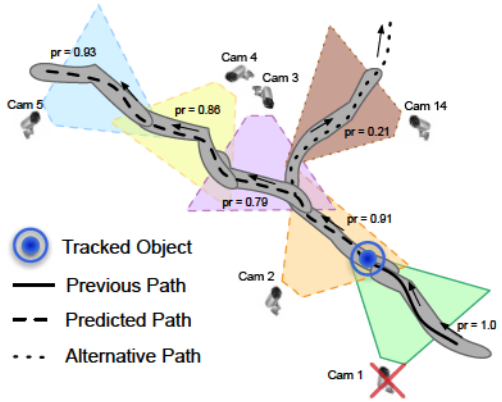
Let Q be the priority queue, cam_j be the j-th camera with FOV_j field of view, then cam_j is pushed in Q if $FOV_j \cap T_i \neq \emptyset$, where T_i is a predicted trajectory. The stream organization component is then in charge to select and organize the elements of the priority queue. It provides to the data display module with the best possible views that help to perform end-users tasks.

3.5.2 Stream organization

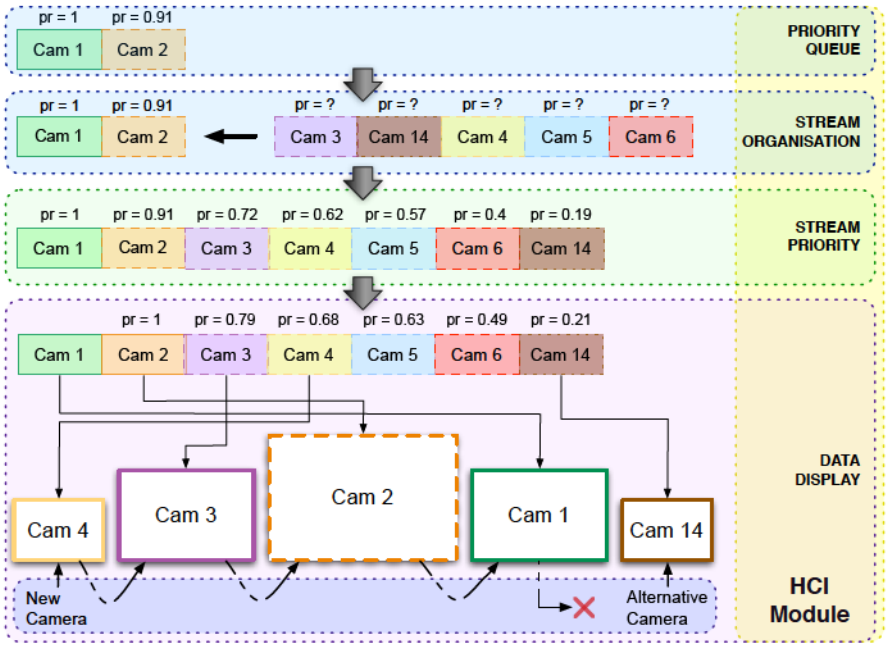
The stream organization component organizes camera views such that only the most relevant views are presented to the end-users. As shown in Figure 3.3, the component achieves its objective re-weighting the streams that have been inserted into the priority queue and sorting the camera views accordingly to their estimated importance.

Streams that have been previously inserted into the priority queue by the stream activation component are evaluated against all the possible object trajectories taking into account the geographical deployment of sensors. Thus, according to the most probable trajectories given by the VAM module, the stream organization component assigns to each view a priority value computed by intersecting the trajectory clusters with each camera FoV that has been inserted into the priority queue.

The stream priority value is computed by traversing the predicted path tree (see Figure 3.2). The edge value $P(C_i|C_j)$, connecting the clusters C_i and C_j, represents



(a)



(b)

Figure 3.3: In (a) the tracked object and the predicted path are shown together with camera FoV. In (b) the corresponding behavior of the HCI module components is shown.

the probability of the object to reach cluster C_i given its previous position in cluster

C_j. Hence

$$P(C_i|C_j, C_{j-1} \ldots, C_k) = P(C_i|C_j) \prod_{l=j}^{k+1} P(C_l|C_{l-1}) \tag{3.4}$$

is the probability that the object will reach the cluster C_i through the path

$$C_i, C_j, C_{j-1}, \cdots, C_k \tag{3.5}$$

Thus, the camera in the queue that covers the cluster C_i is assigned with a priority value equal to $P(C_i|C_j, C_{j-1} \ldots, C_k)$. The camera covering the cluster where the object is currently in is assigned a priority of 1. Once the priority values have been computed, the queue is sorted in order to have higher priority cameras on top. The goal of the final step is to correctly provide the priority information to the data display module such that it can parse and take the streams as fast as possible.

3.5.3 Data display

The data display component introduces a novel information visualization technique that aims to ease surveillance operators tasks exploiting Human-Computer Interaction principles. As Figure 3.4 shows, the proposed UI introduces two main components:

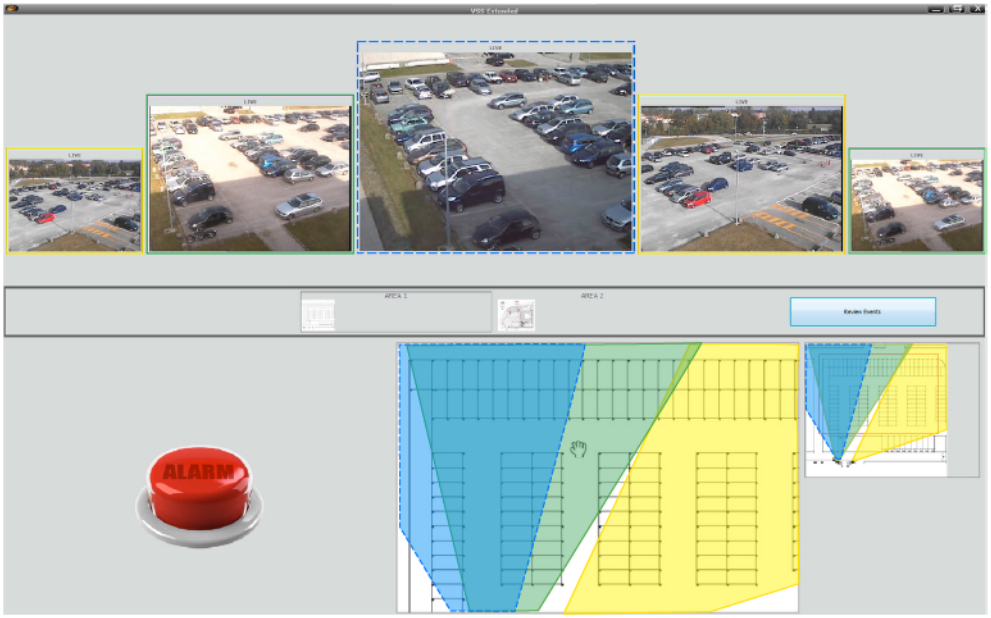


Figure 3.4: Finally proposed User Interface. The top region shows five camera views that are displayed accordingly to the priority queue computed by the VAM. The bottom region shows the map area component together with the active camera fields-of-view.

i) the video streams area and ii) the map area. The video stream area organizes and displays the camera view UI elements inserted in the priority queue to better support end-users tasks. The "switch panel" allows to switch between different areas of the monitored environment and to organize the video streams on the basis of the objects of interest. In case of multiple objects of interest, the operator is able to follow one of them just by switching the active visualization.
The map of the area displays geographical information about cameras positions, cameras FoV and moving objects.

Video streams area

The video streams area introduces a novel information visualization technique to display only the most relevant views. Three main novel features are introduced by this component: i) camera views displacement; ii) camera views animation; iii) camera views representation.

The *camera view displacement* is organized such that, the stream of the sensor with highest priority is displayed at the center of the video streams area (see Figure 3.3). The streams that have lower priority values are displayed either at the left or a the right side -depending on the predicted path of the object- of the main camera view. The previous highest-priority camera is shown on the other side. The goal is to provide the operator the previous and the next camera views that cover the predicted object trajectory.

It could be possible that the object of interest does not follow the most probable estimated trajectory, so, the most probable alternative path is considered. The camera view that has been assigned the highest priority with respect to such alternative path is displayed next to the previous highest-priority camera view (see Figure 3.3).

For instance, let consider Figure 3.3 and let assume that the tracked object is moving -from right to left- along the predicted path. Since the object is moving from Cam2 towards Cam3, Cam3 is displayed at the left of the current main view. As Cam2 is the highest priority camera, Cam1 is displayed at its right. The alternative camera with highest priority, Cam14, is placed next to Cam1. The camera priority is also used to set the size of the displayed camera views. The camera view with the highest priority has the largest size. Other camera views are scaled to 2/3 of the size of the camera view with a higher priority.

The *camera views animation* is introduced to smooth the hand-off between camera views. Accordingly to the camera view displacement feature, the stream of the sensor with the highest priority is displayed at the center of the video streams area. Since objects are moving across a path, the camera views have to be moved to respect the stated objective. If camera views are just switched a "flashing" effect is introduced due to the fact that the relevant streams will be activated/deactivated at different camera views positions. Such behavior cause confusion to the end-users and it does not respect Human-Computer Interaction principles. To sidestep this issue, UI animation effects are introduced.

As shown in Figure 3.5, as long as the object of interest follows the predicted trajectory, the relevant camera views are moved to the opposite direction with respect

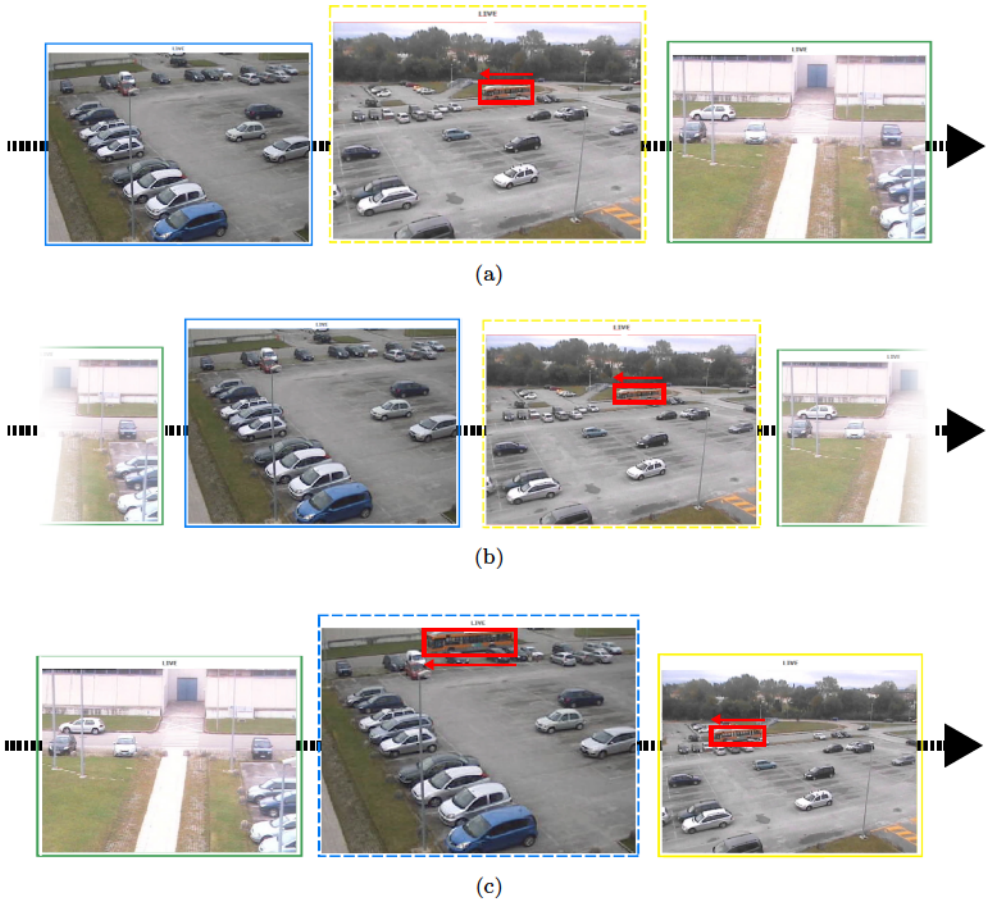
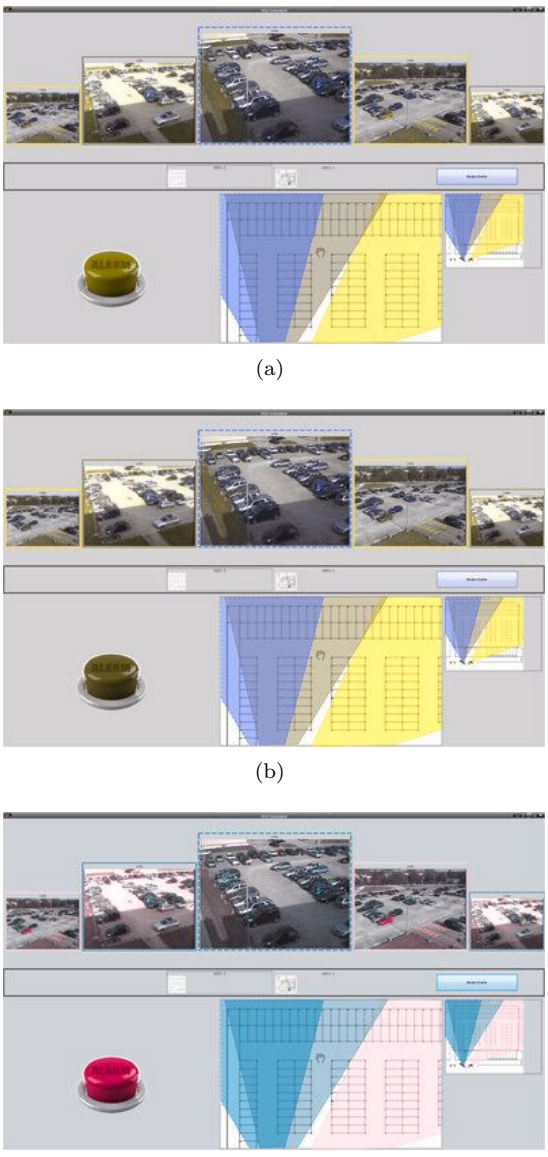


Figure 3.5: Proposed camera views transition. The tracked object is moving from right to left. Camera views are scaling and moving to the right. In (a) the initial camera view UI element position is shown. In (b) the scaling and transition of all the camera views to the right is depicted. Finally, in (c) the updated camera view UI element position after one transition is shown.

to the predicted object trajectory. Given the homography transformation between camera views and the map of the monitored environment (see section 3.4.1), the data display component estimates the velocity of the tracked object at each time instant and moves the UI camera views accordingly. Similarly, as camera views move, they are gradually scaled to the new sizes. Old selected views are scaled down and animated out of the UI.

The *camera views representation* introduces two main representation features: i) the color-coded and ii) the drawing style techniques used to depict the camera view UI elements. To achieve internal coherence the same representation techniques are



(a)

(b)

(c)

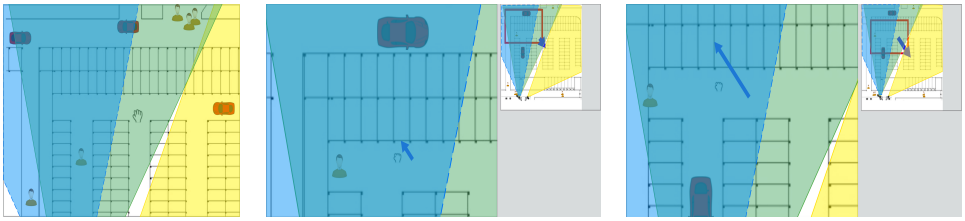
Figure 3.6: Color-blind people vision simulation of the proposed UI: (a) Deuteranope simulation (red/green color deficit) (b) Protanope simulation (red/green color deficit) (c) Tritanope simulation (blue/yellow deficit)

used to depict the camera FoV in the map area component. The colors used to depict UI elements have been selected such that each camera view can be distinguished even

from colour-blind people (see Figure 3.6). To ease the end-users tasks, a different drawing style has been used to highlight the most relevant view. A dashed line is used to represent the camera with the highest priority. This allows to easily link the main camera view representation in the video streams area with its FoV depicted in the map area.

Map area

The map area introduces a component that shows the topological representation of the monitored area. As shown in [50], the map representation of the monitored area improves the ability to follow objects and to analyze their behavior while these are moving across different camera views. Similarly to state-of-the-art video analytics systems, cameras, FoV and objects are represented in the proposed map area (see Figure 3.7). In addition to that, the work introduces a novel map visualization technique. Though the standard scrolling, panning and zooming techniques are useful to explore an information space at different levels of detail, it is often useful to display more than one level of detail at the same time [23]. The *overview plus detail* technique [27] is exploited to achieve such objective. This technique helps users to keep focusing on the details of an information space without loosing the overview of the entire space.



(a) (b) (c)

Figure 3.7: Proposed map area UI component. In (a) a standard map representation together with objects positions is shown. In (b) and (c) the overview plus detail representation is shown. Both the overview and the detail view can be zoomed and panned. The viewfinder (red box) is updated accordingly.

The *overview plus detail* technique is used to display both the detailed map and the context view. The context view displays a downscaled version of the map. It also highlights the portion of the map displayed in the detail view with a rectangular viewfinder. Both the viewfinder and the detailed view can be dragged to navigate the environment. The size and the position of the viewfinder in the context view also provide useful information for navigation, such as details about the scale between the displayed detailed map portion and the whole map. The viewfinder is updated accordingly to the scale used to display the detail view. Thanks to the novel visualization technique, the operator has an overview of the entire area even if it has

zoomed the map view to retrieve more details about an object (see Figure 3.7(b) and Figure 3.7(b)).

3.6 Experimental results

Human-Computer Interaction guidelines have been followed to evaluate the usability performance of the novel system. Four prototypes have been designed respecting the main usability rules defined in [39]. Empirical and non-empirical methods have been employed to evaluate each prototype.

To correctly apply the Human-Computer Interaction principles, information about classes of users and context of use have been identified. Users have been grouped in the following four different classes:

1. operators that use a VSS to monitor a small area for private purposes;
2. operators that use a VSS to monitor a small public environment;
3. operators that use a VSS with a multi-camera setup to monitor a wide area;
4. operators that use a VSS with a multi-camera setup to monitor multiple wide areas.

The second step was to identify the most probable contexts of use of the system. Five contexts of use have been identified:

1. visualize video streams using single-multiple displays;
2. track objects through multiple cameras FoV;
3. fire alarms;
4. automatic recording of interesting events footages;
5. review recorded footages.

All the four proposed prototypes have been validated using empirical and non-empirical tests. The following six different evaluation tasks have been proposed:

1. visualize the real time footage from “AREA 2”;
2. fire the alarm if an anomalous event occurs;
3. associate current visible streams to area sensors;
4. start tracking an object and follow it along its path;
5. start tracking an object, then fire the alarm if an anomalous event occur;
6. start tracking an object, then switch to a different area and start tracking a new object.

Non-empirical evaluations have been performed with the support of four Human-Computer Interaction experts. The non-empirical techniques have been used to obtain an initial evaluation of each prototype. After the evaluation, each prototype has been reviewed accordingly to the reports provided by the HCI experts.

To evaluate the prototypes, the six stated tasks have been performed by the experts. The steps required to reach each given task have been analyzed using two common techniques: i) the heuristic evaluation [108] and ii) the cognitive walkthrough [114]. After the evaluation, the experts provided a review for each of the 10 principles proposed in [107].
The cognitive walkthrough technique has been used to mainly detect the UI design errors that affected the ease of learning. A review for each UI feature, behavior and action involved in the proposed tasks has been given by each HCI expert.

All the recommendations provided by HCI experts have been taken into account to solve the identified problems. The process strongly helps the design and lets the system to perform better in terms of affordance, visibility and coherence with respect to standard video surveillance system UIs.

Empirical evaluations have also been performed to validate the proposed prototypes. Three standard empirical evaluation methods have been used to evaluate the usability performance: i) thinking aloud; ii) video screen recording and iii) usability questionnaires. To perform the empirical evaluations forty pre-identified end-users have been selected (see Table 3.2) and grouped into the four proposed clusters. As for non-empirical evaluation, they have been asked to perform the same six evaluation tasks.

Table 3.2: Forty pre-identified users have been selected to evaluate the performance of the proposed prototypes.

		Years of experience			
		0-1	2-5	5-10	10+
Real Operators	Male	2	4	5	2
	Female	2	4	0	0
Others	Male	4	5	2	1
	Female	6	2	1	0

The test sessions have been conducted in a controlled environment using pre-recorded video data. During such sessions users were supported by the researcher that was not allowed to intervene unless the end-user were not able to achieve the goal or if they had some questions about the UI elements behavior that did not reflect their expectations. Video screen recordings have been captured during test sessions. After completing all the assigned tasks, the usability questionnaires have been given to each user. All the acquired data has been merged and compared to detect and solve the prototypes issues.

To get a quantitative evaluation of each designed UI two indexes have been proposed: i) the mean success rate index and ii) the mean execution time index. Let $p = \{1, 2, \ldots, P\}$ be the proposed interface and let $j = \{1, 2, \ldots, J\}$ be the current

evaluation task. The mean success rate index (MSR) provides information about how well p scales to j. It is given by:

$$MSR(p, j) = \frac{\sum_{i=0}^{n_u} C_{i,j}^{p}}{n_u} \tag{3.6}$$

where $C_{i,j}^{p}$ is a matrix that gives the completion percentage of task j reached by user i using prototype p. n_u is the total number of tests. This index has been evaluated against each single task j and each proposed prototype p in order to see if the current solution improves the previous one.

Similarly to the MSR index the mean execution time index (MET) has been computed to investigate the UI efficiency. The MET is computed as

$$MET(p, j) = \frac{\sum_{i=0}^{n_u} T_{i,j}^{p}}{n_u} \tag{3.7}$$

where $T_{i,j}^{p}$ is a matrix that gives the time required by user i to complete task j using prototype p. In case a user was not able to fully complete the required task, the time assigned to that user is given by $\max T_{k,j}^{p}$ with $k \neq i$. The MET index has been used to provide information about how much time a single user needs to reach a given goal (user failure has been taken into account as well). By analyzing this data, it was possible to identify which were the tasks that required more time. In particular, during the design process, if a given task was requiring too much time the UI elements involved in that process were deeply inspected before evaluating the next prototype.

In all the experiments, the distance threshold required by the clustering algorithm has been empirically fixed to 2. Since the trajectory-to-cluster distance is normalized by the cluster variance, this means that a trajectory matches a cluster if, on average, its distance from the cluster center lies in the 2σ range.

3.6.1 Evaluation of the first prototype

A paper model has been used to design the first UI prototype. As defined by HCI rules, this is a common solution that allows a faster and easier definition of the system UI. As shown in Figure 3.8(a), the model does not introduce any color that allows people to identify cameras and to relate their FoV. This choice allowed to investigate how people associate cameras views and their UI map representation. The task #3 is thus hard to perform under this scenario, and, as results depicted in Figure 3.8(b) and Figure 3.8(c) show, some users did not complete it. The MSR for this specific task is about 81%, and the standard deviation is about 37%.

Even though the proposed UI was completely static, some of the given tasks required to track objects. To sidestep this issue the UI elements behavior has been simulated by moving paper objects. Mainly because of that, users failed to perform actions that required non-static UI elements and live video streams. In particular, task #5 required a higher amount of time to be completed since the anomalous events were displayed at 0'30", 1'14", 2'11', 3'38' and 5'40". Notice that events used in task

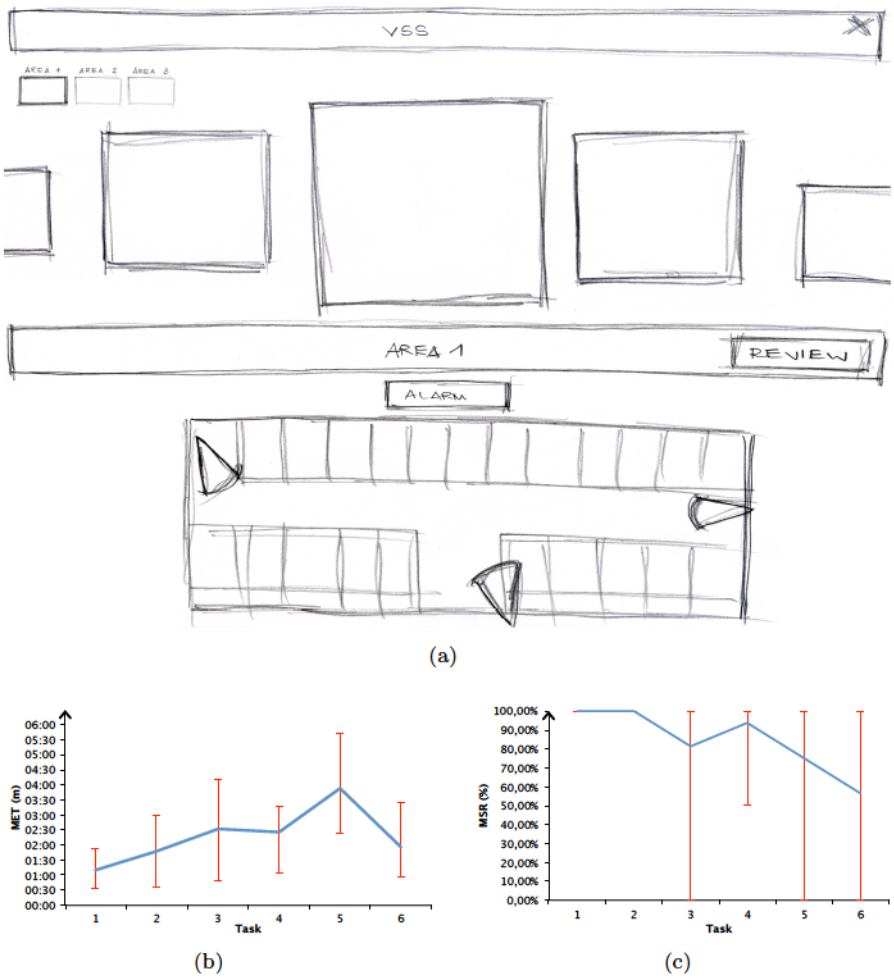


Figure 3.8: First system prototype (paper). The first model lacked of colors and had poor interaction but was equally helpful to detect the initial design issues. (a) Proposed user interface, (b) Mean Execution Time and (c) Mean Success Rate.

#2 were not the same as those used in task #5. Results reported in Figure 3.8(b) shows that all the users had trouble with such task.

As shown in Figure 3.8(c), users failed to reach the required goal for task #3, #5, and #6. In particular, task #6 has the lowest MSR (about 56%).
The main problems that came from the evaluations of the first prototype were:

- the lack of colors and techniques that allowed users to relate cameras depicted on the map with the camera views in the video streams area;

- the lack of video streams and the low interaction;
- the lack of interaction with the map.

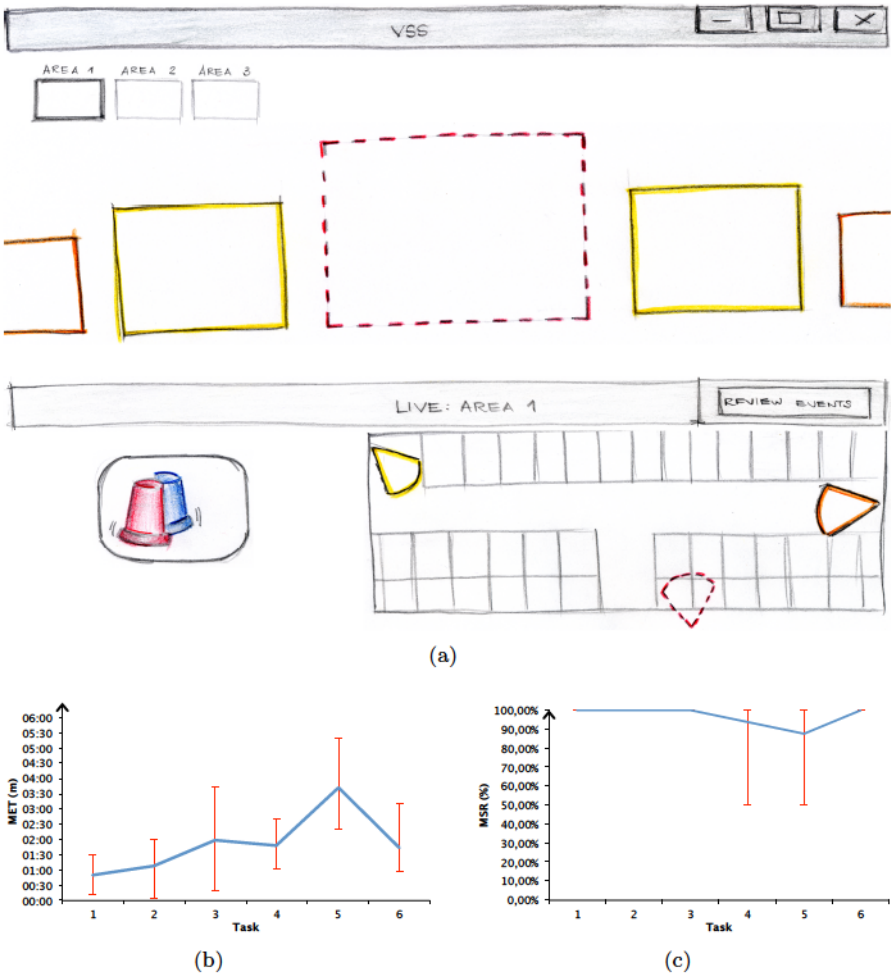


Figure 3.9: Second system prototype (paper). The second model introduced the colors different depiction techniques to associate the camera views in the video stream area and the cameras in the map area. (a) Proposed user interface, (b) Mean Execution Time and (c) Mean Success Rate.

3.6.2 Evaluation of the second prototype

As for the previous prototype, a paper model has been used to design the second UI prototype. To solve the issue detected from the previous evaluation two main novelties

have been introduced: i) usage of colors and ii) change of the *alarm* text-button to an icon-button.

As Figure 3.9(a) shows UI colors have been added. The MET index (see Figure 3.9(b)) shows that such feature did not significantly decrease the average time required to perform the proposed tasks. In particular, as Figure 3.8(b) and Figure 3.9(b) highlight, the task #3 reached a mean execution time of about 2'32" for the first prototype and an average time of 2'01" for this second prototype. The other tasks, if compared to the first prototype, achieved similar MET results even the standard deviation for each of them is about 6% lower. As for the previous evaluation task #5 was the one that required much time to be performed since anomalous events were shown at 0'30", 1'14", 2'11', 3'38' and 5'40". As before, events used in task #2 were not the same as those used in task #5.

Though the MET index doesn't show any significant improvement by the new prototype, the MSR index shows that the proposed colors and the employed depiction techniques solved the previously detected issues. In particular, the mean success rate for the task #3 had strongly increased from about 81.25% to 100%. The same happened for task #6. In both cases, all the users achieved the required tasks reaching a 100% MSR score. But, as shown in Figure 3.9(b) and Figure 3.9(c), task #4 and task #5 were still difficult to perform and required a long time to be completed.
Similarly to the first prototype evaluation, the main issues posed by this second prototype were:

- the lack of video streams and the low interaction;
- the behavior and the representation of the *alarm* icon-button;
- the lack of interaction with the map.

3.6.3 Evaluation of the third prototype

As Figure 3.10(a) shows a more interactive model has been used to design the third prototype. The third prototype has been developed using a presentation software. The slides of the presentation were arranged to simulate a real software. Footages recorded from a real-surveillance scenario have also been added. In order to allow users to perform all tasks, behaviors of UI elements involved by the required tasks have been defined. The main novelties introduced by the third prototype were: i) higher-level interactions and ii) change of the *alarm* button.

Similarly to the previous evaluations, both the MET and the MSR indexes have been computed. As the MET index shows (see Figure 3.10(b)) the amount of time required to perform each single task has decreased with respect to the two previously proposed prototypes. The standard deviation -of the MET index- computed for all the six given tasks has averagely decreased of about 64%. The amount of time required to complete the task #1 was about 0'15". Similarly, the MET for task #3 has decreased from 2'01" (second prototype) to 0'48". The strongest improvement has been achieved by the task #5, where the MET has decreased from 3'41" to 1'20".

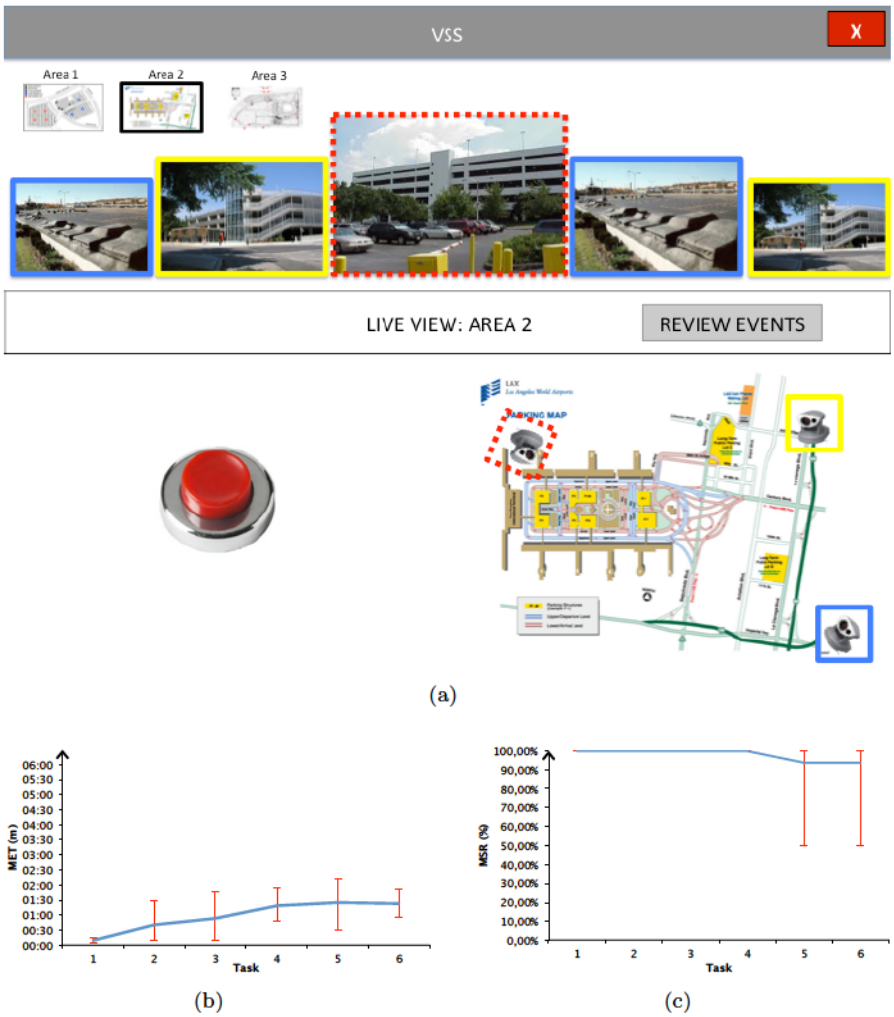


Figure 3.10: Third system prototype (interactive model). The third model introduced video streams and an interactive map. (a) Proposed user interface, (b) Mean Execution Time and (c) Mean Success Rate.

Since the same anomalous events have been used (as in the first and the second prototype evaluation), the achieved results show that most of the users missed only the first anomalous event (at 0'30").
The MSR index shows similar results, to the ones achieved by the second prototype, for task #1, task #2, and task #3. A 100% MSR has been reached by task #4. In contrast with results obtained from the previous evaluation, task #6 was not fully completed by all users (see Figure3.10(c)). A MSR score of 93% has been achieved

by both task #5 and #6. The problem was that the *alarm* icon-button was hard to understand and its behavior was not clear. Users also expected to use the map to select the objects to track.

Results of end-users tests conducted among the third prototype showed that the proposed UI elements had a better affordance, but some issues were still present. Single-user questionnaires inspection and results analysis showed that the main negative aspects posed by the third prototype were:

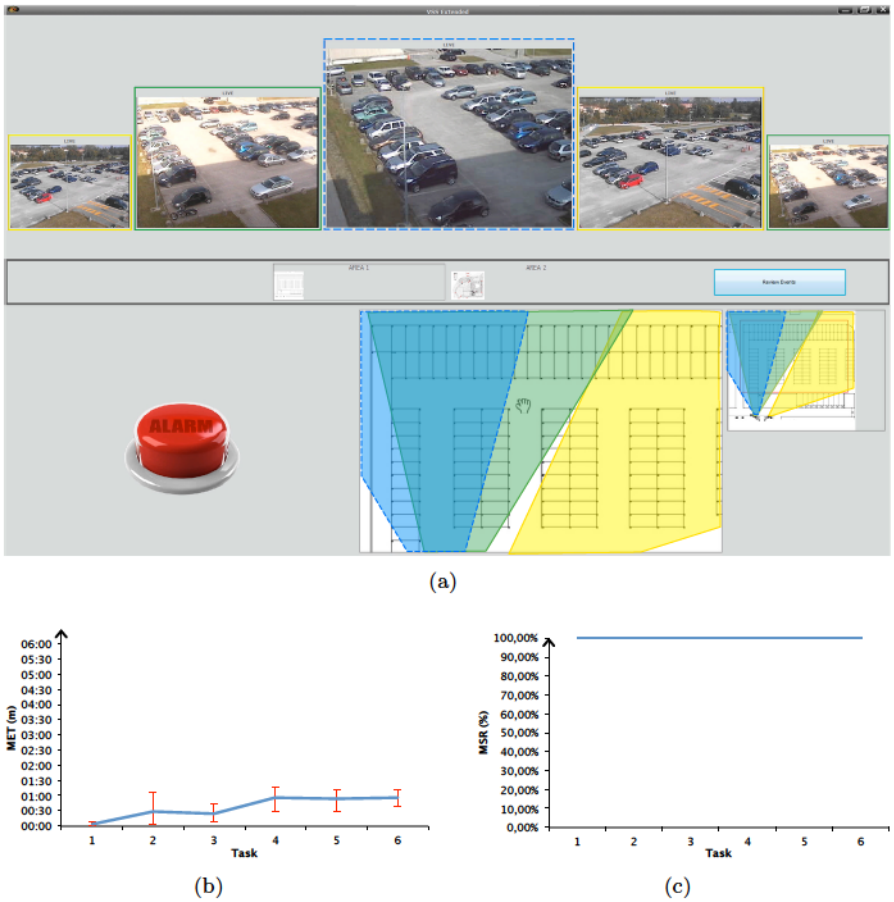
- when the active, the *alarm* button showed visual clues but no sound information was provided;
- the *alarm* button was misunderstood by many users;
- the lack of interaction with video streams. The selected videos came with multiple objects and some users expected to start tracking a chosen object by clicking through it. The prototype was not designed to allow such interaction and it started tracking a different object with respect to the selected one.

3.6.4 Evaluation of the fourth prototype

A software program has been developed as the fourth prototype (see Figure 3.11(a)). As for other prototypes, the same prerecorded data has been used. The main novelties introduced by such prototype were: i) the *overview plus detail* UI element, ii) the depicted cameras FoV and iii) the representation of objects within the map.

As Figure 3.11(b) shows the MET index decreased with respect to the MET computed for the third prototype. Also, the MET standard deviation computed for all the six given tasks decreased of about 47% on average. The MET for task #3 decreased from 0'48" (third prototype) to 0'26". The strongest improvements have been achieved by the task #5 and task #6. In contrast with previous evaluations, task #5 wasn't the one that required the longest time to be performed: all the users catch the first anomalous event. This is very important for surveillance operators and it shows the efficiency of the applied design methodology. The higher MET was achieved by the task #6 since many users had difficulties in selecting an object to track by clicking through it on the video streams UI representations. The MET required to perform each single task was always less than a minute.

The best results come from the MSR index since all the tasks achieved a 100% score. Such results show that the depicted camera FoV and the representations of the object onto the map component ease the surveillance tasks. None of the real-operators nor the novel-users asked to view all other camera streams that were not shown in the given UI. This is a very interesting result if compared to standard VSS UIs where users have to manually switch between camera views to activate them to follow an object of interest.



(a)

(b)

(c)

Figure 3.11: Fourth system prototype (software). Both the VAM and the HCI modules were designed and the same prerecorded data has been used to evaluate the performance of the system prototype. (a) Proposed user interface, (b) Mean Execution Time and (c) Mean Success Rate.

3.7 Conclusions

In this chapter, a novel information visualization technique for Video Surveillance Systems has been introduced. The video analytics system introduces the VAM and the HCI modules to properly visualize only the most important cameras and information contents, thus simplifying surveillance tasks.
The VAM performs video analytics tasks and predicts the possible paths of the objects of interest. Trajectories and cluster trees learned from real-tracking data are used to predict the most probable paths of tracked objects.
The HCI module presents only relevant information to surveillance operators selecting

the streams accordingly to information provided by the VAM module. It introduces three main components to propose a novel information visualization technique for VSS. The video streams data displays selected camera views such that the most relevant view is always displayed at the the center of the UI. The data display component also introduces the map area UI element that exploits the overview plus detail technique to show geographical information about the monitored environment.

Four UI prototypes have been designed and evaluated using standard Human-Computer Interaction techniques. Non-empirical evaluations results have been fused together with two proposed indexes to detect and solve usability issues introduced by each prototype. The results show that the adopted information visualization technique achieves high usability results and supports end-users during their surveillance tasks.

Despite the encouraging results, three main problems still affect the current system. i) The system can be used in situations where the monitored scenario is not overcrowded. ii) In case the size of the display is small, the camera views displayed in the video stream area may be too small and the task of recognizing objects may be hard. iii) If the number of objects to track is very high, the "switching panel" gets overcrowded and users may get confused by that. To address those issues, robust techniques that allows object tracking over crowded environments cameras will be exploited. New displacement methods to better display the camera views in the video stream area and the "switch panel" will be analyzed as well.

3.8 What next?

In this chapter we have introduced a VSS to ease surveillance operators tasks. However, the system has been designed considering a strong assumption. That is, we have assumed that persons can be tracked through the whole monitored environment such as the most probable paths can be extracted and can be later used for preemptive activation of cameras and visualization tasks. In particular, we have not addressed the problem of tracking persons across cameras. In the next chapters three different approaches are introduced to tackle such challenging issue with particular focus on the real scenarios where cameras have disjoint FoVs.

4

Re-Identification by Discriminative Signature Matching

In this chapter, the problem of re-identify a person that moves across the FoVs of disjoint cameras is addressed by means of a discriminative signature based method. First the person and body parts detection tasks are addressed, then the process of extracting local and global features from such body parts is given. Finally, after describing the signature computation and the signature matching methods we present the performance of our method and compare them to state-of-the-art approaches.

4.1 Introduction

In the previous chapters it has been shown that as the monitored site grows different problems arise, from the number of sensors to deploy, to their configuration, to the way they communicate and cooperate to achieve a global objective. As a matter of fact, as the dimension of the monitored environment grows, it quickly becomes hard to deploy a network of video sensors such that there are enough overlapping FoVs to cover every point of the monitored environment. In this context, even though sensors are becoming cheaper, a full coverage of the area is still not affordable due to the amount of human supervision, privacy concerns, and maintenance costs involved. These limitations yield to the development of video analytics systems that provide partial area coverage. Blind areas, called "blind-gaps", are therefore bringing in new problems since no information can be obtained from these areas. This connects to what has been done in the previous chapter and introduces the person re-identification problem.

To address the person re-identification problem without using active cameras, features having properties that are invariant to the common multi-camera issues should be used. According to the literature (see Chapter 2), two main groups of features-based methods have been proposed to address person re-identification challenges: i)

biometric-based methods and ii) *appearance-based methods*. Though the former group exploits biometric features whereas the latter relies on the appearance of the objects, both of them aim to extract features that can be used to describe an object seen under different orientations, poses, etc..

While state-of-the-art methods can achieve good re-identification results, using appearance-based and biometrics-based features, the person re-identification problem in a non-overlapping multi-camera scenario is still an open issue. In this chapter a method to tackle the re-identification challenges by means of discriminative signature matching is introduced [91, 92]. Each sensor in the network exploits camera specific learned models of persons to detect pedestrians and to extract both the whole body silhouette and the different body parts, thus avoiding common change detection or standard background subtraction algorithms that could lead to noisy and spurious detections. Local and global appearance features are extracted from the silhouette and accumulated over multiple images of the same person forming a highly discriminating signature that is finally matched with gallery signatures to perform the re-identification. To evaluate the performance of the proposed method results are compared to state-of-the-art methods using two publicly available benchmark datasets.

The rest of the chapter is organized as follows. An overview of the proposed system is given in section 4.2. The person detection and body part division methods are described in section 4.3. The signature computation process is defined in section 4.4. The signatures matching is described in section 4.5. Experimental results are shown in section 4.6 and conclusions are finally drawn in section 4.7.

4.2 System Overview

The proposed work introduces a signature based method to deal with the person re-identification problem. The overall scheme of the proposed approach is shown in Figure 4.1. The three modules introduced to achieve the final re-identification objective are as follows:

a) For each frame acquired by a camera, the *person detection and body part division module* detects the person in the scene and extracts both the silhouette and the person body parts. The work proposed in [150] has been used to achieve these goals by exploiting a camera-specific learned person model. Even though the proposed algorithm gives information about all the different body parts, the detected parts are not always correctly assigned, so, to minimize these effects, here only the two main body parts,i.e., torso and legs, are used. The adopted person detection algorithm allows to avoid using change detection and background subtraction methods that suffer of noisy and spurious detections.

b) The silhouette and the body parts of the same person computed on multiple frames are input to the *signature computation module*. This accumulates four local and global features to compute a discriminating signature for each person. The four features extracted from the silhouette and the body parts are the

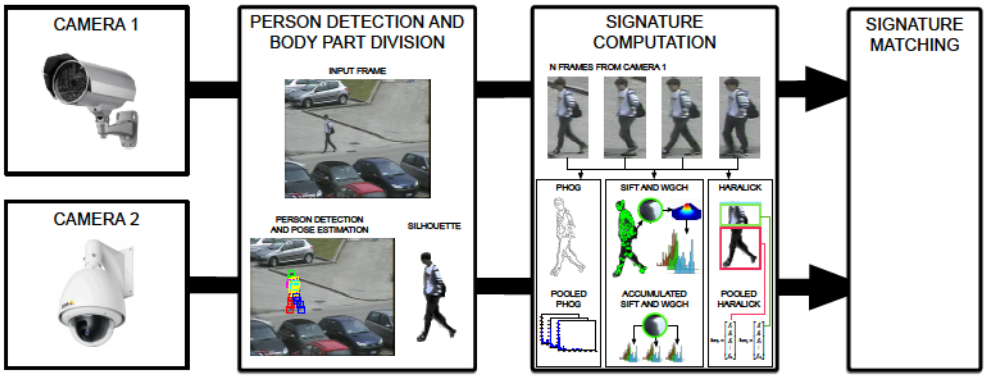


Figure 4.1: System overview. The proposed system uses three modules to address the person re-identification problem. Given an input frame the person and the main body parts are detected using person models learned for each camera. Local and global features are extracted from multiple frames of the same person acquired by a single camera and accumulated to form the discriminating signature. Then, the discriminative signatures from two disjoint cameras are matched using a combination of feature distances.

following: i) Pyramid Histogram of Orientation Gradients (PHOG); ii) Scale-Invariant Feature Transform (SIFT); iii) SIFT-based weighted Gaussian color histogram (WGCH); iv) Haralick texture features. PHOG and Haralick features are accumulated by pooling the feature vectors extracted from all the given frames. That gives a PHOG feature matrix and two Haralick feature vectors, one for each of the two body parts (i.e. torso and legs). The SIFT and the WGCH features extracted from each single frame are compared over all the given frames and accumulated to form the discriminative signature using an iterative procedure. In particular, multiple WGCH are accumulated and associated to one single SIFT feature to capture the variance of localized color patches. This procedure is described in details in section 4.4.

c) The *signature matching module* uses the computed signatures to perform the person re-identification. Given the signatures from two cameras, the distance between them is computed by: i) computing distances between singular features and ii) fusing distances. Single features distances are computed depending on the type of feature considered. PHOG features are matched using a weighted χ^2 distance. The L^2-norm distance is used to match Haralick and SIFT features. The WGCH features associated to matching SIFT features are then compared using a weighted χ^2 distance. Since multiple WGCHs can be associated to a single SIFT features, the computed distances are averaged over all the WGCHs. Finally, the distance between the given signatures is computed as an affine combination of the distances between the considered features.

4.3 Person Detection and Body Part Division

The first step to re-identify a person is to detect the person. This is a challenging task especially in crowded scenes and in large scale camera networks in which multiple and different objects appear simultaneously. Detecting other objects leads to many false matches. When a person is detected its silhouette is computed in order to limit the region for the feature extraction process. In addition, to build a more robust discriminating signature the feature extraction is independently executed on different body parts of the silhouette. For such purposes, we exploit the work proposed in [150]. By combining results achieved using the person detection and pose estimation method with hysteresis thresholding background subtraction and standard filtering methods for noise removal, the person detection, the silhouette extraction and the body part computation can be addressed in a few steps.

According to [150] full-body pose estimation is a difficult task because of the many degrees of freedom. Moreover, how limbs appear varies greatly due to changes in viewpoint orientations as well as in clothing and body shape. The approach introduced in [150] can be used to model the pose of a person as a mixture of non-oriented pictorial structures. The classic spring models are enhanced by adding co-occurrence constraints that favour particular combinations of parts so that constraints can be used to capture notions of local rigidity. Such an approach is briefly introduced in the following by adopting the same notation as the original paper for clarity. Let $\mathcal{I}$ be an image, $p_i = (x, y)$ and t_i, with $i \in 1, \cdots, K$, be the pixel location and the mixture component (or type) of part i respectively. Let $p_i \in 1, \cdots, L$ and $t_i \in 1, \cdots, T$, and let $G = (V, E)$ be a relational graph -in form of a tree- whose edges denote the pairs of parts that are constrained to have consistent relations. Then, the three different models that define the configuration of part types and positions can be computed. Let define the compatibility function for part types as

$$S(t) = \sum_{i \in K} b_i^{t_i} + \sum_{ij \in E} b_{ij}^{t_i, t_j} \tag{4.1}$$

where $b_i^{t_i}$ and $b_{ij}^{t_i, t_j}$ are the two components that favor a particular assign for part i and particular co-occurrences of part types respectively. The appearance model

$$A(\mathcal{I}, p) = \sum_{i \in V} w_i^{t_i} \cdot \phi(\mathcal{I}, p_i) \tag{4.2}$$

gives the score of placing a template $w_i^{t_i}$ for part i and type t_i at location p_i. Here, $\phi(\mathcal{I}, p_i)$ is the HOG feature vector extracted at pixel location p_i. The deformation model

$$D(\mathcal{I}, p) = \sum_{ij \in E} w_{ij}^{t_i, t_j} \cdot \psi(p_i - p_j) \tag{4.3}$$

gives a particular model that controls the relative placement of part i and j by switching between a collection of springs. Each spring is computed for each pair of types (t_i, t_j) and is parameterized by its location and rigidity encoded in $w_{ij}^{t_i, t_j}$. $\psi(p_i - p_j)$

Figure 4.2: Examples of the computed body pose estimation using [150]. Top row shows four query images. Bottom row shows the corresponding detections and the estimated body parts. Lower limbs are depicted using red and blue while upper limbs are shown using cyan and magenta. The torso is shown in yellow and the head in green. The first three column show good results while the fourth column shows a good detection but a wrong pose estimation of the person body parts. Notice that in all four cases a perfect detection is achieved, and shadows and other objects in the scene do not affect the results.

controls the relative location of part i with respect to j. The full score associated with a particular configuration of part types t and positions i is finally computed as

$$S(\mathcal{I}, p, t) = S(t) + A(\mathcal{I}, p) + D(\mathcal{I}, p) \tag{4.4}$$

Then, given the model parameters b and w learned by minimizing a cost function in a supervised learning framework, inference from an image x can be done maximizing $S(x, p, t)$ over p and t. The maximization problem can be addressed in a dynamic programming framework computing

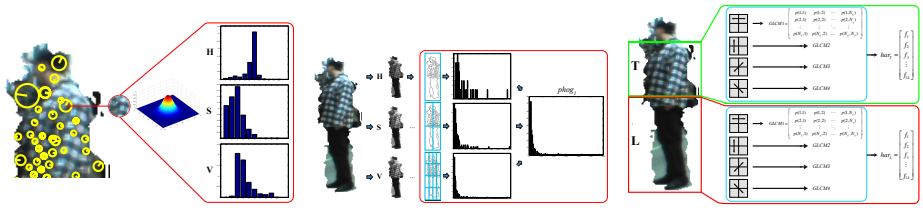
$$score_i(t_i, p_i) = b_i^{t_i} + w_{t_i}^i \cdot \phi(\mathcal{I}, p_i) + \sum_{k \in kids_i} m_k(t_i, p_i) \tag{4.5}$$

$$m_i(t_j, p_j) = \max_{t_i} b_{ij}^{t_i, t_j} + \max score(t_i, p_i) + w_{ij}^{t_i, t_j} \cdot \psi(p_i, p_j) \tag{4.6}$$

for parts i at all pixels locations p_i and part types t_i. $kids_i$ is the set of children of part i in G. After maximizing the score function for a given query image x the set of body part regions is given as $R_i^{t_i}$.

Some examples of the quality of the detection and estimation of body parts in a wide area surveillance scenario are shown in Figure4.2.

Even though the approach in [150] detects multiple articulated body parts through the proposed mixture models method, in this work we only used the two main body parts, namely the torso and legs, to extract the appearance features and to compute



(a) (b) (c)

Figure 4.3: Computed features: In (a) SIFT-based Weighted Gaussian Color Histograms are shown. In (b) PHOG features are shown. In (c) Haralick features for the two detected body parts are shown.

the discriminating signature. This is due to the facts that: i) as shown in [150] state-of-the-art pose estimation algorithms have generally poor performance in case of small person images, especially in detecting limbs; ii) the proposed features are extracted from local interest points and local image patches; iii) the body parts are exploited to reject possible matches between signatures, so the poor estimation of limbs would lead to wrong matches or wrong rejection of matching features. To address such problems the proposed approach groups the detected body part regions $R_i^{t_i}$ into the two main body parts: torso and legs. The head region is discarded because it often consists of a few and less informative pixels.

After computing the estimated body parts $R_i^{t_i}$ for the given input image $\mathcal{I}$, the bounding box $B(\mathcal{I})$ is computed. An hysteresis background subtraction algorithm is then applied to the region $B(\mathcal{I})$ so that the silhouette image region $F(\mathcal{I})$ can be extracted. The wrong estimate positions of limbs are pruned by intersecting $F(\mathcal{I})$ with $R_i^{t_i}$ for all i. Noise removal filters are then applied to get the resulting silhouette $\hat{F}(\mathcal{I})$. Each pixel in $\hat{F}(\mathcal{I})$ is also assigned the corresponding value in p_i so that each pixel is labeled according to its body part. Assuming that people wear the same clothes for the torso and the upper limbs, pixels assigned to the upper limbs are assigned the same torso region label. Finally, labeled pixels are grouped into the two main body parts, i.e. torso and legs, to get the silhouette regions $\hat{F}_T(\mathcal{I})$ and $\hat{F}_L(\mathcal{I})$ corresponding to torso and legs respectively.

4.4 Signature Computation

The problem of re-identifying targets moving across cameras with non-overlapping FoV in a wide area camera network is challenging due to the open issues of multi-camera video analysis such as changes of scale, illumination, viewing angle and pose. The task is even harder when dealing with people due to the non-rigid shape of the human body. To address those issues and build a discriminating signature, four local and global features are extracted and accumulated over multiple frames.

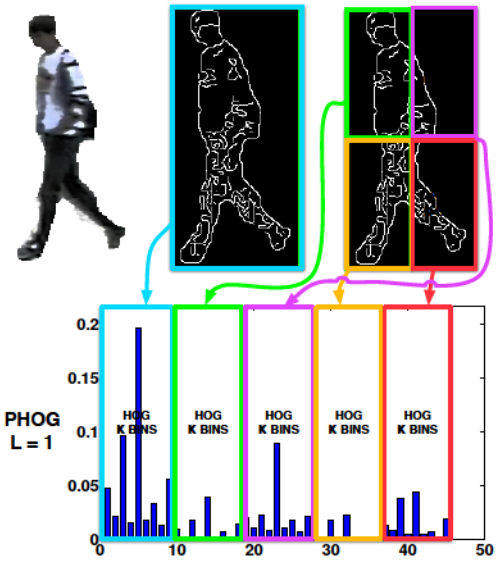


Figure 4.4: Pyramid of Histogram of Oriented Gradients computed using 2 levels ($L = 1$) of the spatial pyramid representation. Here the PHOG is computed for the H component of the HSV color space. The PHOG feature vector is computed concatenating the HOG features extracted for each cell at each level of the spatial pyramid. The resulting feature vector is finally normalized to sum up to 1.

4.4.1 Feature Extraction

As shown in Figure 4.3, the following features are extracted: i) Pyramid Histogram of Orientation Gradients (PHOG); ii) Scale-Invariant Feature Transform (SIFT); iii) SIFT-based weighted Gaussian color histogram (WGCH); iv) Haralick texture features. Each of those features have been properly selected as they capture different information about the given image. PHOG features capture the shape and the spatial layout of the person silhouette. SIFT and WGCH features capture the appearance of the person at specific local regions of interest. Finally, Haralick features capture information about textures. The process of extracting such features is described in details in the following.

Pyramid of Histograms of Oriented Gradients

The Pyramid of Histograms of Orientation Gradients feature [19] captures the local shape and the spatial layout of shape in a given image. To this end the spatial pyramid framework proposed in [79] is exploited. As shown in Figure 4.4, in a spatial pyramid framework, the given image is divided into a sequence of spatial grid cells by repeatedly doubling the number of divisions in each axis direction. That is, the number of points in a grid cell at one level is the sum of the points contained in the

four cells it is divided into at the next level of the pyramid. The number of grid cells at each level of the pyramid gives the number of HOGs that have to be computed at that level.

The PHOG feature vector is computed as a concatenation of all the HOG vectors computed for all the grid cells locations at each level of the spatial pyramid representation, where each bin in the local HOG feature represents the number of edge gradients that have orientations within a certain oriented (i.e. angular) range. The contribution of each gradient to the histogram is weighted by the magnitude of the gradient itself and, similarly to SIFT feature computation, a soft assignment is used to affect neighboring bins. More formally, let K be the number of orientations bins used to compute a single HOG feature vector, and $l \in 0, 1, \cdots, L$ be the level of the spatial pyramid representation such that the number of grid cells at level l of the spatial pyramid is 2^l along each dimension, e.g. at level 0, the concatenated HOG feature vector is of size K. Let HOG^l_K be the concatenation of the HOG feature vectors computed for all the 4^l grid cells. Then the PHOG feature vector for the entire image is a column vector of length $m = K \sum_l^L 4^l$ defined as

$$phog = [HOG^0_K HOG^1_K HOG^2_K \cdots HOG^l_K \cdots HOG^L_K]^T. \tag{4.7}$$

The PHOG feature vector is finally normalized to sum up to unity. Figure 4.5 illustrates this principle showing the PHOG features computed for different values of L.

In our implementation, before extracting the PHOG features from the whole silhouette, $\hat{F}(\mathcal{I})$ is histogram equalized and projected into the HSV color space to achieve illumination invariance. As shown in Figure 4.5, to retain some information about colors, the gradients for each of the hue, saturation and value axes are computed separately only at image locations where an edge is detected by the Canny edge detection algorithm. The PHOG feature matrix $PHOG \in \Re^{m \times 3}$, computed for the given image $\mathcal{I}$ is defined as

$$PHOG(\mathcal{I}) = [phog_h \quad phog_s \quad phog_v] \tag{4.8}$$

where $phog_h$, $phog_s$, and $phog_v$ are the $phog$ feature vectors computed for the hue, saturation and value color components respectively.

SIFT and Weighted Gaussian Color Histogram Features

The SIFT features are jointly used with the Weighted Gaussian Color Histogram features to capture the local chromatic appearance of given person image. Given the silhouette of the whole body $\hat{F}(\mathcal{I})$, the SIFT features are computed by exploiting the proposed cascade filtering approach in which the main four phases are defined: i) detection of scale-space extrema; ii) keypoint localization; iii) orientation assignment; iv) keypoint descriptor building. Then, for each of the detected SIFT features a circular image patch centered at the SIFT keypoint is extracted. The three color components that compose it are separately taken to compute three different histograms weighted by a Gaussian distribution. Due to the robust identification of localized

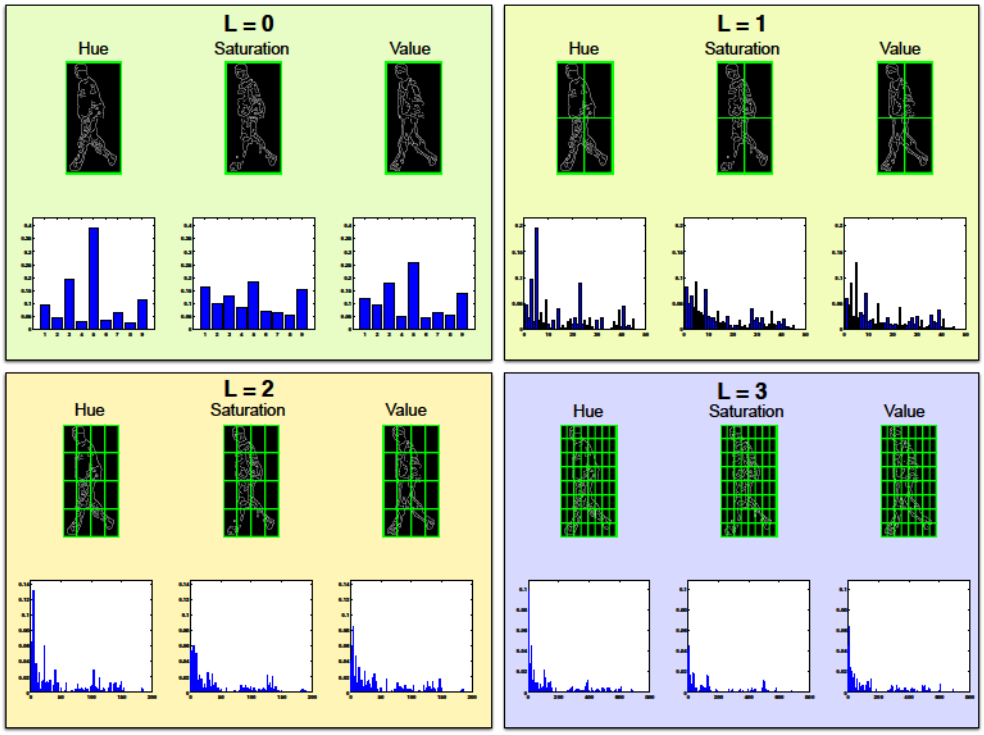


Figure 4.5: Effects of the number of levels used to compute the PHOG feature. PHOG features extracted from the hue, saturation and value color components using different spatial pyramid levels are shown. For each of the four blocks, the top row shows the grid cells (in green) at which the HOG features are extracted. Bottom rows show the final PHOG features for each color component computed concatenating the HOG features extracted at each level of the pyramid.

SIFT keypoints, and to the fact that the farthest part of the patch is given a lower weight, the WGCH captures the local chromatic appearance reducing the occlusion and viewpoint changes issues.

The process of extracting SIFT feature from a given image is not described since it's assumed it's well known. Let define a single SIFT feature as

$$sift = \left\{ sift_{kp}, sift_{hist}, sift_{F} \right\} \tag{4.9}$$

where $sift_{kp} = [x, y]^T$ gives the x and y coordinates of the detected keypoint, $sift_{hist} \in \mathbb{R}^{128}$ is the standard SIFT feature descriptor and $sift_F \in \{T, L\}$ denotes the body part region from which the feature is extracted. All the detected SIFT features are then concatenated to form the feature vector

$$SIFT(\mathcal{I}) = [sift(1) \quad sift(2) \quad \cdots \quad sift(S)] \tag{4.10}$$

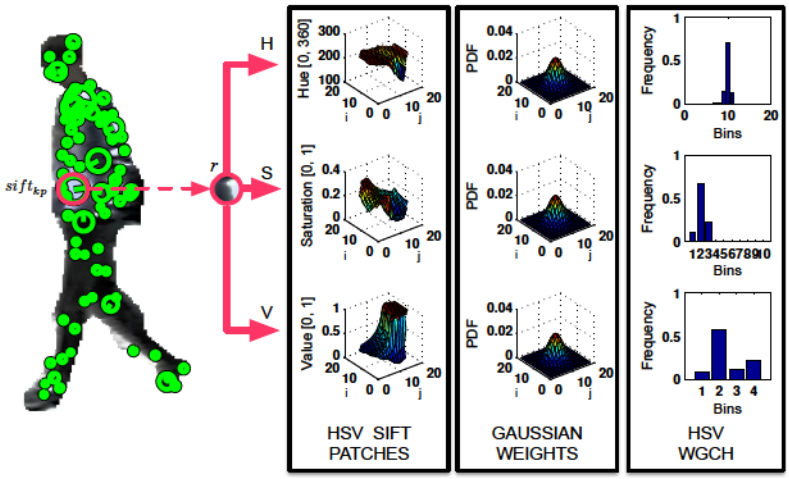


Figure 4.6: Weighted Gaussian Color Histogram (WGCH). The process of computing the WGCH related to a specific SIFT keypoint $sift_{kp}$ is shown. A circular patch of radius r centered at $sift_{kp}$ is extracted and projected to the HSV color space. The first column shows the hue, saturation and value intensities of the given patch. Second column shows the Gaussian weights used to weight the HSV sift patches values. Third column shows the three WGCHs computed for the hue, saturation and value axes using different bin quantizations.

where $sift(k)$ is the k-th SIFT feature extracted from the silhouette $\hat{F}(\mathcal{I})$.

Given a SIFT feature keypoint $sift_{kp}$, the process of computing the related WGCH feature is shown in Figure 4.6. A circular patch R of radius r centered at $sift_{kp}$ is extracted and projected into the HSV color space to better cope with illumination changes and color variations. To compute the WGCH feature vector, here denoted as $wgch$, each element of the patch R at coordinates i, j is weighted by the probability density value at i, j of a Gaussian probability density function with mean $\mu = [r/2, r/2]$ and diagonal covariance $\Sigma \in \mathbb{R}^{2\times 2}$. This can be written as follows. Let $[b, t)$ be a single bin range of the WGCH and $R_{i,j}$ be the pixel value at coordinates i, j of the patch R, then, if $b \leq R_{i,j} < t$

$$wgch(b, t) = wgch(b, t) + \mathcal{N}(\mu, \Sigma)_{i,j} \tag{4.11}$$

where $\mathcal{N}(\mu, \Sigma)_{i,j}$ is the value at location i, j of a Gaussian probability density function. The computed WGCH is then normalized to sum up to 1. Since the WGCH is computed for the hue, saturation and value patches, we end up with three WGCHs denoted as $wgch_h \in \mathbb{R}^{b_h}$, $wgch_s \in \mathbb{R}^{b_s}$, and $wgch_v \in \mathbb{R}^{b_v}$ where b_h, b_s, and b_v are the number of bins used for quantization of the hue, saturation and value components respectively. We denote a single WGCH feature as

$$wgch = [wgch_h, wgch_s, wgch_v]^T . \tag{4.12}$$

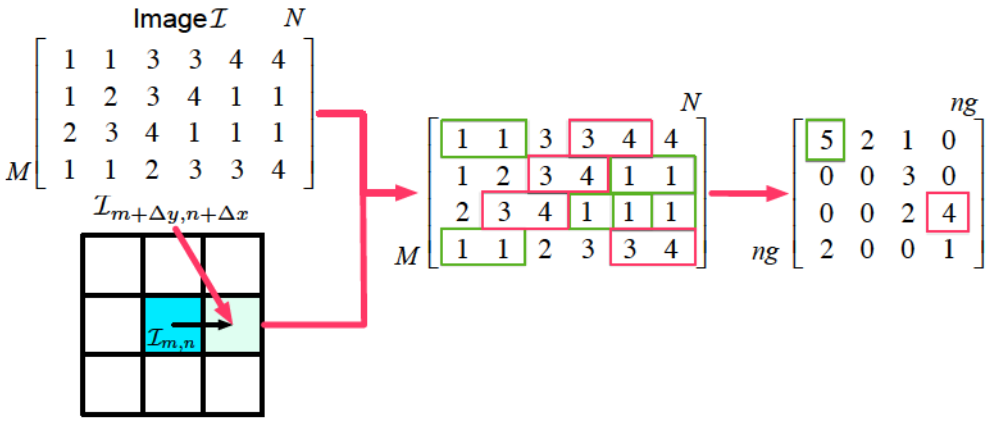


Figure 4.7: Gray level co-occurrence matrix. Given the gray scale input image $\mathcal{I}$ and the adjacency mask (bottom left) the gray level co-occurrence matrix (rightmost) is formed by counting the number of adjacent pixels that have gray intensity level equals to (a, b). Here an example of computing a GLCM using offset $\Delta x = 1$ and $\Delta y = 0$ and $ng = 4$ gray levels is shown. Green boxes show pixels with intensity values $(a = 1, b = 1)$ that are adjacent according to the offset. Red boxes highlight pixels with intensity values $(a = 3, b = 4)$ for the same offset.

As WGCH features are extracted from the previously computed SIFT features, we end up with the WGCH feature matrix

$$WGCH(\mathcal{I}) = [wgch(1) \quad wgch(2) \quad \cdots \quad wgch(S)] \tag{4.13}$$

where $wgch(k)$ is the WGCH associated to the k-th SIFT feature $sift(k)$.

Haralick Features

The Haralick feature captures information about the patterns that emerge in the image texture. In particular, Haralick feature captures information about the image textures such as the homogeneity, the gray level linear dependencies, the contrast, the number and the nature of edges, and the complexity of the image itself. The Haralick texture features are calculated in the spatial domain, and they rely on the assumption that the texture information in an image is contained in the spatial relationship between the image gray levels.

To extract the Haralick texture features, a set of gray level co-occurrence matrix (GLCM) is used. A gray level co-occurrence matrix defined over an image is a matrix that describes the distribution of co-occurring gray level pixel values at a given offset. Such gray-level co-occurrence matrix is a function of the angular relationship between the neighboring pixels in the image as well as a function of the distance between them. As shown in Figure 4.7, given an image $\mathcal{I}$ of size $M \times N$, with $a, b = 1, 2, \cdots, ng$

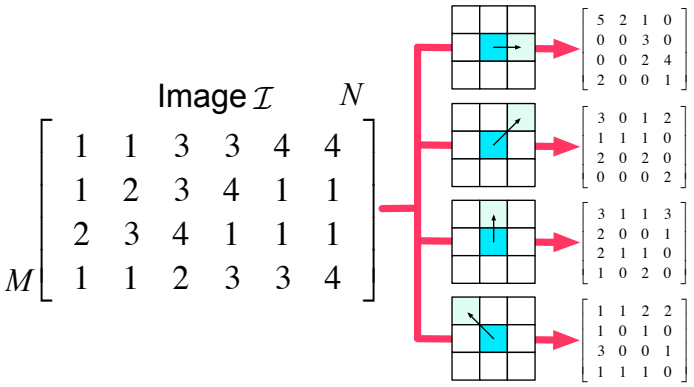


Figure 4.8: Complete toy-example where GLCM are computed using the four adjacency matrix suggested in [57]. First column shows the gray scale input image I. Second column shows the four different offsets. The blue pixel is the considered pixels, while the cyan pixel pointed by the arrow is the adjacent pixel. The four resulting gray level co-occurrence matrices are depicted in the third and last column.

gray levels, the gray level co-occurrence matrix $GLCM \in \mathbb{R}^{ng \times ng}$ defined over $\mathcal{I}$ is parameterized by the adjacency matrix (offsets Δx and Δy). Given such offset values, the GLCM is computed as

$$GLCM_{a,b}^{\Delta x, \Delta y} = \sum_{n=1}^{N} \sum_{m=1}^{M} \begin{cases} 1, & \text{if } \mathcal{I}_{m,n} == a \ \wedge \mathcal{I}_{m+\Delta y, n+\Delta x} == b \\ 0, & \text{otherwise} \end{cases} \tag{4.14}$$

where $\mathcal{I}_{m,n}$ is the gray level pixel intensity of image $\mathcal{I}$ at coordinates (m, n).

Haralick features rely on the assumption that image texture information is contained in the GLCM, so Haralick features are extracted from the computed GLCM. However, the parameters Δx and Δy lead to different GLCM and different values for the pixel intensity pairs (a, b) and (b, a). This would make the GLCM, hence the Haralick features, sensitive to rotation. To deal with this issue, in [57] it was suggested to: i) use the following offset Δx and Δy values: $\Delta x = 1$, $\Delta y = 0$ (0°); $\Delta x = 1$, $\Delta y = 1$ (45°); $\Delta x = 0$, $\Delta y = 1$ (90°); $\Delta x = -1$, $\Delta y = 1$ (135°). ii) tale the GLCM matrix entries as symmetric so that both (a, b) and (b, a) pairings are computed by counting the number of times the value a is adjacent to the value b; iii) advantage of pooling and average the resulting GLCM over multiple images. If so, some invariance to rotations is also achieved.

To use Haralick features to compute a discriminative signature for re-identification we rely on the assumption that most of the people wear different clothes for the bottom and for the lower body parts. In light of this we extract two Haralick texture feature vectors: one for the torso and one for the legs silhouettes respectively. Let $GLCM_T(\mathcal{I})$ and $GLCM_L(\mathcal{I})$ be the GLCM matrices computed for the torso and legs regions of a given person's images $\mathcal{I}$. Then, following the details in [57], those are used to extract

the two 14 dimensional feature vectors $HAR_T(\mathcal{I}) \in \mathbb{R}^{14}$ and $HAR_L(\mathcal{I}) \in \mathbb{R}^{14}$, where $HAR_T(\mathcal{I})$ is the Haralick feature vector computed for the torso region and $HAR_L(\mathcal{I})$ is the Haralick feature vector computed for the legs region.

4.4.2 Feature Accumulation

To form the discriminating signature the signature computation module accumulates the four features extracted from each single frame of a given person. PHOG and Haralick feature are accumulated by pooling the feature vectors, whereas SIFT and WGCH features are accumulated using an iterative procedure. In particular, SIFT feature descriptors are used to match SIFT features extracted from two different frames. If a valid match between them is detected, the associated WGCH are compared and accumulated. The whole feature accumulation process is described hereby.

Let $\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_N$ be the N frames of a same person acquired by camera a camera of the network. The accumulated PHOG feature matrix is given by pooling the PHOG feature vectors computed for all the N frames as

$$PHOG(1, N) = \frac{1}{N} \sum_{n=1}^{N} PHOG(\mathcal{I}_n) \tag{4.15}$$

where $PHOG(\mathcal{I}_n)$ is the PHOG feature vector extracted from the n^{th} frame.

The SIFT and the WGCH features are accumulated through an iterative procedure that applies the following three steps: i) SIFT features matching; ii) WGCH features matching; iii) accumulation of WGCH and SIFT features. The process is defined in algorithm 1 and shown in Figure 4.9.

Let $SIFT(1, N-1)$ denote the set of SIFT features accumulated for frames $\mathcal{I}_1$, $\mathcal{I}_2$, $\cdots$, $\mathcal{I}_{N-1}$. Let d_{L^2} be the L^2-norm distance, S_{N-1} and S_N be the number of SIFT features in $SIFT(1, N-1)$ and $SIFT(N, N)$ respectively. Then, a match i, j, where $i \in \{1, 2, \cdots, S_{N-1}\}$ and $j \in \{1, 2, \cdots, S_N\}$ are the indexes of two SIFT features in $SIFT(1, N-1)$ and $SIFT(N, N)$, is detected if

$$d_{L^2}(sift_{hist}(i), sift_{hist}(j)) \tag{4.16}$$

is lower than a given threshold Th_{sift} and the two SIFT keypoints lie on the same body part.

Given a match i, j between SIFT feature, the corresponding WGCH features $wgch(i)$ and $wgch(j)$ are compared using the d_{χ^2} distance measure and weighted by the Mahlanobis distance computed for the two body parts. Since multiple WGCHs can be related to a single SIFT feature of index i (details in the following) the distance between them is given by the average value of their distances.

Let d_M be the Mahlanobis distance

$$d_M(p, F) = \sqrt{(p - \mu_F) S_F^{-1} (p - \mu_F)} \tag{4.17}$$

where $p = [x, y]^T \in F$ is a pixel coordinates vector, and μ_F and S_F^{-1} are the centroid and the covariance matrix of a silhouette body part region $F \in \{\hat{F}_L, \hat{F}_T\}$. An example

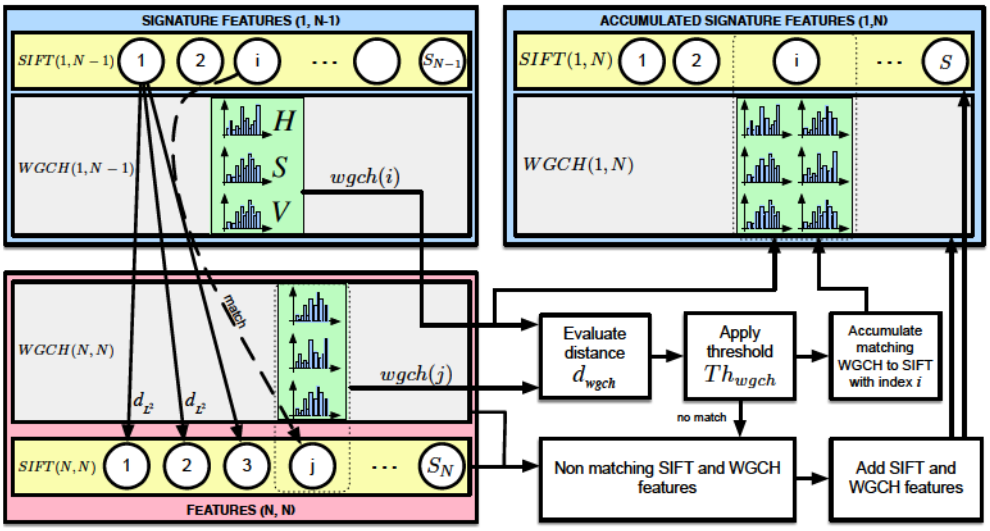


Figure 4.9: Accumulation of SIFT and WGCH features. The i-th SIFT feature in $SIFT(1, N-1)$ that is part of the initial signature is compared with the j-th SIFT features extracted from the N-th frame of a given person using the d_{L^2}-norm distance. Matching SIFT keypoints that lie on the same body part are kept and the related WGCH features are compared using the d_{wgch} distance and thesholded with Th_{wgch}. If the computed distance is lower than the threshold, the i-th SIFT feature descriptor is updated to be the average between the two matching SIFT descriptors and the WGCH related to j-th SIFT feature is assigned to the updated i-th SIFT feature. The SIFT features and the WGCH that do not match are added to the new signature features $SIFT(1, N)$ and $WGCH(1, N)$.

of the Mahlanobis distance computed for the three body parts of a person's silhouette is depicted in Figure 4.10.

Also, let

$$d_{\chi^2}(wgch(i), wgch(j)) = \frac{1}{3}\sum_{a=1}^{3} \lambda_a \chi^2\left(wgch_a(i), wgch_a(j)\right) \tag{4.18}$$

be the weighted distance between WGCHs, where $\chi^2(\cdot,\cdot)$ is the average chi-squared distance computed between the WGCHs $wgch_a(i)$ and $wgch_a(j)$ associated to the i-th and j-th SIFT features respectively. a denotes the color component from which the feature is extracted, and λ_a is a weight factor such that $\sum_a \lambda_a = 1$.

Then, given two matching SIFT features $sift(i)$ and $sift(j)$, the distance between

Algorithm 1: Accumulate SIFT and WGCH features

input : The SIFT feature vectors $SIFT(1, N-1)$ and $SIFT(N, N)$
The WGCH feature vectors $WGCH(1, N-1)$ and $WGCH(N, N)$
output: The accumulated SIFT feature vector $SIFT(1, N)$ and
The accumulated WGCH feature vector $WGCH(1, N)$
Result: Accumulated SIFT and WGCH features

$S_{N-1} \leftarrow$ number of SIFT features in $SIFT(1, N-1)$;
$S_N \leftarrow$ number of SIFT features in $SIFT(N, N)$;
for $i \leftarrow 1$ **to** S_{N-1} **do**
  for $j \leftarrow 1$ **to** S_N **do**
    $match_{sift} \leftarrow d_{L^2}(sift_{hist}(i),\ sift_{hist}(j)) < Th_{sift}$;
    $match_{wgch} \leftarrow d_{wgch}(wgch(i),\ wgch(j)) < Th_{wgch}$;
    if $match_{sift} \wedge sift_F(i) == sift_F(j) \wedge match_{wgch}$ **then**
      `Accumulate`$(sift_{hist}(i),\ sift_{hist}(j),\ wgch(i),\ wgch(j))$;
    else
      `Add`$(sift_{hist}(i),\ sift_{hist}(j),\ wgch(i),\ wgch(j))$;
    end
  end
end

the associated WGCHs $wgch(i)$ and $wgch(j)$ is computed as

$$d_{wgch}(wgch(i), wgch(j)) = \max\Big(d_M(sift_{kp}(i), sift_F(i)), \\ , d_M(sift_{kp}(j), sift_F(j))\Big) \\ \times d_{\chi^2}(wgch(i), wgch(j))$$

where $sift_F(i) = sift_F(j)$ is the body part silhouette region into which SIFT keypoints $sift_{kp}(i)$ and $sift_{kp}(j)$ lie. Recall, a match between SIFT features is allowed only if they lie on the same body part.

If the distance d_{wgch} between the WGCH features associated to the matching SIFT features with index i, j is lower than a given threshold Th_{wgch} a correct match is detected. Then, $sift_{hist}(i)$ is updated such that it is the average between $sift_{hist}(j)$ and itself. $wgch(j)$ is also accumulated and assigned to the updated SIFT feature $sift(i)$ (see Figure 4.9). All the non-matching features in $SIFT(N, N)$ are added to $SIFT(1, N-1)$ such that $SIFT(1, N) = [SIFT(1, N-1), SIFT(N, N)]$. The same applies to the WGCH features, that is $WGCH(1, N) = [WGCH(1, N-1), WGCH(N, N)]$, where, as for $SIFT(N, N)$, $WGCH(N, N)$ here denotes the set of all non matching WGCH features.

As for the PHOG, Haralick features are accumulated by pooling the feature vectors

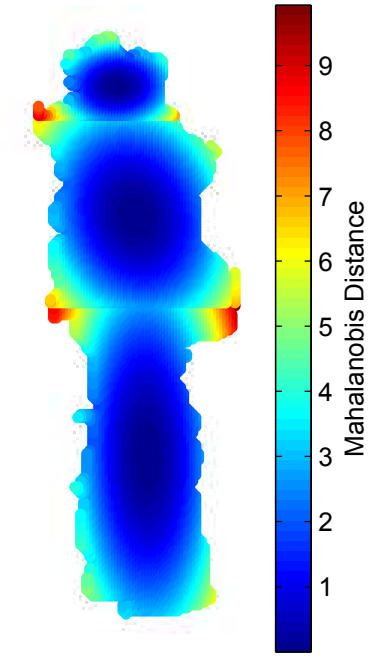


Figure 4.10: Mahalanobis distance computed for the three detected body parts of a person's silhouette.

computed for all the N frames as

$$HAR_T(1,N) = \frac{1}{N}\sum_{n=1}^{N} HAR_T(\mathcal{I}_n) \tag{4.19}$$

$$HAR_L(1,N) = \frac{1}{N}\sum_{n=1}^{N} HAR_L(\mathcal{I}_n) \tag{4.20}$$

where $HAR_T(\mathcal{I}_n)$ and $HAR_L(\mathcal{I}_n)$ are the Haralick feature vectors extracted from the torso and legs region of the n-th frame. According to [57] by pooling such vectors extracted using the same offsets for the gray level co-occurrence matrices across multiple images invariance to rotations is achieved.

Given the accumulated features, the discriminative person signature is defined as

$$\begin{aligned}\Phi(p,c) = \Big\{ & PHOG^{(p,c)}(1,N), SIFT^{(p,c)}(1,N), WGCH^{(p,c)}(1,N), \\ & HAR_T^{(p,c)}(1,N), HAR_L^{(p,c)}(1,N) \Big\}\end{aligned} \tag{4.21}$$

where the superscript of each feature vector denotes that the feature is extracted from person p viewed by camera c.

4.5 Signature Matching

In this section a method to match a probe signature with a gallery signature is introduced.

Given a pair of signatures the signature matching module compares them through a linear weighted distance computed between all the signatures features. The PHOG and the WGCH features are compared using a χ^2 distance metric, the Haralick feature vectors and the SIFT feature descriptors are matched using the L^2-norm distance. In particular, SIFT and WGCH are matched using similar functions to the ones introduced in section 4.4.2.

Let $\mathcal{C} = \{c_1, c_2, \cdots, c_{|\mathcal{C}|}\}$ be the set of all cameras in the network. Let $\Phi(p, c_i)$ be the probe signature of person p viewed by camera $c_i \in \mathcal{C}$ and $\Phi(g, c_j)$ be the gallery signature of person g viewed by camera $c_j \in \mathcal{C}$. The signature distance between $\Phi(p, c_i)$ and $\Phi(g, c_j)$ is computed as a linear combination of i) the d_{PHOG} distance between PHOG signature features, ii) the d_{WGCH} distance between WGCH signature features, and iii) the d_{HAR} distance between Haralick signature features. For the sake of readability, the notation $(1, N)$ that defines the number of images used to compute the accumulated signature is omitted in the following.

Let

$$erf(x) = \frac{2}{\pi} \arctan(\frac{\pi}{2} x) \tag{4.22}$$

be a special function of sigmoid shape that is used in the following to keep the feature distances within the range $[0, 1]$.

The signatures PHOG features $PHOG^{(p,c_i)}$ and $PHOG^{(g,c_j)}$ are compared using a weighted χ^2 distance

$$d_{PHOG}\left(PHOG^{(p,c_i)}, PHOG^{(g,c_j)}\right) = erf\left(\sum_{a=1}^{3} \kappa_a \chi^2\left(phog_a^{(p,c_i)}, phog_a^{(g,c_j)}\right)\right) \tag{4.23}$$

where $phog_a^{p,c_i}$ and $phog_a^{g,c_j}$ are the PHOG feature vectors computed for the a-th HSV component and κ_a is the weight factor, such that $\sum_a \kappa_a = 1$.

All the SIFT features in $SIFT^{(p,c_i)}$ and $SIFT^{(g,c_j)}$ are matched by computing the L^2-norm distance between SIFT descriptors as defined in eq. (4.16). As before, SIFT features that do not lie on the same body part are rejected. If this condition is satisfied, then the same threshold Th_{sift} is used to get all the pairwise matches $k, l \in \mathbb{M}$, where $sift(k) \in SIFT^{(p,c_i)}$ and $sift(l) \in SIFT^{(g,c_j)}$. The matching k, l SIFT features are then used to compute the pooled distance between WGCHs signature features as

$$d_{WGCH}(WGCH^{(p,c_i)}, WGCH^{(g,c_j)}) = erf\left(\frac{1}{\epsilon + |\mathbb{M}|} \sum_{k,l \in matches} d_{wgch}\left(wgch(k), wgch(l)\right)\right) \tag{4.24}$$

where ϵ is a small constant used to avoid the division by zero in case the number of SIFT matches between the two signatures, here denoted as $|\mathbb{M}|$, is equal to zero.

Given the four accumulated Haralick features vectors extracted from the two body parts of the two detected persons, the distance between those features is computed as

$$d_{HAR}(HAR^{(p,c_i)}, HAR^{(g,c_j)}) = erf\Big(d_{L^2}\big(HAR_T^{(p,c_i)}, HAR_T^{(g,c_j)}\big) + \\ + d_{L^2}\big(HAR_L^{(p,c_i)}, HAR_L^{(g,c_j)}\big)\Big) \tag{4.25}$$

Finally, the overall distance between the two signatures $\Phi(p, c_i)$ and $\Phi(g, c_j)$ is computed as an affine combination of the distances computed for all the signatures features. It is defined as

$$\begin{aligned} d\left(\Phi(p,c_i), \Phi(g,c_j)\right) &= \alpha \; d_{PHOG}\left(PHOG^{(p,c_i)}, PHOG^{(g,c_j)}\right) \\ &+ \beta \; d_{WGCH}\left(WGCH^{(p,c_i)}, WGCH^{(g,c_j)}\right) \\ &+ \gamma \; d_{HAR}\left(HAR^{(p,c_i)}, HAR^{(g,c_j)}\right) \end{aligned} \tag{4.26}$$

where α, β and γ are the weight factors.

4.6 Experimental Results

To evaluate the performance of the proposed method two public datasets have been used, namely the ETHZ [128] and the CAVIAR4REID [26] datasets. Each one covers different aspects and challenges for the person re-identification problem. A comparison and details of the used person re-identification datasets is given section 2.2.6.

As commonly suggested by the literature, we report the performance of our method in terms of recognition rate by the Cumulative Matching Characteristic (CMC) curve and the normalized Area Under Curve (nAUC) score for the CMC curve. We report the results of our approach using both a single-shot (i.e. $N = 1$) and a multiple-shot (i.e. $N > 1$) strategy. To compute the CMC curves, the proposed distance is used to match the gallery signatures with the probe signatures. To fairly evaluate our method against state-of-the-art approaches we perform the whole re-identification procedure 10 times using different sample images. We report the CMC curves and nAUC values averaged over the 10 trials.

4.6.1 Implementation Details

The following parameters have been kept fixed for all evaluations and have been selected using 4-fold cross-validation. To obtain better results the parameters could have been differently selected for each dataset, but we averaged them for all the datasets to have a more general re-identification setting. According to this, the following parameters have been set:

i) PHOG features: $K = 9$ orientation bins were used to compute a single HOG vector and $L = 3$ pyramid levels were used;

ii) SIFT and WGCH features: WGCH histograms were extracted from patches having radius $r = N/8$ (where N is the width of the considerd image) and quantized using 16, 10, and 4 bins for the H, S and V channels respectively. The gaussian weights were defined by a normal distribution having $\mu = [r/2, r/2]^T$ and diagonal covariance Σ with non-zero entries equal to r;

iii) Haralick features: the GLCM matrices were computed using the same symmetric offsets suggested in [57].

iv) Feature accumulation: λ was set to $[0.5, 0.3, 0.2]$ so as the value channel is given less importance. The matching thresholds Th_{sift} and Th_{wgch} were set to 0.1 and 0.15 respectively.

v) Signature matching: κ was set to $[0.5, 0.25, 0.25]$. α, β and γ were set to 0.4, 0.2 and 0.4 respectively.

4.6.2 ETHZ Dataset

To make this dataset more challenging, we followed the strategy proposed in [8] by randomly picking a set of 10 consecutive frames from the beginning and from the end of each sequence. Following the evaluation setup in [128] and [13], all images have been resized to 64×32 pixels. We evaluate our method using both a single-shot ($N = 1$) and a multiple-shot strategy ($N \in \{5, 10\}$). We report comparisons to discriminative signature approaches and to state-of-the-art approaches that exploit machine learning techniques. Notice that these kind of approaches uses all the person for training and for testing but with different images samples for the two phases.

In Table 4.1 we report the results achieved by our method using a single-shot strategy. The first 3 rows of the table show the comparisons with state-of-the-art discriminative signature methods. Last 6 rows show the comparison with state-of-the-art approaches that exploit machine learning techniques. Regarding the comparison with discriminative signature methods we achieve similar performance to the two state-of-the-art methods used for comparisons, namely SDALF [13] and eBiCOV [88]. For SEQ.#1 we have have similar performance to SDALF and ICT [4] for rank 1, then we achieve the same performance of eBiCOV and get very close to the recognition percentage achieved at rank 7 by learning based methods. The same behavior is shown for the other two sequences, namely SEQ.#1 and SEQ.#2, where low rank values give similar recognition percentages to the ones achieved by discriminative signature methods, while higher rank values get very close to learning based methods also.

In Table 4.6.2 we report the results of our method using a multiple-shot strategy. Top 7 rows show the results compared with discriminative signature methods where 5 images (top 3 rows) and 10 images (next 4 rows) are used to compute both the gallery and the probe signatures. Last 4 rows show the results of learning based methods. For these multiple-shot scenarios our method gets performance that meet the ones achieved by both discriminative signature and learning based methods. In particular, for SEQ.#3, we achieve 100% recognition rate at rank 3 using 5 images and at rank 2

Table 4.1: Comparison of the proposed method on the ETHZ dataset using a single shot-strategy. Top 3 rows show comparisons with discriminative signature methods, while the last 6 show the results of state-of-the-art methods that involves learning algorithms. Recognition rates for top 7 ranks are shown for each of the three sequences. Best recognition rates for discriminative signature methods are shown in italic. Overall best recognition rates for each rank are shown in boldface font.

Method	SEQ.#1							SEQ.#2							SEQ.#3						
	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
Proposed	67	79	82	*85*	*87*	*89*	*90*	68	78	*84*	*86*	*88*	*90*	*91*	81	*87*	*91*	*93*	*94*	*95*	*96*
SDALF [13]	65	73	77	79	81	82	84	64	74	79	83	85	87	89	76	83	86	88	90	92	93
eBiCOV [88]	*74*	*80*	*83*	*85*	*87*	88	89	*71*	*79*	83	*86*	*88*	*90*	*91*	*82*	*87*	90	92	93	94	95
eLDFV [89]	**83**	**87**	**90**	**91**	**92**	**93**	**94**	79	84	87	90	91	92	93	**91**	94	96	97	97	97	97
eSDC_knn [152]	81	86	89	90	**92**	**93**	**94**	79	84	87	90	91	92	93	90	**95**	96	97	**98**	**98**	**99**
eSDC_ocsvm [152]	80	85	88	90	91	92	93	**80**	**86**	**89**	**91**	**93**	**94**	**95**	89	94	96	97	**98**	**98**	**99**
RPLM [60]	77	83	87	90	91	92	92	65	77	81	82	86	89	90	83	90	92	94	96	96	97
IBML [59]	78	84	87	89	90	91	91	74	81	84	87	89	91	92	**91**	**95**	**97**	**98**	**98**	**98**	**99**
ICT [4]	68	76	82	86	87	89	90	70	82	**89**	**91**	**93**	**94**	**95**	**91**	94	96	97	97	**98**	98

Table 4.2: Comparison of the proposed method on the ETHZ dataset using a multiple-shot strategy. Top 7 rows show comparisons with discriminative signature methods, while the last 4 show the results of state-of-the-art methods that involves learning algorithms. Recognition rates for top 7 ranks are shown for each of the three sequences. Best recognition rates for discriminative signature methods are shown in italic. Overall best recognition rates for each rank are shown in boldface font.

Method	SEQ.#1							SEQ.#2							SEQ.#3						
	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
Proposed (5 images)	90	91	92	93	94	94	95	86	92	94	95	96	96	97	94	99	**100**	**100**	**100**	**100**	**100**
PLS [128](all images)	79	85	86	87	88	89	90	74	79	81	83	84	85	87	77	81	82	84	85	87	89
eBiCOV [88](5 images)	*93*	*94*	*95*	*95*	*96*	96	96	91	94	95	96	97	97	97	*98*	98	99	**100**	**100**	**100**	**100**
Proposed (10 images)	91	92	*95*	*96*	*96*	*97*	*97*	*92*	*95*	*96*	*97*	*98*	*99*	*99*	97	**100**	**100**	**100**	**100**	**100**	**100**
eBiCOV [88](10 images)	*93*	*94*	*95*	*96*	*96*	96	96	91	*95*	*96*	*97*	*98*	*99*	*99*	97	98	99	**100**	**100**	**100**	**100**
SDALF [13](10 images)	91	92	93	94	94	94	94	91	94	96	96	97	97	98	94	95	96	96	96	96	96
AHPE [11] (10 images)	85	89	92	93	94	94	95	80	86	89	92	93	94	95	83	91	92	94	96	97	97
eLDFV [89](5 images)	**96**	**97**	97	97	**98**	**98**	**98**	**97**	**98**	**99**	**100**	**100**	**100**	**100**	**99**	**100**	**100**	**100**	**100**	**100**	**100**
LDC [151](5 images)	92	95	96	97	**98**	**98**	**98**	92	95	97	98	99	99	99	96	97	98	99	99	99	99
ICT [4](5 images)	92	93	94	95	96	96	97	95	**98**	**99**	99	**100**	**100**	**100**	95	96	97	99	**100**	**100**	**100**
BRM [8](10 images)	**96**	**97**	**98**	**98**	**98**	**98**	**98**	94	95	95	95	95	95	96	96	98	**100**	**100**	**100**	**100**	**100**

using 10 images, thus outperforming two very recent learning based methods, namely LDC [151] and ICT [4].

4.6.3 CAVIAR Dataset

To make a fair comparison with state-of-the-art algorithms images have been resized to 128×64. Evaluation performance has been computed with $N \in \{1, 3, 5\}$. In Figure 4.11 results are reported for these three different cases. The comparisons are given for 5 discrminative signature methods, namely SDALF [13], AHPE [12], CPS [26], CI (comb) [78] and MRCG [7], and 2 learning-based methods, namely, LAFT [82] and LDC [151]. Notice that, for learning-based methods, while different image samples are used, all the 50 persons are used both for training and for testing.

In Figure 4.11(a) evaluation performance computed using the single-shot scenario are shown. Under this scenario, while we have worse performance than LAFT, results are better than all discriminative signature methods. For higher ranks we're also getting very close to LAFT performance, where we're achieving 82.5% of correct recognitions at rank 25.

In Figure 4.11(b) evaluation performance are computed using the multiple-shot scenario with $N = 3$. Similarly to the previous results, only LAFT is performing better than our method. Also, we're having very similar results as LAFT for rank 1 and rank 25, where a correct recognition percentage of 17% and 89% are achievd respectively. All other discriminative signature methods are outperformed.

In Figure 4.11(c) evaluation performance are computed using the multiple-shot scenario with $N = 5$. As for the single-shot scenario and for the multiple-shot scenario with $N = 3$ we are getting similar performance to CPS and we outperform all other discriminative signature methods. In this case we're also having very similar performance to LDC where for lower ranks (i.e. from rank 2 to rank 10) the difference in performance is under 2%.

4.6.4 Discussion

In this section results of the proposed approach are compared to state-of-the-art person re-identification methods on the ETHZ and the CAVIAR4REID benchmark datasets. For both the dataset the proposed method has performance that are similar or even better than discriminative signature based state-of-the-art methods. In some cases (see Table 4.6.2 and Figure 4.11(c)) the proposed method is also outperforming learning-based methods that use the same persons for both traning and testing. In particular, results show that good performance are achieved when multiple images of the same person are used. This is a good point as, in a real scenario, it's plausible to assume that multiple images of a same person can be extracted using intra-camera tracking techniques. Notice that, the feature accumulation part and the signature matching mechanism can also be used to deal with this and perform the intra-camera tracking.

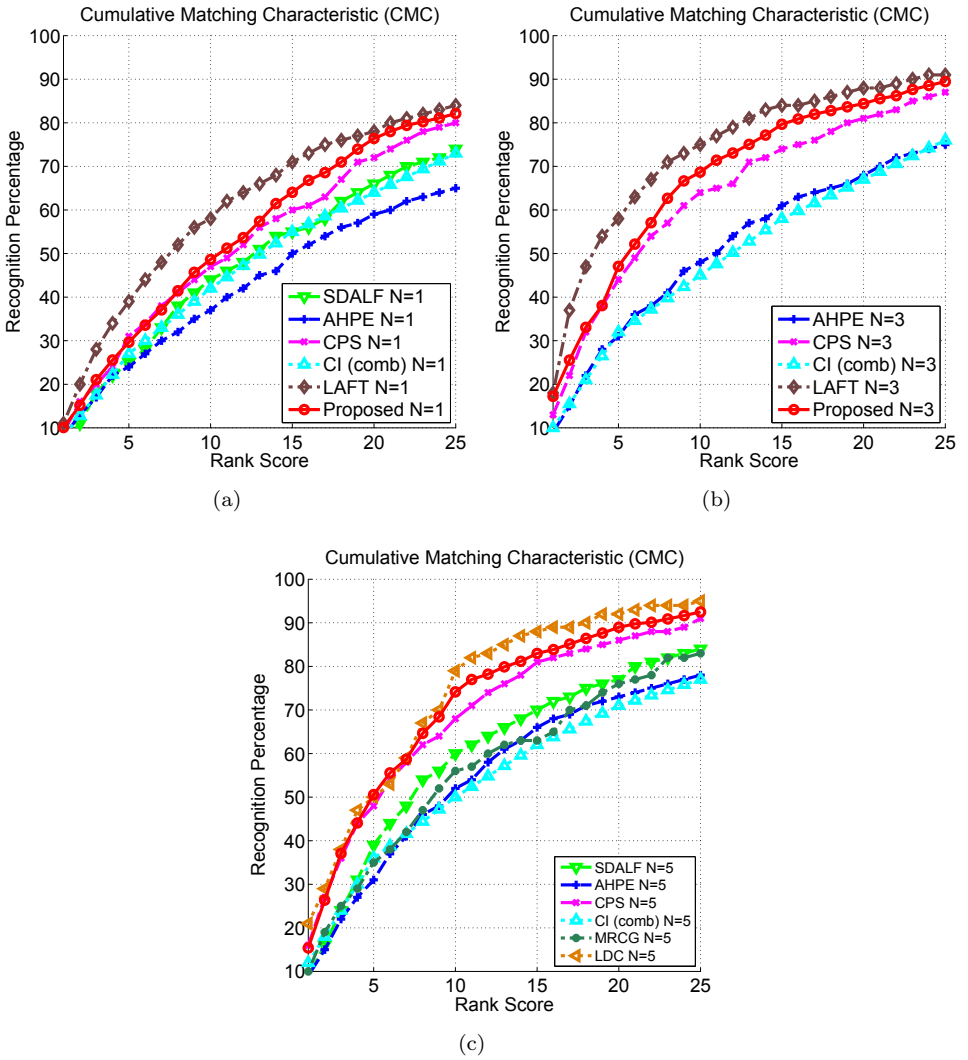


Figure 4.11: Comparison of the proposed approach on the CAVIAR4REID dataset using both a single-shot and a multiple shot approach. In (a) comparisons with other methods using $N = 1$ are shown. In (b) and (c) multiple-shot results with $N = 3$ and $N = 5$ are shown.

4.7 Conclusion

In this chapter, the person re-identification problem is addressed by means of a discriminative signature based method. Local and global features are extracted from the silhouettes of a detected person and accumulated through multiple frames to form the

discriminative signature. The so computed signatures are compared to gallery signatures using an affine combination of feature distances. To show the performance of the proposed method, results are compared to state-of-the-art approaches. Such evaluation shows that it achieves similar or superior performance to those.

4.8 What next?

While being effective yet simple, this methods is based on purely match features across images, thus it misses an important and interesting point in doing this. That is, while being invariant to some issues, appearance features get transformed across cameras. In the next chapters, this interesting problem is considered to address the re-identification challenges.

5

Re-Identification by Classification of Warp Feature Transformation

In this chapter, the person re-identification problem is addressed by studying the nature of the transformation of features across cameras. In the first part of the chapter, a brief introduction about how such transformation can be modeled is given, then a description about how to use such transformation to re-identify persons moving between cameras is given. Results and comparisons with state-of-the-art methods on benchmark datasets are provided at the end of the chapter.

5.1 Introduction

In the previous chapter the problem of person re-identification has been addressed by designing discriminative signatures that are matched between persons acquired by disjoint cameras. However, the method rely on the fact that the individual signatures vary a little from camera to camera. This leads to the fact that, a significant loss of performance is present when strong illumination and color changes occur between different cameras. As a result of these changes, features describing the same person get transformed between cameras. Thus an important aspect of the problem is to understand how features get transformed across cameras. In chapter 2, the existing studies exploiting feature transformation have been introduced. They have tried to learn linear [49] and nonlinear transformation functions [115, 70] between appearance features among pairs of cameras. However, they use the learned transformation function to project the features from one camera to the feature space of the other camera. In a re-identification scenario this may not always be feasible since the mapping may not be unique and it may vary from frame to frame depending on a large number of camera parameters (e.g. illumination, scene geometry, exposure time, focal length, and aperture size). In this chapter the goal is to understand the space of feature transformation functions, termed as the feature *warp function space* (WFS) and re-

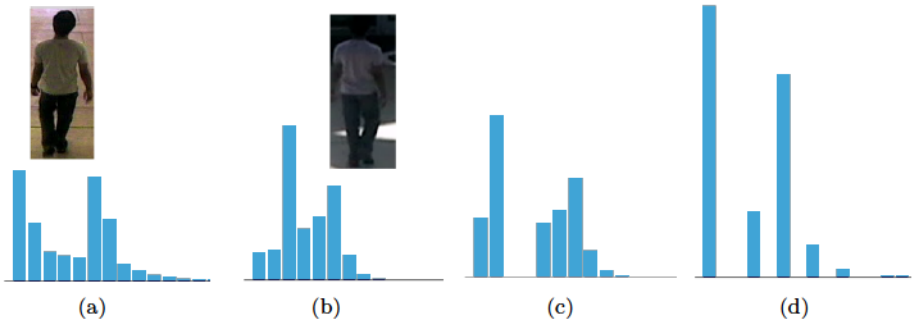
identify targets by learning and classification in this function space of nonlinear warps between features [99].

Considering two non-overlapping cameras, a pair of images of the same target is denoted as a feasible pair, while a pair of images between two different targets is denoted as an infeasible pair. The corresponding warp functions describing the transformation of features are denoted as *feasible* (positive) and *infeasible* (negative) warp functions respectively. The set of infeasible warp functions vary widely as in this set the warps are computed for image pairs consisting of different persons. Even within the set of feasible warps, the transformations are not unique when computed for different feasible pairs. For each of the two sets, the feature transformations may not be well represented by a single warp function in presence of such variabilities. So, the idea is to model the function space capturing all the feasible and infeasible warps between pairs of cameras. The WFS not only allows to model feasible transformation between pairs of instances of the same target, but also to separate them from the infeasible transformations between instances of different targets. This enables to address the re-identification problem as a binary classification problem by discriminating in the WFS. In Figure 5.1 an example of the use of warp functions is shown. Here, the histogram obtained by warping the histogram from Figure 5.1(a) to Figure 5.1(b) is much similar compared to the brightness transfer function (BTF) [70] method. Figure 5.1(e) shows a comparison of the performances achieved using warps and BTFs on the set of images of the same persons in different cameras for the CAVIAR4REID dataset [26]. As shown, the distance is smaller between the features of the same persons if a nonlinear warp function is used compared to BTF or raw (untransformed) features.

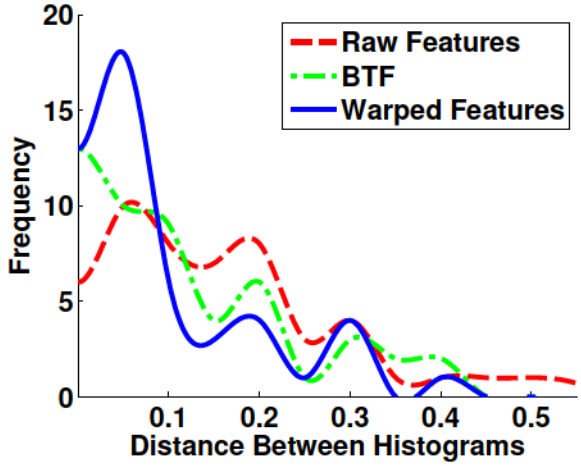
To summarize, in this chapter, the feature transformation is captured by computing a nonlinear mapping (warp) that minimizes a cost function defined as the mismatch between histogram features. A WFS composed of the collection of feasible and infeasible warp functions is built. Onece the WFS is built, a discriminating surface between the sets of feasible and infeasible warps in learnt using a random forest classifier. The re-identification problem is addressed by mapping a test warp function onto the WFS and classifying it as belonging to either the set of feasible or infeasible warps (see Figure 5.2).

The performance of the proposed approach are compared to state-of-the-art person re-identification methods using four publicly available benchmark datasets and a newly collected dataset named RAiD (Re-identification Across indoor-outdoor Dataset) [99]. The new dataset is collected with particular focus on large illumination variation between cameras. Our average performance on different combinations of multiple datasets is higher than other state-of-the-art methods.

The rest of the paper is organized as follows. An overview of the proposed approach is given in section 5.2. The details about the re-identification approach, as feature extraction, warping and WFS are described in section 5.3. Experimental results and comparisons with state-of-the-art methods are shown in section 5.4. Finally, conclusions are drawn in section 5.5.



(a) (b) (c) (d)



(e)

Figure 5.1: Warp functions capture inter camera feature transformations. (a) and (b) show the value feature histograms of the same person viewed in 2 different cameras. In (c) the histogram in (a) is warped to the histogram in (b). The same process is applied in (d) using BTF [70]. Figure (e) shows the distribution of the Bhattacharyya distances between the original value histograms in the second camera and the transformed value histograms using BTF (in green) and warp functions (in blue) computed for all the 50 persons in the CAVIAR4REID dataset. The distribution of the distances computed between the raw value histograms is also shown for comparison (in red).

5.2 Overview of proposed approach

The overall scheme of the proposed person re-identification process is shown in Figure 5.3. Given the frames from two cameras we learn a discriminative model in the WFS to get the probability of a sample feature warp function coming from the same person or not.

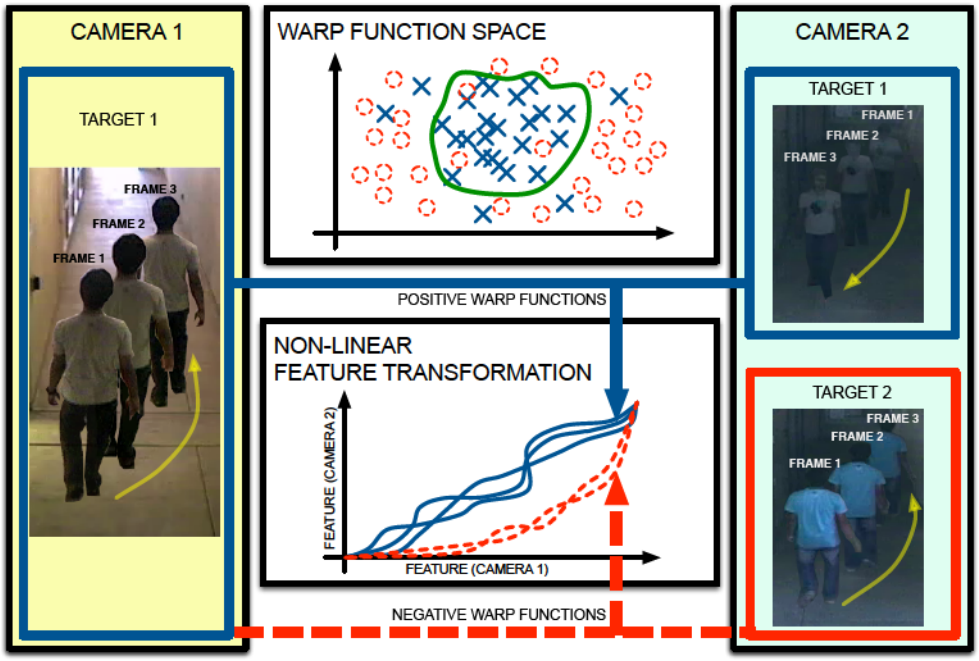


Figure 5.2: Re-identification by discriminating in the warp function space. The warp functions computed between features extracted from images of the same target (i.e. positive warp functions) are shown in solid blue. The warp functions computed between features extracted from different targets (i.e. negative warp functions) are shown in dashed red. A nonlinear decision surface (shown in green) is learnt to separate the two regions.

Towards this objective we first extract features from the person images. The feature extraction module performs the following tasks: a) splitting the image of the detected persons into four main body parts, and b) extracting dense color and texture features from the detected body parts.

For each extracted feature, vector valued warp functions are computed by the warp function space module. All the warp functions (corresponding to different features) are concatenated to form a high dimensional warp function for each image pair. The warp function between the same target in different cameras is denoted as a feasible or positive warp function while the warp function between two different targets is denoted as an infeasible or a negative warp function. The set of all feasible and infeasible warp functions forms the WFS. The dimensionality of the WFS is reduced using Principal Component Analysis (PCA) [62].

Given the WFS, a decision surface discriminating the two sets of warp functions is learnt using a Random Forest (RF) [22] of bagged decision trees. Every component of the warp functions may not be discriminating enough between the two classes of

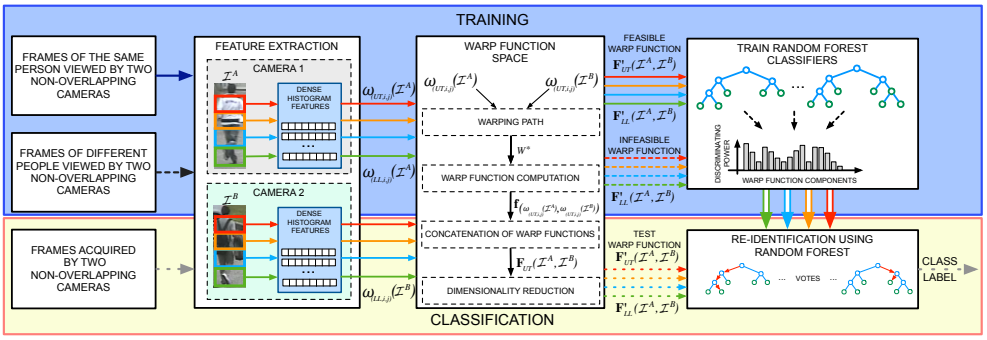


Figure 5.3: System Overview. The feature extraction module takes raw video frames and extracts dense color and texture features from each of the four detected body parts. These are input to the warp function space module that computes the warp function between each of them and reduces the dimensionality of the warp function space. A random forest classifier is trained to discriminate between the feasible and the infeasible warp functions in the WFS. The trained classifier is used to classify the test warp functions.

transformations (feasible/inefasible). The decision trees select the subset of warp function components according to their importance and maximize the discrimination between the feasible and infeasible warp functions in the WFS.

For classification, features are extracted from test image pairs and input to the WFS module to compute the warp functions. Finally, the RF classifies the test warp functions in the WFS as feasible or infeasible.

5.3 Methodology

In this section we describe the different modules of the proposed approach in details.

5.3.1 Feature extraction

The task of re-identifying targets across camera pairs is challenging because of the issues of pose variation, illumination and color changes. State-of-the-art methods for person re-identification have successfully explored different appearance features [85] to tackle these challenges. While existing feature transformation based methods are designed for color features, our framework can be used to study the nature of transformation of any feature which, in turn, can be used for re-identification. In this work we focus not only on color features but also on popular texture features like Local Binary Patterns (LBP), Gabor, Schimid and Leung-Malik features.

Before computing these features, we identify the salient regions like head $\mathcal{I}_H$, torso $\mathcal{I}_T$ and legs $\mathcal{I}_L$ from the given image $\mathcal{I}$ as proposed in [13]. In our approach we only consider $\mathcal{I}_T$ and $\mathcal{I}_L$ since the head region $\mathcal{I}_H$ often consists of a few and

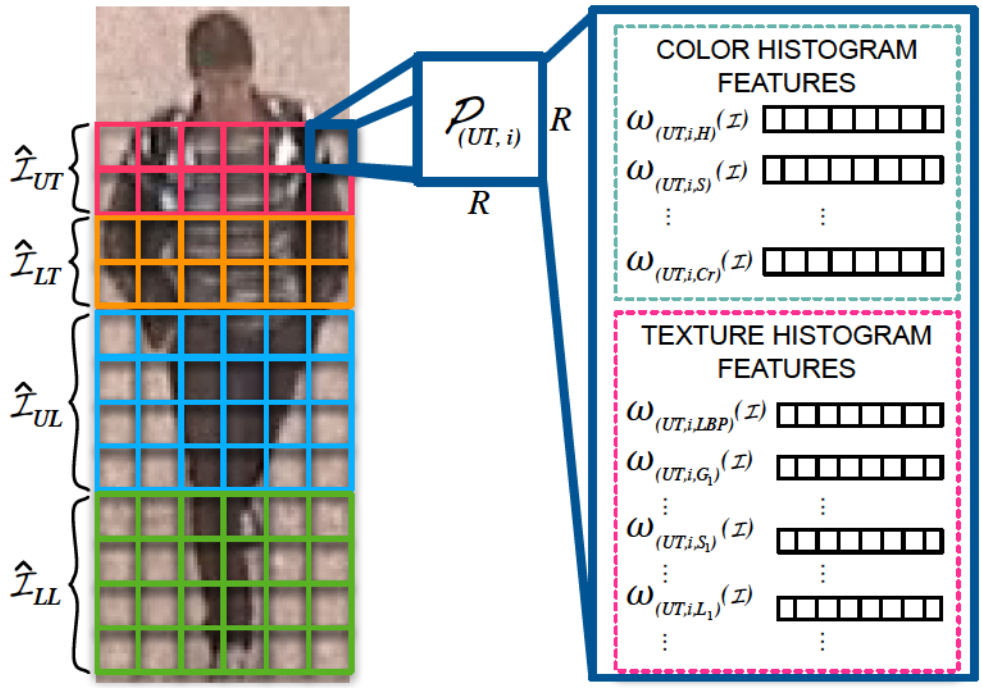
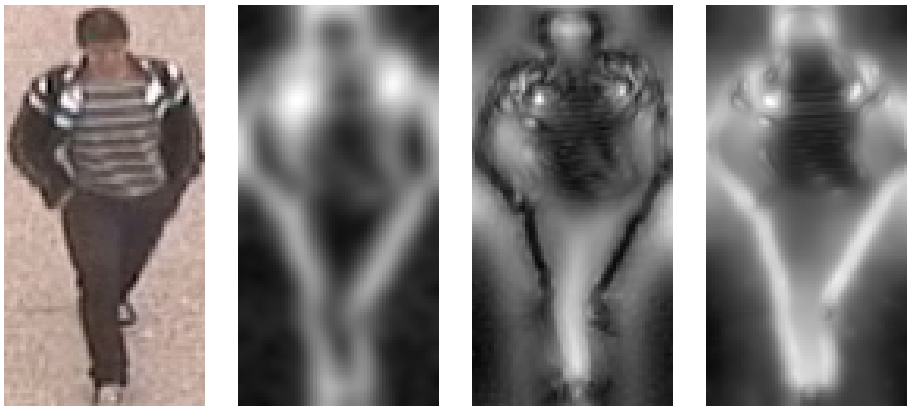


Figure 5.4: Dense image features from the detected body parts. Dense color and texture histogram features are extracted from each of the 4 resized body parts.

less informative pixels. This is especially true for low resolution images typically encountered in surveillance scenarios. We additionally divide both $\mathcal{I}_T$ and $\mathcal{I}_L$ into two horizontal sub-regions based on the intuition that people can wear shorts or long pants and short or long sleeves tops. The four different regions are resized to fixed height and width to extract fixed size dense features from all of them. We denote these resized regions as $\hat{\mathcal{I}}_\phi$ where $\phi \in \{UT, LT, UL, LL\}$ denotes the upper-torso, lower-torso, upper-legs and lower-legs region respectively. The resized regions are further divided into non overlapping patches $\mathcal{P}_{(\phi,1)}, \mathcal{P}_{(\phi,2)}, \cdots \mathcal{P}_{(\phi,n_\phi)}$ of size $R \times R$ each, where n_ϕ denotes the number of patches corresponding to the body part ϕ. Then, for all the patches $\mathcal{P}_{(\phi,i)}$, $i = 1, \cdots, n_\phi$ we extract the following features.

Color: Color histogram features are the most widely used appearance features to represent a person's appearance. State-of-the-art person re-identification methods use color features relying on the assumption that persons do not change their clothes as they move between camera FoVs. According to that, and following the considerations on appearance features suggested in [85], we extract multiple dense color histogram features exploiting the HSV, CIELab, RGB and YCbCr color spaces. Let $c \in \{H, S, V, L, a^*, b^*, R, G, B, Y, Cb, Cr\}$ denotes the color space component. For image $\mathcal{I}$, bodypart ϕ and patch i we extract the histogram $\omega_{(\phi,i,c)}(\mathcal{I}) \in \mathbb{R}^{b_c}$, where b_c



(a) (b) (c) (d)

Figure 5.5: Response images after convolutions with the Gabor, Schmid and Leung-Malik filter banks. All filter responses are sum and scaled for visualization. (a) Input image. (b) Response after convolution of 40 Gabor filters. (c) Response after convolution of 13 Schmid filters. (d) Response after convolution of 48 Leung-Malik filters.

is the number of bins of the feature histogram for colorspace c.

Texture: Similar to the color features, we extract dense texture features to capture the appearance of a person. To deal with object scale and rotation variations, we consider texture features that are invariant with respect to these. Before computing the texture features, the input image is converted to gray-scale.

We use LBP texture feature which is computationally efficient and is robust to both gray-scale variations [58] and rotation [109]. The extracted LBP texture histogram is denoted as $\omega_{(\phi,i,LBP)}(\mathcal{I}) \in \mathbb{R}^{b_{LBP}}$, where b_{LBP} is the number of bins used to quantize the resulting LBP histogram. We also use filter banks to extract texture features. A bank of Gabor filters with different sizes and orientations have been used. After convolving the i-th patch with a single filter we compute the modulus of the response and quantize it in a histogram of b_{Gabor} bins. We denote each resulting histogram as $\omega_{(\phi,i,g)}(\mathcal{I}) \in \mathbb{R}^{b_{Gabor}}$, where $g \in G = \{G_1, G_2, \cdots, G_{N_G}\}$, the set of the N_G Gabor filters. Similarly we use the Schmid filters [127] to compute $\omega_{(\phi,i,s)}(\mathcal{I}) \in \mathbb{R}^{b_{Schmid}}$, where $s \in S = \{S_1, S_2, \cdots, S_{N_S}\}$, the set of N_S Schmid filters. Finally we convolve each image with the Leung-Malik (LM) [81] filter bank $LM = \{L_1, L_2, \cdots, L_{N_{LM}}\}$ composed of N_{LM} filters. After convolving the image with the l-th filter ($l \in LM$) we quantize the response in a histogram $\omega_{(\phi,i,l)}(\mathcal{I}) \in \mathbb{R}^{b_{LM}}$. An example of the responses of the different filter banks is shown in Figure 5.5.

The set of features extracted from patch $\mathcal{P}_{(\phi,i)}$ is given by the set $\{\omega_{(\phi,i,j)}(\mathcal{I})\}$ where $j \in \{c \cup LBP \cup G \cup S \cup LM\}$.

5.3.2 Warp function space

To capture the transformation of the extracted features between cameras we use the principles of Dynamic Time Warping (DTW). DTW [126] has been widely used in many fields such as speech recognition [72], data mining [75], bioinformatics [1], fingerprint verification [77], activity recognition [138, 139], event detection [112], etc. DTW is a dynamic programming algorithm that optimizes the alignment of two time series by non-linearly warping the series so that the sum of the point-to-point distances is minimized. Time sequences are functions of time while feature histograms are functions of the bin numbers. In our approach the bin number axis is warped to reduce the mismatch between feature values of two feature histograms from two cameras.

Let $\mathbf{x}(1, \cdots, m) = \langle x(1), \cdots, x(m) \rangle$ and $\mathbf{y}(1, \cdots, m) = \langle y(1), \cdots, y(m) \rangle$ be two vector valued functions. Let f be a warp function from $\mathbf{x}$ to $\mathbf{y}$, that is

$$y(a) = x(f(a)), f(a) : [1, m] \rightarrow [1, m] \in \mathcal{F} \tag{5.1}$$

where $\mathcal{F}$ is the space of all warp functions, the WFS.

To find the warp function, a cost matrix $C \in \mathbb{R}^{m \times m}$ is generated where the $(a, b)^{th}$ element (denoted as C_{ab}) of the matrix is given by the distance $\delta(x(a), y(b)), \forall a, b \in \{1, 2, \cdots, m\}$. Though any suitable distance function can be used or learned using a metric learning procedure, in general, the magnitude of the difference and the Euclidean distance between elements are adopted due to their simplicity [15]. The warp function is the path giving the lowest cumulative cost between fixed start point, the $(1, 1)^{th}$ cell and fixed end point, the $(m, m)^{th}$ cell of C. Let $\mathbb{W} = \{W_1, W_2, \cdots\}$ be the set of all possible paths between these two fixed points where W_q denotes the q^{th} path. W_q consists of tuples indicating the indices of the cells in C. Then the optimal warp path is given by,

$$W^* = \underset{W_q \in \mathbb{W}}{\operatorname{argmin}} \left(\sum_{(a,b) \in W_q} C_{ab} \right) \tag{5.2}$$

The optimization problem in (5.2) is solved in a dynamic programming framework under suitable monotonicity and continuity constraints [15]. Finding the non-linear warp path W^* does not guarantee that the length of the warp path is same for all feature pairs $\mathbf{x}$ and $\mathbf{y}$. This is due to the fact that the mapping $f(a) : \{1, 2, \cdots m\} \rightarrow \{1, 2, \cdots m\}$, described by the tuples in W^* is, in general, many to many. To get a m length warp function we employ the following rule for all $(a, b) \in W^*$

$$f(a) = \begin{cases} \min(b) & \text{if } a \neq 1, m \\ a & \text{otherwise} \end{cases} \tag{5.3}$$

Gathering the $f(a)$'s for all $a = 1, 2, \cdots, m$ in a vector $\mathbf{f}_{(\mathbf{x},\mathbf{y})}(1, \cdots, m) = \langle f(1), \cdots, f(m) \rangle$ we get the warp function that warps $\mathbf{x}$ to $\mathbf{y}$.

In our approach the warp function $\mathbf{f}$ is computed for each feature and for every dense patch (see Section 5.3.1). In other words, as shown in Figure 5.6, $\mathbf{f}$ is computed

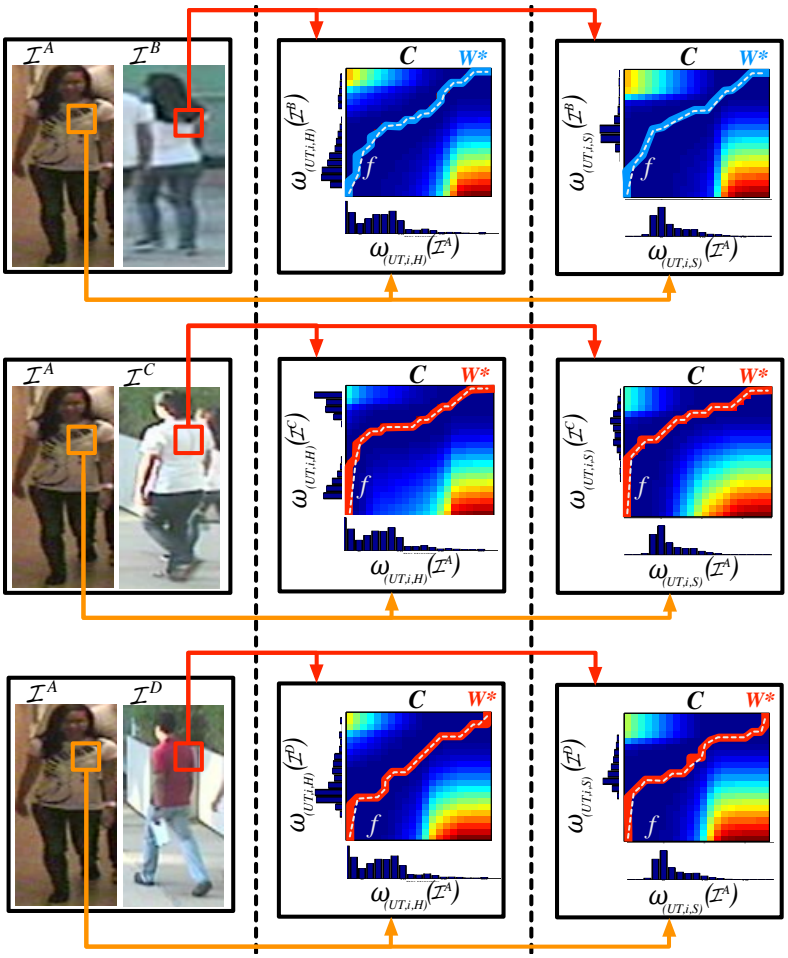


Figure 5.6: Example of computing the warp functions between features extracted from the same patch of two images. The first column shows two images from two cameras. The warp function between the features extracted from the same patches (shown by the orange and red boxes) are computed next. The last two columns show the cost matrices, the optimal warp path W^* and the corresponding warp function f. For convenience of visualization, warp functions computed for the H and S colorspaces only are shown in second and third column respectively. The cost matrix is colorcoded and the cost gets higher as the color goes from blue to red. First row shows the feature warps for the same person. Second and third rows show the warping of features between different persons that have similar and different appearance respectively with the person in the left.

for feature pairs ($\omega_{(\phi,i,j)}(\mathcal{I}^A)$ and $\omega_{(\phi,i,j)}(\mathcal{I}^B)$) for each body part ϕ, patch i and feature j. The vector created by concatenating all such vector warp functions computed for the body part ϕ, is denoted as

$$\mathbf{F}_\phi(\mathcal{I}^A,\mathcal{I}^B) = \left\langle \mathbf{f}_{(\omega_{(\phi,i,j)}(\mathcal{I}^A),\omega_{(\phi,i,j)}(\mathcal{I}^B))} \right\rangle, \qquad \forall i,j \tag{5.4}$$

The set of all $\mathbf{F}_\phi(\mathcal{I}^A,\mathcal{I}^B)$'s computed between two images $\mathcal{I}^A$ and $\mathcal{I}^B$ of the same person forms the feasible or positive set $\mathcal{F}^p_\phi$ (for bodypart ϕ). The same computed between images of two different persons forms the infeasible or negative set $\mathcal{F}^n_\phi$. Both $\mathcal{F}^p_\phi$ and $\mathcal{F}^n_\phi$ together form the WFS which provides the description of the nonlinear feature transformations under different variabilities.

The proposed WFS model allows us to pose the re-identification problem as finding the parameters of the decision surface, that best separates the sets $\mathcal{F}^p_\phi$ and $\mathcal{F}^n_\phi$. Given a pair of candidate images, we classify such images as coming from the same target or not according as the warp functions between the image features lie in the positive or the negative region.

5.3.3 Re-identification in WFS

To re-identify persons moving across camera views we propose to train a binary classifier and classify the warp functions in the WFS as belonging to the feasible or infeasible sets. As discussed in section 5.3.2 we use high-dimensional dense color and texture features to represent the appearance of the targets. While it is advantageous for a richer representation, it comes with the curse of dimensionality. The high dimensionality of the features result in high dimensional warp functions. Accordingly, any nonlinear classifier has to pay high computational and memory complexity in the training phase. This scalability issue makes it nontrivial to train a classifier directly on such high dimensional warp functions for large datasets whose training size is typically far beyond thousands. Therefore, we need to select a low dimensional subspace that can adequately handle the intrinsic dimensionality of the warp functions. Towards this objective, and supported by the recent study on real data discussed in [90], we use PCA [62] to embed the WFS into a low dimensional subspace. In the following we refer to $\mathbf{F}'_\phi(\mathcal{I}^A,\mathcal{I}^B)$ as the low dimensional warp function computed between images $\mathcal{I}^A$ and $\mathcal{I}^B$ for body part ϕ.

Even though PCA is able to reduce the dimensionality of the WFS, each dimension of it may not, still, be discriminating enough between the feasible and infeasible warp functions. Thus a classifier giving more importance to the more discriminative dimension is desirable. A random forest (RF) [22] is a popular and efficient classifier based on bootstrapped aggregation ideas. It is a combination of many binary decision trees built using several bootstrap samples. At each node of each tree a subset of the warp function dimensions is randomly chosen and the best split is calculated only within this subset. This randomization of the warp function dimensions effectively chooses the dimensions according to their importance in separating the feasible and the infeasible warp functions in the WFS. This coupled with the reduction of

overfitting error makes RF a suitable choice to learn the parameters of the decision boundary.

In the classification phase the warp function between the features of two candidate images from two different cameras is computed. The trained RF classifies the warp function as coming from the same target or not according as it lies in the positive or the negative region.

Let $\mathcal{I}^{A_1}, \cdots, \mathcal{I}^{A_N}$ be the N images of a given person $\mathcal{A}$ and $\mathcal{I}^{B_1}, \cdots, \mathcal{I}^{B_M}$ be the M images of another person $\mathcal{B}$ in another camera. As commonly accepted in the field of person re-identification, if N=1 and M=1, then the approach is defined to be a *single-shot* approach, otherwise, if both N and M are greater than 1, it is named a *multiple-shot* approach. As the total number of possible warp functions that can be computed for a single body part ϕ is $N \times M$, we have $|\phi| \times N \times M$ predicted probabilities for a target pair, where $|\phi|$ denotes the number of parts into which the body of a person is divided. The probability of $\mathcal{A}$ and $\mathcal{B}$ being the same person is computed by averaging all the $|\phi| \times N \times M$ probabilities obtained from the classifier.

5.4 Experiments

We evaluated our approach on four publicly available datasets, the ETHZ dataset [40], the CAVIAR4REID dataset [26], the WARD dataset [92], the VIPeR dataset [54] and one new dataset (RAiD), introduced in this work. We have choses these datasets because they provide many challenges faced in real world person re-identification applications, e.g., viewpoint, pose and illumination changes, different backgrounds, image resolutions, occlusions, etc. Of these, WARD and RAiD are specifically geared towards large illumination change. More details about each dataset are reported in section 2.2.6. We report the results for both single-shot ($N = 1$) and multiple-shot ($N > 1$) strategies. For all multiple-shot strategies we use $N = M$. Results are shown in terms of recognition rate as Cumulative Matching Characteristic (CMC) curves and normalized Area Under Curve (nAUC) values, as commonly performed in the literature. For each dataset the evaluation procedure is repeated 10 times using independent random splits. We reported the average results on these 10 splits.

5.4.1 Implementation Details

In our implementation we used the following settings:

- Image pairs of the same or different person(s) in different cameras were randomly picked to compute the positive and negative warp functions (samples) respectively;
- $\hat{\mathcal{I}}_{UT}$, $\hat{\mathcal{I}}_{LT}$, $\hat{\mathcal{I}}_{UL}$ and $\hat{\mathcal{I}}_{LL}$ have been resized as follows:
  - For the ETHZ dataset: $\hat{\mathcal{I}}_{UT} = \hat{\mathcal{I}}_{LT} = \hat{\mathcal{I}}_{UL} = \hat{\mathcal{I}}_{LL} = 32 \times 16$;
  - For the CAVIAR, WARD and RAiD dataset: $\hat{\mathcal{I}}_{UT} = \hat{\mathcal{I}}_{LT} = \hat{\mathcal{I}}_{UL} = \hat{\mathcal{I}}_{LL} = 64 \times 32$

  - For the VIPeR dataset: $\hat{\mathcal{I}}_{UT} = \hat{\mathcal{I}}_{LT} = \hat{\mathcal{I}}_{UL} = \hat{\mathcal{I}}_{LL} = 48 \times 32$;

- The size of each dense patch has been selected to be $R \times R = 8 \times 8$ pixels.
- The color histograms extracted from the dense patches were quantized using $b_c = 10$ bins for each color space component c.
- Texture features have been extracted using the following parameters:
  - LBP: we followed the same protocols used in [109]. LBP histograms were quantized into $b_{LBP} = 10$ bins.
  - Gabor: we used Gabor filters at 8 orientations and 5 scales. b_{Gabor} was set to 16.
  - Schmid: the same filter settings as [127] have been used. b_{Schimd} was set to 16.
  - Leung-Malik: the same filter bank defined in [81] consisting of 36 oriented filters with 6 orientations, 3 scales and 2 phases, 8 Laplacian of Gaussian (LoG) filters, and 4 Gaussians was used. b_{LM} was set to 16.
- δ was taken as the Euclidean distance between the feature values.
- While doing PCA, we selected the largest principal components such that the 99% of the original variance is retained.
- The RF parameters such as the number of trees, the number of features to consider when looking for the best split, etc. were selected using 4-fold cross validation.

The proposed method is, first, evaluated on 3 challenging benchmark datasets, namely ETHZ, CAVIAR4REID and VIPeR. Since WARD and RAiD contain a large illumination variation, we show the performance on these two datasets separately in the next sub-section.

5.4.2 Comparative Evaluation on Benchmark Datasets

ETHZ Dataset

To make this dataset more challenging, we followed the strategy proposed in [8] by randomly picking a set of 10 consecutive frames from the beginning and from the end of each sequence. Following the evaluation setup in [128, 13], all images have been resized to 32×64 pixels. We evaluate our method using both single-shot and multiple-shot strategies. Similar to [60, 59], for the single-shot scenario, we randomly sample two images per person to build a training set, and another two images to build the test set. The test images from one camera constitute the probe and the those from the other camera create the gallery set.

In Table 5.1 we present the performance of our method using both single-shot and multiple-shot strategies. The first 9 rows show the performance comparison with 8

Table 5.1: Comparison of the proposed method on the ETHZ dataset using both a single shot-strategy (top 9 rows) and a multiple-shot strategy (last 10 rows). Recognition rates for top 7 ranks are shown for each of the three sequences. The best recognition rates for each rank are shown in boldface font

Method	SEQ.#1							SEQ.#2							SEQ.#3						
	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
Proposed (1 image)	**84**	**88**	**91**	**93**	**94**	**95**	**96**	**81**	**86**	**90**	**93**	**95**	**96**	**97**	**91**	**97**	**99**	**99**	**99**	**99**	**100**
eLDFV [89](1 image)	83	87	90	91	92	93	94	79	84	87	90	91	92	93	**91**	94	96	97	97	97	97
SDALF [13](1 image)	65	73	77	79	81	82	84	64	74	79	83	85	87	89	76	83	86	88	90	92	93
eBiCOV [88](1 image)	74	80	83	85	87	88	89	71	79	83	86	88	90	91	82	87	90	92	93	94	95
eSDC_knn [152](1 image)	81	86	89	90	92	93	94	79	84	87	90	91	92	93	90	95	96	97	98	98	99
eSDC_ocsvm [152](1 image)	80	85	88	90	91	92	93	80	**86**	89	91	93	94	95	89	94	96	97	98	98	99
RPLM [60](1 image)	77	83	87	90	91	92	92	65	77	81	82	86	89	90	83	90	92	94	96	96	97
IBML [59](1 image)	78	84	87	89	90	91	91	74	81	84	87	89	91	92	**91**	95	97	98	98	98	99
ICT [4](1 image)	68	76	82	86	87	89	90	70	82	89	91	93	94	95	**91**	94	96	97	97	98	98
Proposed (5 images)	94	95	96	97	**98**	**98**	**99**	**98**	**99**	**99**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
PLS [128](all images)	79	85	86	87	88	89	90	74	79	81	83	84	85	87	77	81	82	84	85	87	89
eBiCOV [88](5 images)	93	94	95	95	96	96	96	91	94	95	96	97	97	97	98	98	99	**100**	**100**	**100**	**100**
eLDFV [89](5 images)	**96**	**97**	**97**	97	**98**	**98**	98	97	98	**99**	**100**	**100**	**100**	100	99	**100**	**100**	**100**	**100**	**100**	**100**
LDC [151](5 images)	92	95	96	97	**98**	**98**	98	92	95	97	98	99	99	99	96	97	98	99	99	99	99
ICT [4](5 images)	92	93	94	95	96	96	97	95	98	**99**	99	**100**	**100**	**100**	95	96	97	99	**100**	**100**	**100**
SDALF [13](10 images)	91	92	93	94	94	94	94	91	94	96	96	97	97	98	94	95	96	96	96	96	96
AHPE [11] (10 images)	85	89	92	93	94	94	95	80	86	89	92	93	94	95	83	91	92	94	96	97	97
eBiCOV [88](10 images)	93	94	95	96	96	96	96	91	95	96	97	98	99	99	97	98	99	**100**	**100**	**100**	**100**
BRM [8](10 images)	**96**	**97**	**98**	**98**	**98**	**98**	98	94	95	95	95	95	96	96	96	98	**100**	**100**	**100**	**100**	**100**

different methods when 1 single image has been used to build the gallery and the probe sets. The last 10 rows show the performance comparison with 9 different methods using a multiple-shot strategy. For the single shot scenario our performance is either superior to or same with that of all the 8 methods for each of the 3 sequences. For the multiple-shot scenario the same settings of experiments as in [151, 88] were used with N=5. In this scenario, the BRM [8] approach has superior performances only from rank 1 to rank 4 for SEQ.#1 . Similarly the eLDFV [89] method has superior performance compared to our method for rank 1 to 3. Our method is the only one that achieves the 99% of correct recognition for this sequence within the top 7 rank scores. On SEQ.#2 we outperform all other methods as we reach 100% correct recognition within top 4 matches. Similarly, on SEQ.#3 our method has the best performance and recognizes all the persons at rank 1. Notice that in these experiments we are using N=5 images, whereas the results for SDALF, AHPE, eBiCov and BRM were reported using N=10 images. For all the three sequences in the ETHZ dataset our method is the only one that achieves the 99% of correct recognition within the top 7 matches.

CAVIAR4REID Dataset

It is common to split the CAVIAR4REID dataset both in terms of people [4, 110] and not [82, 13]. We conducted experiments following both these protocols to fairly compare against methods following either of these two. Following the same setup as in [4] first, the 50 people are equally divided into training and test sets of 25 persons each. In this setup we compare against LF [110] and ICT [4] who use a multishot strategy with N=5 and N=10 images respectively. In Figure 5.7(a) we show that our algorithm outperforms both the methods and reaches as high as 40.9% rank 1 score when a multishot strategy with N=10 is employed. In the second set up following the same protocol as in [82], we do not split the dataset in terms of persons. Pairs of images are randomly selected in different views for training. The probe and the gallery sets are formed by randomly selecting images from the remaining ones for each person. In this scenario we compare against the methods who have adopted the same strategy of split. Namely the methods are AHPE [11], SDALF [13], CI [78], CPS [26], LAFT [82] and LDC [151]. Figure 5.7(b) shows the CMC curves for the single shot scenario. Figure 5.7(c) and (d) show the comparison with the multi-shot strategy. While for single shot scenario we meet the state-of-the-art performance of LAFT and outperform the rest, for both the multishot scenarios we have superior performance over all the compared methods.

VIPeR Dataset

Although images from the same camera are not always taken from the same viewpoint and thus do not fully fit our framework, still we compare our results with other methods to show that the proposed approach achieves good results in such a scenario too. To evaluate our method we followed the same normalization approach as in [13, 4, 152], resizing all the images to 48×128 pixels. To compare our approach to state-

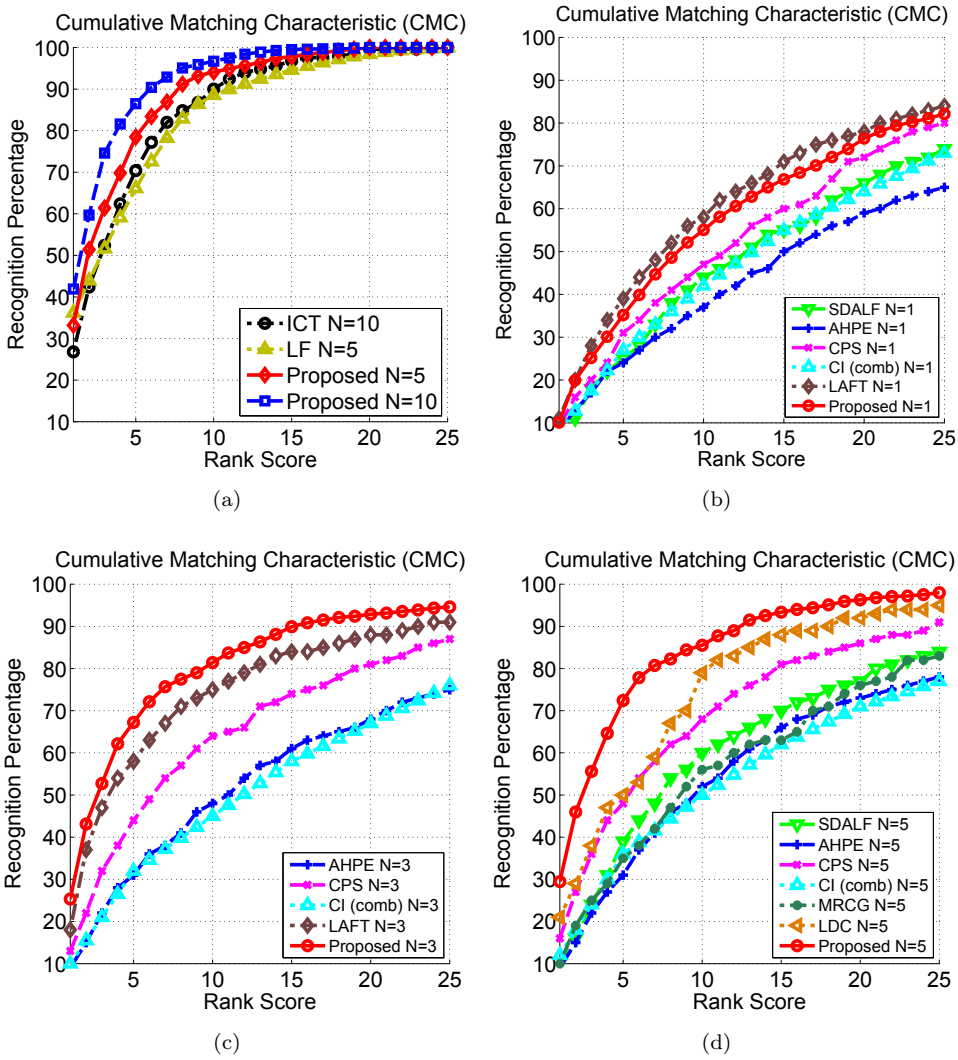


Figure 5.7: CMC curves for CAVIAR4REID dataset. In (a) results are shown when the dataset is split in terms of persons. In (b), (c) and (d) comparisons are shown for the case where the dataset is not split in terms of persons with N=1, N=3 and N=5 respectively.

of-the-art methods we used the same evaluation protocol proposed in [53]. We split the dataset in terms of persons and used 316 of them for training and the remaining 316 for testing. As the VIPeR dataset is a single-shot dataset, we used N=1 images per person to form the training and test sets.

Table 5.2: Comparison of the proposed method on the VIPeR dataset. Top 100 rank matching rate (percent) is shown.

Rank Score	1	10	20	50	100
Proposed	25.81	69.56	83.67	95.12	98.89
RCCA [3]	**30.00**	**75.00**	**87.00**	96.00	**99.00**
LAFT [82]	29.60	69.30	81.34	**96.80**	**99.00**
LF [110]	24.18	67.12	81.38	94.12	
TML [83]	19.00	61.00	74.00	91.00	97.00
KISSME [76]	19.60	62.20	74.92	91.80	98.00
RPLM [60]	27.00	69.00	83.00	95.00	**99.00**
IBML [59]	22.00	63.00	78.00	93.00	98.00
ELF [53]	12.00	43.00	60.00	81.00	93.00
SDALF [13]	19.87	49.73	65.73	84.80	
PRSVM [119]	14.60	53.90	70.10	85.00	94.00
CPS [26]	21.84	57.21	71.00	88.10	
PRDC [154]	15.70	53.86	70.09	87.00	
LMNN-R [36]	23.70	68.00	80.00	93.00	**99.00**
eBiCOV [88]	20.66	56.18	68.00	84.90	
eLDFV [89]	22.34	60.04	71.00	88.92	**99.00**
eSDC.knn [152]	26.31	58.86	72.77	79.30	
eSDC.ocsvm [152]	26.74	62.37	76.36	82.10	
CI [78]	18.00	50.00	62.00	81.00	
ICT [4]	15.90	57.20	78.30	91.00	95.00
ARLTM [87]	21.00	52.00	68.00	86.00	

In Table 5.2 we report the recognition performance for the top 100 ranks and compared the results with 20 state-of-the-art methods for person re-identification. The table shows that the proposed method does achieve a performance better than most of the state-of-the-arts as far as the performance corresponding to rank 1 is considered. It is behind the top performer only by 4.19% for rank 1. The performance continuously improves with higher ranks. The rank 100 performance is either the same or better than all the methods. According to [4] the performance at higher ranks is, sometimes, more significant as this reflects the algorithm's performance for difficult cases. Thus, in this challenging dataset with only one image per person in two non-static cameras the proposed method does achieve competitive performance as that of the state-of-the-arts.

5.4.3 Comparative Evaluation with Large Appearance Variation

Since our focus is to understand the space of transformation of features, we provide the performance of the proposed method for 2 datasets which posses significant appearance variation.

WARD Dataset

We conducted the experiments for all the three different camera pairs, denoted here as camera pairs 1-2, 1-3, and 2-3. The proposed approach is compared with the methods for which either the CMC performance on this dataset is reported in literature or the code is available. Namely the methods are SDALF [13], WACN [92] and ICT [4]. Figure 5.8(a), (b) and (c) compare the performance adopting a multishot strategy with N=10 for camera pairs 1-2, 1-3, and 2-3, respectively. The 70 people in this dataset are equally divided into training and test sets of 35 persons each. For all 3 camera pairs the proposed method outperforms the rest with rank 1 recognition percentage as high as 51.6% for the camera pair 2-3. The next runner up has the recognition percentage of 29.5% for rank 1. For all the camera pairs 97% recognition performance is reached within top 10 matches.

RAiD Dataset

We collected this new dataset with large illumination variation that is not present in most of the publicly available benchmark datasets. To make sure there is enough variation of appearance between cameras, subjects were asked to walk through 3 cameras of which 2 are outdoor and 1 is indoor. We name the dataset as Re-identification Across indoor-outdoor Dataset (RAiD) [99]. 6060 images of 43 persons walking through 1 indoor (denoted as camera 1) and 2 outdoor cameras (denoted as camera 16 and camera 22) are collected. Sample images showing the variation of illumination between the cameras are shown in Figure 5.9.

The proposed approach is compared with the methods for which the code is available. Namely the methods are SDALF [13], WACN [92] and ICT [4]. The dataset

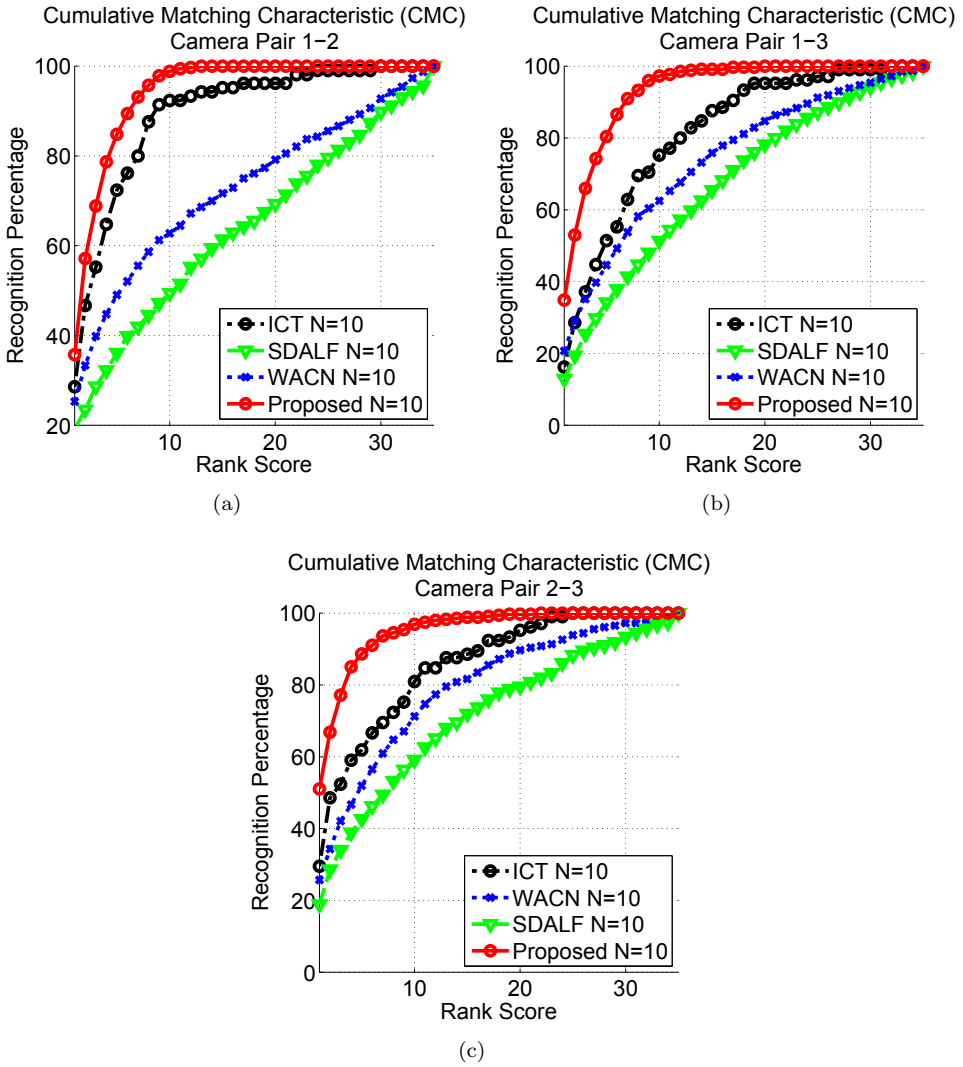


Figure 5.8: CMC curves for the WARD dataset. Results and comparisons in (a), (b) and (c) are shown for the camera pairs 1-2, 1-3, and 2-3 respectively. All the results are reported for the case where the dataset is split in terms of persons with N=10.

was split in terms of persons with 22 persons forming the training set and the rest 21 persons forming the test set. Figure 5.10(a), (b) and (c) compare the performance adopting a multishot strategy with N=10 for camera pairs 16-22, 1-16 and 1-22 respectively. We see that the proposed method is superior to all the rest for both the cases when there is not much appearance variation (camera pair 16-22) and when

Figure 5.9: Sample images of persons from the RAiD dataset showing the variation of appearance between the indoor and the outdoor cameras.

there is significant lighting variation (for camera pairs 1-16 and 1-22). Expectedly, for camera pair 16-22 the performance is the best achieving 55.7% rank 1 performance. For the other two difficult cases too, the proposed method is superior to all the rest achieving 46.4% and 53.9% rank 1 performances for camera pairs 1-16 and 1-22 respectively. The second best performance is that of ICT which achieves 29.5% and 37.3% rank 1 performances for camera pairs 1-16 and 1-22 respectively. Figure 5.11 shows a comparison of re-identification performances with ICT [4] (achieving the next best performance). The comparison is done on 10 randomly selected persons. For viewing convenience only the top 15 candidates are shown. The green bounding box highlights the ground truth match for each of the query persons. The ground truth match is within top 3 ranked matches for 9 out of the 10 examples while 6 out of these 10 persons are the highest ranked matches too. For the same set of persons the ground truth match is within top 3 ranked matches for 2 out of the 10 examples in ICT. None of them is the highest ranked match.

5.4.4 Average Performance across Multiple Datasets

Having shown the performance of the proposed method on separate datasets with different challenges, in this sub-section we show that the proposed method gives the most consistent performance across different datasets each having multiple different challenges. The performance is measured in terms of average nAUC values across different combinations of the 4 publicly avialable benchmark datasets (ETHZ, WARD, CAVIAR4REID and VIPeR). We compare with 14 state-of-the-art methods for which either the code is available or results for at least 2 of these 4 datsets are reported. The nAUC values for different methods are either taken from the reported results or computed from the reported CMC curves. To make a fair comparison we consider

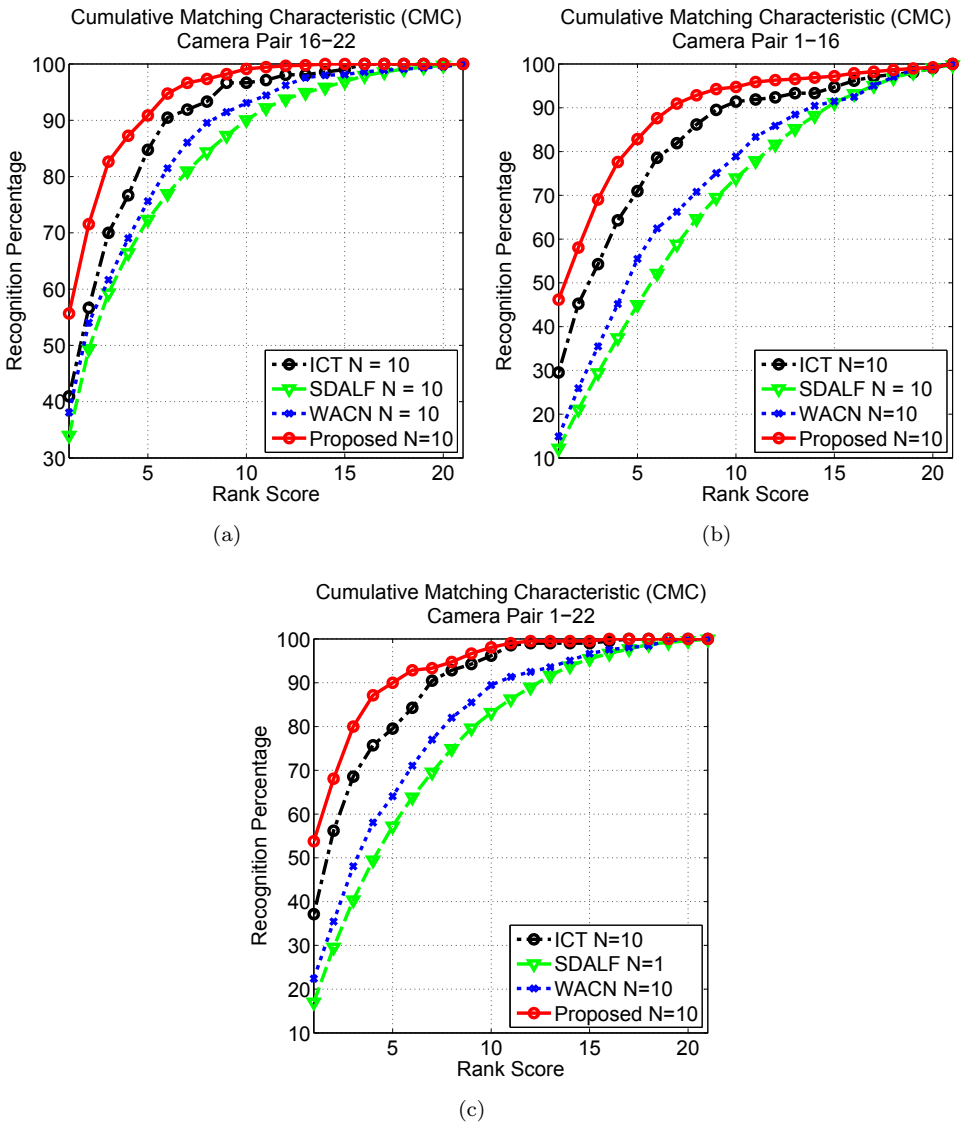


Figure 5.10: CMC curves for RAiD dataset. In (a), (b) and (c) comparisons are shown for the camera pairs 16-22, 1-16 and 1-22 respectively.

all combinations of 2 or more datsets and compare our performance by averaging over the datsets separately for each combination. Table 5.3 shows the performance comparison. The proposed method has the highest average nAUC value for 10 out of the 11 possible combinations. The only case (combination of ETHZ and CAVIAR)

Figure 5.11: Visual comparison of matches using feature warps for camera pair 1-16 of the RAiD dataset. First column is the probe image. Second and third columns show the top 15 matches computed using the proposed method and ICT [4] respectively.

where the proposed method is the runner up, the nAUC value changes only at the 3^{rd} decimal place. The superior performance of the proposed method on any combination of these datasets establishes the fact that the proposed method is not tuned to any specific dataset and can address varied number of challenges across different datasets better than the state-of-the-art.

5.5 Conclusions

In this work we have addressed the problem of multi-camera target re-identification by finding a nonlinear warp function between features from two cameras. Given a pair of feature vectors we have shown that we can learn the decision surface best separating the feasible and infeasible set of warp functions in the WFS. The target re-identification problem is posed as classifying a test warp function as belonging to the set of feasible or infeasible warp functions. We have shown that our approach is robust with respect to severe illumination and pose variations by evaluating the

Table 5.3: Comparison of average performance across different datasets

Number of datasets	4	3				2					
	ETHZ WARD CAVIAR VIPeR	ETHZ WARD CAVIAR	ETHZ WARD VIPeR	ETHZ CAVIAR VIPeR	WARD CAVIAR VIPeR	ETHZ WARD	ETHZ CAVIAR	ETHZ VIPeR	WARD CAVIAR	WARD VIPeR	CAVIAR VIPeR
Proposed	**0.9377**	**0.9292**	**0.9666**	**0.9352**	**0.9200**	**0.9683**	0.9211	**0.9772**	**0.8983**	**0.9544**	**0.9072**
ICT [4]	0.9115	0.8977	0.9302	0.9298	0.8883	0.9188	0.9182	0.9669	0.8561	0.9048	0.9042
SDALF [13]	0.8230	0.7898	0.8538	0.8697	0.7786	0.8195	0.8433	0.9393	0.7066	0.8026	0.8264
RPLM [60]	-	-	-	-	-	-	-	0.9566	-	-	-
IBML [59]	-	-	-	-	-	-	-	0.9549	-	-	-
CPS [26]	-	-	-	0.9062	-	-	0.8914	0.9674	-	-	0.8600
eBiCOV [88]	-	-	-	-	-	-	-	0.9394	-	-	-
eLDFV [89]	-	-	-	-	-	-	-	0.9622	-	-	-
eSDC.knn [152]	-	-	-	-	-	-	-	0.9335	-	-	-
eSDC.ocsvm [152]	-	-	-	-	-	-	-	0.9402	-	-	-
LDC [151]	-	-	-	-	-	-	**0.9250**	-	-	-	-
AHPE [11]	-	-	-	-	-	-	0.8245	-	-	-	-
LAFT [82]	-	-	-	-	-	-	-	-	-	-	0.8820
LF [110]	-	-	-	-	-	-	-	-	-	-	0.8980
CI (comb) [78]	-	-	-	-	-	-	-	-	-	-	0.7948

performance on five datasets. Our approach outperforms the existing state-of-the-art methods for person re-identification.

5.6 What next?

While the given results show that the proposed method is capable of dealing with complex transformation functions that occur between features computed by two disjoint cameras, the computational complexity of the algorithm, mainly guided by the very high dimensionality of the features, makes such an algorithm very computationally expansive. This is especially true for the RF training part. To address such challenge, in the next chapter, we build upon the idea that, as features get transformer across cameras, so are the distances between them.

6

New Directions: Feature Dissimilarities and Distributed Techniques

In this chapter, we extend the two main person re-identification ideas presented in the previous chapters. First, we study the nature of the transformation of feature distances. We introduce which features will be studied in such an approach, then we describe how those can be used to re-identify targets moving across disjoint camera FoVs. Results on benchmark datasets are provided at the end of the chapter. Then, we extend simple camera-to-camera re-identification approaches to a wide area camera network. Towards this end, we first extend the signature matching approach proposed in Chapter 4, then by using such information we propose the novel distributed re-identification framework.

6.1 Re-Identification by Classification of Feature Dissimilarities

In the previous chapter we've addressed the person re-identification problem by modeling the transformation of features across disjoint cameras. However, the representation we used in the classification framework was very high-dimensional, even it was reduced by using standard methods like PCA, thus introducing the problem of the curse of dimensionality. Motivated by the recent success of metric learning methods and feature transformation approaches we propose a person re-identification approach to address these challenges. The core novelty of this work is a method that aims to model not the way features are transformed across camera, but to advantage of invariant features and proper distance metrics to model how the distances between such features are transformed across cameras [96]. To achieve this goal we extract the feature vectors from a pair of targets viewed in different cameras, then we compute the distance between such features and use the distances to form the distance feature

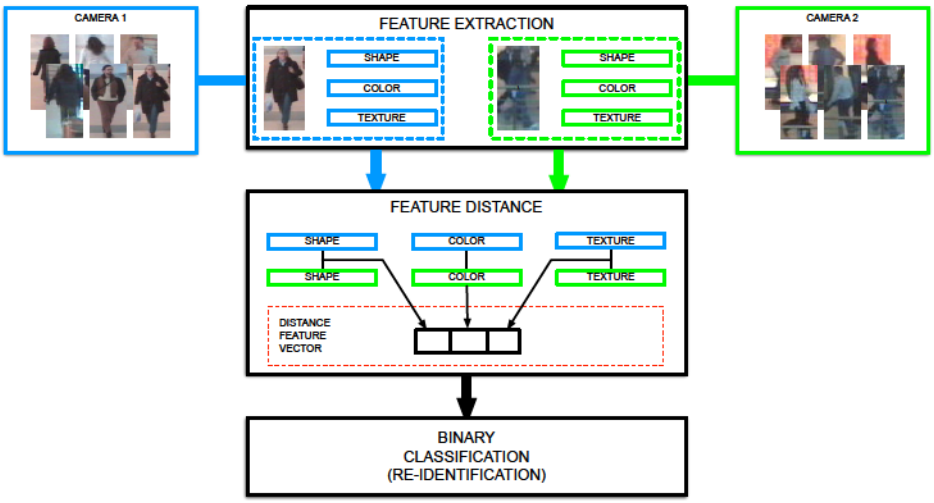


Figure 6.1: An overview of the proposed person re-identification approach. From each given image, we extract shape, color and texture features, then we compute the pairwise distances between the feature vectors extracted for targets viewed by different cameras. The computed distances form the DFV. The DFV from a pair of images of the same person is a positive sample, while the DFV from a pair of images of different persons is a negative sample. The DFVs are used to train a binary classifier. The trained classifier is used to re-identify targets by classifying test DFV.

vector (DFV). The DFVs from the same person form the set of positive samples, while the DFVs from different persons form the negative set. Using the positive and negative DFVs we re-identify persons in a supervised classification framework.

To validate the proposed method we compare the performance of our approach to state-of-the-art methods for person re-identification using two publicly available benchmark datasets.

6.1.1 The Approach

An overview of the proposed approach is shown in Figure 6.1. Given a pair of images from non-overlapping cameras, we extract multiple local and global features (Section 6.1.1) and we compute pairwise distances between them (Section 6.1.1). The computed distances form the DFV. To use this feature for classification we train a binary classifier to classify new examples.

Feature Extraction

Here we describe the feature extraction methods used to build a discriminative representation of the image of a person.

Motivation: The task of re-identifying targets across camera pairs is challenging because of the issues of pose variation, illumination and color changes. State-of-the-art methods for person re-identification have successfully explored different appearance features [85] to tackle these challenges. Inspired by that, to obtain a robust feature representation of an image across cameras, we considered, shape, color and texture features invariant to the stated issues.

Shape: To capture the shape of a given person we used the Pyramid Histogram of Oriented Gradients (PHOG) feature. The PHOG feature is computed exploiting the spatial pyramid technique. Let $l = 0, \cdots, L$ be the level of the spatial pyramid, and 4^l the number of cells in which the image is divided at each level l. The PHOG feature Φ is the concatenation of the HOG computed at the different levels and for different cells of the spatial pyramid. The final PHOG feature vector is of size $b \sum_{l=0}^{L} 4^l$, where b is the number of bins used to compute the HOG features.

Color: Color histogram features are the most widely used features to describe a person image. All state-of-the-art person re-identification methods use color features relying on the assumption that persons do not change their clothes as they move between camera Fields-of-view. According to that, we extract six different color histogram features from each given image. We consider that most of the persons wear different clothes for the upper and lower body part, so, before computing the color features we detect the three salient body parts (i.e., head, torso and legs) using a derivation of the approach proposed in [38]. We discard the head region from the feature computation since it generally contains few and not informative pixels. To achieve illumination invariant properties we equalize the histograms of the two regions and project them into the Lab color space. Then, we extract a histograms for each color channel c for both the torso and legs regions. The histograms for the two regions are denoted $\Upsilon_T \in \mathbb{R}^{n_c}$ and $\Upsilon_L \in \mathbb{R}^{n_c}$, respectively. We use different bin quantizations n_c, such that the lightness component of the color space has a coarse representation.

Texture: As for the color features, we use texture features to capture the appearance of a person. To deal with object scale and rotation variations, we consider texture features that have invariant properties with respect to these issues. We used a bank of Gabor filters with different sizes and orientations (see Figure 6.3(a)). After convolving each image with a single filter we computed the modulus of the response and we quantized it in a histogram with g bins. We denote the set of all such histograms as $\{\Gamma_i\}_{i=1}^{I}$, where i indicates the i^{th} Gabor filter. Similarly we used the Schmid filters (Figure 6.3(b)) to get the set of histograms $\{\Psi_j\}_{j=1}^{J}$, each of which has s bins. Finally we convolve each given image with the Leung-Malik (LM) filter bank consisting of first and second derivatives of Gaussians at 6 orientations and 3 scales, 8 Laplacian of Gaussian (LoG) filters, and 4 Gaussians (Figure 6.3(c)). After convolving the image with a single filter we quantized the response in a histogram with m bins. $\{\Lambda_k\}_{k=1}^{K}$ is the set of all such histograms, where k indicates the k^{th} LM filter. An example of the responses of the different filter banks is shown in Figure 6.4.

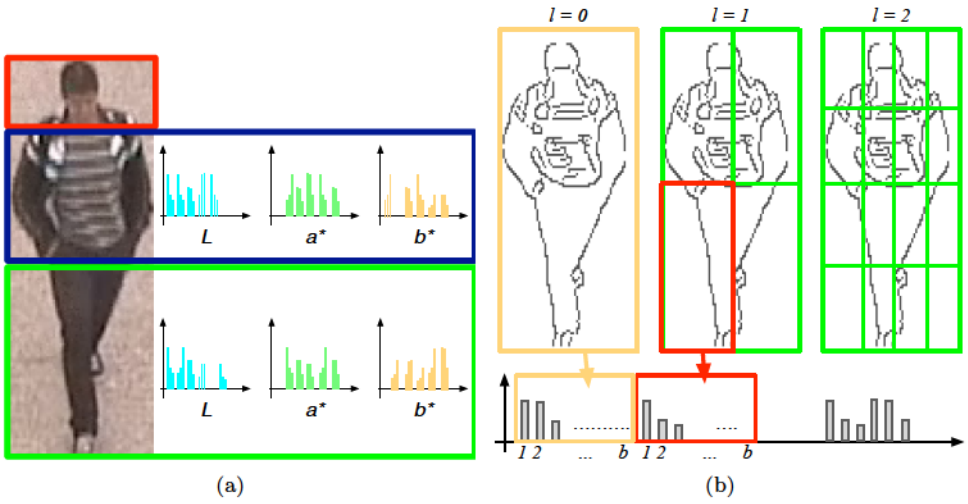


Figure 6.2: Color and shape features. (a) Color histogram features extracted from the torso and legs body parts. (b) PHOG features extracted from the whole body at three different levels of the spatial image pyramid ($L = 2$).

Distance Feature Vector

Intuition. In the image representation discussed in Section 6.1.1, we compute color, shape and texture features resulting in a very high-dimensional feature vector for each image. Using such a large number of features is advantageous because they can provide a richer representation and capture more subtle visual distinctions between different persons. However, the feature vector may contain non-discriminative elements (some features might capture uninformative features). Even though some invariant properties hold, projecting the feature vector to the feature space of a different camera and match features through proper distances is not always sufficient for finding a good correspondence between persons images. Therefore, we need to find a better way to use the invariant properties of such features and to find the most discriminating elements of the feature vector that allows us to perform a robust re-identification. Towards this objective we propose not to use the distance metrics to find direct correspondences between persons across cameras, but we used the pairwise distance between feature vectors as a new feature.

Distances: To form the DFV for a pair of images we compute pairwise distances for all the considered features. Given two images A and B and the corresponding features extracted as described in Section 6.1.1, we define the following pairwise distances.

- PHOG: $d_\Phi(A^\Phi, B^\Phi)$, where A^Φ and B^Φ are the PHOG features for the image A and image B respectively.
- Color: histograms are compared using distances between feature vectors ex-

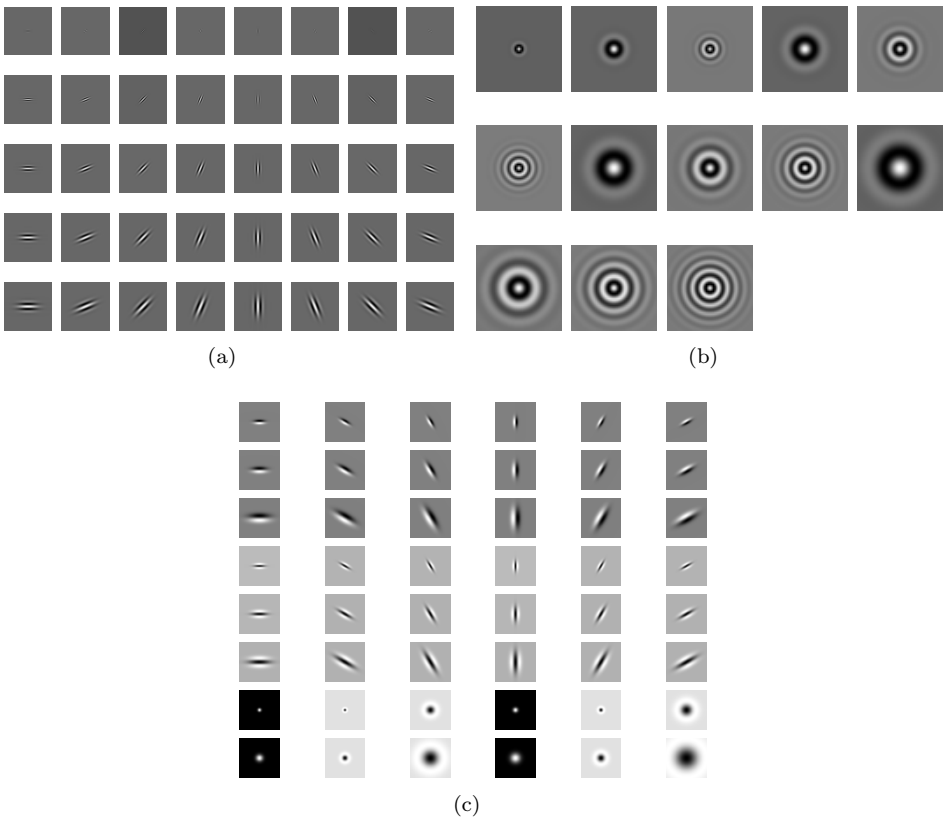


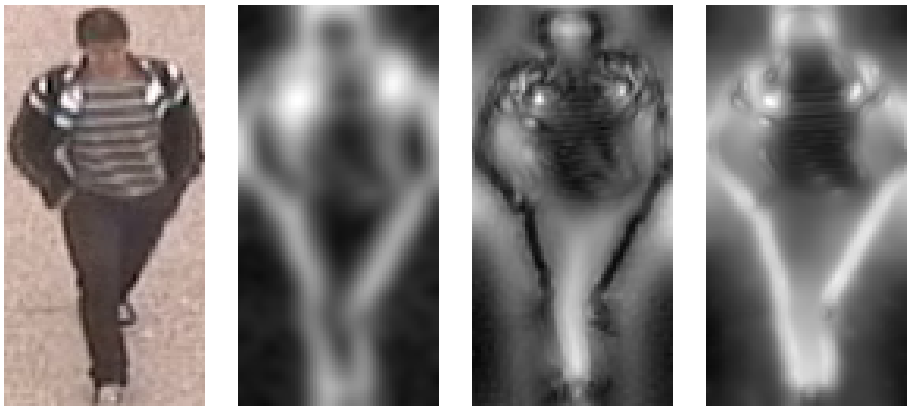
Figure 6.3: (a) Gabor filter bank with 8 orientations and 5 sizes; (b) Standard Schimid filter bank; (c) Leung-Malik filter bank. The set consists of first and second derivatives of Gaussians at 6 orientations and 3 scales making a total of 36; 8 Laplacian of Gaussian filters; and 4 Gaussians.

tracted from the same body part for for each of the three channels as $d_{\Upsilon_T}(A^{\Upsilon_T}, B^{\Upsilon_T})$ and $d_{\Upsilon_L}(A^{\Upsilon_L}, B^{\Upsilon_L})$.

- Gabor: $d_\Gamma(A^{\Gamma_i}, B^{\Gamma_i})$, for $i = 1, \cdots, I$.
- Schmid: $d_\Psi(A^{\Psi_j}, B^{\Psi_j})$, for $j = 1, \cdots, J$.
- LM filters: $d_\Lambda(A^{\Lambda_k}, B^{\Lambda_k})$, for $k = 1, \cdots, K$.

Notice that here we do not specify any particular distance measure since the algorithm can be used with different metrics.

Classification: The DFV computed for a pair of images of the same person is considered as a positive sample, while the DFV computed for a pair of images of different



(a) (b) (c) (d)

Figure 6.4: Response images after convolutions with the three different filter banks shown in Figure 6.3. All filter responses are sum and scaled for visualization. (a) Input image. (b) Response after convolution of 40 Gabor filters. (c) Response after convolution of 13 Schmid filters. (d) Response after convolution of 48 Leaung-Malik filters.

persons is a negative sample. We use our novel pairwise image representation to discriminate in the distance feature space training a random forest classifier [22].

Let A and B be two images, all the computed distances are concatenated to form the DFV $V_{A,B} = \langle d_{\Upsilon_T}, d_{\Upsilon_L}, \cdots, d_{\Lambda} \rangle$. Then, the goal of classification is to learn a mapping from the feature space of V, to the label space, $Y = \{-1, +1\}$.

The random forests algorithm builds a large collection of de-correlated trees exploiting the bagging idea, where the objective is to reduce the variance of an estimated prediction function by pooling many noisy but approximately unbiased models. Trees are ideal candidates for bagging as they capture complex interaction structures in the data and have low bias. Also, trees are very noisy, hence they benefit greatly from the pooling procedure. As shown in [22], an average of N i.i.d. random variables, each with variance σ^2, has variance $1/N\sigma^2$. If the variables are simply i.d. (identically distributed, but not necessarily independent) with positive pairwise correlation ρ, the variance of the average is $\rho\sigma^2 + \frac{1-\rho}{N}\sigma^2$. As N increases, the second term disappears, but the first remains, and hence the size of the correlation of pairs of bagged trees limits the benefits of pooling. The idea in random forests is to improve the variance reduction of bagging by reducing the correlation between the trees, without increasing the variance too much. This is achieved in the tree-growing process through random selection of the input variables.

To learn the parameters of the decision surface that separates positive and negative DFVs we trained a random forest classifier using the steps given in Algorithm 2. Once the model has been trained, a new sample DFVs $V_{A,B}$ is assigned a class label

Algorithm 2: Random Forest for Classification of DFVs

Input : Training DFVs
Output: Trained ensemble of trees

for $n \leftarrow$ **to** N **do**
  Draw a bootstrap sample $\mathbf{Z}^*$ of size S from the training data;
  Grow a random-forest tree T_n to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size s_{min} is reached:
  i. Select m variables at random from the p variables.
  ii. Pick the best variable/split-point among the m.
  iii. Split the node into two child nodes.
end
Output the ensemble of trees $\{T_n\}_{n=1}^{N}$;

$\hat{C}^N(V_{A,B}) = majority\,vote\{\hat{C}_n(V_{A,B})\}_{n=1}^{N}$ where $\hat{C}_n(V_{A,B}) = \{-1, +1\}$ is the class prediction of the n-th random-forest tree.

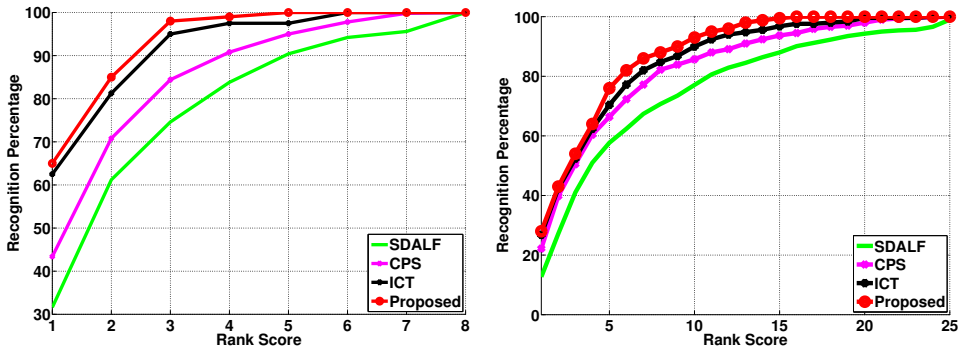
6.1.2 Experimental Results

We evaluate the performance our method using two publicly available benchmark datasets: CAVIAR4REID [26] and Wide Area Re-Identification Dataset (WARD) [92]. To show the achieved performance, we computed the Cumulative Matching Characteristic (CMC) curve.

Implementation Details

In our current framework, we selected the following settings for all the experiments using 4-fold cross validation.

- PHOG: features are extracted for $L = 4$ levels of the spatial pyramid; the HOG histograms computed for each cell have been quantized into $b = 9$ bins.
- Color histograms: The histograms for the torso and legs body parts have been computed using 20, 30, and 30 bins for the L*, a*, and b* channel respectively.
- Gabor: filters at 8 orientations and 5 scales have been used.
- Schmid: We used the 13 standard Schmid filters.
- Leung-Malik: for LM filters we considered the following. The four basic Gaussians have scales $\sigma = \{\sqrt{2}, 2, 2\sqrt{2}, 4\}$. The first and second derivatives of Gaussians occur at the first three scales with an elongation factor of 3. Finally, the 8 Laplacian of Gaussian filters have been defined using the same σ and 3σ.



(a) (b)

Figure 6.5: Comparison of the proposed algorithm with state-of-the-art methods for person re-identification on CAVIAR dataset. In (a) 42 persons have been used for training and 8 person for testing. In (b) 25 persons have been used for training and 25 person for testing.

- Distances: we used the χ^2 distance to compute all the distances given in Section 6.1.1.
- Datasets: we followed a standard image normalization procedure on the datasets and we re-sized all the images to 64×128. We tested the performance of our approach using 40 positive and 40 negative samples per person.

CAVIAR Dataset

We compare our results with those achieved by SDALF [13], CPS [26] and ICT [4], as reported in [4]. To fairly evaluate our approach we used the same two setups proposed in [4] showing the relative performance as a function of the size of the training data. We run 10 independent trials for each setup and average the achieved results.

Figure 6.5(a) shows the performance of the method when 42 persons have been used to form the training set. The remaining 8 persons form the test set. Using 42 persons as training data our approach achieves similar performance to ICT and it outperforms both the other two methods used for comparison. We achieved 65% rank 1 correct matches and we re-identify all the persons in the test set in the first 5 ranks thus outperforming all other methods.

In figure 6.5(b), the recognition performance are computed using a training set and a test set of 25 persons. As for the previous scenario, the performances of the our approach are similar to those of ICT. We achieved a recognition percentage of about 78% for a rank score of 5. For the same rank score, a recognition percentage of 72%, 67% and 56% is achieved by ICT, CPS and SDALF, respectively. Similarly as before, we achieve the 100% recognition percentage with a lower rank score value than all other methods. In particular, we recognize all the persons in the test set when the

rank score is 15.

WARD Dataset

We compare our results with those achieved by WACN [92] and SDALF [13], then we deeply investigate our performance under two different setups. For each result we run 10 independent trials and we show the average performance for the three camera pairs, here denoted camera pair 1-2, 1-3 and 2-3.

Figure 6.6(a), 6.6(b) and 6.6(c) show the performance of our method compared to RWACN and SDALF. The recognition performance are computed using a training set and a test set of 35 persons. For all the three camera pairs we outperform the methods used for comparisons. For camera pair 1-2 (see Fig. 6.6(a)), we achieve a recognition percentage of 84% for a rank score of 5, while, for the same rank score, RWACN and SDALF achieve a recognition percentage of 48% and 36% respectively. Similarly, for camera pair 1-3 (see Fig. 6.6(b)), a recognition percentage of 86% is achieved for a rank score of 5. The other state-of-the-art methods achieve the same recognition percentage for a rank score of 19 and 23 respectively. Finally, for camera pair 2-3 (see Fig. 6.6(c)), we achieve a recognition percentage of more than 50% for a rank score of 1 thus outperforming both methods used for comparison.

Figure 6.7(a), 6.7(b) and 6.7(c) show the relative performance of the approach as a function of the size of the training data. The CMC curves have been computed using all the color, shape and texture features as described in section 6.1.1. Notice that the maximum rank for each curve is given by the number of persons used for testing. For all the three curves the performance are not decreasing that much even if only 50% of the persons in the dataset are used for training and the remaining 50% of persons for testing. In such case, the worst performances still lead to a recognition rate higher than 33% for the rank 1 score. The best recognition percentage is achieved for the camera pair 2-3, where a recognition rate of 52% is achieved. For all the three camera pairs, the recognition rate strongly improves as the number of persons used for training increases. When 63 persons out of 70 are used for training a recognition rate higher than 60% is achieved for rank 1 for all the camera pairs. In particular, a recognition rate of 81% is reached for rank 1 score for camera pair 1-2.

Figure 6.8(a), 6.8(b) and 6.8(c) show the performance of the method when 56 persons out of 70 have been used to form the training for all the three cameras in the dataset. We show different CMC curves for the remaining 14 persons when only some of the proposed features are used for re-identification. For all the three cameras the combination of all the proposed features achieves the best overall performances. Despite of that it's worth noticing some facts. For the first and the third camera pair, the most discriminative features are the color and the shape features, while for the second camera pair the color features have weaker performance than texture and shape features. Considering the combination of all features we achieved about 50% rank 1 correct matches for the first and second camera pair. For the third camera pair the performance increase significantly and a 70% rank 1 is achieved. Finally, a visual comparison of the achieved results among all the three cameras in shown in Figure 6.9.

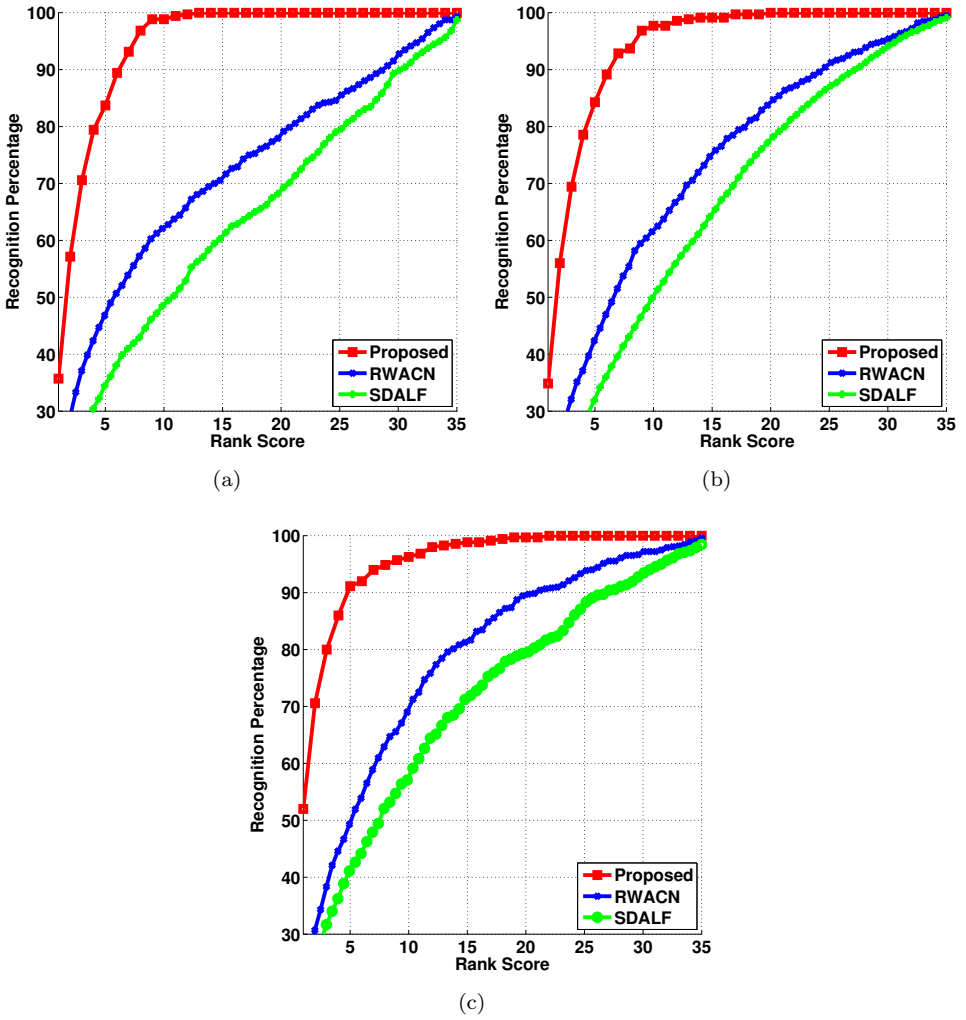


Figure 6.6: Comparison of the proposed algorithm with state-of-the-art methods for person re-identification on WARD dataset. (a) Recognition performance for camera pair 1-2. (b) Recognition performance for camera pair 1-3. (c) Recognition performance for camera pair 2-3.

6.1.3 Conclusions

In this work we presented an approach for person re-identification in a non-overlapping multi-camera scenario. We introduced a method that models not the way features are transformed across camera, but exploits invariant features and robust distance metrics to model how the distances between such features are transformed across

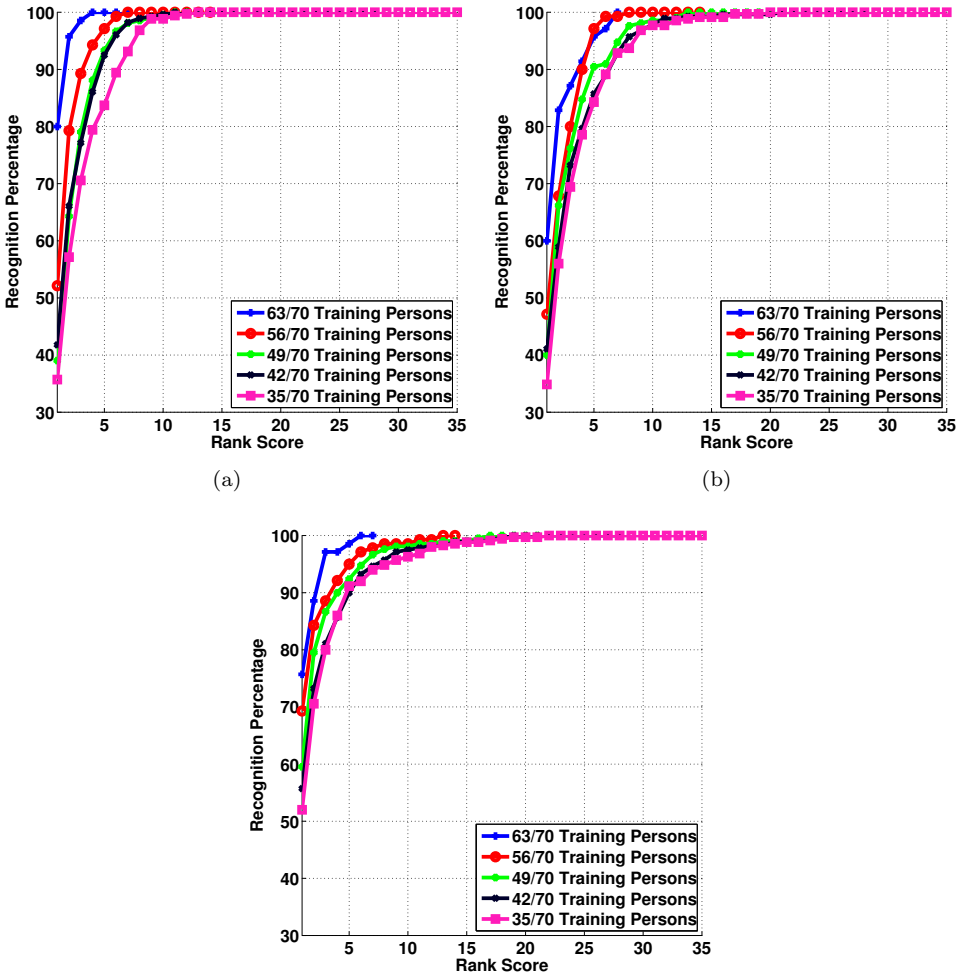


Figure 6.7: Performance on the WARD dataset for varying train and test dataset sizes. Recognition performance for camera pairs 1-2, 1-3 and 2-3 are shown in (a), (b) and (c) respectively.

non-overlapping cameras. Towards this objective we extracted feature vectors from pairs of persons images viewed in different cameras, and we computed the distance between them to form the DFV. We trained a binary classifier to discriminate between DFVs and to perform the re-identification. To validate the proposed method we compared the performance of our approach to state-of-the-art methods using two publicly available benchmark datasets.

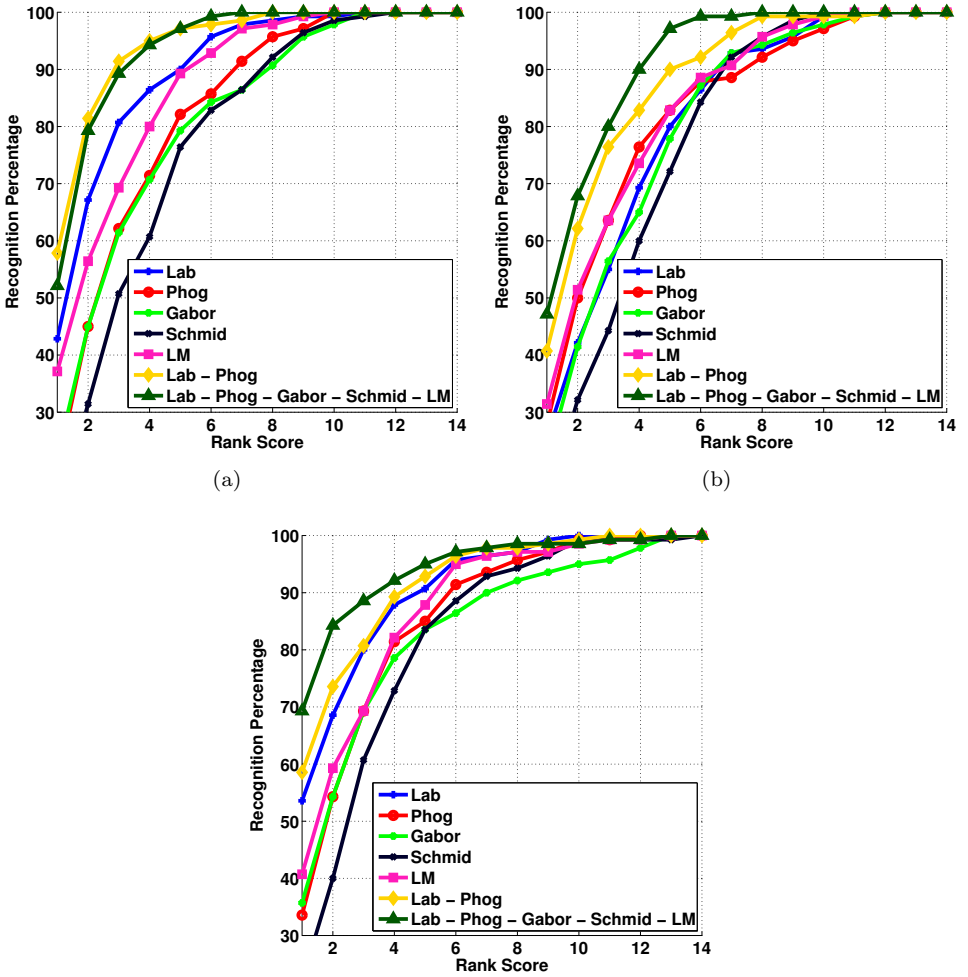


Figure 6.8: Performance on the WARD dataset using different combination of the proposed features. Recognition performance for camera pairs 1-2, 1-3 and 2-3 are shown in (a), (b) and (c) respectively.

6.1.4 What next?

While the three approaches presented so far have high re-identification performance, they still do not consider the re-identification problem by the camera network viewpoint. Indeed, each approach perform the re-identification between camera pairs only. In the next section we propose to shift this viewpoint and focus on the re-identification problem in a camera network. Towards this end we'll introduce a distributed frame-

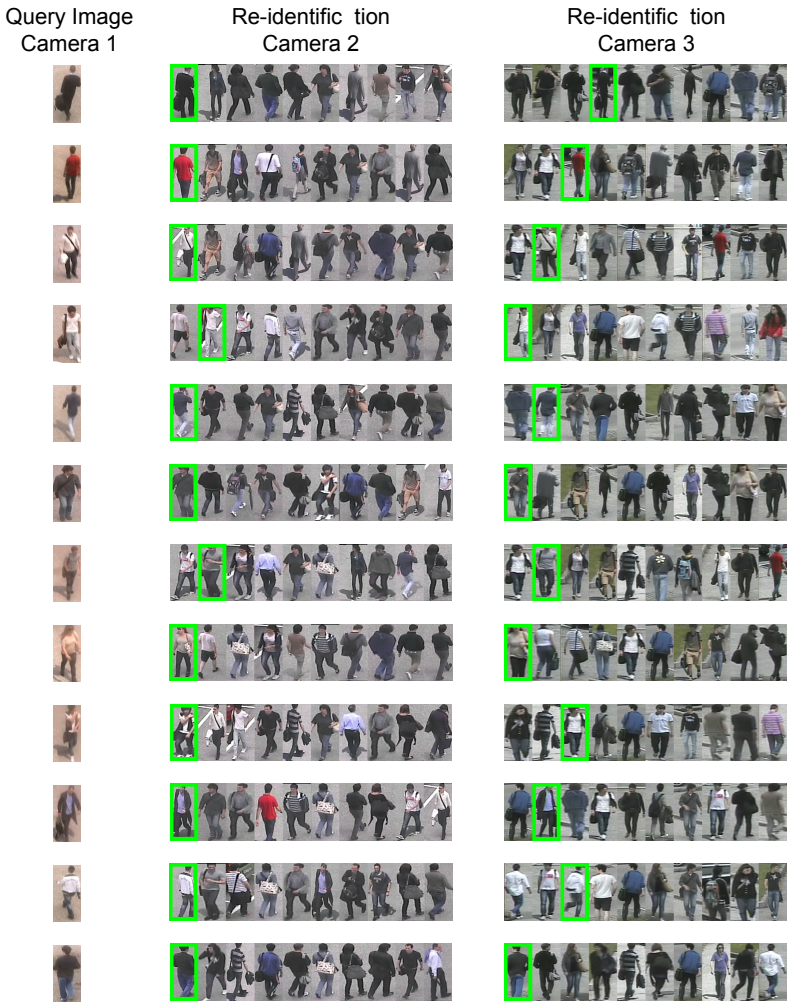


Figure 6.9: Visual comparison of matches using the proposed method for camera pair 1-2 and 1-3 of the WARD dataset. First column is the probe image. Second and third columns show the top 10 matches computed from camera 2 and 3 respectively

work that, while presented from the point of view of only one of the proposed methods, can be easily extended to all other ones.

6.2 Efficient Person Re-Identification in a Camera Network

While the person re-identification methods introduced in the previous chapters and section achieved high performance on the re-identification task, none of them has considered a wider approach in which the re-identification is extended to the whole network. This gives rise to very interesting problem, such as the networking and computational costs involved in the process,etc.. To the best of our knowledge the only works that have address the challenges of re-identification considering networking and computational capabilities have been proposed in [94, 141]. In this section we want to introduce a more general framework which can be used by any re-identification method. Towards this goal, we first extend the basic matching mechanism introduced in section 4.5 by introducing a more efficient method that advantages of the proposed feature accumulation process. Then, we consider such mechanism to introduce the novel distributed re-identification framework.

6.2.1 Efficient Signature Matching

In the following, as we propose a method to extend the pure signature matching proposed in section 4.5, we use the same notation as the one in Chapter 4. We have defined $\Phi(p, c)$ as the signature of a person p acquired by camera c computed by accumulating features from N images. In the section 4.5 a method to match a probe signature $\Phi(p, c_i)$ with a gallery signature $\Phi(g, c_j)$ has been introduced. One might think of using the proposed method in two different ways: i) wait for all the available images of person p before computing the probe signature and sending it to camera c_j to ask for a match gallery signature; ii) send a new probe signature every time a new frame of person p is available. However, neither of these two solutions is efficient in terms of computational and networking costs. So, in the following we introduce a novel efficient signature matching technique. The whole process is shown in Figure 6.10.

Let $\Phi(p, c_i)$ be the probe signature formed by accumulating multiple features extracted from the N frames of person p acquired by camera c_i. Let also suppose that we want to re-identify person p in camera c_j, and let $\mathcal{G}_j$ be the set of all gallery persons in camera c_j. After c_i computes the initial signature, this is sent to camera c_j that is in charge to match $\Phi(p, c_i)$ with all the gallery persons $g \in \mathcal{G}_j$. This first round of matches gives rise to the set $\mathcal{G}_j^* \subseteq \mathcal{G}_j$ composed by all signatures $\{\Phi(g, c_j) : g \in \mathcal{G}_j \wedge [d\left(\Phi(p, c_i), \Phi(g, c_j)\right) < Th_{high}]\}$.

Then, either a match is detected or not, the camera c_i is asked to process more frames of the same probe person p. In particular, in case a correct match is detected the probe camera is asked to extract features from P frames, otherwise it is asked to extract features from $W < P$ frames. This two parameters allow to save more resources, as a low number of images may be sufficient to re-identify the person if a valid match has already been detected. Camera c_i extracts the features from the new images and sends only those to camera c_j as it has already received the initial probe

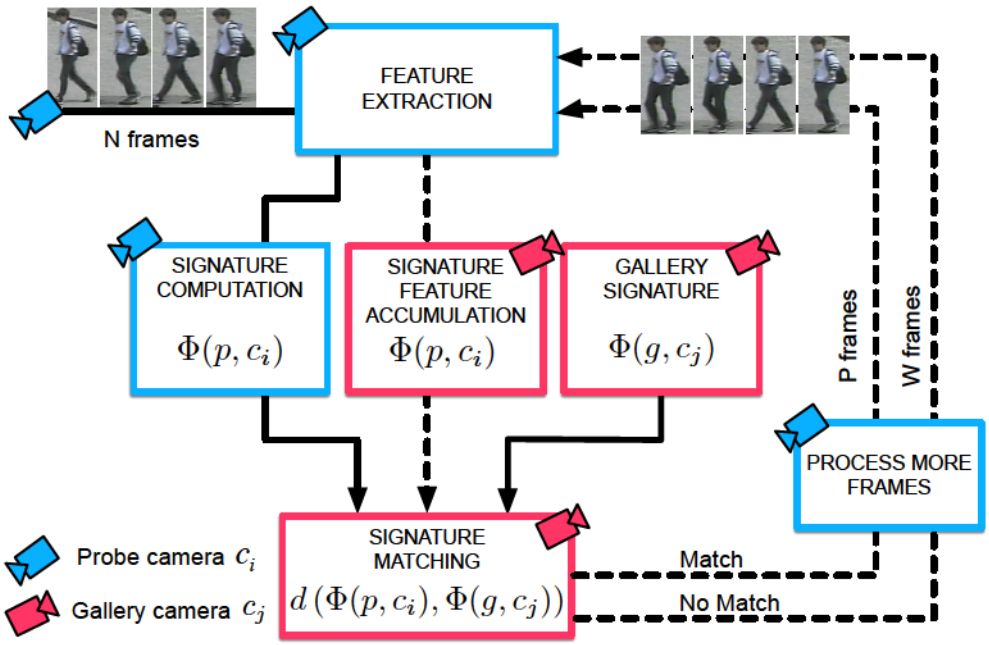


Figure 6.10: Efficient signature matching. N frames are acquired by the camera c_i to compute the initial probe signature. The signature is sent to camera c_j that matches the probe signature with all gallery signatures. If a match is detected P more frames are processed and the extracted features are sent to c_j that updates the old signature with the new features, otherwise, W frames are processed if no match is detected. The process repeats until no more images for person p are available or a single signature matches with the one for person p.

signature. Camera c_j accumulates the new features and updates the given query signature $\Phi(p, c_i)$.

Then, if at the previous step a match has been detected, the query signature is compared against the previously matched signatures only, that is, with all the gallery persons $g \in \mathcal{G}_j^*$. The matches are computed with all the persons $g \in \mathcal{G}_j$ if $\mathcal{G}_j^*$ is the empty set. In both cases, as more features are used to compose the probe signature, a threshold value $Th_{low} < Th_{high}$ is used to get the final matching set $\mathcal{G}_j^{**} = \{\Phi(g, c_j) : d(\Phi(p, c_i), \Phi(g, c_j)) < Th_{low}\}$. The process is repeated until no more images are available to update the probe signature $\Phi(p, c_i)$ or the set $\mathcal{G}_j^{**}$ consists of a single signature, that is the re-identified person signature. If all the available images have been used and $\mathcal{G}_j^{**}$ consists of more than a single signature, the person with the lowest signature distance (see eq. (4.26)) is considered as the re-identified person.

6.2.2 Distributed Re-Identification

Having a single central unit processing all the incoming data from sensors is a common bottleneck for systems that requires real-time performance. Not only that, the information traveling through the network may be useless for many of the nodes, while it is of interest for just a few of them. This is particularly true for the person re-identification problem as the topology of the monitored environment can constrain the path of persons. Also, in many situation a person cannot move from the FoV of one camera to the FoV of a different camera without being viewed from another camera located in the middle of the path. Thus, it is not worth to exchange all available information/signatures between the two cameras and not send any data to the camera in the middle. To prevent network overloading with useless information and, at the same time, to take into account the topology of the network to tackle the person re-identification problem, we first introduce a camera matching cost hand over measure, then we propose a derivation of the distance vector algorithm to perform the re-identification only within a subset of the nodes of the network and to ask for matches in a priority fashion. The procedure is depicted in Figure 6.11.

Let c_i be one camera in $\mathcal{C}$, then the camera matching cost hand-over measure for c_i is denoted as $\Omega_{c_i} \in \mathbb{N}^{|\mathcal{C}|}$. Let $\Omega_{c_i}(c_j)$ be the matching cost from camera c_i to camera c_j. Also let $\mathcal{G}_i = \{g_{i,1}g_{i,2}, \cdots, g_{i,K}\}$ and $\mathcal{G}_j = \{g_{j,1}g_{j,2}, \cdots, g_{j,L}\}$ be the set of gallery persons with known identities available for cameras c_i and c_j respectively. During the off-line training phase, all the gallery signatures available from all cameras are broadcasted through the network to compute the initial camera matching costs for all c_i, c_j in $\mathcal{C}$ as

$$\Omega_{c_i}(c_j) = \sum_{g_{i,k=1}}^{K} \begin{cases} -1, & \text{if } \Omega_{c_i}(c_j) > 0 \wedge \mathcal{G}_j^* \neq \emptyset \\ +1, & \text{otherwise} \end{cases} \tag{6.1}$$

where N images are used to compute both the signatures for camera c_i and c_j, and $\mathcal{G}_j^*$ is the set of all matching signatures from c_j (in this case, defined as in section 6.2.1). In the experimental section we show that using such camera matching cost we can infer the topology of the network thus constraining the re-identification to a subset of nodes (i.e. cameras).

Let now introduce the distance vector routing algorithm and explain how it can be used together with the camera matching cost to perform a distributed re-identification. The distance vector routing algorithm (also known as the distributed Bellman-Ford routing algorithm or the Ford-Fulkerson algorithm) is derived from the fact that routes are advertised as vectors of (destination, distance), where the former denotes the preferred outgoing line, while the latter is the distance to that destination. Each router is assumed to know the distance to each of its neighbors, and as the network exchanges data through the network, each router learns routes from its neighboring routers' perspectives and then advertises the routes from its own perspective.

The distance vector routing algorithm is used to route the probe signatures and ask for matches to cameras in the network in a priority fashion as follows. Let $\Phi(p, c_i)$ be a probe signature computed from camera c_i by accumulating features from N

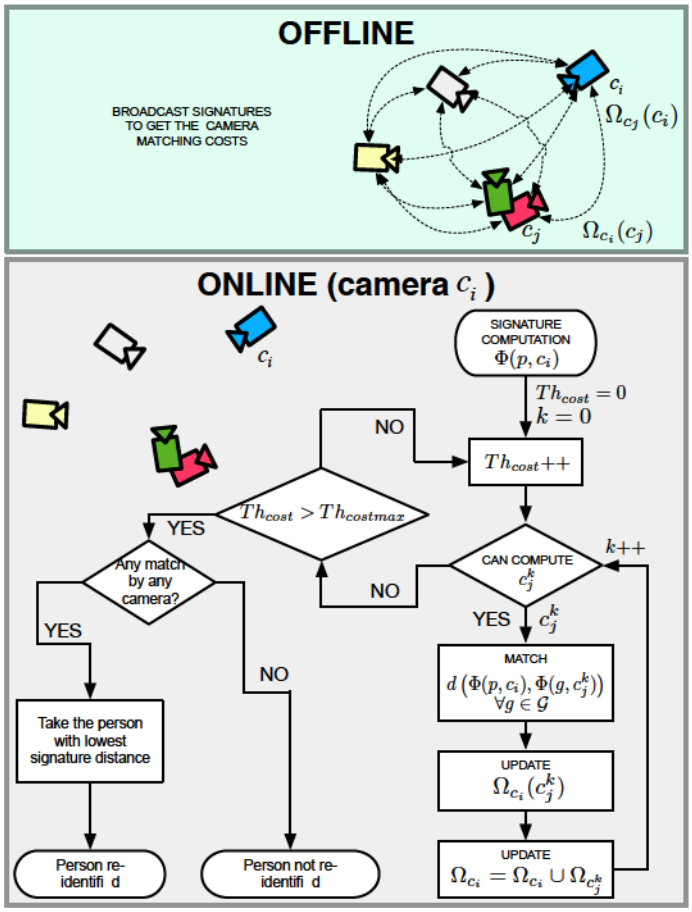


Figure 6.11: Distributed re-identification flowchart.

images of same person p. Let $Th_{cost} \in \mathbb{N}$ be a threshold that is used to select the camera of the network to which the probe signature has to be sent, and $Th_{costmax}$ the maximum cost that is allowed to pay to match the probe signature with gallery signatures computed by cameras in the network.

Having defined $Th_{costmax}$, and initialized $Th_{cost} = 0$ and $k = 0$, the iterative procedure shown in Figure 6.11 starts. Let c_j^k be the camera to which the probe signature is sent be computed as

$$c_j^k = \left\{c_j^k \quad : \quad c_j^k \in \left\{\mathcal{C} \backslash \{c_j^{k-t}\}_{t=1}^{k-1}\right\} \quad \wedge \quad \Omega_{c_i}(c_j^k) < Th_{cost}\right\}_1 . \tag{6.2}$$

Where the subscript 1, denotes that, if the set is not empty, the first element of it is taken. Notice, that if there's no camera c_j^k satisfying all the given conditions, the algorithm checks if the maximum cost is reached. If that is not the case, the threshold

Th_{cost} is increased and c_j^k is computed again. On the contrary, if c_j^k can be computed, the probe signature is routed to such camera that matches all its gallery signatures $g \in \mathcal{G}_j$. After all the possible matches have been computed, the camera matching cost between c_i and c_j^k is updated as in eq. (6.1).

After updating the matching cost $\Omega_{c_i}(c_j^k)$, the distance vector updating rule comes into picture. As a router learns routes from its neighbors, the current probe camera c_i learns the matching cost from the gallery camera c_j^k. In particular we update the camera matching cost of camera c_i using the information given by the camera matching cost of camera c_j^k as

$$\Omega_{c_i} \cup \Omega_{c_j^k} = \min\left(\Omega_{c_i}(c_l), \Omega_{c_i}(c_j^k) + \Omega_{c_j}(c_l)\right), \forall c_l \in \mathcal{C}. \tag{6.3}$$

Then, process is repeated $(k++)$ and the probe signature is sent through all the cameras in the network that satisfy the conditions in eq. (6.2). When the conditions fail, Th_{cost} is increased and the process starts again. Notice that k is not changing here as we do not what to send the probe signature to a camera that has already received it. Once the maximum allowed matching cost is reached,i.e. $Th_{cost} > Th_{costmax}$, the distributed re-identification procedure stops. If no match has been given by any of the c_j^k cameras that have received the probe signature, person p is added to the set of gallery signatures in c_i. Otherwise, if one or more matches are given by the c_j^k cameras, the person that has the lowest signature distance is considered as the re-identified person. Finally, if the signature with lowest distance has been computed by camera $c_j^k == c_i$, then such matching gallery signature is updated with the probe signature $\Phi(p, c_i)$.

While the proposed process is used to save network and computational resources, the distance vector and the camera matching costs also allow to reach another important goal. It is a matter of fact that, due to illumination changes, preferred paths, etc. the re-identification performance between two nodes of the network may change during the day. Using the distance vector algorithm, each camera of the network gets updated with the camera matching costs coming from other cameras in the network each time a re-identification is performed. This allows the network to perform the re-identification in a robust and adaptive fashion.

6.2.3 Experimental Results

To evaluate the performance of the proposed method we consider two public benchmark datasets WARD [92] and DANA36 [111]. Each one covers different aspects and challenges for the person re-identification problem. A comparison and details of the used re-identification datasets is given in section 2.2.6. As commonly suggested by the literature, we report the performance of our method in terms of recognition rate by the Cumulative Matching Characteristic (CMC) curve.

We report the results of our approach using both a single-shot (i.e. $N = 1, P = 0, W = 0$) and a multiple-shot (i.e. $N > 1, P \geq 0, W \geq 0$) strategy. In both cases N images were used to compute the gallery signatures. To compute the CMC curves, the proposed distance is used to match the gallery signatures with the probe signatures.

To fairly evaluate our method against state-of-the-art approaches we perform the whole re-identification procedure 10 times using different sample images. We report the CMC curves averaged over the 10 trials.

The novel distributed re-identification mechanism has been used to route each probe signature in turn to the camera c_j^k as defined in eq. (6.2). The distributed re-identification mechanism is influenced by the value of $Th_{costmax}$ that determines how many cameras should receive the probe signature. For the WARD dataset, as there are only 3 cameras, we set it so as the probe signature is sent to all other cameras, while for the DANA36 dataset, that has images from 36 cameras, we provide a method to automatically select such value by analyzing the matching cost matrix.

Implementation Details

The same parameters as those used in section 4.6 have been used. However, as we're using the efficient matching mechanism discussed in section 6.2.1 we need to specify two additional parameters, that are Th_{high} and Th_{low}. In the following experiments these have been set to 0.15 and 0.1 respectively.

WARD Dataset

The WARD dataset has 4786 images of 70 different people acquired in a real surveillance scenario by three non-overlapping cameras. This dataset is of particular interest because it has a huge illumination variation apart from resolution and pose changes. We conducted the experiments for all the three cameras and report the results for camera pairs 1-2, 1-3, and 2-3. This is done to make a fair comparison with the methods for which either the CMC performance on this dataset is reported in literature or the code is available. Namely the methods are SDALF [13] and DSF [94]. As for other dataset we report the results computed using both a single-shot and a multiple-shot strategy.

In Figure 6.12 we show the results of the proposed method on the WARD dataset and compare them with the ones achieved by SDALF [13] and DSF [94]. For each camera pair we also show the performance of our approach varying the values of N, P and W.

In Figure 6.12(a) results are reported for camera pair 1-2. Using just a single image to compute both the gallery and the probe signature we achieve similar performance to the one achieved by SDALF, which uses $N = 5$ images to compute the gallery and the probe signature. Then, by increasing the number of initial images used to compute the signatures to $N = 3$ we achieve better performance than both the state-of-the-art methods used for comparisons. By increasing the number of initial images used to compute the signatures and the number of images used to update the signatures in case of correct match/no correct match, we outperform all of them. In particular, using the combination of $N = 5$, $P = 3$ and $W = 3$ we achieve a 80% correct recognition percentage for rank score of 20. The same recognition percentage is achieved at rank 51 and at rank 42 by SDALF and DSF respectively.

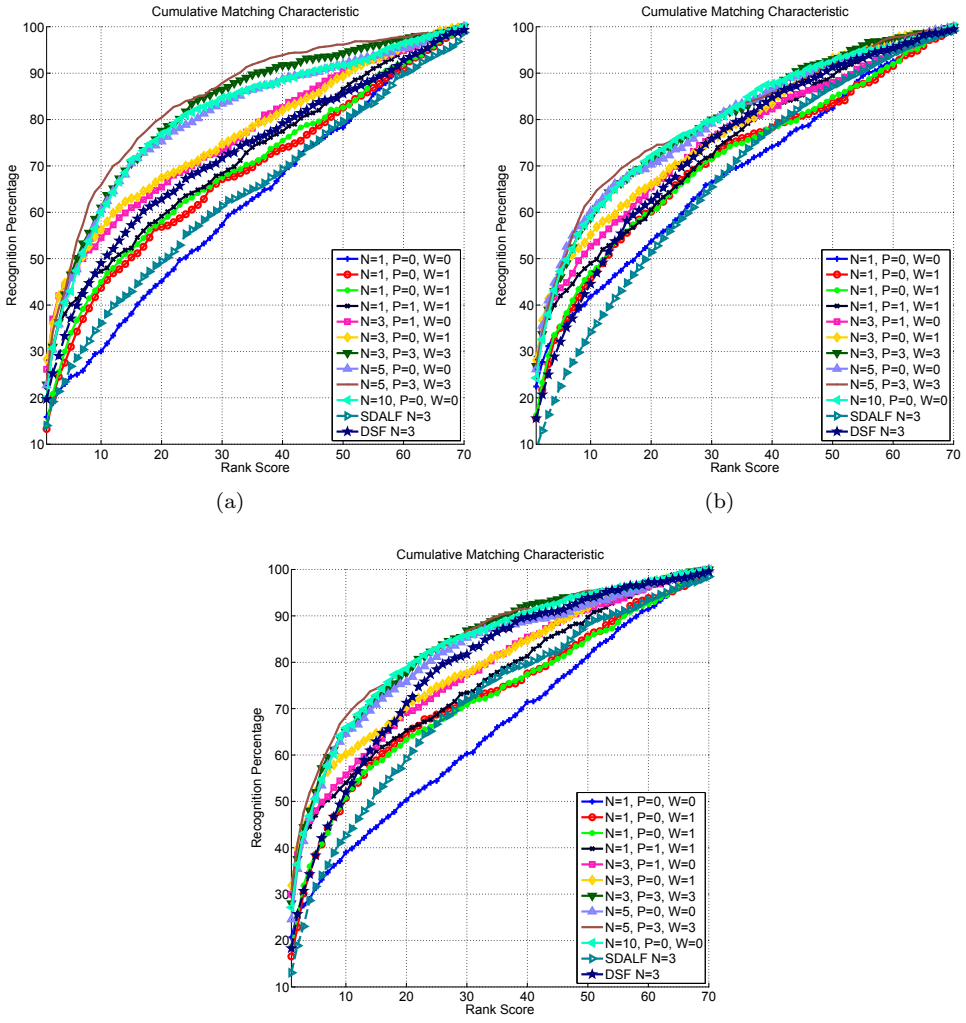


Figure 6.12: Results and comparisons on the WARD dataset. In (a), (b) and (c) results are reported for camera pairs 1-2, 1-3 and 2-3 respectively.

In Figure 6.12(b) results are reported for camera pair 1-3. Similarly to the previous reported results our method has similar results to SDALF when we use $N = 1$ images to compute the signatures. However, in this case, the performance of our approach are very similar to those of DSF when more images are considered to build the initial signature and to update them. Considering the loper ranks, from rank 1 to rank 10, using $N = 3$ images to build the initial signatures, as done by DSF, we outperform

it by 11% on average. Then, for higher tanks, the performance are becoming more similar.

Finally, in Figure 6.12(c) results are reported for camera pair 2-3. In this case SDALF achieves better performance than us when we're using $N = 1$ images to build the initial signature. But, using the same configuration with $N = 3$ images we outperform it. DSF, instead achieves better performance than us using $N = 3$ images only for ranks higher than 16. For rank 1, in fact, we outperform it achieving a recognition rate of about 30% while, DSF has only a recognition rate of 18%. Similarly to the other camera pairs, when the number of images used to compute the initial signature increases we outperform all the methods used for comparisons.

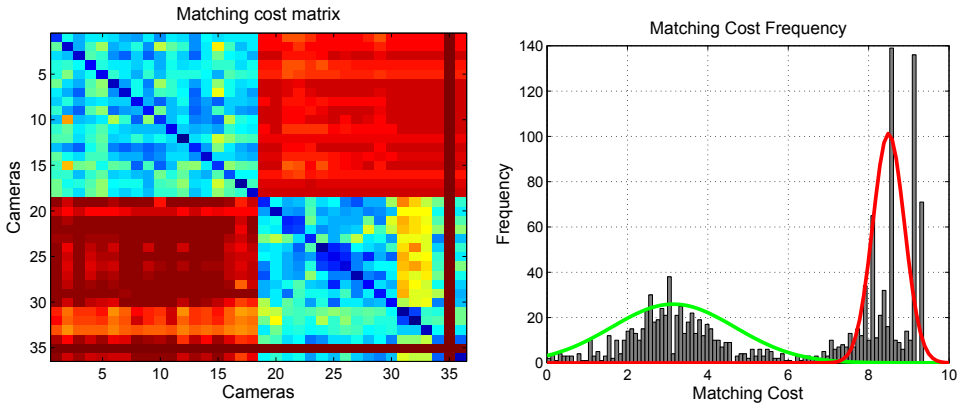
DANA36 Dataset

The DANA36 dataset consists of 23,641 images, depicting 15 persons and nine vehicles. The dataset was acquired from 36 stationary camera views using a variety of surveillance cameras with resolutions ranging from standard VGA to three mega-pixel. 27 cameras observed the persons and vehicles in an outdoor environment, while the remaining 9 observed the same persons indoors. Due to variety of camera locations, vantage points and resolutions, the dataset provides means to adjust the difficulty of the re-identification task in a controlled and documented manner.

While this dataset cannot be considered as representative for a real scenario as only 15 persons are observed, it has images coming from 36 cameras, so we used it to show that, using the proposed camera matching cost, we can reduce the computational and networking costs needed to perform the re-identification in a camera network. This is done by first learning the topology of the environment though re-identification. Towards this objective, we split the dataset as follows. 15 images of 7 out of 15 persons are taken from camera 1 to camera 18, 15 images for the remaining 8 persons are taken from each of the remaining cameras, namely camera 19 to camera 36. As no persons are acquired by camera 35, the resulting dataset contains 3,372 images of 15 person acquired by 35 cameras.

The distributed re-identification mechanism is influenced by the maximum allowable matching cost $Th_{costmax}$ that controls which cameras in the network have to match a probe signature. Here we show that the optimal value of this parameter can be found by analyzing the matching cost matrix. Let consider Figure 6.13, in which we show the initial matching cost matrix computed by broadcasting all the gallery signatures to all the cameras in the network and the corresponding distribution of matching costs. In particular, such distribution of matching costs can be used to compute the optimal threshold $Th_{costmax}$. We propose to perform such operation by applying the histogram entropy-based thresholding method proposed in [74]. This allows to automatically determine $Th_{costmax}$, hence, the cameras to which the probe signatures have to be sent. In the reported case, we found that $Th_{costmax} = 5$. Such value is used in the following to present the re-identification results.

Having computed the maximum threshold $Th_{costmax} = 5$ we can evaluate the performance of the method using the distributed approach and compare them with the case $Th_{costmax}$ is not used, i.e. all the cameras in the network are asked to match



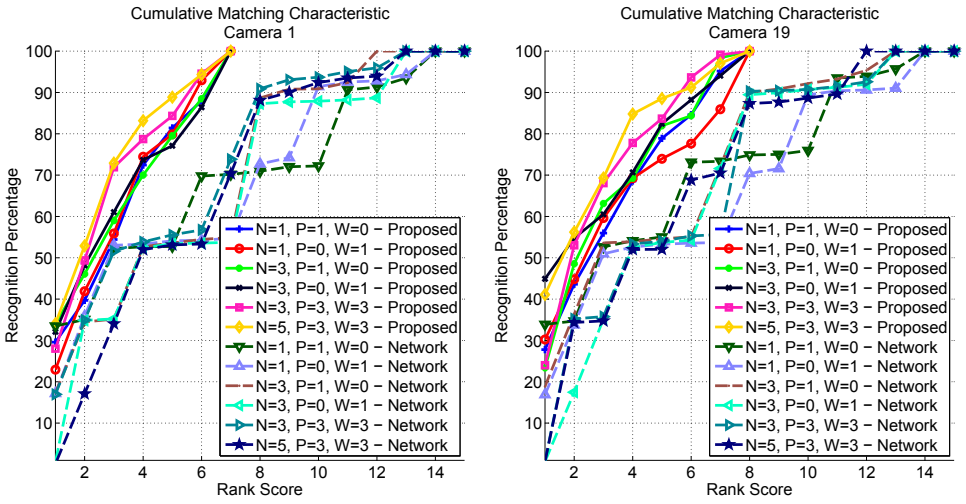
(a) (b)

Figure 6.13: In (a) the initial matching cost matrix computed by broadcasting all the gallery signatures to all the cameras in the network is shown. In the color coded plot, red values mean high cost, while blue values mean low cost. In (b) the distribution of the matching cost values is plot and two Gaussian probability density functions have been fit.

any probe signature. In both cases multiple cameras are asked to match the probe signatures, so, in the following we report the results in terms of CMC curves averaged over all such cameras.

In Figure 6.14 we show the effects of the efficient signature matching approach for two probe cameras, namely camera 1 and camera 19. Here the curves labeled as 'Proposed' are the ones computed using the distributed approach, whereas the curves labeled as 'Network' are the ones computed by matching the probe signatures with all the cameras in the network. Let first consider Figure 6.14(a) where the results are reported for camera 1. As for the results reported for the WARD dataset, by using the efficient matching mechanism together with the proposed distributed approach, we achieve good performance and we have a 100% of correct recognition within rank 7. In particular, using the combination of $N = 5$, $P = 3$ and $W = 3$ images to match the probe signatures, we achieve a 33% correct recognition percentage for rank 1. On the other hand, if the probe signatures are sent to all the cameras in the network, the results significantly decrease and, considering the same combination of $N = 5$, $P = 3$ and $W = 3$ only a correct recognition of 2% is achieved for the same rank 1. Not only that, the 100% of correct recognition is reached at rank 13.

The same situation occurs in Figure 6.14(b) where the CMC curves are computed for camera 19. Using the distributed approach and the efficient matching we achieve a 42% of correct recognition at rank 1 using $N = 5$, $P = 3$ and $W = 3$. For all the configurations, the 100% of correct recognition is achieved at rank 8. If the proposed distributed approach is not exploited, using the same combination of $N = 5$, $P = 3$ and $W = 3$ only a correct recognition of 3% is achieved at rank 1 and the 100% of



(a) (b)

Figure 6.14: CMC curves computed by applying the proposed distributed matching and by searching for a correct match through all the cameras in the network. In (a) the CMC curve is computed with respect to camera 1. In (b) the CMC curve is computed with respect to camera 19.

correct recognition is reached at rank 12.

As shown, if we use the distributed approach not all the cameras of the network are required to match the probe signatures. In fact, using the distributed approach, in the first case (see Figure 6.14(a)) the probe signatures are matched with only 7 gallery signatures (i.e. the 7 persons that have been captured by cameras 1 to 18), whereas in the second case (see Figure 6.14(b)) the probe signatures are matched with only 8 gallery signatures (i.e. the 8 persons that are present in cameras 19 to 36). That is the reason why a 100% of correct recognition is achieved with rank 7 and rank 8 for the two reported cases respectively. So, the proposed distributed approach not only decreases the network bandwidth requirements but it also increases the re-identification performance.

Computational costs

In the previous analysis we evaluated the proposed method and reported the results in terms of performance (i.e. using the CMC values). Apart from the good re-identification performance in that sense, the distributed mechanism brings advantages in terms of computational costs.

In Table 6.1 we report the results of the proposed method when the distributed matching mechanism is used and when it's not. The results are for camera 1. As shown, at the current state, the proposed methods cannot be deployed on embedded

Table 6.1: Computational times for the proposed method where the distributed matching technique is used (first 6 rows) and when it is not used (last 6 rows). Results are reported for camera 1.

Configuration	Feature Extraction Time (s)	Match Searching Time (s)	Overall Re-Identification Time (s)
N=1, P=1, W=0 - Proposed	42.45	37.15	84.15
N=1, P=0, W=1 - Proposed	43.32	31.18	76.15
N=3, P=1, W=0 - Proposed	85.09	38.94	126.01
N=3, P=0, W=1 - Proposed	83.95	35.55	124.73
N=3, P=3, W=3 - Proposed	132.24	42.39	177.32
N=5, P=3, W=3 - Proposed	184.87	39.04	225.07
N=1, P=1, W=0 - Network	42.41	98.12	144.51
N=1, P=0, W=1 - Network	44.21	91.42	136.74
N=3, P=1, W=0 - Network	86.45	118.34	206.18
N=3, P=0, W=1 - Network	84.32	125.35	211.45
N=3, P=3, W=3 - Network	139.44	135.90	270.27
N=5, P=3, W=3 - Network	182.24	127.41	314.66

smart cameras. However using the proposed technique the times get strongly reduced. Indeed, using the distributed approach to match an individual takes three times less the time needed to match an individual across the whole network by searching for a match with all the deployed sensors.

6.2.4 Conclusion

In this section we have introduced a general framework for distributed re-identification methods. Towards this goal, we have first extended the basic matching mechanism introduced in section 4.5 by introducing a more efficient method that advantages of the proposed feature accumulation process. Then, we have considered such mechanism to introduce the novel distributed re-identification framework. To achieved such goal we have introduced a camera matching cost measure and a derivation of the Distance Vector routing algorithm to send a probe signature only to a subset of the cameras in the network. Results on two benchmark dataset showed the efficiency of the proposed method. Not only that, it has been shown to be able to automatically select the cameras to which send the query signature by analyzing the distribution of the camera matching costs.

Conclusions

In this thesis, we have addressed two main related problems: the problem of visualizing the proper information to surveillance operators, and the person re-identification problem.

Presenting the useful information to the human operators is a challenging task as images from many different cameras should be displayed at the same time. This is a well established problem, as it's known that, these systems require a prohibitive amount of human resources and the operator's attention quickly decreases through time. To tackle these issues we have applied our knowledge in the field of computer vision and Human-Computer Interaction to propose an advanced VSS that supports the surveillance operators tasks. In particular we focused on the task of tracking persons that are moving within the monitored environment.

In the proposed VSS, we have assumed that persons moving across cameras can be re-identified so as the tracking task can be performed. However, this is a challenging and open problem, known as the person re-identification problem. So, we also proposed three different approaches to attack the re-identification problem between camera pairs. Namely, we have considered a discriminative signature based method, a feature transformation based method and an error transformation based method. While being effective to re-identify targets between camera pairs, the proposed methods require high computational resources that has to be shared to achieve the final objective. So, this makes the process of extending the re-identification problem to the whole network intractable if a centralized approach is adopted. To address this issues we have introduced a novel framework that can be used by any of the proposed method to perform the re-identification in a camera network in a fully distributed fashion.

The following were the main contributions of the thesis.

Advanced Human Interface for a VSS

In Chapter 3 we have introduced an effective and powerful information visualization technique. The idea was to properly visualize only the most important cameras and information contents to simplify the operators' tasks. The main novelty was the dynamic organization, activation and switching of the UI elements based on the output of video analytics algorithms. Rather than displaying all available camera views, only most probable streams, i.e. those that will be involved with the objects motion, are presented. The results showed that the proposed information visualization technique achieves high usability results and supports the operators during their surveillance tasks.

Re-Identification by Discriminative Signature Matching

In Chapter 4 we have introduced a method to tackle the re-identification challenges by means of discriminative signature matching approach. Each sensor in the network exploited camera specific learned models of persons to detect pedestrians and to extract both the whole body silhouette and the different body parts. Then, local and global appearance features have been extracted from the silhouette and accumulated over multiple images of the same person forming a highly discriminating signature that was finally matched with gallery signatures to perform the re-identification. Comparisons with state-of-the-art approaches have shown that the proposed method achieves similar or superior performance to those.

Re-Identification by Classification of Warp Feature Transformation

In Chapter 5 we have built upon the results of the previous approach. In particular we have focused on the significant loss of performance when strong illumination and color changes occur between different cameras. Inspired by this we aimed to understand how features get transformed across cameras. Differently from state-of-the-art methods, we haven't learned a transformation function to project the features from one camera to the feature space of the other camera, but we have understood the space of feature transformation functions, termed as the feature *warp function space* (WFS) and re-identify targets by learning and classification in this function space of nonlinear warps between features. We have shown that our approach is robust with respect to severe illumination and pose variations by evaluating the performance on five datasets. Comparisons with existing state-of-the-art methods have shown that the proposed approach outperform them.

New Directions

In Chapter 6 we have introduced two novel directions of research.

In the *Re-identification by Classification of Feature Dissimilarities* section we have extended the idea proposed in the Chapter 5. The core novelty of the work was a method that aims to model not the way features are transformed across camera, but to advantage of invariant features and proper distance metrics to model how the distances between such features are transformed across cameras. To achieve this goal we have extracted the feature vectors from a pair of targets viewed in different cameras, then we have computed the distance between such features and used the distances to form the distance feature vector (DFV). Using the positive and negative DFVs we re-identified persons in a supervised classification framework. Comparisons of our approach have been presented using two publicly available benchmark datasets. Those showed that our method, while being simple, it met and outperformed state-of-the-art methods.

In the *Distributed Re-Identification* section we have introduced a novel distributed re-identification framework to extend the single camera-camera re-identification approaches previously proposed. While the proposed person re-identification methods

have achieved high performance on the re-identification task, none of them has considered a wider approach in which the re-identification is extended to the whole network. To address the camera network challenges, by preventing network overloading with useless information and, at the same time, to take into account the topology of the network and tackle the person re-identification problem, we have introduced a camera matching cost hand over measure, then we have proposed a derivation of the distance vector algorithm to perform the re-identification only within a subset of the nodes of the network. Results on benchmark datasets have shown that using the proposed method we could save network and computational resources.

Future Work

Being able to follow and object through all the cameras in the network by achieving a perfect re-identification is a very interesting and challenging task that is far from being solved. However, the approaches proposed in this thesis to address the re-identification challenges can also be extended to other tasks as shown in [98] and [95]. So, our work opens up to a very wide range of applications in numerous other areas. Among of them we'll outline two directions which can lead to future work, one dealing with the transformation of features, the other with its applications.

Feature Transformation

Having features invariant to pose, illumination changes, viewpoint variations, rotations,etc. is the holy grail of computer vision as it can lead to the solution of an enormous variety of problems like recognition, tracking, etc.. Despite the huge effort put by the community, such kind of features have not been discovered yet. But, we introduced two approaches that may help in this process as understanding the transformation of features or the error between those can have a strong impact in this direction. In fact, by understanding the transformation between features we may also try to find a way to make this transformation the identity of the feature space in the sense that the features do not get transformed between cameras.

Similarly, the study on the transformation of the distance between features may be used to find a distance such that it is not affected by the transformation that undergoes between features across two cameras. That is, the distance between a feature and the transformed feature is zero.

Applications

The concept and the ideas that have been discussed in this thesis can be used and applied in many different fields. For instance, the process of extracting local and global features that can bs used to represent a person, can also be used to represent other kind of objects as done in [98] and in [95]. Those are just two examples of such applications, but we can also think about other application fields, like face recognition, etc..

Bibliography

[1] J. Aach and G. M. Church. Aligning gene expression time series with time warping algorithms. *Bioinformatics*, 17(6):495–508, June 2001.

[2] I. Ahmad, Z. He, M. Liao, F. Pereira, and M.-T. Sun. Special Issue on Video Surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(8):1001–1005, Aug. 2008.

[3] L. An, M. Kafai, S. Yang, and B. Bhanu. Reference-Based Person Re-Identification. In *Advanced Video and Signal-Based Surveillance*, 2013.

[4] T. Avraham, I. Gurvich, M. Lindenbaum, and S. Markovitch. Learning Implicit Transfer for Person Re-identification. In *European Conference on Computer Vision, Workshops and Demonstrations*, volume 7583 of *Lecture Notes in Computer Science*, pages 381–390, Florence, Italy, 2012.

[5] S. Bak, E. Corvee, F. Brémond, and M. Thonnat. Person Re-identification Using Haar-based and DCD-based Signature. In *International Conference on Advanced Video and Signal Based Surveillance*, pages 1–8, Boston, MA, Aug. 2010. IEEE.

[6] S. Bak, E. Corvee, F. Bremond, and M. Thonnat. Person Re-identification Using Spatial Covariance Regions of Human Body Parts. In *International Conference on Advanced Video and Signal Based Surveillance*, pages 435–440. IEEE, Aug. 2010.

[7] S. Bak, E. Corvee, F. Brémond, and M. Thonnat. Multiple-shot Human Re-Identification by Mean Riemannian Covariance Grid. In *International Conference on Advanced Video and Signal Based Surveillance*, pages 179–184, Klagenfurt, Austria, 2011.

[8] S. Bak, E. Corvée, F. Brémond, and M. Thonnat. Boosted human re-identification using Riemannian manifolds. *Image and Vision Computing*, 30 (6-7):443–452, June 2012.

[9] D. Baltieri, R. Vezzani, and R. Cucchiara. 3DPeS: 3D People Dataset for Surveillance and Forensics. In *International ACM Workshop on Multimedia access to 3D Human Objects*, pages 59–64, 2011.

[10] M. Bauml, K. Bernardin, M. Fischer, H. K. Ekenel, and R. Stiefelhagen. Multi-pose Face Recognition for Person Retrieval in Camera Networks. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, number i, pages 441–447, Boston, MA, Aug. 2010. IEEE.

[11] L. Bazzani, M. Cristani, A. Perina, M. Farenzena, and V. Murino. Multiple-Shot Person Re-identification by HPE Signature. In *International Conference on Pattern Recognition*, pages 1413–1416, Istanbul , Turkey, Aug. 2010. IEEE.

[12] L. Bazzani, M. Cristani, A. Perina, and V. Murino. Multiple-shot person re-identification by chromatic and epitomic analyses. *Pattern Recognition Letters*, Nov. 2011.

[13] L. Bazzani, M. Cristani, and V. Murino. Symmetry-driven accumulation of local features for human characterization and re-identification. *Computer Vision and Image Understanding*, 117(2):130–144, Nov. 2013.

[14] A. Bellet, A. Habrard, and M. Sebban. A Survey on Metric Learning for Feature Vectors and Structured Data. *ArXiv e-prints*, 2013.

[15] D. J. Bemdt and J. Clifford. Using Dynamic Time Warping to Find Patterns in Time Series. In *Working Notes of the Knowledge Discovery in Databases Workshop*, pages 359–370, 1994.

[16] J. Ben-Arie, P. Pandit, and S. Rajaram. Human activity recognition using multidimensional indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1091–1104, Aug. 2002.

[17] A. Bissacco and S. Soatto. Hybrid Dynamical Models of Human Motion for the Recognition of Human Gaits. *International Journal of Computer Vision*, 85(1): 101–114, May 2009.

[18] W. Blaschof and T. Caelli. Scene understanding by rule evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(11):1284–1288, 1997.

[19] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *International Conference on Image and video retrieval*, pages 401–408, Amsterdam, The Netherlands, 2007. ACM Press.

[20] P. Bottoni, M. D. Marsico, S. Levialdi, G. Ottieri, M. Pierro, and D. Quaresima. A Dynamic Environment for Video Surveillance. *Human-Computer Interaction INTERACT 2009*, 5727:892–895, 2009.

[21] M. Bramberger, A. Doblander, A. Maier, B. Rinner, and H. Schwabach. Distributed Embedded Smart Cameras for Surveillance Applications. *Computer*, 39(2):68–75, Feb. 2006.

[22] L. Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

[23] S. Burigat, L. Chittaro, and S. Gabrielli. Navigation Techniques for Small-screen Devices: an Evaluation on Maps and Web pages. *International Journal of Human-Computer Studies*, 66(2):78–97, Feb. 2008.

[24] E. Chang and Y.-F. Wang. Introduction to the special issue on video surveillance. *Multimedia Systems*, 10(2):116–117, Aug. 2004.

[25] K.-w. Chen, C.-c. Lai, and Y.-p. Hung. An adaptive learning method for target tracking across multiple cameras. *International Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[26] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino. Custom Pictorial Structures for Re-identification. In *Procedings of the British Machine Vision Conference*, pages 68.1–68.11. British Machine Vision Association, 2011.

[27] A. Cockburn, A. Karlson, and B. B. Bederson. A review of overview+detail, zooming, and focus+context interfaces. *ACM Computing Surveys*, 41(1):1–31, Dec. 2008.

[28] N. Colineau, J. Phalip, and A. Lampert. The delivery of multimedia presentations in a graphical user interface environment. In *Proceedings of the 11th international conference on Intelligent user interfaces - IUI '06*, pages 279–282, New York, New York, USA, 2006. ACM Press.

[29] R. T. Collins, A. J. Lipton, and T. Kanade. Introduction to the special section on video surveillance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):745–746, 2000.

[30] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson. A System for Video Surveillance and Monitoring. Technical Report 4, Carnegie Mellon University, Pittsburgh, PA, Dec. 2000.

[31] A. Dantcheva and J.-L. Dugelay. Frontal-to-side face re-identification based on hair, skin and clothes patches. In *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 309–313. IEEE, Aug. 2011.

[32] A. Datta, L. M. Brown, R. Feris, and S. Pankanti. Appearance Modeling for Person Re-Identification using Weighted Brightness Transfer Functions. In *International Conference on Pattern Recognition*, number Icpr, 2012.

[33] H. M. Dee and S. A. Velastin. How Close Are We to Solving the Problem of Automated Visual Surveillance? A Review of Real-world Surveillance, Scientific Progress and Evaluative Mechanisms. *Machine Vision and Applications*, 19(5-6):329–343, May 2007.

[34] B. Dieber, C. Micheloni, and B. Rinner. Resource-Aware Coverage and Task Assignment in Visual Sensor Networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(10):1424–1437, 2011.

[35] B. Dieber, L. Esterle, and B. Rinner. Distributed Resource-aware Task Assignment for Complex Monitoring Scenarios in Visual Sensor Networks. In *International Conference on Distributed Smart Cameras*, pages 1–6, 2012.

[36] M. Dikmen, E. Akbas, T. S. Huang, and N. Ahuja. Pedestrian Recognition with a Learned Metric. In *Asian conference on Computer vision*, pages 501–512, 2010.

[37] G. Doretto, T. Sebastian, P. Tu, and J. Rittscher. Appearance-based person reidentification in camera networks: problem overview and current approaches. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–25, Jan. 2011.

[38] M. Eichner, M. Marin-Jimenez, a. Zisserman, and V. Ferrari. 2D Articulated Human Pose Estimation and Retrieval in (Almost) Unconstrained Still Images. *International Journal of Computer Vision*, 99(2):190–214, Mar. 2012.

[39] Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems. ISO Standard 9241-210:2010, 2010.

[40] A. Ess, B. Leibe, and L. Van Gool. Depth and Appearance for Mobile Scene Analysis. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, Rio De Janeiro, Brazil, Oct. 2007. IEEE.

[41] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *International Conference on Computer Vision and Pattern Recognition*, pages 2360–2367, San Francisco, June 2010. IEEE.

[42] M. Fischer, H. K. Ekenel, and R. Stiefelhagen. Person re-identification in TV series using robust face recognition and user feedback. *Multimedia Tools and Applications*, 55(1):83–104, 2010.

[43] G. Foresti, C. Micheloni, L. Snidaro, P. Remagnino, and T. Ellis. Active video-based surveillance system: the low-level image and video processing techniques needed for implementation. *IEEE Signal Processing Magazine*, 22(2):25–37, Mar. 2005.

[44] G. L. Foresti, C. S. Regazzoni, and R. Visvanathan. Special issue on video communications, processing, and understanding for third generation surveillance systems. *Proceedings of the IEEE*, 89(10):1355–1539, 2001.

[45] G. L. Foresti, C. Micheloni, and C. Piciarelli. Detecting moving people in video streams. *Pattern Recognition Letters*, 26(14):2232–2243, Oct. 2005.

[46] A. C. Gallagher and C. Tsuhan. Clothing cosegmentation for recognizing people. In *International Conference on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, June 2008. IEEE.

[47] R. Garg, S. M. Seitz, D. Ramanan, and N. Snavely. Where's Waldo: Matching people in images of crowds. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1793–1800. IEEE, June 2011.

[48] N. Gheissari, T. Sebastian, and R. Hartley. Person Reidentification Using Spatiotemporal Appearance. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1528–1535. IEEE, 2006.

[49] A. Gilbert and R. Bowden. Tracking objects across cameras by incrementally learning inter-camera colour calibration and patterns of activity. In *European Conference Computer Vision*, 2006.

[50] A. Girgensohn, D. Kimber, J. Vaughan, T. Yang, F. Shipman, T. Turner, E. Rieffel, L. Wilcox, F. Chen, and T. Dunnigan. DOTS: Support for Effective Video Surveillance. In *in Proceedings of the 15th International Conference on Multimedia*, pages 423–432, Augsburg, Germany, Sept. 2007.

[51] A. Girgensohn, F. Shipman, T. Turner, and L. Wilcox. Effects of presenting geographic context on tracking activity between cameras. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*, pages 1167–1176, San Jose, California, USA, 2007. ACM Press.

[52] A. Girgensohn, F. Shipman, and L. Wilcox. Determining activity patterns in retail spaces through video analysis. In *Proceeding of the 16th ACM international conference on Multimedia - MM '08*, pages 889–892, Vancouver, British Columbia, Canada, 2008. ACM Press.

[53] D. Gray and H. Tao. Viewpoint Invariant Pedestrian Recognition with an Ensemble of Localized Features. In *European Conference on Computer Vision*, pages 262–275, Marseille, France, 2008.

[54] D. Gray, S. Brennan, and H. Tao. Evaluating appearance models for recongnition, reacquisition and tracking. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, Rio De Janeiro, Brazil, Oct. 2007.

[55] A. Hampapur, R. Bobbitt, L. Brown, M. Desimone, R. Feris, R. Kjeldsen, M. Lu, C. Mercier, C. Milite, S. Russo, C.-f. Shu, and Y. Zhai. Video Analytics in Urban Environments. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 128–133, Genova, IT, Sept. 2009. IEEE.

[56] J. Han and B. Bhanu. Individual recognition using gait energy image. *IEEE transactions on pattern analysis and machine intelligence*, 28(2):316–22, Feb. 2006.

[57] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(6): 610–621, Nov. 1973.

[58] M. Heikklä and M. Pietikäinen. A texture-based method for modeling the background and detecting moving objects. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):657–62, Apr. 2006.

[59] M. Hirzer, P. M. Roth, and H. Bischof. Person Re-identification by Efficient Impostor-Based Metric Learning. In *Advanced Video and Signal-Based Surveillance*, pages 203–208, 2012.

[60] M. Hirzer, P. M. Roth, K. Martin, and H. Bischof. Relaxed Pairwise Learned Metric for Person Re-identification. In *European Conference Computer Vision*, volume 7577 of *Lecture Notes in Computer Science*, pages 780–793, 2012.

[61] J. Hoey and J. J. Little. Value-directed human behavior analysis from video using partially observable Markov decision processes. *IEEE transactions on pattern analysis and machine intelligence*, 29(7):1118–32, July 2007.

[62] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.

[63] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, and S. Maybank. Principal axis-based correspondence between multiple cameras for people tracking. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):663–71, Apr. 2006.

[64] G. Iannizzotto, C. Costanzo, F. La Rosa, and P. Lanzafame. A multimodal perceptual user interface for video-surveillance environments. In *Proceedings of the 7th international conference on Multimodal interfaces - ICMI '05*, pages 45–52, Trento, 2005. ACM Press.

[65] IBM. IBM Smart Surveillance System. URL http://www.research.ibm.com/peoplevision/.

[66] Ipsotek. Tag and Track, 2011. URL http://www.ipsotek.com/?q=en/news/48.

[67] Y. Ivanov and A. Bobick. Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):852–872, 2000.

[68] Y. Iwashita, R. Baba, K. Ogawara, and R. Kurazume. Person Identification from Spatio-temporal 3D Gait. In *International Conference on Emerging Security Technologies*, pages 30–35, Canterbury, Sept. 2010. Ieee.

[69] O. Javed and K. Shafique. Appearance Modeling for Tracking in Multiple Non-Overlapping Cameras. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 26–33. Ieee, 2005.

[70] O. Javed, K. Shafique, Z. Rasheed, and M. Shah. Modeling inter-camera space-time and appearance relationships for tracking across non-overlapping views. *Computer Vision and Image Understanding*, 109(2):146–162, Feb. 2008.

[71] N. Johnson and D. Hogg. Learning the distribution of object trajectories for event recognition. *Image and Vision Computing*, 14(8):609–615, Aug. 1996.

[72] J. Junqua, J. Haton, and H. Wakita. *Robustness in Automatic Speech Recognition — fundamentals and applications*. Kluwer Academic Publishers, 1995.

[73] J. Kai and M. Arens. View-invariant Person Re-identification with an Implicit Shape Model. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 197–202, Klagenfurt, Austria, 2011.

[74] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics and Image Processing*, 29(3):273–285, 1985.

[75] E. Keogh. Exact indexing of dynamic time warping. In *28th International Conference on Very Large Data Bases*, pages 406–417, Hong Kong, 2002.

[76] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *International Conference on Computer Vision and Pattern Recognition*, number Ldml, pages 2288–2295, 2012.

[77] Z. Kovacs-Vajna. A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1266–1276, 2000.

[78] I. Kviatkovsky, A. Adam, and E. Rivlin. Color Invariants for Person Re-Identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1622–1634, 2013.

[79] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2:2169–2178, 2006.

[80] L. Lee, R. Romano, and G. Stein. Monitoring activities from multiple video streams: establishing a common coordinate frame. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):758–767, Aug. 2000.

[81] T. Leung and J. Malik. Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. *International Journal of Computer Vision*, 43(1):29–44, 2001.

[82] W. Li and X. Wang. Locally Aligned Feature Transforms across Views. In *International Conference on Computer Vision and Pattern Recognition*, 2013.

[83] W. Li, R. Zhao, and X. Wang. Human Reidentification with Transferred Metric Learning. In *Asian Conference on Computer Vision*, pages 31–44, 2012.

[84] Z. Lin and L. S. Davis. Learning Pairwise Dissimilarity Profiles for Appearance Recognition in Visual Surveillance. In *International Symposium on Advances in Visual Computing*, volume 5358 of *Lecture Notes in Computer Science*, pages 23–34, Las Vegas, NV, 2008.

[85] C. Liu, S. Gong, C. C. Loy, and X. Lin. Person Re-identification : What Features Are Important ? In *European Conference on Computer Vision, Workshops and Demonstrations*, pages 391–401, Florence, Italy, 2012. Springer Berlin Heidelberg.

[86] C. Liu, S. Gong, and C. C. Loy. On-the-fly Feature Importance Mining for Person Re-Identification. *Pattern Recognition*, Nov. 2013.

[87] X. Liu, M. Song, Q. Zhao, D. Tao, C. Chen, and J. Bu. Attribute-restricted latent topic model for person re-identification. *Pattern Recognition*, 45(12): 4204–4213, Dec. 2012.

[88] B. Ma, Y. Su, and F. Jurie. BiCov: a novel image representation for person re-identification and face verification. *British Machine Vision Conference*, pages 57.1–57.11, 2012.

[89] B. Ma, Y. Su, and F. Jurie. Local Descriptors Encoded by Fisher Vectors for Person Re-identification. In *European Conference on Computer Vision, Workshops and Demonstrations*, pages 413–422, Florence, Italy, 2012.

[90] L. J. P. V. D. Maaten, E. O. Postma, and H. J. V. D. Herik. Dimensionality Reduction: A Comparative Review. *Journal of Machine Learning Research*, 10 (February):1–41, 2009.

[91] N. Martinel and G. L. Foresti. Multi-signature based person re-identification. *Electronics Letters*, 48(13):765, 2012.

[92] N. Martinel and C. Micheloni. Re-identify people in wide area camera network. In *International Conference on Computer Vision and Pattern Recognition Workshops*, pages 31–36, Providence, RI, June 2012. IEEE.

[93] N. Martinel, C. Micheloni, and C. Piciarelli. Pre-Emptive camera activation for Video Surveillance HCI. In *Intenational Conference on Image Analysis and Processing*, pages 189–198, Ravenna, RA, Sept. 2011.

[94] N. Martinel, C. Micheloni, and C. Piciarelli. Distributed Signature Fusion for Person Re-Identification. In *International conference on Distributed Smart Cameras*, pages 1–6, Hong Kong, Hong Kong, 2012.

[95] N. Martinel, C. Micheloni, and G. L. Foresti. Robust Painting Recognition and Registration for Mobile Augmented Reality. *IEEE Signal Processing Letters*, 20(11):1022–1025, Nov. 2013.

[96] N. Martinel, C. Micheloni, and C. Piciarelli. Learning Pairwise Feature Dissimilarities for Person Re-Identification. In *International conference on Distributed Smart Cameras*, Palm Springs, CA, 2013.

[97] N. Martinel, C. Micheloni, C. Piciarelli, and G. L. Foresti. Camera Selection for AdaptiveHuman–Computer Interface. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pages 1–1, 2013.

[98] N. Martinel, M. Vernier, G. L. Foresti, and E. Lamedica. Image Processing Supports HCI in Museum Application. In *International Conference on Computer Vision Theory and Applications*, Barcellona, Spain, Feb. 2013.

[99] N. Martinel, A. Das, C. Micheloni, and A. K. Roy-Chowdhury. Re-Identification in the Function Space of Feature Warps. *Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[100] R. Mazzon, S. F. Tahir, and A. Cavallaro. Person re-identification in crowd. *Pattern Recognition Letters*, Mar. 2012.

[101] M. McCahill and C. Norris. CCTV in London. Technical report, Centre for Criminology and Criminal Justice, University of Hull, Cottingham Road, HU6 7RX Hull, UK, 2002.

[102] G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia. Event detection and analysis from video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):873–889, 2001.

[103] C. Micheloni, P. Remagnino, H.-L. Eng, and J. Geng. Intelligent Monitoring of Complex Environments. *IEEE Intelligent Systems*, 25(3):12–14, May 2010.

[104] C. Micheloni, B. Rinner, and G. Foresti. Video Analysis in Pan-Tilt-Zoom Camera Networks. *IEEE Signal Processing Magazine*, 27(5):78–90, Sept. 2010.

[105] A. Mignon and F. Jurie. PCCA : A New Approach for Distance Learning from Sparse Pairwise Constraints. In *International Conference on Computer Vision and Pattern Recognition*, pages 2666–2672, 2012.

[106] B. T. Morris and M. M. Trivedi. Contextual Activity Visualization from Long-Term Video Observations. *IEEE Intelligent Systems*, 25(3):50–62, May 2010.

[107] J. Nielsen. Heuristic Evaluation. In J. Nielsen and R. L. Mack, editors, *Usability Inspection Methods*, page 448. John Wiley & Sons, New York, New York, USA, 1 edition, 1994.

[108] J. Nielsen and R. Molich. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI '90*, CHI '90, pages 249–256, New York, New York, USA, 1990. ACM Press.

[109] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, July 2002.

[110] S. Pedagadi, J. Orwell, and S. Velastin. Local Fisher Discriminant Analysis for Pedestrian Re-identification. In *International Conference on Computer Vision and Pattern Recognition*, pages 3318–3325, 2013.

[111] J. Per, V. S. Kenk, R. Mandeljc, M. Kristan, and S. Kovacic. Dana36: A Multi-camera Image Dataset for Object Identification in Surveillance Scenarios. In *International Conference on Advanced Video and Signal-Based Surveillance*, pages 64–69. IEEE, Sept. 2012.

[112] C. Piciarelli and G. L. Foresti. Online Trajectory Clustering for Anomalous Event Detection. *Pattern Recognition Letters*, 27:1835–1842, 2006.

[113] C. Piciarelli, C. Micheloni, and G. L. Foresti. Occlusion-aware multiple camera reconfiguration. In *Proceedings of the Fourth ACM/IEEE International Conference on Distributed Smart Cameras - ICDSC '10*, pages 88–94, Atlanta, GA, USA, 2010. ACM Press.

[114] G. P. Polson, C. Lewis, J. Rieman, and C. Wharton. Cognitive walkthroughs: a method for theory-based evaluation of user interfaces. *International Journal of Man-Machine Studies*, 36(5):741–773, May 1992.

[115] F. Porikli and M. Hill. Inter-Camera Color Calibration Using Cross-Correlation Model Function. In *IEEE International Conference on Image Processing (ICIP)*, pages 133–136, 2003.

[116] F. Porikli, F. Bremond, S. L. Dockstader, J. Ferryman, A. Hoogs, B. C. Lovell, S. Pankanti, B. Rinner, P. Tu, and P. L. Venetianer. Video surveillance: past, present, and now the future. *IEEE Signal Processing Magazine*, 30(3):190–198, May 2013.

[117] B. Prosser, S. Gong, and T. Xiang. Multi-camera Matching using Bi-Directional Cumulative Brightness Transfer Functions. In *British Machine Vision Conference*, Leeds, UK, Sept. 2008.

[118] B. Prosser, S. Gong, and T. Xiang. Multi-camera Matching under Illumination Change Over Time. In *ECCV Workshop on Multi- Camera and Multi-Modal Sensor Fusion Algorithms and Applications*, pages 1–12, 2008.

[119] B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang. Person Re-Identification by Support Vector Ranking. In *Procedings of the British Machine Vision Conference 2010*, pages 21.1–21.11. British Machine Vision Association, 2010.

[120] F. Z. Qureshi and D. Terzopoulos. Planning Ahead for PTZ Camera Assignment and Handoff. In *Third ACM/IEEE International Conference on Distributed Smart Cameras 2009*, pages 1–8, Como, Italy, 2009.

[121] C. Regazzoni, A. Cavallaro, Y. Wu, J. Konrad, and A. Hampapur. Video Analytics for Surveillance: Theory and Practice [From the Guest Editors. *IEEE Signal Processing Magazine*, 27(5):16–17, Sept. 2010.

[122] B. Rinner and W. Wolf. Introduction to Distributed Smart Cameras. *Proceedings of the IEEE*, 96(10):1565–1575, Oct. 2008.

[123] G. Robertson, D. Ebert, S. Eick, D. Keim, and K. Joy. Scale and complexity in visual analytics. *Information Visualization*, 8(4):247–253, Jan. 2009.

[124] G. Robertson, D. Ebert, S. Eick, D. Keim, and K. Joy. Scale and complexity in visual analytics. *Information Visualization*, 8(4):247–253, Jan. 2009.

[125] A. K. Roy-Chowdhury and B. Song. *Camera Networks: The Acquisition and Analysis of Videos over Wide Areas*, volume 3. Jan. 2012.

[126] S. Salvador and P. Chan. FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space. In *KDD Workshop on Mining Temporal and Sequential Data*, 2004.

[127] C. Schmid. Constructing models for content-based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–39–II–45, Montbonnot, France, 2001. IEEE Comput. Soc.

[128] W. R. Schwartz and L. S. Davis. Learning Discriminative Appearance-Based Models Using Partial Least Squares. In *Brazilian Symposium on Computer Graphics and Image Processing*, pages 322–329, Rio De Janeiro, Brazil, Oct. 2009. Ieee.

[129] C.-f. Shu, A. Hampapur, L. Brown, J. Connell, A. Senior, and T. YingLi. IBM smart surveillance system (S3): a open and extensible framework for event based surveillance. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 318–323. IEEE, 2005.

[130] N. T. Siebel and S. J. Maybank. The ADVISOR Visual Surveillance System. In *Workshop Applications of Computer Vision (ACV)*, pages 103–111, Prague, CZ, 2004.

[131] C. Siebler, B. Keni, and R. Stiefelhagen. Adaptive Color Transformation for Person Re-identification in Camera Networks. In *International conference on Distributed Smart Cameras*, number April, pages 199–205, 2010.

[132] Siemens. Siveillance Vantage - a command and control solution for critical infrastructure, 2012. URL http://www.buildingtechnologies.siemens.com/bt/global/en/security-solution/siveillance-vantage-command-control/Pages/siveillance-vantage-command-control.aspx.

[133] Siemens. Siveillance SiteIQ Wide Area - a better way to view automated video surveillance, 2012. URL http://www.buildingtechnologies.siemens.com/bt/global/en/security-solution/intelligent-video-surveillance-analytics/siveillance-site-iq/Pages/siveillance-site-iq-wide-area.aspx.

[134] C. Stauffer and W. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (8):747–757, 2000.

[135] D.-N. Truong Cong, L. Khoudour, C. Achard, C. Meurie, and O. Lezoray. People re-identification by spectral classification of silhouettes. *Signal Processing*, 90 (8):2362–2374, Aug. 2010.

[136] P. H. Tu, G. W. Brooksby, G. Doretto, D. W. Hamilton, N. Krahnstoever, J. B. Laflan, X. Liu, K. A. Patwardhan, T. Sebastian, Y. Tong, J. Tu, F. W. Wheeler, C. M. Wynnyk, Y. Yao, and T. Yu. Video Analytics for Force Protection. In B. Bhanu, C. V. Ravishankar, A. K. Roy-Chowdhury, H. Aghajan, and D. Terzopoulos, editors, *Distributed Video Sensor Networks*, chapter 27, pages 408–425. Springer London, London, 1 edition, 2011.

[137] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea. Machine Recognition of Human Activities: A Survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488, Nov. 2008.

[138] A. Veeraraghavan, A. K. Roy-Chowdhury, and R. Chellappa. Matching shape sequences in video with applications in human movement analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1896–1909, Dec. 2005.

[139] A. Veeraraghavan, A. Srivastava, A. K. Roy-Chowdhury, and R. Chellappa. Rate-invariant recognition of humans and their activities. *IEEE Transactions on Image Processing*, 18(6):1326–1339, June 2009.

[140] S. Velastin. CCTV Video Analytics: Recent Advances and Limitations. In H. Badioze Zaman, P. Robinson, M. Petrou, P. Olivier, H. Schröder, and T. K. Shih, editors, *Visual Informatics: Bridging Research and Practice*, volume 5857 of *Lecture Notes in Computer Science*, pages 22–34. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[141] M. Vernier, N. Martinel, C. Micheloni, and G. L. Foresti. Remote Feature Learning for Mobile Re-Identification. In *International conference on Distributed Smart Cameras*, Palm Springs, CA, 2013.

[142] R. Vezzani, D. Baltieri, and R. Cucchiara. People Re-identification in Surveillance and Forensics: a Survey. *ACM Computing Surveys*, 46(2), 2014.

[143] T. Wada and T. Matsuyama. Multiobject behavior recognition by event driven selective attention method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):873–887, 2000.

[144] J. Wang, M. S. Kankanhalli, W. Yan, and R. Jain. Experiential Sampling for video surveillance. In *First ACM SIGMM international workshop on Video surveillance - IWVS '03*, page 77, New York, New York, USA, 2003. ACM Press.

[145] L. Wang, T. Tan, S. Member, H. Ning, and W. Hu. Silhouette analysis-based gait recognition for human identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1505–1518, Dec. 2003.

[146] X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu. Shape and Appearance Context Modeling. *International Conference on Computer Vision*, pages 1–8, Oct. 2007.

[147] Y. Wu, M. Minoh, M. Mukunoki, W. Li, and S. Lao. Collaborative Sparse Approximation for Multiple-Shot Across-Camera Person Re-identification. In *Advanced Video and Signal-Based Surveillance*, pages 209–214. Ieee, Sept. 2012.

[148] L. Yang and R. Jin. Distance Metric Learning : A Comprehensive Survey. Technical report, Michigan State University, 2006.

[149] M. Yang, L. Zhang, J. Yang, and D. Zhang. Robust sparse coding for face recognition. *International Conference on Computer Vision and Pattern Recognition*, (1):625–632, June 2011.

[150] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1385–1392. IEEE, June 2011.

[151] G. Zhang, Y. Wang, J. Kato, T. Marutani, and M. Kenji. Local distance comparison for multiple-shot people re-identification. In *Asian conference on Computer Vision*, volume 7726 of *Lecture Notes in Computer Science*, pages 677–690, 2013.

[152] R. Zhao, W. Ouyang, and X. Wang. Unsupervised Salience Learning for Person Re-identification. In *International Conference on Computer Vision and Pattern Recognition*, 2013.

[153] W.-S. Zheng, S. Gong, and T. Xiang. Associating Groups of People. In *British Machine Vision Conference*, number 1, pages 1–11, London, UK, Sept. 2009.

[154] W.-S. Zheng, S. Gong, and T. Xiang. Re-identification by Relative Distance Comparison. *IEEE transactions on pattern analysis and machine intelligence*, 35(3):653–668, June 2013.

[155] X. S. Zheng, J. Kiekebosch, and R. Rauschenberger. Attention-aware Human-Machine Interface to Support Video Surveillance Task. In *Human Factors and Ergonomics Society Annual Meeting*, pages 1818–1822, Princeton, NJ, Sept. 2011.