Fast person re-identification based on dissimilarity representations

Riccardo Satta, Giorgio Fumera, Fabio Roli

Dept. of Electrical and Electronic Eng., Univ. of Cagliari
Piazza d'Armi, 09123 Cagliari, Italy

Abstract

Person re-identification is a recently introduced computer vision task that consists of recognising an individual who was previously observed over a video-surveillance camera network. Among the open problems, in this paper we focus on computational complexity. Despite its practical relevance, especially in real-time applications, this issue has been overlooked in the literature so far. In this paper, we address it by exploiting a framework we proposed in a previous work. It allows us to turn any person re-identification method, that uses multiple components and a body part subdivision model, into a dissimilarity-based one. Each individual is represented as a vector of dissimilarity values to a set of visual prototypes, that are drawn from the original non-dissimilarity representation. Experiments on two benchmark datasets provide evidence that a dissimilarity representation provides very fast re-identification methods. We also show that, even if the re-identification accuracy can be lower (especially when the number of candidates is low), the trade-off between processing time and accuracy can nevertheless be advantageous, in real-time application scenarios involving a human operator.

Keywords: person re-identification, computational complexity, processing time, dissimilarity-based representation, multiple instances

Email address: {riccardo.satta,fumera,roli}@diee.unica.it (Riccardo Satta, Giorgio Fumera, Fabio Roli)

URL: http://prag.diee.unica.it (Riccardo Satta, Giorgio Fumera, Fabio Roli)

1. Introduction

Person re-identification consists of recognising an individual who has previously been observed over a camera network. It is a challenging computer vision task that can provide useful tools for many security applications of video-surveillance, e.g., on-line tracking of individuals over different, non-overlapping cameras, and off-line retrieval of the video sequences containing an individual of interest, whose image is given as a query (also called "probe"). Typically, the low resolution of the images, and the variety of possible poses, make face recognition techniques ineffective. A widely used approach is thus to build a descriptor of the global appearance of individuals (*appearance-based* person re-identification). Given the descriptor of an individual of interest (the probe), the task is to rank a set of descriptors (templates) of individuals observed over a camera network, with respect to their similarity to the probe.

Attaining robustness to pose variations, occlusions, and illumination changes is a widely recognised open problem in person re-identification. A common approach to address pose variations and occlusions is to extract multiple local image *instances* (e.g. patches, points of interest) and/or to divide the body into different parts. The issue of illumination changes is often addressed by colour equalization techniques. Computational complexity is another relevant issue, and plays a crucial role in real applications, especially real-time ones.

In a previous work [1], we developed a unifying framework for appearance-based person re-identification, named Multiple Component Matching (MCM). It is based on a multiple instance and multiple part representation, that is commonly used in the literature. MCM can frame, partially or completely, most of the existing methods. It provides a common foundation for the analysis and improvement of such methods, and can guide the development of new ones. In [1] we also implemented a specific re-identification method following MCM, that pursues robustness to illumination changes through a new algorithm based on adding fictitious instances to templates, to simulate such changes.

In this work, we focus on the issue of computational complexity of person re-identification methods. Despite its practical relevance, this issue has been overlooked in the literature so far, and we argue that the processing time of most of the existing methods may be too high for practical applications. In [2] we proposed a general approach for dealing with this issue, as an exten-

sion of the MCM framework. It consists of turning a given appearance-based representation, based on multiple instances and body part subdivision, into a dissimilarity-based one [3], in which an individual is represented as a vector of dissimilarity values to predefined *visual prototypes*. Prototypes themselves are defined as set of instances, that correspond to local appearance characteristics shared by *different* individuals. This allowed us to define the Multiple Component Dissimilarity (MCD) framework, which provides a dissimilarity version of any re-identification method that can be framed in MCM [2]. Dissimilarity-based descriptors are vectors of real numbers. Therefore, they are much more compact than, e.g., bags of multiple feature vectors coming from different interest points. Such compactness drastically reduces their matching time. Accordingly, the dissimilarity-based representation of MCD allows one to implement very fast re-identification methods.

In this paper we extend our previous work of [2] as follows: (i) we improve the MCD-based fast re-identification method of [2]; (ii) we discuss the kind of application where trading a lower re-identification accuracy for a lower processing time can be advantageous, and develop a quantitative model to evaluate it; (iii) we give a thorough experimental evaluation of our MCD-based re-identification method on two benchmark datasets.

The paper is structured as follows. In Sect. 2 we survey previous works on appearance-based person re-identification, and describe our MCM framework, on which MCD is based. In Sect. 3 we summarise the MCD framework, describe the fast person re-identification method based on it, and discuss the trade-off between accuracy and computational complexity. In Sect. 4 we give an extensive experimental evaluation of the proposed method. Final remarks are discussed in Sect. 5, together with future research directions.

2. Background

In this section we provide an overview of the literature on appearance-based person re-identification methods, and describe our MCM framework, which underlies MCD.

2.1. Previous works on person re-identification

Person re-identification is usually modelled as a matching problem, whose goal is to rank templates in a gallery with respect to their similarity to a given probe individual. We focus here on appearance-based methods, namely the ones that rely only on image features, and do not use contextual information.

They represent individuals using descriptors of the clothing appearance, that are built from local or global features [4]. Other approaches include spatio-temporal reasoning to exploit additional spatial and temporal cues (e.g., the spatial layout of the cameras [5]), or other soft biometric traits like gait [6]. Contrary to these approaches, appearance-based methods can usually work even with a single template image per individual.

In [7], the body is subdivided according to its symmetry properties: anti-symmetry separates torso and legs, while symmetry divides left and right parts. The descriptor is made up of three different features: colour histograms in the HSV colour space; *maximally stable colour regions* (MSCR); *recurrent high-structured patches* (RHSP). All local features are extracted from torso and legs separately. To obtain MSCR and RHSP, several patches are sampled at random, mainly near symmetry axes, and then clustered to find the most significant ones.

In [8], a body parts detector is used to find fifteen non-overlapping square cells, that have proven to be "stable regions" of the silhouette. For each cell a covariance descriptor based on colour gradients is computed. Colour histogram equalisation was performed to achieve a better robustness to the varying lighting conditions. Descriptor generation and matching is performed through a pyramid matching kernel.

In [9] two methods are proposed. In the first, Haar-like features are extracted from the whole body, while in the second the body is divided into upper and lower part, each described by the MPEG7 Dominant Colour descriptor. Inter-camera colour calibration was used to address changing lighting conditions. Learning is performed in both methods, respectively to choose the best features and to find the most discriminative appearance model. The training set for each individual consists of different frames (if multiple frames are available) or different viewpoints obtained by sliding a window over the image (if only one frame is available) as positive examples, and of everything which is not the object of interest as negative examples.

An approach based on harvesting SIFT-like interest points from different frames of a video sequence is described in [10]. Different frames are used also in [11], where two methods are proposed. One is based on interest points, that are selected by the Hessian-Affine interest operator. The other exploits a body part subdivision based on decomposable triangulated graphs. Each part is described by features based on colour and shape.

SURF local features are used in [12], where an approach to re-identify individuals as well as different objects is presented.

In [13] several features based on colour (histograms in different colour spaces) and texture (Schmid and Gabor filters), are extracted from randomly taken strips, and their weights are computed by a boosting algorithm.

In [14] global color descriptors (histograms, spatiograms, color/path-length) are computed from the whole body. Changing lighting conditions were addressed by colour histogram normalisation techniques. A graph-based method was used to reduce their dimensionality.

In [15] person re-identification is considered as a relative ranking problem, exploiting a discriminative subspace built through an Ensemble RankSVM. Colour and texture-based features were extracted from six fixed horizontal regions.

Despite their differences, most of the above methods use a *multiple instance* representation, obtained by taking several patches, strips, or interest points, and/or exploit some *body part subdivision*. These two observations provide the foundation for the MCM framework of [1], which is described in Sect. 2.2.

We point out that most of the previous works did not address the issues of processing time and storage requirements. The overall time and memory needed to perform the experiments was reported only in [15]. The general adoption of complex descriptors and matching schemes suggests that most of them are not suitable for application scenarios where a low processing time is required, e.g., real-time re-identification of all the individuals detected in the frames taken by a camera.

2.2. The Multiple Component Matching framework

Our MCM framework provides an unifying view of appearance-based person re-identification methods, embedding the common concepts of multiple instance representation and body part subdivision. For this reason, MCM was used as the underlying paradigm for our dissimilarity framework described in the next section.

In MCM, individuals are represented as bags of instances, named "sets of components". Such components can be any kind of local features, like patches or interest points, extracted from the image of an individual. If a body part subdivision is used, a different set of components is extracted from each part. The rationale behind this representation is to gain robustness to partial occlusions and pose variations.

Formally, let

$$\mathcal{T} = \{\mathbf{T}_1, \ldots, \mathbf{T}_N\} \tag{1}$$

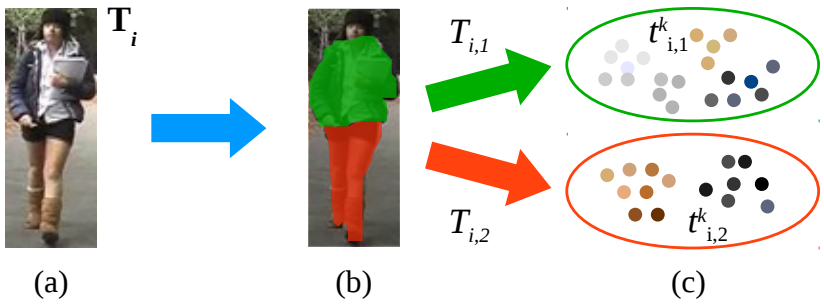

Figure 1: An example of the MCM representation. (a) The image of an individual. (b) The body is subdivided into two parts: upper (in green) and lower body (in red). (c) Each part is composed by a set of components (e.g., image patches), represented here as coloured dots.

be a *template gallery* of N individuals. Each template $\mathbf{T}_i$ is represented via an ordered sequence of M sets, corresponding to M body parts ($M \geq 1$):

$$\mathbf{T}_i = \{T_{i,1}, \ldots, T_{i,M}\}. \tag{2}$$

According to the multiple instance representation, each part $T_{i,m}$ is a set of $n_{i,m}$ of components described by feature vectors $\mathbf{t}^k_{i,m}$:

$$T_{i,m} = \{\mathbf{t}^1_{i,m}, \ldots, \mathbf{t}^{n_{i,m}}_{i,m}\},\ \mathbf{t}^k_{i,m} \in \mathbb{X}, \tag{3}$$

where $\mathbb{X}$ denotes the feature space (without losing generality, we assume that all sets are represented with the same features). An example of the MCM representation is shown in Fig. 1.

Given a probe $\mathbf{Q}$, represented as a sequence of parts as well, the task is to find the most similar template $\mathbf{T}^* \in \mathcal{T}$, according to a similarity measure $D(\cdot,\cdot)$:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}_i} D(\mathbf{T}_i, \mathbf{Q}). \tag{4}$$

Given the above representation, D must be defined as a similarity measure between ordered sequences of sets. In [1] it was defined as a function of similarity measures $d(\cdot,\cdot)$ between M pairs of sets:

$$D(\mathbf{T}_i, \mathbf{Q}) = f\big(d(T_{i,1}, Q_1), \ldots, d(T_{i,M}, Q_M)\big). \tag{5}$$

For instance, D can be defined as a linear combination of the M distances $d(T_{i,1}, Q_1), \ldots, d(T_{i,M}, Q_M)$. The measure d can be in turn any similarity

measure between sets. A suitable one is the k-th *Hausdorff Distance* d_{H} [16], which was used in [1]. Given two sets X and Y, d_H is defined as the k-th ranked distance among the minimum distances between all pairs of elements from X and Y:

$$d_{\mathrm{H}}(X,Y) = \max\left\{h_k(X,Y), h_k(Y,X)\right\}, \tag{6}$$

where

$$h_k(X,Y) = \underset{x\in X}{k\text{-th}} \min_{y\in Y}(\|x-y\|), \tag{7}$$

while $\|\cdot\|$ denotes any distance metric between the elements of the sets. We point out that a proper choice of the parameter k can help attaining robustness to outlying components, that can come from partial occlusions.

Different implementations of MCM can be defined by a specific choice of its parameters, namely, the part subdivision, the components extracted from each part and their representation, and the similarity measures d and D.

In [1], a possible implementation of MCM, named MCMimpl, was proposed. It adopts a two-part subdivision of the body into torso and legs as in [7], and extracts multiple, partly overlapping, rectangular patches from each part, described by colour histograms. MCMimpl inherits robustness to partial occlusions and pose variations from the multiple component/multiple parts representation of MCM. Moreover, it adopts a simple simulation technique to attain robustness to illumination changes, whose effectiveness was experimentally evaluated in [1]. Further details on MCMimpl are given in Sect. 4.1.

3. Fast person re-identification based on dissimilarity representations

In this section, we first summarise the MCD framework of [2]. Then, we present a re-identification method based on MCD, which improves the one of [2]. Finally, we discuss its trade-off between re-identification accuracy and computational complexity.

3.1. Multiple component dissimilarity representation of individuals

In [2] we showed that the MCM framework suggests a general method to reduce the computational complexity of appearance-based re-identification methods. We observed that, regardless of the specific descriptor (i.e., the

body part model and the kind of local components), if the clothing of different individuals share similar local characteristics, their bag-of-instance representation exhibits some redundancy, due to similar local components. Such redundancy can be reduced by turning any bag-of-instance representation into a dissimilarity-based one [3], which consists of representing each individual as a vector of dissimilarity values to pre-defined *visual prototypes*. We proposed to define also prototypes as sets of bag of instances, each corresponding to a local characteristic shared by different individuals.

This allowed us to define the Multiple Component Dissimilarity (MCD) framework. MCD provides a dissimilarity-based version of any re-identification method that can be framed in MCM, adopting a very compact representation of individuals (basically, a small set of dissimilarity vectors). As such, matching two descriptors becomes very fast, as it reduces to compare two vectors of real numbers. Dissimilarity-based descriptors of MCD try to keep the same discriminative capability and robustness of the original re-identification method. In fact, the same part subdivision is used, and local characteristics of the clothings are represented via the same local features, but grouped in homogeneous clusters. The MCD framework is summarised in the rest of this section.

The first step of MCD is prototype construction. Prototypes are constructed from a given gallery of, say, N individuals, starting from their MCM descriptors, denoted as:

$$\mathcal{I} = \{\mathbf{I}_1, \ldots, \mathbf{I}_N\} \ . \tag{8}$$

According to Sect. 2.2, each descriptor $\mathbf{I}_i \in \mathcal{I}$ is an ordered sequence of M elements, one for each body part. The m-th element is a set of n_m feature vectors:

$$\mathbf{I}_i = \{I_{i,1}, \ldots, I_{i,M}\}, \quad I_{i,m} = \{\mathbf{x}_m^1, \ldots, \mathbf{x}_m^{n_m}\}, m = 1, \ldots, M \ . \tag{9}$$

The procedure for prototype construction is the following.

For each body part $m = 1, \ldots, M$:

1. Merge the feature vectors of the m-th part of each $\mathbf{I}_i \in \mathcal{I}$ into a set X_m:

$$X_m = \bigcup_{j=1}^{N} I_{j,m} \ . \tag{10}$$

2. Cluster the set X_m into p_m clusters $\mathbf{P}_m = \{P_{m,1}, \ldots, P_{m,p_m}\}$. Take each cluster as a prototype for the m-th body part.

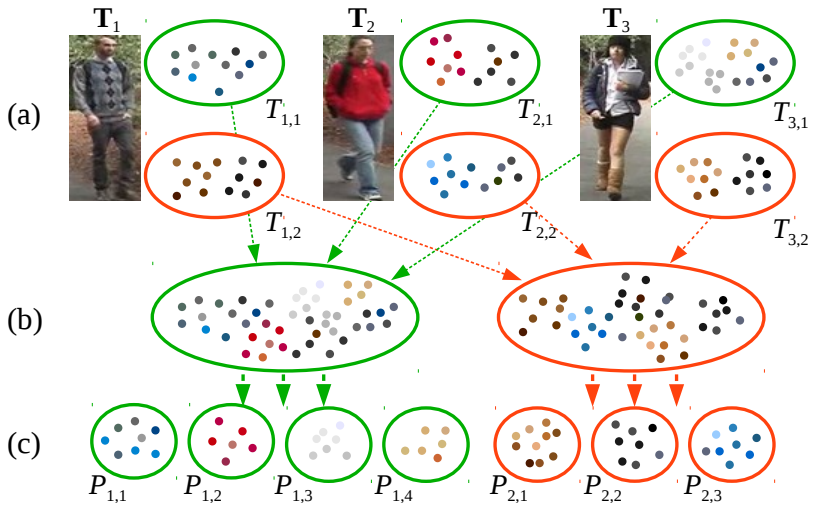

Figure 2: Generation of the prototype gallery in MCD. In this example, the body is subdivided into two parts: upper (in green) and lower body (in red). (a) A template gallery of three individuals, represented according to MCM. (b) All the components of the same part are merged. (c) A clustering algorithm is applied, and a number of prototypes (clusters) are generated for each part.

The above procedure returns M sets of prototypes, one for each body part:

$$\mathcal{P} = \{\mathbf{P}_1, \ldots, \mathbf{P}_M\} \ . \tag{11}$$

Fig. 2 summarises the process of prototypes generation, for $M = 2$ parts.

Once the prototypes have been constructed, given the MCM descriptor of any individual, $\mathbf{I} = \{I_1, \ldots, I_N\}$, its MCD descriptor is obtained (Fig. 3) as a sequence of M dissimilarity vectors $\mathbf{I}^{\mathrm{D}} = \{I_1^{\mathrm{D}}, \ldots, I_M^{\mathrm{D}}\}$, where:

$$I_m^{\mathrm{D}} = \big(d(I_m, P_{m,1}) \ldots d(I_m, P_{m,p_m})\big), m = 1, \ldots, M \ . \tag{12}$$

In principle, any clustering method can be adopted in step 1 of the above procedure. Since prototypes made up by a small number of components may be preferable, to reduce computational and memory requirements, a two-stage clustering procedure can be used: first, the components belonging to each individual are separately clustered; then, a second clustering is carried out on the centroids obtained at the first stage. The number n_m of resulting prototypes for each body part depend on the parameters of the clustering

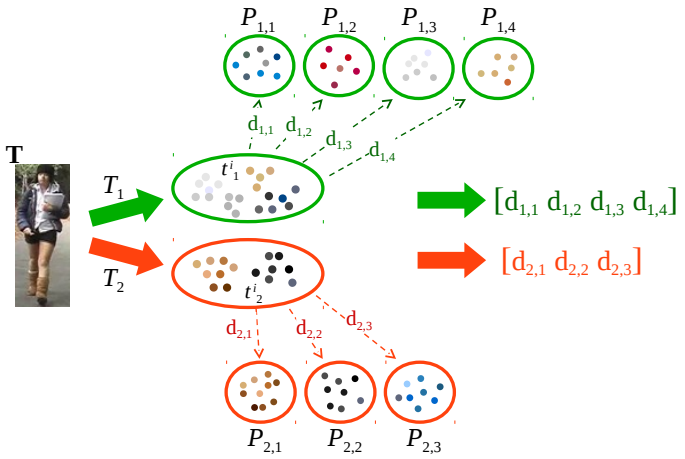

Figure 3: Computation of dissimilarity vectors in MCD. The sets of components that represent each body part of an individual are compared to the prototypes associated to that part. The obtained dissimilarity values will form one dissimilarity vector per part.

algorithm. However, in the experiments of Sect. 4.3 we will show that the choice of such parameters is not crucial.

The advantage of defining prototypes as sets of components, which is the same representation used in MCM for each body part, is that dissimilarities between prototypes and a given individual can be computed using the same distance measure $d(\cdot,\cdot)$ between sets of components adopted by the target MCM-based method (see Eq. (5)). This allows one to easily and directly define a dissimilarity version of any MCM-based method. Simpler dissimilarity measures between descriptors and prototypes can also be defined, for instance considering only the centroid of each prototype.

The proposed dissimilarity-based representation exhibits two clear advantages over complex descriptors used by other methods. One is a considerable reduction in storage requirement: only a small set of dissimilarity vectors (one for each body part) for each individual, and the set of prototypes, need to be stored. The memory required to store the prototypes can be reduced, for example by adopting the two-stage clustering scheme mentioned above, or by considering only the centroids of the corresponding clusters. The other advantage is a reduction in processing time: comparing descriptors becomes as simple as computing distances between vectors of real numbers, which is very fast with modern CPUs. This can enable several useful applications, like real-time re-identification of an individual, among a huge number of

candidates.

We point out that an analogous dissimilarity-based approach was exploited in [17] for object classification tasks. A seemingly analogous representation is used also in *visual words* methods, widely used in scene categorization (e.g., [18]). In these methods, a visual codebook is first built off-line, then the *frequency* (count of the occurrences) of each visual word *inside* each sample is considered. Differently, in the dissimilarity paradigm the *degree* of similarity of each visual prototype to the *whole* sample is considered.

3.2. A fast person re-identification method based on MCD

In this section we develop an improved version of the re-identification method based on MCD proposed in [2].

Let $\mathcal{T}$ be a template gallery of N individuals, and $\mathbf{Q}$ be a probe individual. We denote their MCD-based representation respectively as $\mathcal{T}^{\mathrm{D}} = \{\mathbf{T}_1^{\mathrm{D}}, \ldots, \mathbf{T}_N^{\mathrm{D}}\}$ and $\mathbf{Q}^{\mathrm{D}}$. Note that the same set of prototypes must be used for all the templates and for the probe. In a person re-identification task, the gallery of individuals used to construct the prototypes (see Sect. 3.1) can be either the same template gallery $\mathcal{T}$, a subset of $\mathcal{T}$, or even a different set of individuals, depending on the specific application at hand. The effect of these three choices on the re-identification accuracy will be evaluated in Sect. 4.3.

The problem of re-identifying the individual $\mathbf{Q}^{\mathrm{D}}$ can be formulated similarly to Eq. (4), as:

$$\mathbf{T}^{*\mathrm{D}} = \arg\min_{\mathbf{T}_i^{\mathrm{D}}} D(\mathbf{T}_i^{\mathrm{D}}, \mathbf{Q}^{\mathrm{D}}) \ . \tag{13}$$

The distance measure $D(\cdot,\cdot)$ can be the same as in Eq. (5), while $d(\cdot,\cdot)$ has to be a distance measure between dissimilarity vectors. To highlight this, we will denote it as $d^{\mathrm{D}}(\cdot,\cdot)$. Accordingly, the dissimilarity between any template $\mathbf{T}^{\mathrm{D}}$ and $\mathbf{Q}^{\mathrm{D}}$ is computed as:

$$D(\mathbf{T}^{\mathrm{D}}, \mathbf{Q}^{\mathrm{D}}) = f\left(d^{\mathrm{D}}(T_1^{\mathrm{D}}, Q_1^{\mathrm{D}}), \ldots, d^{\mathrm{D}}(T_M^{\mathrm{D}}, Q_M^{\mathrm{D}})\right) \ . \tag{14}$$

With respect to the method of [2], the improvements are the following. First, we apply to the prototypes the simulation algorithm that was proposed in [1], to attain robustness to illumination changes. Second, we further reduce computational requirements by defining prototypes as the centroid of the corresponding cluster, as suggested in Sect. 3.1.

The third improvement is related to the distance measure $d^{\mathrm{D}}(\cdot,\cdot)$ between dissimilarity vectors. In preliminary experiments, we found that common

distance measures, like the Euclidean, cosine, and cross-correlation distances, work reasonably well. In [2] the Euclidean distance was used. However, none of the above measures properly captures the concept that underlies the proposed dissimilarity representation, that is, each dissimilarity value represents a degree of *presence* (and then, of *relevance*) of the corresponding prototype. Thus, every element of a dissimilarity vector carries a different amount of information in representing the sample of interest. In particular, lower dissimilarity values carry more information than higher values, and thus encode the most relevant characteristics of the sample.

Based on the above arguments, we propose a weighted Euclidean distance between a pair of dissimilarity vectors $\mathbf{x}$ and $\mathbf{y}$ associated to two given objects, where each weight reflects the importance of the corresponding prototype with respect to such objects:

$$d^{\mathrm{D}}(x, y) = \Big(\sum_i \frac{w_i}{W} |x_i - y_i|^2 \Big)^{1/2}, \tag{15}$$

where W is a normalization factor such that $\frac{1}{W} \sum_i w_i = 1$, and the weights w_i are defined as

$$w_i = \tan\Big(\frac{\pi}{2} \overline{w}_i\Big), \tag{16}$$

$$\overline{w}_i = 1 - \min(x_i, y_i), \tag{17}$$

assuming that dissimilarity values x_i and y_i are in the range $[0, 1]$. Accordingly, $\overline{w}_i$ is the minimum dissimilarity of the i-th prototype with respect to both objects. In other words, the higher the relevance (the lower the dissimilarity) of the i-th prototype to at least one of the objects, the higher the $\overline{w}_i$. This guarantees to give higher weights to the prototypes that carry more information about at least one of the objects, while lowering the weight of prototypes that are less important for both of them. The final weights w_i are then computed as the tangent of $\overline{w}_i$, to strongly differentiate relevant prototypes from non-relevant ones.

3.3. Trade-off between re-identification accuracy and matching time

The results reported in Sect. 4, as well as the ones of [2], that were obtained using a previous version of the the dissimilarity-based re-identification method of Sect. 3.2, show that this method exhibits a much lower matching time and memory requirement than its non-dissimilarity-based version, as expected. However, this is sometimes attained at the expense of a lower

re-identification accuracy (a further investigation of this issue is given in Sect. 4.3). In such a case, re-identification methods obtained by our MCD framework are not advantageous over their non-dissimilarity-based counterparts, in application scenarios where a higher accuracy is more important than a lower processing time. For instance, this can be the case of off-line forensics investigations (e.g., looking for an individual of interest among a dataset of videos previously recorded).

However, trading a lower accuracy for a lower processing time can be advantageous in other scenarios. As an example, consider a real time application in which individuals observed by different, non-overlapping cameras are automatically tracked, and a human operator can select an individual of interest from one of the videos, and ask the system to re-identify it (again, in real time). In this case, the template gallery containing the descriptors of all tracked individuals can be automatically constructed and updated in real time. When the operator sends a probe image to the system, it first builds the corresponding descriptor, then matches such descriptor against all the ones in the template gallery, and returns to the operator the list of templates ranked for decreasing similarity with the probe. Finally, the operator scrolls such list to search for the individual of interest (see Fig. 4). Clearly, a re-identification method A with a lower accuracy than another method B results in a higher average search time spent by the operator to find the individual of interest in the ranked list provided by the system (assuming the operator has a 100% accuracy). However, if the higher search time of method A is balanced by a lower processing time, the overall re-identification time between submitting the probe and finding the corresponding individual in the ranked list (namely, the sum of the processing and search times) can be lower for method A than for B. Therefore, in a real time scenario like the one considered above, method A can be preferable to B, although its re-identification accuracy is lower.

In the following we give a simple quantitative model to evaluate the overall re-identification time, that will be used in the experimental evaluation of Sect. 4. Let us denote with t_{d} the average time required to construct the descriptor of the image of an individual, using a given method, and with t_{m} the average matching time between two descriptors. If the template gallery contains the descriptors of N individuals, the average processing time t_{p} is given by the time needed for constructing the probe descriptor plus the time needed to match it to the N template descriptors (as explained above, we assume that template descriptors are constructed during tracking, and are

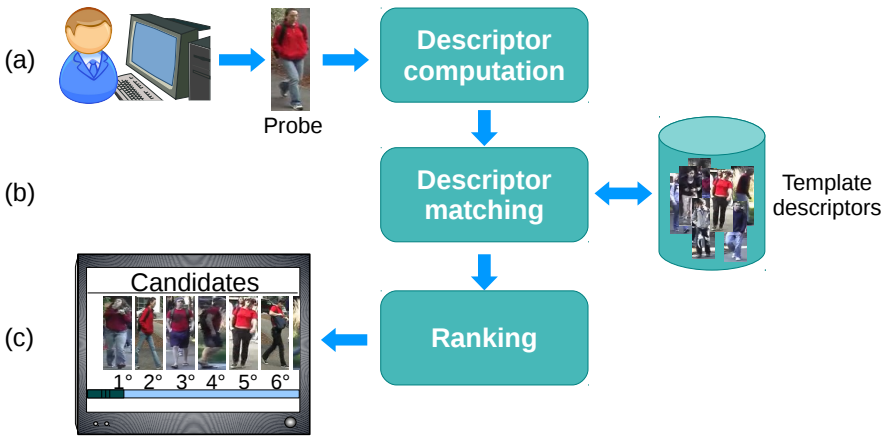

Figure 4: A typical on-line, real-time application scenario of person re-identification. a) The security operator sends a probe image to the system. b) The system builds a descriptor of the probe, and c) matches it against all the descriptors stored in the template gallery, that are constructed on-line. c) The system returns to the operator the list of templates ranked for decreasing similarity to the probe.

thus already available at this time):

$$t_{\mathrm{p}} = t_{\mathrm{d}} + N t_{\mathrm{m}} \ . \tag{18}$$

Let us now denote with t_{c} the average time spent by the operator to compare the probe image with a template image, and with $R \in \{1, \ldots, N\}$ the random variable defined as the rank of the query individual in the list provided by the system.[1] The average search time t_{s} spent by the operator is given by:

$$t_{\mathrm{s}} = t_{\mathrm{c}} E\{R\} \ , \tag{19}$$

where $E\{R\}$ is the expected rank of the probe individual. The value of $E\{R\}$ can be computed from the Cumulative Matching Characteristics (CMC) curve, which is a widely used measure of ranking accuracy of re-identification methods. It is defined as the cumulative distribution of R: $P(R \leq r)$, $r = 1, \ldots, N$, namely the probability that the template image of the query individual is among the top-r ranked images. Using the standard notation

[1] We are assuming that the template gallery always contains the correct match. If this is not the case, for the purposes of this section the domain of R can be extended by adding a value $N + 1$, to denote the absence of the correct match in the template gallery.

$CMC(r)$ for $P(R \leq r)$, it is easy to see that $E\{R\}$ is given by:

$$\begin{aligned} E\{R\} = & \sum_{r=1}^{N} rP(R=r) \\ = & CMC(1) + \sum_{r=2}^{N} r\big(CMC(r) - CMC(r-1)\big) \\ = & N \cdot CMC(N) - \sum_{r=1}^{N-1} CMC(r) \ . \end{aligned} \tag{20}$$

The overall average re-identification time t_{r} can be finally obtained as:

$$\begin{aligned} t_{\mathrm{r}} = t_{\mathrm{p}} + t_{\mathrm{s}} = & \ t_{\mathrm{d}} + Nt_{\mathrm{m}} + \\ & t_{\mathrm{c}} \left[N \cdot CMC(N) - \sum_{r=1}^{N-1} CMC(r) \right] \ . \end{aligned} \tag{21}$$

4. Experimental evaluation

In this section, we give an experimental evaluation of a re-identification method based on our MCD framework, obtained from an existing appearance-based method that can be framed in MCM. To this aim, we chose MCMimpl, which attained a state-of-the-art performance in [1]. We first describe the implementation of its dissimilarity-based version.

4.1. Application of the MCD framework to MCMimpl

We give first a brief description of MCMimpl. First, background and foreground are separated through a STEL generative model [19]. Then the body is divided into $M = 2$ parts, torso and legs, exploiting its anti-symmetry properties. From each part, a set of 80 partly overlapping patches is randomly extracted and represented via HSV colour histograms. To achieve robustness to changing lighting conditions, we added four *simulated* patches, generated from each original one, by varying brightness and contrast (see [1] for details on the simulation algorithm). The distance between two sets of patches X and Y corresponding to the same part is evaluated by the k-th Hausdorff distance of Eq. (6), with $k = 10$. The final matching score of Eq. (5) between a probe $\mathbf{Q}$ and a template $\mathbf{T}$ is computed as the average of the distances between the two parts:

$$D(\mathbf{T}, \mathbf{Q}) = \frac{1}{2}\big(d(T_1, Q_1) + d(T_2, Q_2)\big) \ . \tag{22}$$

To obtain a dissimilarity version of MCMimpl, we followed the procedure described in Sect. 3.1, using the two-stage clustering scheme to obtain the prototype gallery. The Mean-Shift algorithm [20] was used at the first

stage, to separately cluster the components of each individual (excluding the simulated patches), while k-means was applied at the second stage on the resulting centroids. Each prototype was finally associated to a bag containing 1) the patch nearest to each centroid, and 2) the series of simulated patches created from that patch. Note that this procedure is different from that used in [2], where prototypes were associated to *all* the original patches in the corresponding clusters, and the simulated patches were discarded. The new approach attained a better performance in preliminary experiments.

The bandwidth parameter of Mean-Shift, which governs the spread of each cluster, was set to $BW = 0.3$. The number p_m of prototypes for the m-th body part corresponds to the k value of the k-means algorithm, and therefore it must be set in advance. Although this seems a drawback (as in practice it is difficult to guess a suitable value for p_m), we found that the choice of k is not crucial. Mean-Shift, which does not require to define the desired number of clusters beforehand, turned out to produce too unbalanced clusters at the second stage instead, as many of them were composed by only one or two components. An example is given in Fig. 5, which shows the prototypes obtained from ten individuals of the VIPeR dataset (see below). In this example, $k = 8$ was used. Note that some prototypes look quite similar (for example $P_{1,4}$ and $P_{1,7}$, or $P_{2,7}$ and $P_{2,8}$); this means that the chosen number k of clusters is larger than the number of different visual characteristics. However, all the different visual characteristics are reasonably well captured by distinct prototypes. This suggests that any reasonably high value of k should provide a proper prototype gallery.

Concerning the distance measure $d^{\mathrm{D}}(\cdot, \cdot)$ between dissimilarity vectors (Eq. (14)), we adopted the weighted euclidean distance of Eqs. (15)–(17). Finally, the overall matching distance $D(\cdot, \cdot)$ of Eq. (14) was computed similarly to Eq. (22), as the average of the distances of the single parts.

The number and size of the patches mentioned above were shown in [1] to be good choices for MCMimpl. We found however that the performance of MCMimpl$^{\mathrm{Dis}}$ improved by using more patches with a reduced size, thus obtaining an higher granularity that was more effective in capturing visual characteristics. We extracted therefore 300 patches from both torso and legs. The patch width and height were defined as 15% of the width and height of the corresponding part.

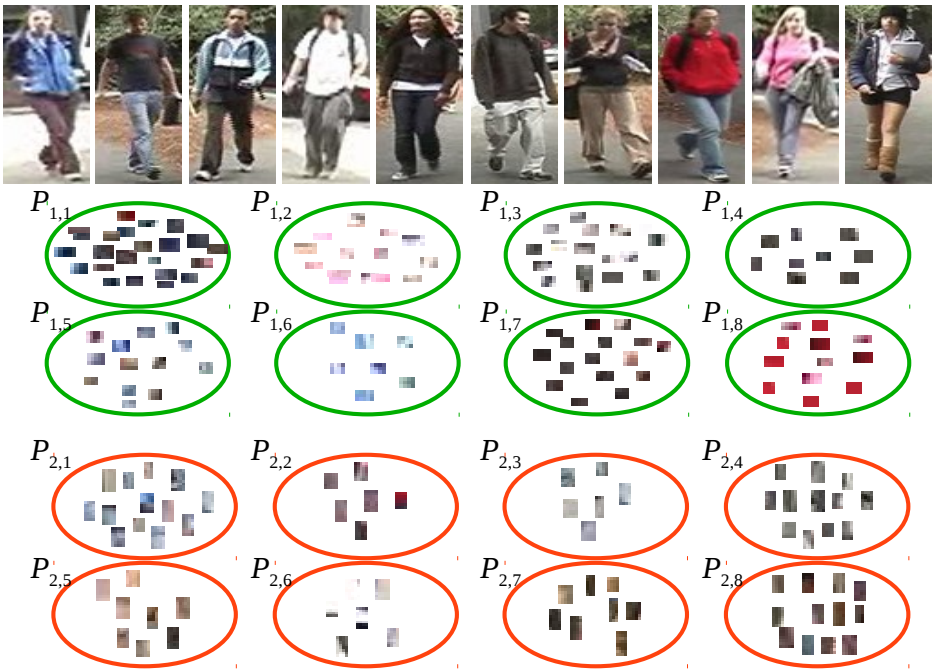

Figure 5: An example of the prototypes obtained from ten individuals of the VIPeR dataset, from two body parts: torso (green) and legs (red). The number of desired prototypes was set to 8 for both parts.

4.2. Experimental setup

Experiments were carried out on two benchmark datasets used in many previous works: VIPeR [21], and a set of images taken from the i-LIDS MCTS video dataset [22].

The VIPeR dataset is made up of two non-overlapping views of 632 different pedestrians, taken from two different cameras, under different poses, viewpoint and lighting conditions. It is the most challenging dataset currently available for person re-identification. The first and second view of each pedestrian were used respectively as the template gallery and the probe gallery. We carried out experiments on three different subdivisions of this dataset. One of them was used in many previous works: the images of the 632 pedestrians are split into ten, partially overlapping folds of 316 individuals, to carry out ten different runs of the experiments. We used the same folds as in [1], which in turn are the same originally used in [7], to obtain comparable results. Since a template gallery of 316 individuals is relatively

small for some real applications, we considered two further subdivisions of VIPeR: ten, partially overlapping folds of 474 individuals randomly sampled from the whole dataset, and a single fold made up of all the 632 available individuals. This also allowed us to evaluated the trade-off between accuracy and re-identification time as a function of the number of templates. We refer to the above three versions of VIPeR respectively as VIPeR-316, VIPeR-474, and VIPeR-632. Note that in VIPeR-632 only one run of the experiments was carried out.

The i-LIDS dataset contains 476 images of 119 different pedestrians, taken at an airport arrival hall from different non-overlapping cameras. We adopted the same experimental setup as [22]: one image for each person was randomly selected to build the template gallery, while the other images formed the probe gallery. We thus obtained a template gallery of 119 images, and a probe gallery of 357 images. The whole procedure was repeated ten times. The folds originally used in [22] were not available, therefore we randomly extracted them.

The folds we used for VIPeR-474 and i-LIDS are available upon request.

The re-identification accuracy was evaluated using the CMC curve defined in Sect. 3.3. The re-identification time was evaluated using the model described in the same section.

4.3. Results

We first evaluated the raw processing time, memory requirements, and re-identification performance of MCMimpl$^{\text{Dis}}$. Then, we evaluated the trade-off between accuracy and computational time on the real-time application scenario depicted in Sect. 3.3. Finally, we assessed the relevance of two critical aspects of our approach, namely, the number of prototypes and the gallery of individuals used to construct them.

4.3.1. Computational requirements and re-identification accuracy

First, we evaluated the processing time and memory requirements of MCMimpl$^{\text{Dis}}$, attained on the same 2.4 GHz CPU used in [1], and compared them with MCMimpl. Results are shown in Table 1. Processing times are averaged over ten runs of the experiments, except for VIPeR-632. The average time for prototype construction is reported for four different sizes of the template gallery, corresponding to the four different datasets considered (i-LIDS, VIPeR-316, VIPeR-474, and VIPeR-632). The average total time required for a single run of the experiments is also reported for each

	MCMimpl	MCMimpl$^{\text{Dis}}$
Avg time for template descriptor creation	93.7 ms$^{(1)}$	17.5 ms
Avg time for probe descriptor creation	6.8 ms$^{(1)}$	17.5 ms
Avg time for prototypes creation, 119 templates	-	2447.3 ms$^{(2)}$
Avg time for prototypes creation, 316 templates	-	6083.2 ms$^{(2)}$
Avg time for prototypes creation, 474 templates	-	12384.8 ms$^{(2)}$
Avg time for prototypes creation, 632 templates	-	16270.7 ms$^{(2)}$
Avg time for dissimilarity vector creation	-	110.3 ms$^{(2)}$
Avg time for a single match	28.6 ms	0.004 ms
Avg total time for a single run (i-LIDS)	2719.1 sec	63.5 sec
Avg total time for a single run (VIPeR-316)	2887.6 sec	87.2 sec
Avg total time for a single run (VIPeR-474)	6521.0 sec	134.5 sec
Avg total time for a single run (VIPeR-632)	11550.6 sec	179.4 sec
Size of the descriptor	96 KB	1.2 KB$^{(2)(3)}$
Size of the prototype gallery	-	48 KB$^{(2)(3)}$

Table 1: Comparison of the computational and memory requirements of MCMimpl and MCMimpl^{Dis}. Notes: (1) in MCMimpl, the construction of a template descriptor includes the generation of simulated patches, and thus requires a higher time than the construction of a probe descriptor; (2) these values refer to 150 prototypes for both the considered body parts (torso and legs); (3) 32 bit floating point values.

dataset: it comprises creation and matching of template and probe descriptors, and also prototypes creation and dissimilarity vectors construction, for MCMimpl$^{\text{Dis}}$.

As expected, MCMimpl$^{\text{Dis}}$ clearly outperforms MCMimpl in terms of processing time and memory usage. In particular, a speed-up of four orders of magnitude is attained for descriptors matching. The average overall time required to perform a run of the experiments is much lower as well, and the difference increases as the size of the template gallery grows.

Regarding re-identification accuracy, the average CMC curves of MCMimpl and MCMimpl$^{\text{Dis}}$ on the four datasets are reported in Fig. 6(a)–(d). MCMimpl$^{\text{Dis}}$ attained a worse recognition performance than MCMimpl on i-LIDS and VIPeR-316, that correspond to the smallest template galleries, respectively 117 and 316 templates. However, the accuracy gap diminished on VIPeR-474, that exhibits a larger template gallery (see Fig. 6(c), and almost vanishes on VIPeR-632, that corresponds to the largest template gallery, and is this in the most challenging and most realistic scenario. This suggests that,

when the number of templates is very high, as in many practical applications, the dissimilarity-based version of a re-identification method obtained through MCD can attain the same performance as the original, not dissimilarity-based method, while requiring much lower computational and storage resources.

The possible reason behind the reduction of the accuracy gap as the template gallery grows, is twofold. On the one hand, an MCM-based approach evaluates the global similarity between one probe and one template, and therefore may give a low relevance to local differences, if any. As the template gallery grows, the likelihood that any template exhibits a high global similarity with other templates, except for local differences, increases, which may reduce re-identification accuracy. On the other hand, an MCD-based approach is in principle capable to capture also small local differences between individuals. In fact, these peculiar visual characteristics should be encoded by different prototypes, and the weighting algorithm of Eqs. ((15))–((17)) should give them a higher weight. This allows them to be detected.

4.3.2. Trade-off between accuracy and processing time

The above results showed that the dissimilarity-based version of a re-identification method can perform worse than the original one. Here we evaluate whether the resulting trade-off between accuracy and processing time can be nevertheless advantageous, in the real-time application scenario described in Sect. 3.3. To this aim, we evaluated the overall re-identification time t_r of MCMimpl and MCMimpl$^{\mathrm{Dis}}$, through Eq. (21). The expected rank of Eq. (19) was computed through the CMC curves of Fig. 6. To evaluate the time t_d required by MCMimpl$^{\mathrm{Dis}}$ for creating one descriptor, we consider both the time needed to build the MCM descriptor, and the time to build the corresponding dissimilarity representation.

For the sake of completeness, we evaluated t_r for all the four datasets: the i-LIDS and VIPeR-316 datasets, where the MCMimpl$^{\mathrm{Dis}}$ attained a lower accuracy than MCMimpl; and the VIPeR-474 and VIPeR-632 datasets, were the accuracy of the two methods was similar. The results are reported in Table 2.

The overall re-identification time is the sum of two quantities, the processing time t_p (i.e., the time required by the system to rank templates in respect to a probe) and the search time t_s (i.e., the time spent by the operator to find the individual in the ranked list of templates). As expected, the processing time of MCMimpl$^{\mathrm{Dis}}$ is lower than the one of MCMimpl. The search time of Eq. (19) is given by t_c (i.e., the average time the operator spends in compar-

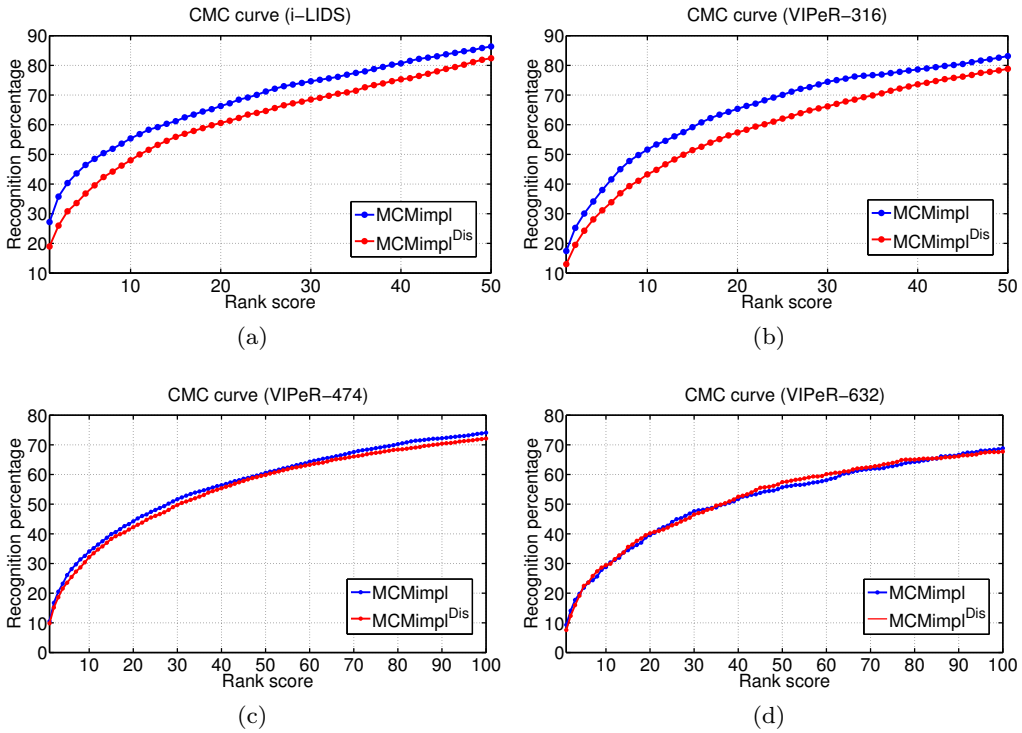

Figure 6: CMC curves attained by MCMimpl and MCMimpl$^{\text{Dis}}$ on the four datasets used in the experiments. Note that in plots (a)–(c) average CMC curves over ten runs of the experiments are reported, while plot (d) refers to a single run. Note also that the range of rank scores (X axis) is $[1, 50]$ in plots (a) and (b), and $[1, 100]$ in plots (c) and (d), since the latter plots correspond to datasets with a larger number of templates.

ing the probe image with one template image) times the expected rank. The latter turned out to be higher for MCMimpl$^{\text{Dis}}$, on i-LIDS and VIPeR-316, due to the lower accuracy. It was slightly higher also for VIPeR-474 and VIPeR-632, although very close to the one of MCMimpl. This means that the overall re-identification time of MCMimpl$^{\text{Dis}}$ will be lower than the one of MCMimpl, for t_c lower than a given value t_c^*, and higher for $t_c > t_c^*$.

Accordingly, we first computed the value of t_c^*. Table 2 shows that the re-identification time of MCMimpl$^{\text{Dis}}$ is lower, if t_c is below about 0.8 seconds for i-LIDS, and 1.3 seconds for VIPeR-316. Since it is likely that in a real-time application scenario like the one considered here t_c is lower than these values, these results show that MCMimpl$^{\text{Dis}}$ can be considered advantageous over MCMimpl, despite the lower accuracy. Note finally that in VIPeR-474

	MCMimpl	MCMimpl$^{\text{Dis}}$
i-LIDS		
Processing time t_{p}	3.497 sec	0.128 sec
Search time t_{s} (with $t_{\text{c}} = 0.5$ sec)	10.103 sec	12.203 sec
Re-identification time t_{r} (with $t_{\text{c}} = 0.5$ sec)	13.600 sec	12.331 sec
t_{c}^*	0.802 sec	
VIPeR-316		
Processing time t_{p}	9.044 sec	0.129 sec
Search time t_{s} (with $t_{\text{c}} = 0.5$ sec)	13.224 sec	16.601 sec
Re-identification time t_{r} (with $t_{\text{c}} = 0.5$ sec)	21.268 sec	16.730 sec
t_{c}^*	1.320 sec	
VIPeR-474		
Processing time t_{p}	13.564 sec	0.129 sec
Search time t_{s} (with $t_{\text{c}} = 0.5$ sec)	42.475 sec	43.084 sec
Re-identification time t_{r} (with $t_{\text{c}} = 0.5$ sec)	56.039 sec	43.213 sec
t_{c}^*	11.021 sec	
VIPeR-632		
Processing time t_{p}	18.082 sec	0.130 sec
Search time t_{s} (with $t_{\text{c}} = 0.5$ sec)	55.941 sec	57.700 sec
Re-identification time t_{r} (with $t_{\text{c}} = 0.5$ sec)	74.023 sec	57.830 sec
t_{c}^*	5.101 sec	

Table 2: Comparison of processing time, search time, and overall re-identification time of MCMimpl versus MCMimpl$^{\text{Dis}}$ (see the text for the details).

and VIPeR-632 t_{c}^* is considerably higher.

We also evaluated the re-identification time t_{r}, for a realistic reference value of $t_{\text{c}} = 0.5$ seconds. It can be seen that t_{r} is always lower for MCMimpl$^{\text{Dis}}$, and the difference with respect to MCMimpl increases as the template gallery size increases.

Finally, we point out that the processing time of MCMimpl$^{\text{Dis}}$, namely the delay between the request of the operator and the response of the system, is almost independent on the template gallery size, and exhibits the very low value of about 0.13 seconds. In contrast, MCMimpl requires a much higher processing time, which grows with the number of templates. This difference

is due to the extremely fast matching attained by MCMimpl$^{\text{Dis}}$. We remark that such high matching speed can be attained by any dissimilarity-based re-identification method based on MCD, as the comparison of dissimilarity vectors is always a fast operation.

To sum up, the above results provide evidence that a dissimilarity-based version of an appearance-based re-identification method can attain an advantageous trade-off between accuracy and processing time.

4.3.3. Effect of changing the source and the number of the prototypes

The processing time of MCMimpl$^{\text{Dis}}$, as well as of any dissimilarity-based method obtained via MCD, is affected by prototype construction. This can be a problem, especially in applications where new templates can be added on-line during system operation. For instance, they can correspond to new individuals that are observed by a camera network. It is thus very interesting to investigate whether the prototype gallery can be constructed using only a subset of the whole template gallery, or even using gallery of individuals *different* than the template gallery. This can avoid to re-build the prototype gallery (and thus, the dissimilarity representation of the existing templates) each time a new template is added to the system. In particular, in the latter case prototypes can be generated off-line, prior to system operation. To this aim, it would be desirable to use a dataset with a wide range of different clothing characteristics. To assess the performance that can be attained when the prototype gallery is built either from a subset of the template gallery, or from a different gallery, we conducted two further experiments: 1) we evaluated the recognition performance in the VIPeR-316 dataset with respect to the percentage of templates used to build the prototype gallery, and 2) we repeated the experiment on i-LIDS (Fig. 6(b)) using VIPeR to construct the prototypes, taking into account that VIPeR exhibits a relatively wider range of clothing characteristics.

Results are reported in Fig. 7(a) and 7(b). They show that re-identification accuracy remains almost the same, 1) if at least 60% of the templates in the original gallery are used to construct prototypes (see Fig. 7(a)), and, most importantly, 2) if prototypes are constructed using a gallery of individuals different from the template gallery (see Fig. 7(b)).

Finally, we evaluated the accuracy and processing time on the the VIPeR-316 dataset, as a function of the number of prototypes per part p.

The accuracy was concisely evaluated as the portion of the area under the CMC curve corresponding to the first 20% of the ranks, denoted as

$AUC_{20\%}$. This is the part of the curve of most interest. The results (shown in Fig. 7(c), 7(d) and 7(e)) provide evidence that the number of prototypes affects performance only slightly.

Fig. 7(e) shows that, as the number of prototypes increases, the $AUC_{20\%}$ initially grows, then reaches a nearly stable value. This behaviour can be easily explained: once the number of prototypes is enough so that the great part of the distinctive visual characteristics have been captured by different clusters, increasing the number of prototypes has mainly the effect of splitting some of the previous clusters into two or more similar ones. Consequently, no further information is embedded in the new prototypes. This effect can be observed in Fig. 5, where some of the prototypes look very similar. On the other hand, Figs. 6(c) and 6(d) show that increasing the number of prototypes slows down both prototype construction and dissimilarity vector computation. Note that all the plots of Fig. 6 correspond to $p = 150$.

5. Conclusions and future work

In this work, we have addressed the open issue of the computational complexity of person re-identification methods, which has been overlooked so far in the literature.

We have exploited the MCD framework previously proposed by the authors, that allows one to turn a given appearance-based re-identification method into a dissimilarity-based one. We have showed that MCD drastically reduces the processing time as well as memory requirements. Also, it can attain a similar accuracy as the original method, especially when the size of the template gallery is high.

Moreover, even if its accuracy is lower, the trade-off attained between accuracy and processing time can be advantageous in terms of the overall re-identification time, in real-time application scenarios. Finally, we have shown that the visual prototypes needed by a dissimilarity-based method can be constructed either using a subset of the template gallery, or even a different gallery, without affecting re-identification accuracy. This is very relevant for real-time applications as well.

We argue that dissimilarity-based representations can be exploited to address another practical problem of current person re-identification methods, namely how to accumulate different images of the same person in a single descriptor, to improve robustness to pose, lighting changes, and partial occlusions. Most of the current methods use only one template image per

person. Instead, since in dissimilarity-based representations each image is represented as a vector of fixed size, different images of the same individual can be used as training samples for a statistical classifier, to learn a more general model of his appearance. This is an interesting research issue we are currently investigating.

References

[1] R. Satta, G. Fumera, F. Roli, M. Cristani, V. Murino, A multiple component matching framework for person re-identification, in: Proceedings of the 16th International Conference on Image Analysis and Processing (ICIAP), Vol. 2, 2011, pp. 140–149.

[2] R. Satta, G. Fumera, F. Roli, Exploiting dissimilarity representations for person re-identification, in: Proceedings of the 1st International Workshop on Similarity-Based Pattern Analysis and Recognition (SIMBAD), 2011, pp. 275–289.

[3] E. Pekalska, R. P. W. Duin, The Dissimilarity Representation for Pattern Recognition: Foundations And Applications (Machine Perception and Artificial Intelligence), World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2005.

[4] G. Doretto, T. Sebastian, P. Tu, J. Rittscher, Appearance-based person reidentification in camera networks: problem overview and current approaches, Journal of Ambient Intelligence and Humanized Computing 2 (2011) 127–151.

[5] T. Gandhi, M. M. Trivedi, Panoramic appearance map (pam) for multi-camera based person re-identification, in: 2006 IEEE International Conference on Video and Signal Based Surveillance (AVSS), 2006, pp. 78–83.

[6] L. Wang, T. Tan, H. Ning, W. Hu, Silhouette analysis-based gait recognition for human identification, IEEE Transactions on Pattern Anallisys and Machine Intelligence 25 (2003) 1505–1518.

[7] M. Farenzena, L. Bazzani, A. Perina, V. Murino, M. Cristani, Person re-identification by symmetry-driven accumulation of local features, in: Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2360–2367.

[8] S. Bak, E. Corvee, F. Bremond, M. Thonnat, Person re-identification using spatial covariance regions of human body parts, in: Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2010, pp. 435–440.

[9] S. Bak, E. Corvee, F. Bremond, M. Thonnat, Person re-identification using haar-based and dcd-based signature, in: Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2010, pp. 1–8.

[10] O. Hamdoun, F. Moutarde, B. Stanciulescu, B. Steux, Interest points harvesting in video sequences for efficient person identification, in: Proceedings of the 8th International Workshop on Visual Surveillance (VS), 2008.

[11] N. Gheissari, T. B. Sebastian, R. Hartley, Person reidentification using spatiotemporal appearance, in: Proceedings of the 2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 2, 2006, pp. 1528–1535.

[12] I. de Oliveira, J. de Souza Pio, Object reidentification in multiple cameras system, in: 4th International Conference on Embedded and Multimedia Computing (EM-Com), 2009, pp. 1–8.

[13] D. Gray, H. Tao, Viewpoint invariant pedestrian recognition with an ensemble of localized features, in: Proceedings of the 10th European Conference on Computer Vision (ECCV), 2008, pp. 262–275.

[14] D. N. Truong Cong, C. Achard, L. Khoudour, L. Douadi, Video sequences association for people re-identification across multiple non-overlapping cameras, in: Proceedings of the 15th International Conference on Image Analysis and Processing (ICIAP), 2009, pp. 179–189.

[15] B. Prosser, W. Zheng, S. Gong, T. Xiang, Person re-identification by support vector ranking, in: Proceedings of the British Machine Vision Conference (BMVC), 2010, pp. 21.1 – 21.10.

[16] J. Wang, J.-D. Zucker, Solving the multiple-instance problem: A lazy learning approach, in: Proceedings of the 17th International Conference on Machine Learning (ICML), 2000, pp. 1119–1125.

[17] A. Carli, U. Castellani, M. Bicego, V. Murino, Dissimilarity-based representation for local parts, in: Proceedings of the 2nd IEEE International Workshop on Cognitive Information Processing (CIP), 2010, pp. 299–303.

[18] J. Yang, Y.-G. Jiang, A. G. Hauptmann, C.-W. Ngo, Evaluating bag-of-visual-words representations in scene classification, in: Proceedings of the international workshop on Workshop on multimedia information retrieval (MIR), 2007, pp. 197–206.

[19] N. Jojic, A. Perina, M. Cristani, V. Murino, B. Frey, Stel component analysis: Modeling spatial correlations in image class structure, Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009) 2044–2051.

[20] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 24 (2002) 603–619.

[21] D. Gray, S. Brennan, H. Tao, Evaluating appearance models for recognition, reacquisition, and tracking, in: Proceedings of the 10th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), 2007.

[22] W.-S. Zheng, S. Gong, T. Xiang, Associating groups of people, in: Proceedings of the British Machine Vision Conference (BMVC), 2009.

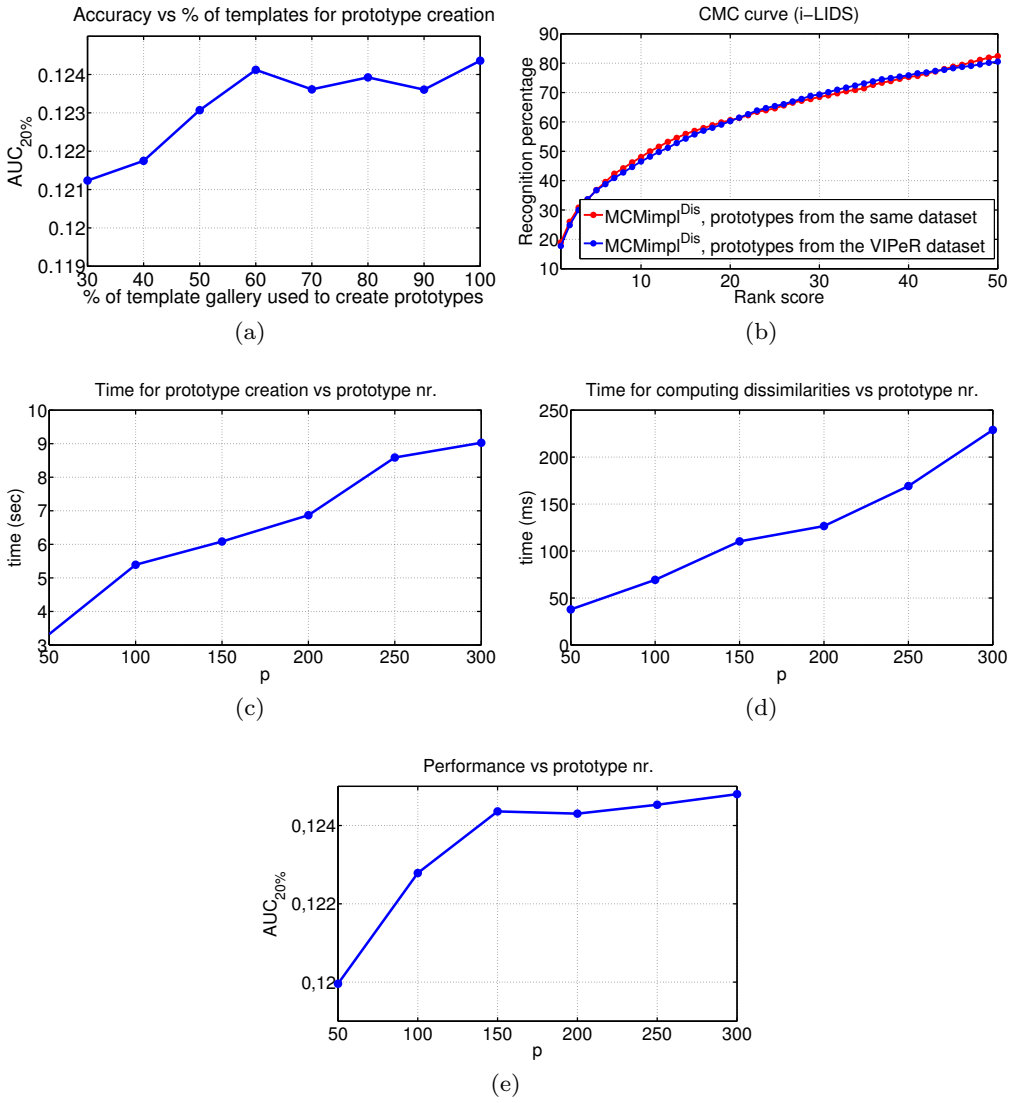

Figure 7: (a) Recognition performance of MCMimpl$^{\text{Dis}}$ on the VIPeR-316 dataset, measured as the $AUC_{20\%}$, versus the percentage of the template gallery used to build prototypes. (b) Comparison between the CMC curves of MCMimpl$^{\text{Dis}}$ on the i-ILIDS dataset, attained by constructing the prototypes using either the same dataset, or the VIPeR dataset. (c) Average time for creating prototypes from a dataset of 316 images, versus the number of prototypes per part, p. (d) Average time for computing dissimilarity vectors for a single individual, versus the number of prototypes per part, p. (e) Recognition performance of MCMimpl$^{\text{Dis}}$ on the VIPeR-316 dataset, measured as the $AUC_{20\%}$, versus the number of prototypes per part, p.