Person re-identification employing 3D scene information

Slawomir Bak, Francois Bremond

Person re-identification employing 3D scene information

Sławomir Bąk, François Brémond

INRIA Sophia Antipolis, STARS team, 2004, route des Lucioles, BP93
06902 Sophia Antipolis Cedex - France

Abstract. This paper addresses the person re-identification task applied in a real-world scenario. Finding people in a network of cameras is challenging due to significant variations in lighting conditions, different colour responses and different camera viewpoints. State of the art algorithms are likely to fail due to serious perspective and pose changes. Most of existing approaches try to cope with all these changes by applying metric learning tools to find a transfer function between a camera pair, while ignoring the body alignment issue. Additionally, this transfer function usually depends on the camera pair and requires labeled training data for each camera. This might be unattainable in a large camera network. In this paper we employ 3D scene information for minimising perspective distortions and estimating the target pose. The estimated pose is further used for splitting a target trajectory into the reliable chunks, each one with a uniform pose. These chunks are matched through a network of cameras using a previously learned metric pool. However, instead of learning transfer functions that cope with all appearance variations, we propose to learn a generic metric pool that only focuses on pose changes. This pool consists of metrics, each one learned to match a specific pair of poses, not being limited to a specific camera pair. Automatically estimated poses determine the proper metric, thus improving matching. We show that metrics learned using only a single camera can significantly improve the matching across the whole camera network, providing a scalable solution. We validated our approach on publicly available datasets demonstrating increase in the re-identification performance.

Keywords: re-identification, pose estimation, metric learning.

Address all correspondence to: Sławomir Bąk, recently moved to Disney Research Pittsburgh, PA, USA
E-mail: slawomir.bak@disneyresearch.com

1 Introduction

Person re-identification is a well known problem in computer vision community. This task requires finding a target appearance in a network of cameras with non-overlapping fields of view. The changes in person pose together with different camera viewpoints and different colour responses make the task of appearance matching extremely difficult.

Current state of the art approaches focus either on *feature-modelling*[1–4] that designs descriptors invariant to camera changes or on *metric learning*[5–11] that uses training data to search for matching strategies minimizing the appearance changes (intra-class variations), while highlighting distinctive properties of the target (maximising inter-class variation).

Feature-modelling approaches concentrate on feature representation which should be invariant to pose and camera changes. These approaches usually assume *a priori* an appearance model, focusing on designing novel features for capturing the most distinctive aspects of an individual. However, designing novel features, we need to look for a trade-off between their discriminative power and invariance through cameras. This task is particularly hard, especially, as this trade-off varies from data to data.[12]

Metric learning approaches are often the one that achieve the best performance in re-identifying people. These approaches learn a distance function that transfers the feature space from one camera to the other such that relevant dimensions are emphasized while irrelevant ones are discarded. Although, this transfer function boosts the recognition accuracy, it is usually camera pair dependent and requires large training data (hundreds of annotated image pairs with the same individual) for each camera pair. This might be unattainable in a large camera network. Moreover, metric learning can lose the performance while directly computing the difference between two images (feature vectors) without aligning them first.

Most of appearance-based approaches are usually evaluated using cropped images that are manually extracted from images taken at eye level (no significant perspective changes). In this case slight pose changes can usually be approached by adopting perceptual principles of symmetry and asymmetry of the human body.[2] The extracted features are then weighted using the idea that features closer to the bodies' axes of symmetry are more robust against scene clutter. However in a real world scenario a person pose and a camera view can change significantly due to serious perspective changes, thus having noticeable impact on recognition performance.

In this paper we address the re-identification in a real camera network, where pose and camera viewpoint change significantly. To improve alignment and minimize perspective changes, we offer

a simple but efficient affine transformation using 3D scene information.

The taxonomy of appearance-based techniques distinguishes the *single-shot* and the *multiple-shot* approaches. The former class extracts appearance using a single image,[3,13,14] while the latter employs multiple images of the same object to obtain a robust representation.[2,4,10,15,16] The main idea of multiple-shot approaches is to take advantage of several frames for extracting a more reliable representation.[15,17,18] Although *multiple-shot* approaches employ several frames for generating the signature, in the end they usually produce a single (averaged) representation ignoring the possible pose and view-angle changes that can occur while tracking the target in a single camera. Those approaches either select more frequent features or blend the appearance by averaging features. Color-soft[19] computes color histograms from segmented head, torso and legs. Color variations across cameras are reduced by employing soft-binning, where each pixel might contribute to several bins based on its proximity to the center of each bin. Multiple frames are incorporated into the model by averaging soft-binned histograms. On the contrary, we propose first to estimate the target pose in each frame and then split the trajectory w.r.t. the estimated pose. Instead of generating a single averaged signature per subject, we propose to compute multiple signatures for each trajectory that reflect multiple appearance of the target as a function of its changing pose.

Given two images/video chunks with subjects and their estimated poses, it is highly desirable to develop the strategy that could exploit pose information for improving the matching accuracy. In our previous work,[20] we proposed to employ an Epanechnikov kernel as a function of orientation that assigns higher weights to features corresponding to the frontal appearance. In this work instead of having a *hand-crafted* weighting strategy, we learn a similarity function that consists of a *pool of metrics*, each one learned to match a specific pair of poses. In this paper we propose to train our model using only a single camera. We believe that this avoids over-fitting the metric to the given

camera pair. Once the metric pool is learned, it can be applied to any camera pair. While matching two images, we select the proper metric based on automatically estimated pose of the target image and of the record image in the gallery (database). The selected metric reflects the transformation of the feature space between two given poses, thus improving matching. The proposed solution showed to be effective and scalable. In summary, this paper makes the following contributions:

- We eliminate perspective distortions by applying a simple affine transformation (rotation) on cropped images containing a person of interest. This transformation is based on 3D scene information (section 3).

- We propose a pose estimation algorithm that uses 3D scene information and motion of the target. Pose is further used for splitting the trajectory into the reliable video chunks, each one with a uniform pose distribution. Those video chunks are used for retrieval that is improved by employing the pose cues (see section 4).

- Finally, instead of learning a single metric that tackles all difficulties related to appearance changes, we focus on learning a *metric pool* that tackles pose changes. This pool consists of metrics, each one learned to match a specific pair of poses, not being limited to a specific camera pair. Automatically estimated poses determine the proper metric, thus improving matching. We show that metrics learned using only a single camera can significantly improve the matching across the whole camera network, providing a scalable solution (see section 5).

The outline of the paper is the following. An overview of our approach is presented in section 2. Sections 3 and 4 focus on target alignment and pose estimation, respectively. Metric learning is detailed in section 5. We validate all steps of our approach in section 6 before discussing perspectives and concluding.

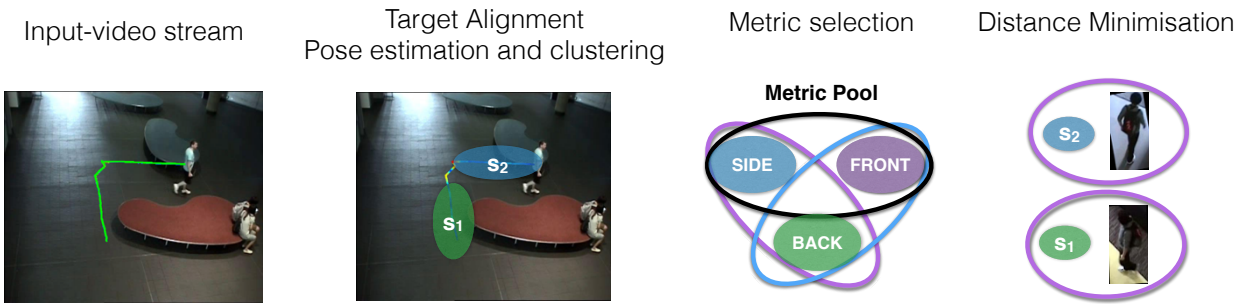


Fig 1: Person re-identification using pose priors. Input: video stream, target detections, trajectory and 3D camera information. Processing: target alignment; pose estimation and pose-driven clustering. Retrieval: distance minimisation using the previously learned metric pool.

2 Overview of the approach

The input of our approach is a video stream with already detected person of interest and its trajectory (see figure. 1). Based on the motion of the target (*i.e*. trajectory) and 3D scene information (*i.e*. camera calibration), in every frame we align person detections (see section 3) and estimate its pose orientation (see section 4). Further, we split the video into the chunks with a uniform pose distribution. We believe that trajectory parts with uniform pose contain reliable information that further can be used for retrieval, *e.g*. in figure 1 we split the trajectory into two with estimated back pose and side pose, generating an appearance representation as a **M**ultiple **S**ignature: $MS = \{S_1, S_2\}$, where S_1 and S_2 correspond respectively to back pose and side pose signatures. Estimated pose of the target signature together with the estimated pose of a matching candidate determine the proper metric for computing the similarity between two video chunks/images. For example for matching side and back pose we will use a different metric than for matching side and front pose. The metric is selected from a metric pool beforehand learned using a single camera (see section 5). This metric pool consists of metrics, each one learned to match a specific pair of poses. Final retrieval is based on minimizing distance among all target signatures and the corresponding signature in the

database.

3 Target alignment

Changing viewpoint in a network of cameras might significantly distort the visual appearance of a person. This problem has a direct impact on the re-identification accuracy. Eliminating perspective distortions in an image region containing the target is often called *image rectification*.[21] Although employing rectification methods gives satisfactory results in pedestrian detection tasks, we observed that the extracted homography between the image and the ground planes can still produce significant distortions in the texture inside the target appearance. As a result, instead of employing rectification, we propose to minimize perspective distortions by rotating the cropped image with the person by angle α. α is extracted using 3D scene information, by mapping the vector orthogonal to the ground plane living in real world coordinates to the vertical of a given image. This mapping is achieved by employing the calibration information of the camera (*i.e.* we employ Tsai calibration.[22]).

We compute rotation angle α in the following way. Given a detected person (a rough bounding box around the person, see figure 2(b)), we select the center point of the detection (point C). Having a pixel location of this point in the image plane (x_c^i, y_c^i), we compute its corresponding point in the world coordinate system (x_c^r, y_c^r) using calibration information and a fixed height of a person $h = 1.7$ m. From this point, we can easily compute the orthogonal to the ground plane in the world coordinate system meeting the head point (x_h^r, y_h^r) that has its corresponding location (x_h^i, y_h^i) in the image plane (point H). The rotation angle can be computed by

$$\alpha = \arctan\left(\frac{x_h^i - x_c^i}{y_c^i - y_h^i}\right). \tag{1}$$

(a) trajectory (b) cropped (c) rotated

Fig 2: Affine transformation of the target image: (a) trajectory of the target (color of the trajectory illustrates the reliability of the detected pose; see section 4.2 for details); (b) the cropped image obtained by the detection algorithm; (c) the rotated image.

Figure 2(c) illustrates the result of the rotation.

4 Pose estimation

This section introduces the method for extracting the pose by using 3D scene information (Tsai calibration[22]) and the motion of the target (section 4.1). Using pose, we split the trajectory into video chunks with a uniform pose (section 4.2) and generate multiple signatures for the trajectory, one signature for each pose.

4.1 Pose orientation

Given detection results for n frames, we compute a set of central points $\mathcal{C}^r = \{(x^r_{c,1}, y^r_{c,1}), \ldots, (x^r_{c,n}, y^r_{c,n})\}$ that lie in the real world coordinates system and correspond to the center of the detections in the image plane. Using calibration information, we extract the position of the camera projected on the ground plane (x^r_{cam}, y^r_{cam}). Let us define the motion vector as a difference between two consecutive real world location coordinates of the target. For each position $k \in [2, n]$ on the trajectory, motion vector $\mathbf{m}_k$ can be expressed as

$$\mathbf{m}_k = \left[x^r_{c,k} - x^r_{c,k-1}, y^r_{c,k} - y^r_{c,k-1}\right], \tag{2}$$

9° 16° 26° 46° 47° 50° 56° 59° 90° 93° 178° 181° 184° 185° 189° 217° 221° 235° 265° 296° 298° 309°

Fig 3: Sample aligned images for different orientations. Estimated θ is provided below each picture.

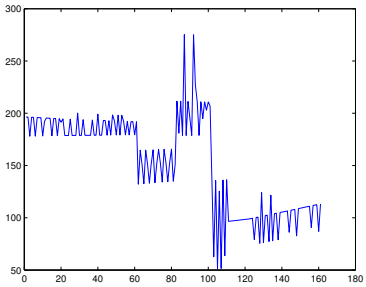
where k is the frame number. We also define view point vector $\mathbf{v}_k$ as a difference between the real world location coordinates of the target in the scene and the real world location coordinates of the camera, both projected on the ground plane

$$\mathbf{v}_k = \left[x^r_{cam} - x^r_{c,k-1}, y^r_{cam} - y^r_{c,k-1}\right]. \tag{3}$$

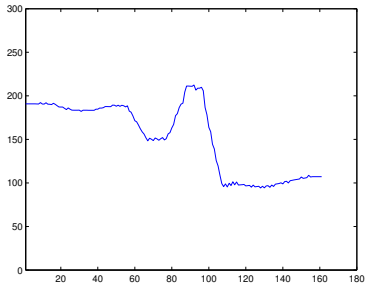
Pose orientation θ_k is computed as a dot product between viewpoint vector $\mathbf{v}_k$ and motion vector $\mathbf{m}_k$.

$$\theta_k = \arccos\left(\frac{\mathbf{v}_k \cdot \mathbf{m}_k}{|\mathbf{v}_k||\mathbf{m}_k|}\right). \tag{4}$$

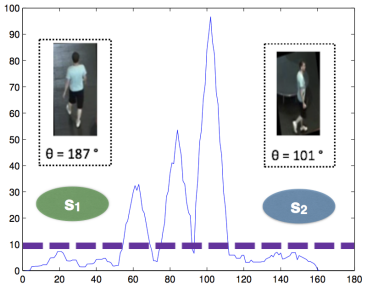
Employing this simple but effective method, we obtain sufficiently accurate information for estimating the pose. This information is used to select the data for training a metric pool for pose changes (see section 5) and then it can be used to select the proper metric while matching different poses. Figure 4(a) presents θ values for the given trajectory (figure 4(d)) and figure 3 shows sample images for different orientation angles.



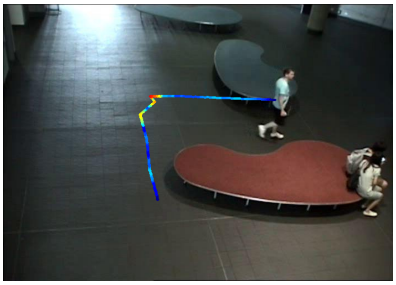
(a) original data: θ



(b) smooth data: θ^s



(c) derivative: δ



(d) orientation reliability

Fig 4: Orientation driven trajectory clustering: (a) original pose orientation θ estimated from the video; (b) the result of the smoothing operation; (c) our control derivative function; (d) the reliability of the trajectory (red color indicates low reliability, while blue stands for the highest).

4.2 Orientation-driven clustering

Figure 4(a) illustrates θ values w.r.t. time. To minimize noise we smooth the data by

$$\theta_k^s = \sum_{l=k-z}^{k+z} \frac{\theta_l}{2z+1}, \tag{5}$$

where z is a smoothing parameter (we set $z = 5$ in experiments). This operation provides us more reliable pose orientation cues (figure 4(b)).

Further, we split the trajectory into the set of chunks (clusters) to obtain multiple appearances of the target with the reliably estimated orientation. We believe that we can detect significant changes in the appearance, by detecting abrupt changes in orientation. We estimate the pose changes using

the control derivative function δ_k defined as

$$\delta_k = \max_{l=k-z...k+z} \frac{d\theta_k^s(t+l)}{dt}, \tag{6}$$

where t reflects frame/time change. δ_k can be seen as a speed of a pose change. We use this function to measure the reliability of the orientation θ. We assume that *peaks* in δ (see figure 4(c)) and their neighbourhoods might provide unreliable information. Figure 4(d) illustrates the trajectory and its reliability. We can notice that the trajectory is unreliable during the pose change (the color red indicates low reliability, while blue – the highest). Frames with estimated unreliable orientation ($\delta > 10$) are removed from the trajectory, determining gaps in the trajectory, thus clustering it into the multiple appearances (see the dashed line in figure 4(c)). Each appearance cluster is labeled with its *mean* orientation (*e.g.* the person in figure 4(d) was separated into two clusters, labeled with orientation 187° and 101°). For each cluster we compute the appearance representation – signature that is equipped with its estimated pose orientation (θ). The next step is to learn the matching strategy that employs pose information.

5 Metric pool

Given two signatures with estimated poses of the subject, we develop the matching strategy that exploits pose information. This strategy consists in generating a pool of metrics, each one learned to match a specific pair of poses. For learning metrics we employ Mahalanobis-like metrics, which have recently attracted a lot of research interest in computer vision.

5.1 Mahalanobis metric

As state of the art we can consider KISS metric,[5] Large Margin Nearest Neighbor Learning (LMNN),[8] Information Theoretic Metric Learning (ITML)[6] and Logistic Discriminant Metric Learning (LDML).[7] These machine learning algorithms learn a distance or similarity metric based on the class of Mahalanobis distance functions. Having data points $x_i, x_j \in \mathbb{R}^d$, we are looking for a similarity function, in which similar data points should be closer than dissimilar points. For training we need a similarity label $y_{ij} : y_{ij} = 1$ for similar pairs, *i.e.* if the samples share the same class label ($y_i = y_j$) and $y_{ij} = 0$ otherwise. Mahalanobis-like metric measures the squared distance between two data points x_i and x_j:

$$d^2_{\mathbf{M}}(x_i, x_j) = (x_i - x_j)^T \mathbf{M}(x_i - x_j), \tag{7}$$

where $\mathbf{M} \geq 0$ is a positive semi-definite matrix. Note that label y_i for x_i is not usually needed for training but the pair-wise relation (label y_{ij}). In the results, training data is given as a set of positive and negative pairs.

5.2 Learning a pose-change metric

Let us assume that the subject's pose can be described by the orientation angle between the motion vector of the target and the viewpoint vector of the camera (angle θ – see section 4). Thus, for each image we have given the pose as the angle in the range of $[0^\circ, 360^\circ)$. We divide this range into n bins, *i.e.* pose $p \in P$, where $|P| = n$. Given n bins of estimated poses, the idea is to learn metrics that stand for transfer functions between pairs of two poses. In the result, the metric pool will consist of $\binom{n}{2}$ metrics, each one learned to match specific pair of poses. Note that bins might

not be necessary continues, *e.g.* images with poses $\theta = 90^\circ$ and $\theta = 270^\circ$ can be assigned to the same bin due to symmetry by vertical flipping.

Learning is performed in the following way. For each pair of poses, we automatically select subjects that support a given pose transfer. In experiments we illustrate that this transfer can be learned using a single camera to avoid including any color transfer. In the result, the learned metric can be applied to uncorrelated camera pair (see section 6).

While learning metrics, we follow a well known scheme based on image pairs, containing two desired poses of the same target. Let us assume that we want to learn the metric for the pose change from pose a to pose b. In this case $y_{ij} = 1$ only if it is the same subject ($y_i = y_j$) and only if it supports the pose change ($p_i = a \wedge p_j = b$), $y_{ij} = 0$ otherwise.

For learning metrics we employ the previously mentioned metric learning tools.[5–8] As the learning is performed offline, the time complexity is not the main concern. Usually metric learning approaches rely on an iterative optimization scheme which can get computationally expensive for large datasets. In contrast to these methods, KISS metric is a non-iterative method that builds on a statistical inference perspective of the space of pairwise differences.[5] In the result, it is orders of magnitudes faster than comparable metric learning approaches. Thus, if the reader is interested in training on a large dataset we recommend the KISS metric.[5] In experiments we compare the performance of all these approaches.

The set of learned metrics stands for the metric pool. As we employ images only from a single camera, the metric pool is not dependent on a camera pair. Thus, once the pool for matching poses is learned, it can be applied to any pair of cameras.

While matching two images given from different (or the same) camera, we first align subjects and estimate their poses. Having poses, we select the corresponding metric from the metric pool.

This metric is used to compute the similarity between given subjects that is used in the final ranking. As the selected metric reflects the transformation of the feature space between two given poses, it improves the recognition accuracy.

6 Experimental Results

In this section we validate our approach on two datasets: VIPER[23] and SAIVT-SOFTBIO.[19] VIPER dataset has particularly been designed for evaluating algorithms handling pose changes and contains a single image per subject per camera. SAIVT-SOFTBIO is a good dataset for evaluating multiple-shot approaches. It provides a highly unconstrained environment reflecting a real-world scenario. We selected this dataset as it allows us to evaluate all steps of our approach and show the performance impact of each step. The results are analysed in terms of recognition rate using a standard re-identification measure that is the cumulative matching characteristic curve (CMC).[23] The CMC curve represents the expectation of finding the correct match in the top n matches.

6.1 Viper dataset[23]

VIPER[23] dataset contains two views of 632 pedestrians. Each image is cropped and scaled to be 128×48 pixels. Images of the same pedestrian have been taken from different cameras, under different viewpoints, poses and lighting conditions. The primary motivation of Gray *et al.*[23] was to propose a dataset which can be used for learning and evaluation of the viewpoint invariant approaches. Thus, the dataset contains pairs which viewpoint angle changes from 45° up to 180°. The quality of the images varies. The video was compressed before processing and as a result, the images have spatially sub-sampled chrominance channels, as well as some minor interlacing and compression artifacts.

In the ground truth for each image we have given the estimated θ and we found that 240 image pairs support transfer from $\theta = 0^\circ$ to $\theta = 90^\circ$. In the result we propose the following evaluation setup.

6.1.1 Appearance model and learning

Select randomly 120 pairs from all 632 pairs for learning metric $\mathbf{M}$ (simulating case of missing pose information) and select randomly 120 pairs from 240 pairs supporting the transfer for learning metric $\mathbf{M}_{pose}$. The remaining 120 pairs from the set of 240 we use as the testing set. From each image we extract a dense grid of rectangular patches with 8×16 size and 8 pixels step in horizontal and vertical direction. Each patch is represented by mean values computed from colour LAB channels and a HOG descriptor. Learning is performed using KISSME framework[5] that provides several metric learning tools; KISS metric (KISSME),[5] Mahalanobis distance with similar pairs (MAHAL), Information Theoretic Metric Learning (ITML)[6] and Large Margin Nearest Neighbor Learning (LMNN).[8] IDENTITY label corresponds to the diagonal metric $\mathbf{M}$ that is the Euclidean distance (L_2 metric). We repeat experiments 10 times to obtain reliable statistics.

6.1.2 Results

Figure 5 illustrates the averaged CMC curves for different metric approaches. Metrics that have been learned on images that support the transfer are denoted by index p. From the results, it is apparent that all metric learning approaches consistently improve the performance while being learned using the transfer data. We can notice that the performance increases when metrics are learned on more specific data. Applying our pose estimation algorithm we can automatically select the training data and learn the metrics for particular pose changes, thus generating the metric

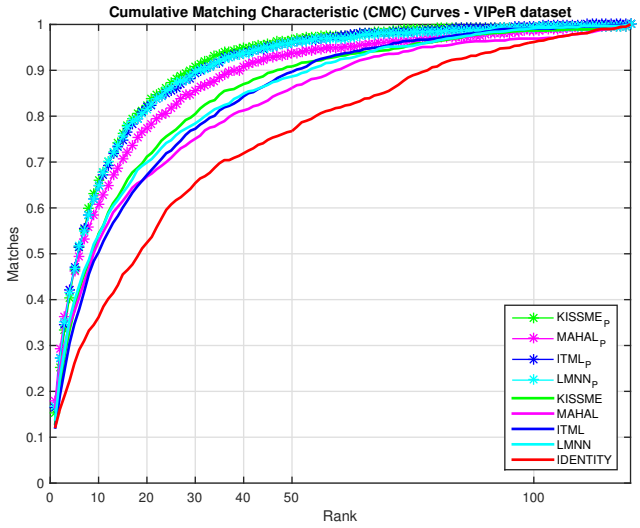


Fig 5: Person re-identification driven by the estimated pose: CMC curves obtained on VIPER dataset. Metric learning approaches with index p correspond to learning using pose orientation (M_{pose}).

pool. In this experiment we were learning a pose transfer using 2 cameras. However, as we have mentioned before, we propose to learn the metric pool using only a single camera, providing a scalable solution. This case is evaluated in the next section as well as all steps of our approach.

6.2 SAIVT-SOFTBIO dataset[19]

This dataset consists of 152 people moving through a network of 8 cameras. Subjects travel in an uncontrolled manner, thus most of subjects appear only in a subset of the camera network. This provides a highly unconstrained environment reflecting a real-world scenario. In average, each subject is registered by 400 frames spanning up to 8 camera views in challenging surveillance conditions (significant illumination, pose and viewpoint changes). Provided annotations given by coarse bounding boxes indicate locations of the subjects in each frame. The centers of these bounding boxes build trajectories of the subjects. Thanks to trajectories and 3D scene information we can evaluate the target alignment and the pose estimation algorithm.

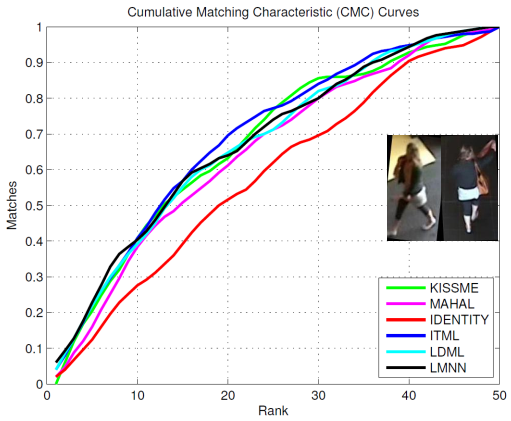
6.2.1 Appearance model and learning

Every cropped and aligned image is scaled into a fixed size window of 128×64 pixels. A set of rectangular patches is produced by shifting 8×16 regions with a 8 pixels step. From each patch, we extract RGB colour and HOG histograms. We minimise colour dissimilarities caused by camera illumination changes by applying *histogram equalization* to each color channel. By this operation we try to avoid a dependency of our metric on the camera colour spectrum.

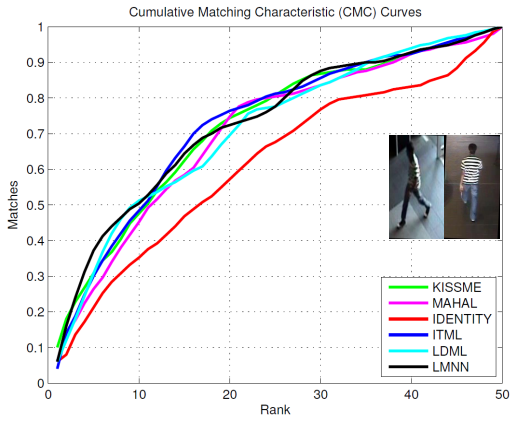
For learning a metric we selected the camera 5. We divide pose orientation into $n = 3$ bins: the front pose, the back pose and the left side pose (right side is flipped to the left). The centers of bins are 0° (front), 180° (back) and 270° (side). The image is classified in one of the poses based on the nearest neighbour strategy. We learn a transfer from side ($\theta = 270^\circ$) to back pose ($\theta = 180^\circ$) that is supported by the sufficient number of subjects (37) and images (279) coming from only camera 5.

By using a single camera we want to avoid including any colour transfer in our metric, thus producing independent to camera pair the metric pool. Training data was obtained using our target alignment and pose estimation algorithms (see section 3).

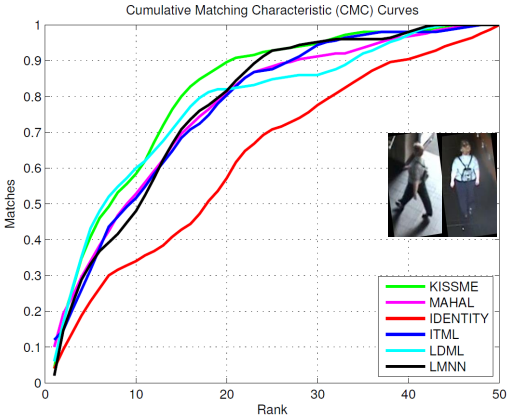
For learning we again used KISSME framework[5] employing KISS metric (KISSME),[5] Mahalanobis distance with similar pairs (MAHAL), Information Theoretic Metric Learning (ITML),[6] Logistic Discriminant Metric Learning (LDML)[7] and Large Margin Nearest Neighbor Learning (LMNN).[8] For testing we randomly select a single image from each camera for each subject that supports the given pose change (single-shot case). All camera pairs are evaluated using 50 subjects. The procedure is repeated 10 times to obtain reliable statistics.



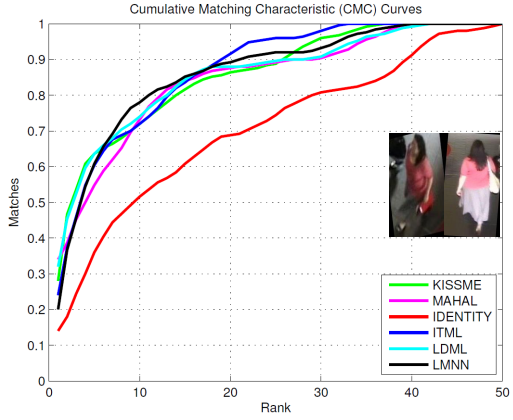
(a) Cam 4 to 3



(b) Cam 4 to 8



(c) Cam 5 to 3



(d) Cam 5 to 8

Fig 6: **Person re-identification by pose change metric**: CMC curves obtained on the SAIVT-SOFTBIO dataset on different camera pairs. The metric for the change from the side pose to the back pose was learned using images from the camera 5.

6.2.2 Metric scalability

Fig. 6 illustrates the results of re-identification on different camera pairs. From the results, it is clear that learning a metric for the specific pose change improves the recognition for all camera pairs. Each metric learning method shows improvement w.r.t. the L_2 metric (IDENTITY). This result is very promising, especially in Fig. 6(a,b) where we can notice the improvement even when the camera pair does not contain the training camera 5.

From Fig. 6(c,d) it seems that the improvement in matching is higher when the testing cam-

era pair contains the camera 5. We believe this is due to the metric dependency on the training data (*e.g.* the selected model is dependent on the features available in the camera 5). In the result, while learning a metric pool we should properly select the training data to obtain sufficiently general metrics. Alternative solution would be to represent the image by such features that are independent on the camera parameters. Unfortunately, more invariant descriptor has usually less discriminative power. Designing a new appearance model, we need to look for a trade-off between its discriminativity and invariance through cameras. This task is particularly hard, especially, as this trade-off varies from data to data.[12]

Finally, for illustrating the difference between learning a metric within a single camera and across different cameras, we set the following experiment. We used the previously learned metric using 37 subjects and 279 images coming from camera 5 and we additionally learned a metric using 40 subjects and images from camera 5 and camera 8. For learning both metrics we used KISS metric (KISSME).[5] We evaluated both metrics matching randomly selected 10 subjects, which were not included in training data. Averaged CMC curves of 10 experiments are illustrated in figure 7. We can notice that learning a metric across different cameras gives better re-identification accuracy. Nevertheless, for training the metric across different cameras we need to annotate the same subjects appearing in different cameras. Note, that training within a single camera can be automatic if cameras are calibrated and a short-term tracker is available.

6.2.3 Full chain evaluation

From the previous results it is apparent that learning a metric for the specific transfer improves the recognition accuracy. In this section we evaluate all steps of our approach illustrating the impact of each step on the recognition performance.

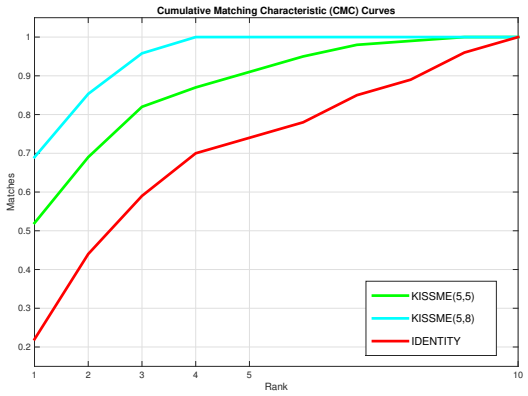
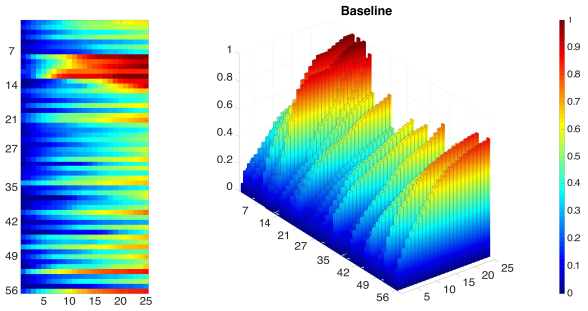
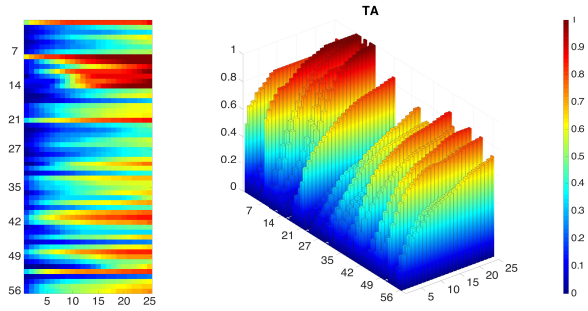


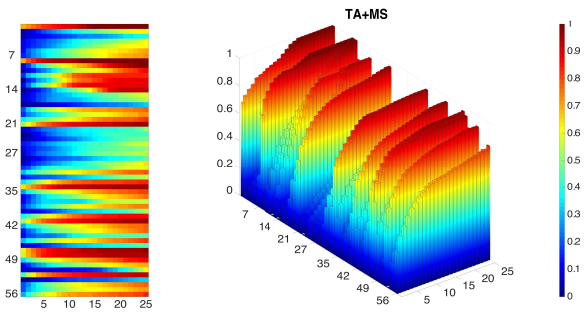
Fig 7: Person re-identification by metrics learned using only images from camera 5 (KISSME(5,5)) and using images from two cameras 5 and 8 (KISSME(5,8)).



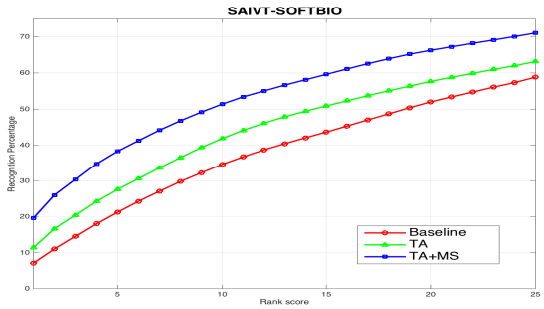
(a) $nAUC = 0.71$



(b) $nAUC = 0.74$



(c) $nAUC = 0.79$



(d)

Fig 8: Evaluation of target alignment and pose clustering on 56 camera pairs of SAIVT-SOFTBIO dataset; (a)-(c) 3D bar charts and their top views illustrating the recognition accuracy as a function of rank and the camera pair; we provide averaged nAUC values that are a weighted areas under CMC curves; (d) presents averaged CMC curves over all 56 pairs of cameras.

First we run experiments on the full SAIVT-SOFTBIO dataset without employing any metric learning approach. As SAIVT-SOFTBIO consists of several cameras, we display the CMC results using 3D bar-charts (see figure 8). The heights of bars illustrates the recognition accuracy as a

function of the camera pair and rank. The results are displayed for 56 camera pairs (*i.e*. having 8 cameras we actually can produce 56 CMC bar series that present recognition accuracy for each camera pair) and first 25 ranks. We also colour the CMC bars with respect to recognition accuracy and display it as a top-view image of 3D bar (left side of each 3D bart chart). In the result we can see that re-identification accuracy might be strongly associated with a particular pair of cameras (similar/non-similar camera view, resolution, the number of registered subjects). For example we can notice high recognition accuracy for rows 7-14 that actually correspond to results of querying camera 2 in which only few subjects were registered (29 of 152), thus high recognition accuracy is due to a small number of subjects. In the rest of cameras the number of subject is more balanced (about 100 subjects per camera). In this figure we illustrate the impact of each step of our algorithm on the re-identification accuracy. BASELINE corresponds to signatures extracted using randomly selected $N = 10$ subsequent frames. Labels TA, MS correspond respectively to the given contributions: **T**arget **A**lignment (section 3) and pose orientation clustering into **M**ultiple **S**ignature (section 4.2). From the results it is clear that each step of the algorithm has a significant impact on the performance. We consistently increase the recognition for all ranks employing target alignment and orientation-driven pose clustering (figure 8(d) illustrates averaged CMC curves over all 56 camera pairs). We can notice that the pose orientate clustering has a higher performance impact than target alignment (*i.e*. for $r = 1$ we can notice an increase of about 9% in the recognition accuracy).

Finally, we show performance of our full framework employing the metric pool. We use the metric learned on camera 5 and compare the performance with Color-soft[19] and our previous approach[20] on matching subjects from camera 5 to camera 8. Figure 9 shows clearly that our approach significantly outperforms state of the art performance.

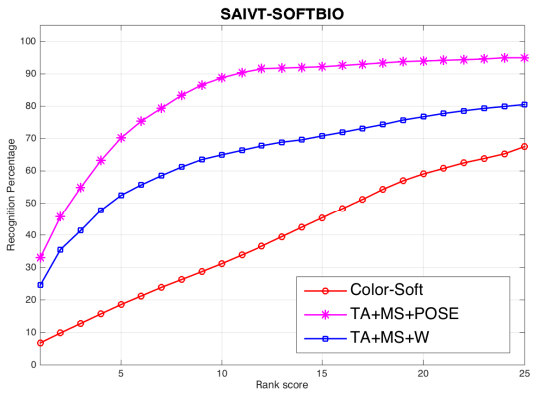


Fig 9: Performance comparison while matching signatures from camera 5 with signatures from camera 8: TA+MS+W[20] and COLOR-SOFT.[19] TA+MS+POSE corresponds to our approach while employing the proposed processing chain: target alignment, pose estimation, pose orientation clustering and the metric pool.

7 Conclusion and perspectives

This paper tackles several issues related to the person re-identification employed in a real world scenario. We proposed to use the motion of the target and 3D information for: (1) eliminating perspective distortions in the target appearance by aligning cropped images w.r.t. the camera view; (2) estimating target poses and splitting the trajectory into video chunks with a uniform pose; (3) learning a general metric pool to match a specific pair of poses. We learned the transfer functions employing Mahalanobis metrics using only a single camera. This allowed us to apply the metric to uncorrelated camera pairs, providing the scalable solution for large camera networks. Experiments on various datasets and various camera viewpoints demonstrated that our method consistently improves the re-identification accuracy. In future, we will further explore the generalization capability of the pose-driven metric pool. Different training schemes will be tested to obtain more general metrics. Additionally, we plan to analyze the correlation between the number of bins in the metric pool and the re-identification accuracy, while employing a finer mapping from the image to the pose using a depth sensor.

Acknowledgments

This work has been supported by PANORAMA, CENTAUR and MOVEMENT European projects.

References

1 S. Bak, E. Corvee, F. Bremond, and M. Thonnat, "Person re-identification using spatial covariance regions of human body parts," in *AVSS*, (2010).

2 M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *CVPR*, (2010).

3 X. Wang, G. Doretto, T. Sebastian, J. Rittscher, and P. Tu, "Shape and appearance context modeling," in *ICCV*, (2007).

4 L. Bazzani, M. Cristani, A. Perina, M. Farenzena, and V. Murino, "Multiple-shot person re-identification by hpe signature," in *ICPR*, (2010).

5 M. Koestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *CVPR*, (2012).

6 J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *ICML*, (2007).

7 M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? metric learning approaches for face identification," in *ICCV*, (2009).

8 K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *NIPS*, (2006).

9 M. Dikmen, E. Akbas, T. S. Huang, and N. Ahuja, "Pedestrian recognition with a learned metric," in *ACCV*, (2010).

10 W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by probabilistic relative distance comparison," in *CVPR*, (2011).

11 W. Li and X. Wang, "Locally aligned feature transforms across views," in *CVPR*, (2013).

12 M. Varma and D. Ray, "Learning the discriminative power-invariance trade-off," in *ICCV*, (2007).

13 D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *ECCV*, (2008).

14 U. Park, A. Jain, I. Kitahara, K. Kogure, and N. Hagita, "Vise: Visual search engine using multiple networked cameras," in *ICPR*, (2006).

15 N. Gheissari, T. B. Sebastian, and R. Hartley, "Person reidentification using spatiotemporal appearance," in *CVPR*, (2006).

16 B. Prosser, W.-S. Zheng, S. Gong, and T. Xiang, "Person re-identification by support vector ranking," in *BMVC*, (2010).

17 S. Bak, G. Charpiat, E. Corvee, F. Bremond, and M. Thonnat, "Learning to match appearances by correlations in a covariance metric space," in *ECCV*, (2012).

18 O. Oreifej, R. Mehran, and M. Shah, "Human identity recognition in aerial images," in *CVPR*, (2010).

19 A. Bialkowski, S. Denman, S. Sridharan, C. Fookes, and P. Lucey, "A database for person re-identification in multi-camera surveillance networks," in *DICTA*, (2012).

20 S. Bak, S. Zaidenberg, B. Boulay, and F. Bremond, "Improving Person Re-identification by Viewpoint Cues," in *AVSS*, (2014).

21 Y. Li, B. Wu, and R. Nevatia, "Human detection by searching in 3d space using camera and scene knowledge," in *ICPR*, (2008).

22 R. Tsai, "An efficient and accurate camera calibration technique for 3d machine vision.," in *CVPR*, (1986).

23 D. Gray, S. Brennan, and H. Tao, "Evaluating Appearance Models for Recognition, Reacquisition, and Tracking," in *PETS*, (2007).

List of figure captions

Figure 1. Person re-identification using pose priors. Input: video stream, target detections, trajectory and 3D camera information. Processing: target alignment; pose estimation and pose-driven clustering. Retrieval: distance minimisation using the previously learned metric pool.

Figure 2. Affine transformation of the target image: (a) trajectory of the target (color of the trajectory illustrates the reliability of the detected pose; see section 4.2 for details); (b) the cropped image obtained by the detection algorithm; (c) the rotated image.

Figure 3. Sample aligned images for different orientations. Estimated θ is provided below each picture.

Figure 4. Orientation driven trajectory clustering: (a) original pose orientation θ estimated from the video; (b) the result of the smoothing operation; (c) our control derivative function; (d) the reliability of the trajectory (red color indicates low reliability, while blue stands for the highest).

Figure 5. Person re-identification driven by the estimated pose: CMC curves obtained on VIPER dataset. Metric learning approaches with index p correspond to learning using pose orientation (M_{pose}).

Figure 6. Person re-identification by pose change metric: CMC curves obtained on the SAIVT-SOFTBIO dataset on different camera pairs. The metric for the change from the side pose to the back pose was learned using images from the camera 5.

Figure 7. Person re-identification by metrics learned using only images from camera 5 (KISSME(5,5)) and using images from two cameras 5 and 8 (KISSME(5,8)).

Figure 8. Evaluation of target alignment and pose clustering on 56 camera pairs of SAIVT-SOFTBIO dataset; (a)-(c) 3D bar charts and their top views illustrating the recognition accuracy as a function of rank and the camera pair; we provide averaged nAUC values that are a weighted areas under CMC curves; (d) presents averaged CMC curves over all 56 pairs of cameras.

Figure 9. Performance comparison while matching signatures from camera 5 with signatures from camera 8: TA+MS+W[20] and COLOR-SOFT.[19] TA+MS+POSE corresponds to our approach while employing the proposed processing chain: target alignment, pose estimation, pose orientation clustering and the metric pool.

Biography

Sławomir Bąk is an Associate Research Scientist at Disney Research Pittsburgh. He was a Research Engineer at the STARS team at INRIA Sophia Antipolis. He received his PhD degree from INRIA, University of Nice in 2012 for a thesis on *person re-identification*. He obtained his Master degree in 2008 at Poznan University of Technology in GRID computing.

François Brémond is a Research Director at INRIA Sophia Antipolis. He created the STARS team on the 1st of January 2012 and was previously the head of the PULSAR INRIA team in September 2009. In 2007 he obtained his HDR degree from Nice University on Scene Understanding: perception, multi-sensor fusion, spatio-temporal reasoning and activity recognition. In 1997 he obtained his PhD degree from INRIA in video understanding.