Proc. Int. Joint Conf. on Neural Networks (IJCNN 2015), Killarney, Ireland, pp. 469-476, IEEE 2015

Evaluation of Multi Feature Fusion at Score-Level for Appearance-based Person Re-Identification

Markus Eisenbach
Ilmenau University of Technology
98684 Ilmenau, Germany
markus.eisenbach@tu-ilmenau.de

Alexander Kolarow
Alexander Vorndran
Ilmenau University of Technology
98684 Ilmenau, Germany

Julia Niebling
Horst-Michael Gross
Ilmenau University of Technology
98684 Ilmenau, Germany

***Abstract*—Robust appearance-based person re-identification can only be achieved by combining multiple diverse features describing the subject. Since individual features perform different, it is not trivial to combine them. Often this problem is bypassed by concatenating all feature vectors and learning a distance metric for the combined feature vector. However, to perform well, metric learning approaches need many training samples which are not available in most real-world applications. In contrast, in our approach we perform score-level fusion to combine the matching scores of different features. To evaluate which score-level fusion techniques perform best for appearance-based person re-identification, we examine several score normalization and feature weighting approaches employing the the widely used and very challenging VIPeR dataset. Experiments show that in fusing a large ensemble of features, the proposed score-level fusion approach outperforms linear metric learning approaches which fuse at feature-level. Furthermore, a combination of linear metric learning and score-level fusion even outperforms the currently best non-linear kernel-based metric learning approaches, regarding both accuracy and computation time.**

I. Introduction and Related Work

In the past years, the need for automated person re-identification has significantly increased. Biometric features like fingerprint or iris are very robust and therefore commonly used to identify a person. Nevertheless, these features require close interaction and are therefore not applicable for frequent or large distance re-identification scenarios, like tracking persons in multiple non overlapping cameras [1] or service robotic applications [2]. In these or similar cases, appearance-based person re-identification can be used to re-identify a person in a set of gallery images. Compensating differences in location, view, resolution, lighting, and pose of persons using non-biometric features, like color, textures, and style of clothing, makes this a very hard problem. Usually only a combination of multiple diverse features performs well on this task. Because the individual re-identification performance is different for each features, the fusion of the features becomes a hard task, too. Often this problem is bypassed by concatenating all feature vectors and learning a distance metric for the combined feature vector (e.g. [3], [4], [5], [6], [7]). This has a great disadvantage since some powerful features (e.g. MSCR [8]) cannot be fused, due to varying feature sizes for different input images. Furthermore, learning an appropriate distance metric on a high dimensional concatenated feature vector requires many samples, which are not available in common datasets. Additionally, combining the features at this level is very computational expensive in the training as well as in the matching phase. These problems can be bypassed by fusing the features at score-level. Score-level fusion has the advantage to perform well and fast on high-dimensional varying feature vectors sizes, even if only few samples are available. Additionally, the feature set can be easily extended, e.g. by biometric features. Therefore, in this paper we will compare different score-level fusion techniques and provide an answer which methods perform best for appearance-based person re-identification. Additionally, we will compare score-level fusion with feature-level fusion techniques (i.e. feature concatenation and distance metric learning) and provide a framework with publicly available source code[1] for further comparison.

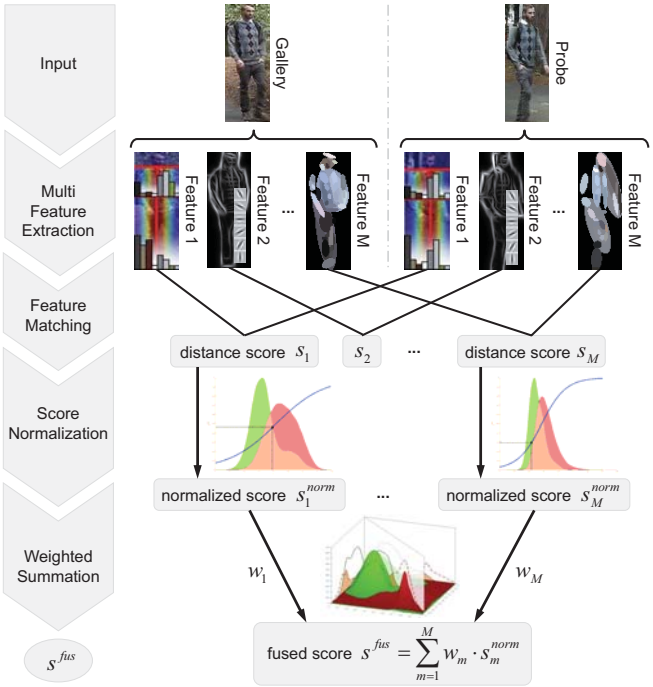


Fig. 1. Workflow for Score-Level-Fusion.

[1] http://www.tu-ilmenau.de/neurob/data-sets-code/score-level-fusion/

II. Score-Level Fusion

Score-level fusion aims to fuse information at an abstract level. Therefore, it combines distance scores from matched feature vectors (see Fig. 1). The goal is to get a fused score that is suitable to calculate a ranking. This is done in three steps: First, the scores for all features are normalized to make them comparable. Additionally, (non-linear) normalization helps to increase the separability between genuine scores (distance scores for image pairs that represent the same person) and impostor scores (distance scores for image pairs showing different persons). The second step is to calculate the weights for each feature. Finally, in the third step, the fused score is calculated as weighted sum. All these steps include only few and simple calculations. Therefore, score-level fusion can be performed much faster than feature-level fusion (e.g. concatenation in combination with metric-learning).

A. Score Normalization

In order to apply score-level fusion, all scores have to be in the same value domain. This is usually achieved by normalizing the value range. Fig. 2 shows a combined systematization of score-level fusion approaches, as described in Maltoni *et al.* [9], Ross *et al.* [10], and Ulery *et al.* [11]. These methods can be categorized in three overall categories: Density-based, Transformation-based, and Classifier-based. The last one can only be used for verification (not identification), since it does not calculate a fused score, but formulates the fusion as a binary decision problem. Therefore, it is not closer examined. In the following, we will describe approaches for performing normalization based on probability density functions or transformations.

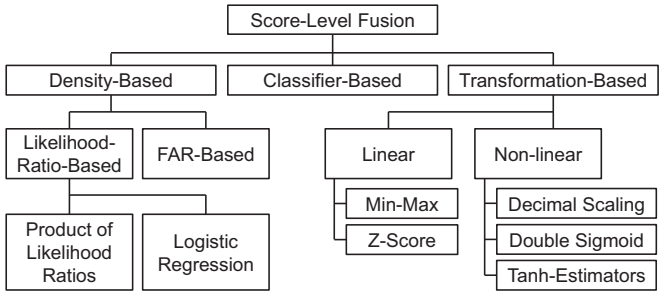


Fig. 2. Categorization of Score-Level Fusion approaches.

Density-based approaches model the genuine (gen) and impostor (imp) score distributions (see Fig. 3). The fused score is defined as the probability of observing a genuine score $s^{fus} = P(gen|\mathbf{s})$ using joint scores $\mathbf{s} = s_1 \dots s_M$ for M features and given score distributions. Using the Bayes theorem, an "a posteriori probability can be expressed in terms of the joint probability densities" [9], as follows

$$P(gen|\mathbf{s}) = \frac{P(\mathbf{s}|gen)P(gen)}{P(\mathbf{s}|gen)P(gen) + P(\mathbf{s}|imp)P(imp)}. \quad (1)$$

Assuming $P(gen)$ and $P(imp)$ are equal, Eq. 1 can be simplified to

$$P(gen|\mathbf{s}) = \frac{P(\mathbf{s}|gen)}{P(\mathbf{s}|gen) + P(\mathbf{s}|imp)}, \quad (2)$$

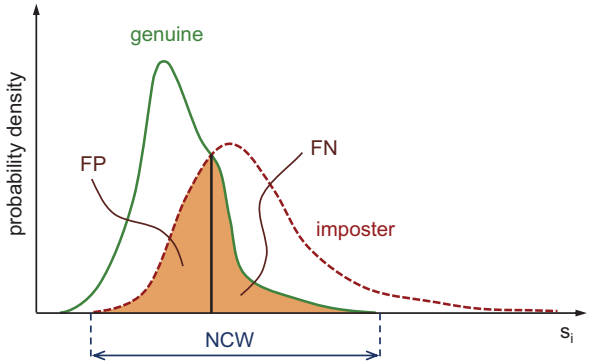


Fig. 3. Exemplary genuine impostor score distribution (MSCR feature [8] on VIPeR dataset [12]). Scores in highlighted area will produce errors (false positives (FP) and false negatives (FN)) when threshold is chosen at the intersection point of genuine and impostor scores as marked. The Non-Confidence Width (NCW) measures the width of this critical overlap area.

where the remaining probabilities $P(\mathbf{s}|gen)$ and $P(\mathbf{s}|imp)$ can be determined from the modeled distributions.

Modeling joint probability distributions with only few training samples is nearly impossible in a high-dimensional space. Therefore, the joint distributions are usually approximated as product of its M marginals

$$P(\mathbf{s}|k) = \prod_{m=1}^{M} P(s_m|k), \quad (3)$$

where k is either gen or imp. This approximation assumes statistical independence of the features. This is not the case in our scenario, since all features are extracted from the same images and belong to the same objects. However, evaluations of Nandakumar *et al.* [13] showed that correlation of features "does not adversely affect the performance of the LR fusion scheme, especially when the individuals matchers are accurate and the difference between genuine and impostor correlation is not high." We verified, that the latter condition holds for appearance-based features. Also the first condition is true for most used features. Only the matching accuracy for some texture features may cause problems.

Using this simplification, the Likelihood Ratio normalization is done by modeling the marginal genuine and impostor score distributions for each feature separately. Modeling is done by Kernel Density Estimation (KDE) with variable bandwidth kernels as in [11]. The density function is therefore

$$P(s_i|k) = \frac{1}{\sqrt{2\pi} \cdot h_{\mathbf{s}^{(k)}}} \cdot \frac{1}{N} \cdot \sum_{j=1}^{N} \exp\left(\frac{(s_i^{(k)} - s_j^{(k)})^2}{2h_{\mathbf{s}^{(k)}}^2}\right), \quad (4)$$

with k as gen or imp, N the number of genuine/impostor samples in the training set, training samples $\mathbf{s}^{(k)} = s_1^{(k)} \dots s_N^{(k)}$, s_i a sample of test set, and the variable bandwidth h of the kernel is chosen by the formula of Silverman [14]

$$h_{\mathbf{s}^{(k)}} = \sigma_{\mathbf{s}^{(k)}} \left(\frac{4}{3}\right)^{\frac{1}{5}} N_{\mathbf{s}^{(k)}}^{-\frac{1}{5}} w(\mathbf{s}^{(k)}), \quad (5)$$

where σ is the distribution's standard deviation, and $w \geq 1$ is chosen such that the kernels' width increases in the boundary

areas of the distribution. Optimization criteria for w are smoothness of both distributions and a monotonic decreasing likelihood ratio. The resulting Likelihood Ratio normalization rule is

$$s_i^{norm_{LR}} = P(gen|s_i) = \frac{P(s_i|gen)}{P(s_i|gen) + P(s_i|imp)}. \quad (6)$$

Since frequent KDE calculations (Eq. 4) are very time consuming and contrary to our real-time condition, the transformation is calculated only once on the training data set and stored as lookup table for the execution phase.

The biggest problem of the Likelihood Ratio method is the need for modeling the genuine distribution accurately, since genuine training samples (image-pairs of same persons under varying environmental conditions) are rare in most datasets, as well as in real-world applications. Logistic Regression bypasses this problem. It does not try to accurately model the distributions, but models the ratio of genuine and impostor distributions instead. Therefore, a rough approximation of both distributions (KDE with fixed bandwidth) is calculated to estimate the logarithmic ratio of genuine and impostor score distribution (see Fig. 4 top). In a trust region, where

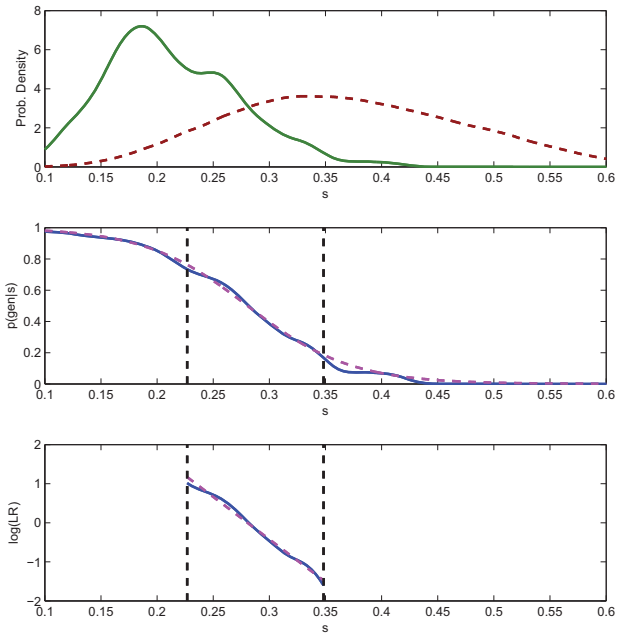


Fig. 4. Normalization by Logistic Regression. Top: Approximated genuine (solid green) and impostor (dashed red) plots. Center: Probability $P(gen|s_i)$ modeled by ratio of distributions (solid blue line) and by logistic regression (dashed magenta line). Bottom: Ratio of genuine and impostor score distribution in log space, where line is fitted.

approximation errors are rare, the log-likelihood is fitted by a polynomial of low degree [11] (Fig. 4 bottom). We evaluated polynomials of degree 1 to 9. A simple line (degree 1) showed the best normalization results. This is expected, since most features showed nearly a line in the log likelihood plot (as it is also the case in Fig. 4 for wHSV feature [8]). However, it should be mentioned, that the log-likelihood-plot for some features showed more complex functions. These features are probably not well suited for logistic regression normalization. When using a fitted line, the approximated probability, representing the normalized score, is calculated by

$$s_i^{norm_{reg}} = \tilde{P}(gen|s_i) = \frac{\exp(m \cdot s_i + n)}{1 + \exp(m \cdot s_i + n)}, \quad (7)$$

where m and n are the slope and y-intercept of the fitted line in log-space (The dashed magenta line in Fig. 4 (center) shows the approximated probability $\tilde{P}(gen|s_i)$).

Another way to bypass the need for modeling the genuine distribution is to formulate the normalization as a probability related to only impostor scores. Commonly, this is done by using the probability of accepting an impostor score [15], which equals the False Acceptance Rate (FAR) for a threshold $\tau = s_i$. Thus the normalization is

$$s_i^{norm_{FAR}} = \text{FAR}(s_i). \quad (8)$$

Since frequent calculations of FAR are time consuming, the transformation is approximated by a lookup table.

Probability-density-based normalization is known to generally perform very well. However, it may lead to errors when score distributions are not modeled accurately. Therefore, often much simpler transformation-based normalization approaches are applied.

Transformation-based approaches can be subdivided into linear and non-linear methods (see Fig. 2). Linear approaches only normalize the range of scores for each feature without changing the shape of the score distributions. This can be done by using the minimum and maximum of all scores $\mathbf{s}$ for image-pairs $i = 1 \ldots N$ in training set and scale the scores to range [0, 1] by

$$s_i^{norm_{mm}} = \frac{s_i - \min(\mathbf{s})}{\max(\mathbf{s}) - \min(\mathbf{s})}, \quad (9)$$

or by using the mean μ and standard deviation σ of all scores and calculate the z-normalization as

$$s_i^{norm_z} = \frac{s_i - \mu_{\mathbf{s}}}{\sigma_{\mathbf{s}}}, \quad (10)$$

to get zero mean and standard deviation.

Non-linear normalization methods do not only scale the scores, but also change the shape of genuine and impostor distributions. A well suited method to normalize exponentially distributed scores is Decimal Scaling. To apply this method, we first transform the scores to a logarithmic scale

$$\mathbf{s}^{(Log)} = \log_{10}(1 + \mathbf{s}), \quad (11)$$

and then normalize the logarithmic scores to range [0, 1] with

$$s_i^{norm_{dec}} = \frac{s_i^{(Log)}}{10^n}, \quad (12)$$

where $n = \log_{10} \max(\mathbf{s}^{(Log)})$ [10].

Capelli *et al.* [16] introduced the Double Sigmoid normalization defined as

$$s_i^{norm_{DS}} = \begin{cases} \frac{1}{1+\exp\left(-2\left(\frac{s_i-\tau}{\alpha_1}\right)\right)} & \text{if } s_i < \tau \\ \frac{1}{1+\exp\left(-2\left(\frac{s_i-\tau}{\alpha_2}\right)\right)} & \text{otherwise} \end{cases}, \quad (13)$$

where τ is the operation point where one sigmoid function fades into the other, and α_1 respectively α_2 defines the steepness of the functions. We derive the parameters from the genuine-impostor distributions. Therefore, we choose τ as the intersection point of genuine and impostor score distribution s.t. $P(genuine|\tau) = 0.5$, α_1 as left border of the overlap s.t. $P(genuine|\alpha_1) = 1 - \beta$, and α_2 as right border of the overlap s.t. $P(genuine|\alpha_2) = \beta$, with potential outliers excluded, controlled by parameter β. Evaluations show that setting $\beta = 0.05$ leads to best normalization results.

Hampel *et al.* [17] introduced the tanh-estimators, which did show good fusion results in biometric context. The normalization is given as

$$s_i^{norm_{\tanh}} = \frac{1}{2}\left\{\tanh\left[0.01\left(\frac{s_i - \mu^{(\psi)}_{\mathbf{s}^{(gen)}}}{\sigma^{(\psi)}_{\mathbf{s}^{(gen)}}}\right)\right] + 1\right\}, \quad (14)$$

where μ and σ are the estimated mean and standard deviation of the genuine score distribution using a Hampel estimator ψ with weights

$$w_{Ha}(u_i) = \begin{cases} 1 & |u_i| \le a \\ \frac{a}{|u_i|} & a < |u_i| \le b \\ \frac{a}{|u_i|} \cdot \left(\frac{c-|u_i|}{c-b}\right) & b < |u_i| \le c \\ 0 & |u_i| > c \end{cases}, \quad (15)$$

where $u_i = s_i - \text{median}(\mathbf{s}^{(gen)})$. We parametrized the Hampel estimator as Jain *et al.* [18] with $a = \text{quantile}_{0.7}(|\mathbf{u}|)$, $b = \text{quantile}_{0.85}(|\mathbf{u}|)$, and $c = \text{quantile}_{0.95}(|\mathbf{u}|)$.

B. Feature Weighting

After all scores are normalized to the same value domain, they can be combined. In order to calculate the fused score as weighted sum, a weight w is computed for each feature by using a test set of normalized distance scores $\mathbf{s}^{norm}$ (in case the normalization leads to similarity scores $\mathbf{s}^{sim}$, we transfer them to distance scores $\mathbf{s}^{dist} = 1 - \mathbf{s}^{sim}$). Thus, the fused scores are calculated by

$$s_i^{fus} = \sum_{m=1}^{M} w_m s_{i,m}^{norm}, \quad (16)$$

where i is the index of the image-pair's distance score s, m is the index of the feature, and M is the number of features to be fused.

A common way to calculate the weights is weighting all features equally. In this case, each weight is calculated by

$$w_m^{(equ)} = \frac{1}{M}. \quad (17)$$

To achieve varying weights, a performance measure function derived from the genuine-impostor-plot is often used. Hereby, the weights are gathered by computing a performance measure on scores for image-pairs of a training set.

A common performance measures for computation of weights is the equal error rate (EER). EER is defined as point in the ROC curve where false acceptance rate (FAR) and false rejection rate (FRR) are equal. The weight for feature m is calculated as

$$w_m^{(EER)} = \frac{\frac{1}{EER_m}}{\sum_{k=1}^{M}\frac{1}{EER_k}}. \quad (18)$$

Instead of using only genuine and impostor scores, an additional ranking can be computed from the training set in person re-identification. Therefore, we evaluated weights as function of rank 1, or rank 10 statistics of the CMC curve, and as function of the area under this curve (nAUC). In this case the weight for feature m can be computed as

$$w_m^{(Perf)} = \frac{Perf_m}{\sum_{k=1}^{M} Perf_k}, \quad (19)$$

where $Perf$ stands for the performance measure and can either be replaced by rank 1, rank 10, or nAUC.

Another way of computing the weights is related to the genuine-impostor score distribution. Methods of this category try to measure how well genuine and impostor scores are separated, since a large overlap of genuine and impostor scores indicates a large amount of false decisions in the re-identification system (see Fig. 3).

A statistical approach to measure the separation is D-Prime [19]

$$d_m = \frac{\mu_m^{(imp)} - \mu_m^{(gen)}}{\sqrt{\left(\sigma_m^{(imp)}\right)^2 + \left(\sigma_m^{(gen)}\right)^2}}, \quad (20)$$

where $\mu_m^{(imp)}$ and $\mu_m^{(gen)}$ are the impostor's and genuine's mean and $\sigma_m^{(imp)}$ and $\sigma_m^{(gen)}$ are their standard deviations. The weights are therefore:

$$w_m^{(DP)} = \frac{d_m}{\sum_{k=1}^{K} d_k}. \quad (21)$$

This measure assumes normal distributions for both genuine and impostor score distributions. We examined all matching score distributions for appearance-based features used in our experiments and found that the assumption holds for only few impostor distributions and none of the genuine distributions.

To avoid the assumption of normal distributions, Chia *et al.* [19] suggest to measure the width of the overlap region, named non-confidence width (NCW). An examplary visualization of this measure can be seen in Fig. 3. As done for the other measures, the weights are a direct function of the NCW

$$w_m^{(NCW)} = \frac{\frac{1}{NCW_m}}{\sum_{k=1}^{K}\frac{1}{NCW_k}}. \quad (22)$$

All methods above compute the weights directly from a quality measure for each feature separately. Therefore, these methods do not calculate the optimal weights to minimize the overall re-identification error. That is because these methods were suggested for biometric context, where joint scores for multiple features (e.g. finger prints and face templates) are often not available. Since in our scenario all features are related to the same images, we have additional information

about joint genuine-impostor distributions. To make use of this information, we recommend to formulate the computation of weights as a pairwise optimization problem: The weights w_1 and w_2 for two features define a vector on which the scores of two features are projected to get the fused score. W.l.o.g. these weights can be expressed as $k \cdot w_1 = \cos(\phi)$ and $k \cdot w_2 = \sin(\phi)$, with ϕ being the angle between the x-axis and the projection vector (see Fig. 5 for visualization). Then

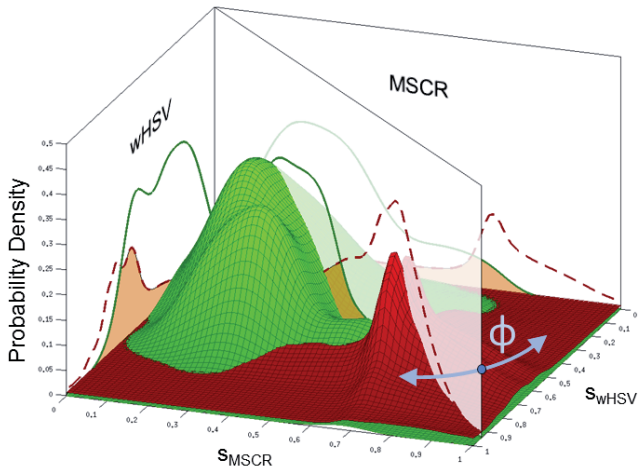


Fig. 5. Weighting formulated as optimization problem. The projection vector (displayed as semi-transparent plane) depends only on ϕ. Notice, that marginal probability densities are scaled at z-axis to visually highlight the relationship to the joint probability density distributions.

the fused genuine and impostor distributions are a function of the marginal distributions (normalized scores) and the angle of the projection vector ϕ. Therefore, finding good weights is the task to find ϕ, where an error measure is minimized. We evaluated the NCW and the overlap of genuine and impostor distribution. NCW leads to a bumpy error landscape. This is a bad condition for an optimization algorithm, since often only local minimums are found. The overlap however, produces a smooth error curve. Therefore we decided in favor of the overlap as error function. The optimization function is thus

$$\phi^{best} = \text{argmin}_{\phi=0}^{\frac{\pi}{2}} overlap\left(\mathbf{s}_{fus}^{(gen)}, \mathbf{s}_{fus}^{(imp)}\right), \tag{23}$$

with

$$\mathbf{s}_{fus}^{(gen)} = \cos(\phi)\mathbf{s}_{m_1}^{(gen)} + \sin(\phi)\mathbf{s}_{m_2}^{(gen)}, \tag{24}$$

and

$$\mathbf{s}_{fus}^{(imp)} = \cos(\phi)\mathbf{s}_{m_1}^{(imp)} + \sin(\phi)\mathbf{s}_{m_2}^{(imp)}. \tag{25}$$

Since we estimate the genuine and impostor distributions by KDE, the overlap calculates as

$$overlap\left(\mathbf{s}^{(gen)}, \mathbf{s}^{(imp)}\right) = \int_{-\infty}^{t(\phi)} KDE\left(\mathbf{s}^{(imp,w)}\right) + \int_{t(\phi)}^{\infty} KDE\left(\mathbf{s}^{(gen,w)}\right), \tag{26}$$

with $t(\phi)$ being the interception point of genuine and impostor distribution in the projection and the integral over the KDE defined as

$$\int_a^b KDE\left(\mathbf{s}\right) = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{2}\left(1 + \text{erf}\left(\frac{a - s_i}{\sqrt{2}h}\right)\right) - \frac{1}{2}\left(1 + \text{erf}\left(\frac{b - s_i}{\sqrt{2}h}\right)\right)\right], \tag{27}$$

where the kernel bandwidth h is calculated by the formula of Silverman [14] (Eq. 5). Because the error curve of the optimization problem appears to be very smooth with a single optimum for every feature combination, we have chosen the fast logarithmic search algorithm for finding the global minimum, while other heuristic optimization algorithms could have been used, too. Due to the commutative property of the summands in the calculation of the fused score (Eq. 16), pairwise weight computation is no restraint. In the experiments we will refer to this weighting method as PROPER (pairwise optimization of projected genuine-impostor overlap).

C. Combination with Feature-Level Fusion

Since score-level fusion only needs a relatively small amount of training data, it is a powerful tool to fuse large ensembles of features (experiments in Sec. III confirm this thesis). However, for small feature ensembles score-level fusion performs only moderately. In contrast, feature-level fusion, which concatenates feature vectors and applies distance metric learning, performs well on small and medium-size feature vectors[2]. In this subsection, we briefly discuss how to combine these two approaches to improve fusion performance.

To apply metric learning in combination with score-level fusion, first the high-dimensional feature vector must be divided into smaller parts (A). Then for each part a distance metric is learned (B). Finally, the matching scores for each feature vector part are combined by score-level fusion (C).

(A) To divide the feature vector, which is a combination of different features, into several medium-size parts, we use the underlying structure of the feature set[3] (see Sect. III-B for used features), i.e. we group feature vectors of the same feature type extracted from different body parts. We do not group different feature types. (B) For each group, we concatenate the feature vectors and learn an adequate distance measure for comparison of concatenated feature vectors. Therefore, we apply the best performing methods evaluated in [7]. (C) To combine the matching scores of all groups, we apply the methods of Sect. II-A and II-B.

Since all of the three processing steps are supervised, information is only lost in a controlled way, which leads to a notable increase in fusion performance. This can be seen in the next section.

[2]Metric learning can also be applied to large feature vectors. But often therefore a dimensionality reduction is needed as preprocessing step. Usually this is done by PCA, which is unsupervised, and thus potentially important information may get lost.

[3]We found, that partitioning has a large influence on the fusion performance, but we do not further examine this aspect in this paper. However, automatic partitioning will be the focus of our future work.

III. Experiments

To evaluate the performance of the methods presented in the previous section, we first examine the performance of all sub-components and then compare the proposed score-level fusion method to state-of-the-art feature-level fusion and to a combination of both fusion schemes.

A. Dataset

We evaluate our methods on the widely used and very challenging VIPeR dataset [12]. It consists of 632 persons, with two images each, taken from disjoint camera views, showing them under very different angles and lighting conditions (see Fig. 6). The images are all normalized to a size of 128×48 pixels. To obtain comparable results we follow the 10-fold cross-validation protocol of [8]. For each of the ten folds, 316 of the 632 available persons are chosen for testing. The images of the 316 remaining persons are used for training. Images of persons in test set from camera A represent the gallery, while camera B provides the corresponding probe images.

Fig. 6. Sample image pairs of VIPeR dataset [12]. Top: Gallery images (cam A). Bottom: Corresponding probe images of the same subjects (cam B).

B. Feature Set

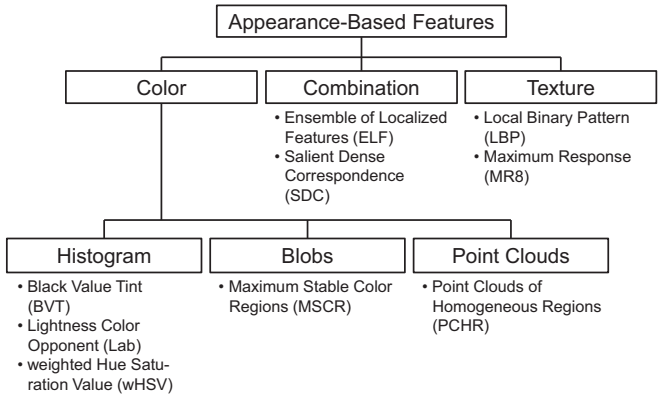


Fig. 7. Categorization of appearance-based features for person re-identification included in experiments.

In our experiments, we use nine different features, which represent the human appearance, namely by color and texture in different, complementary ways. We aim to use widely known and evaluated descriptors [3], [20], [8], [21], [22], which are known to be suitable for person re-identification. For the feature extraction processes the authors' implementations were used. Those were either publicly available or made available by the authors on request.

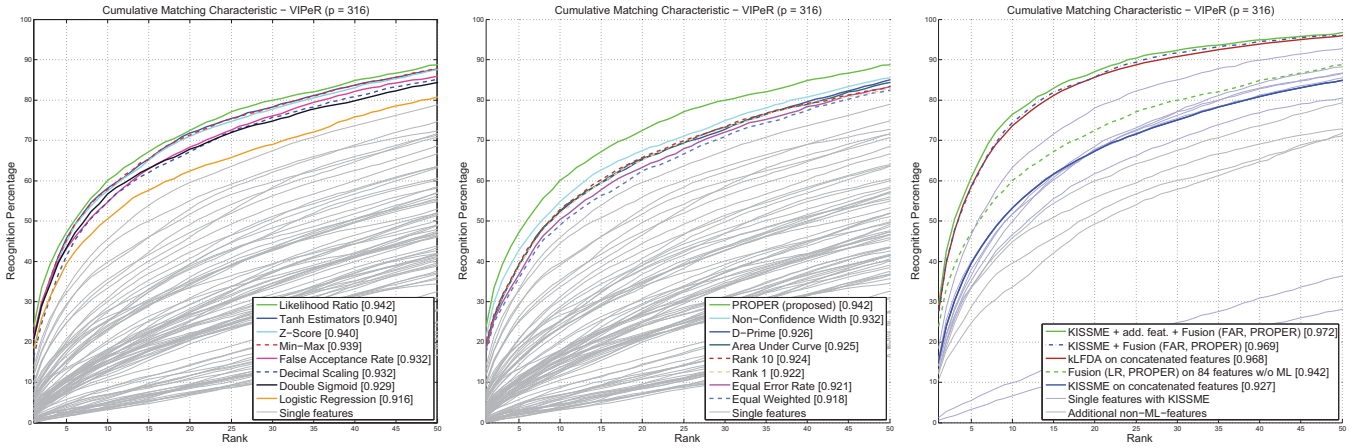
The used features can be categorized as shown in Fig. 7. The first category are features that rely completely on color histograms. Similar to Figueira *et al.* [3], we extracted a color histogram of the image in the Lab color space with ten non-uniform bins[4] per channel. Following the procedure of [3] a *Black Value Tint histogram* (BVT) [20] was extracted for each image. BVT histograms are formed in the HSV color space and handle dark and unsaturated pixels in a separate gray value histogram. This minimizes their influence on the color histogram. Other than the plain HSV histogram in [3], we decided to use the widely-used weighted color histogram in HSV color space as introduced by Farenzena *et al.* [8]. For these histograms, the weight of a single pixel is defined by a Gaussian kernel centered at symmetry lines found in the upper and lower body. We refer to them as *weighted HSV histograms* (wHSV).

Additionally, we extracted *Maximally Stable Color Regions* (MSCR) [20], [8] and *Point Clouds of Homogeneous Regions* (PCHR) [24] that were developed for fast object tracking. In contrast to the methods above, MSCR does not form a fixed length feature vector and PCHR templates can not be directly compared due to the position variability in the point clouds. This means, both features do not suit for *feature-level fusion* and can only be fused with other features at *score-level*.

Another category of features we use is based on texture (see Fig. 7). Therefore, *Local Binary Pattern* (LBP) [25] and *Maximum Response filter* (MR8) [26] are utilized. Uniform LBP encode texture as a histogram of binary patterns. These binary patterns encode darker and lighter pixel around a center point, which is shifted to every possible position in the source image. The representation formed by MR8 is generated by a filter bank consisting of two anisotropic filters (edge and bar filters), each of them at six orientations and three scales, as well as two rotation invariant filters (Gaussian and Laplacian of Gaussian). Only eight filter responses are used by taking the maximal response of the anisotropic filters across all orientations at each scale. Similar to Figueira *et al.* [3], we applied a histogram computation with non-uniform binning with ten bins per response to reduce the dimension of this representation even more.

We also extracted more complex features which combine color and texture descriptors such as *Ensemble of Localized Features* (ELF) [21], [22] and *Salient Dense Correspondence* (SDC) [28]. ELF is a combination of eight color histograms (RGB, HS, YCbCr) with 16 bins per channel and the filter responses of 21 texture filters (13 Gabor filters [29] and eight Schmid filters [30]). All the histograms are concatenated into a 464 dimensional feature vector. For each single histogram, we apply non-uniform binning using the same procedure as for the Lab histograms. SDC extracts SIFT features, encoding texture, and color histograms (32 uniform bins per channel) in Lab color space, densely sampled using overlapping patches.

[4]The limits specifying the bin ranges were chosen, such that the average histogram would resemble a uniform distribution. This was done using the INRIA dataset [23], normally used as benchmark and training dataset for person detection algorithms.



(a) Score Normalization (b) Feature Weighting (c) Combination with Metric Learning

Fig. 8. Re-identification performance on VIPeR dataset. Images in the probe set are compared with gallery images and a ranking is computed based on the similarity of the pictured subjects. The Cumulative Matching Characteristic (CMC) curve shows how often the correct match appears within the the first n ranks ($n = 1 \dots 316$). Normalized area under CMC curve (nAUC) is given in squared brackets. (a) Score-level fusion with presented score normalization approaches (weighting is done by best performing method PROPER). Single feature performances is shown as gray lines. (b) Score-level fusion with presented feature weighting methods (score normalization is done by best performing Likelihood Ratio). (c) Combination of score-level fusion and linear metric learning (KISSME [27]) outperforms currently best state-of-the-art non-linear metric learning approach (kLFDA [7]).

Using the masks of a part detector [20], we extracted BVT, LBP, MR8, Lab and wHSV histograms as well as ELF features for each body part. ELF features are additionally extracted on six stripes as introduced by Prosser *et al.* [22]. We denote them as *SELF*. In addition to the part-based wHSV, we also used the wHSV histograms as used by Farenzena *et al.* [8]. The masks of the part detector were also used for MSCR as designed by Cheng *et al.* [20]. In contrast, the PCHR features were extracted using an average person mask as in [24]. Summarized, our feature set is composed of 84 feature vectors, with an accumulated dimensionality of 242,109 on average (MSCR varies, $\sigma = 16$).

C. Score Normalization

To evaluate the best configuration for score-level fusion, we benchmarked all 64 combinations of *score normalization* and *feature weighting* methods. The best recognition performance was observed for *likelihood ratio* (LR) score normalization with the proposed *PROPER* feature weighting. Fig. 8(a) shows the Cumulative Matching Characteristic (CMC) curves for the score normalization methods in combination with the best performing feature weighting method PROPER. It can be seen, that all score normalization methods are capable to improve the recognition percentage in comparison to the performance of every single feature (gray lines). As known from biometrics (e.g. [11]), LR normalization performed slightly better than the other score normalization methods.

D. Feature Weighting

Fig. 8(b) shows the CMC curves for feature weighting methods in combination with the best performing LR score normalization. The figure shows, that the proposed PROPER feature weighting (normalized area under CMC curve $nAUC = 0.942$) clearly outperforms all state-of-the-art weighting methods by a significant margin. The second best performance for feature weighting w/o PROPER was observed with the non confidence width (NCW) criterion in combination with LR normalization ($nAUC = 93.2$). The expected rank (ER) for PROPER is 19.23, which means, that the correct match can be found on rank 19 on average (out of 316; $\sigma_{Rank} = 29.65$). This is more than three ranks better than NCW ($ER = 22.41$; $\sigma_{Rank} = 31.85$). It becomes apparent, that using additional information with PROPER increases the re-identification rate considerably.

E. Score-Level vs. Feature-Level Fusion

Obviously, score-level fusion is a powerful tool to combine multiple features. However, state-of-the-art approaches usually fuse at feature-level by concatenating all feature vectors and applying distance metric learning. Therefore, the performance of these two fusion techniques, as well as a combination, is evaluated in the following.

In Fig. 8(c) the performance of the score-level fusion with LR and PROPER is shown as dashed green line. The state-of-the-art linear metric learning method KISSME [27] (solid blue line) performs worse on the concatenated features due to a unsupervised PCA preprocessing step. However, on a subset of features (solid light blue lines), KISSME is able to perform better than score-level fusion (less information get lost by PCA). This shows, that the performance of metric learning methods drops for high dimensional feature vectors. Therefore, we decided to run KISSME on multiple subsets of features and fuse them at score-level. The performance of the combined feature- and score-level fusion is visualized as dash-dotted blue line. This combination even outperforms the best non-linear metric learning method, the Kernel Local Fisher Discriminant Analysis [7] (kLFDA), on the concatenated feature vector (red line) and on any subset of features (not shown for clarity reasons), while being much faster in the execution phase, since no kernel computation is necessary. Including additional

features to the score-level fusion, that are not suited for feature-level fusion (see Sect. III-B), improves the recognition rate further (solid green line; $nAUC = 0.972$). The influence of each feature on the overall performance is shown in Fig. 9. Using kLFDA (instead of KISSME) for metric learning on feature subsets and apply score-level fusion resulted in a only slightly better performance ($nAUC = 0.973$), but is not worth the complexer calculation when real-time constraints have to be respected. Therefore, when a large ensemble of features for person re-identification has to be fused, we recommend to use KISSME in combination with the proposed score-level fusion.

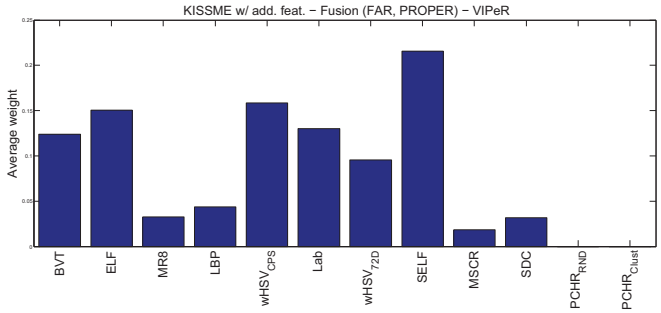


Fig. 9. Learned weights to fuse features at score-level, when metric learning is applied per feature, show the influence of each feature for the overall performance (see Fig. 8(c) for CMC curve).

IV. Conclusion

We evaluated score-level fusion techniques for appearance-based person re-identification features. As known from biometrics, score normalizing with the likelihood ratio method performed best. For weighting the features, we proposed the pairwise optimization scheme PROPER, which outperforms state-of-the approaches. When fusing a large ensemble of features, score-level fusion with likelihood ratio score normalization and pairwise weight optimization outperforms linear metric learning approaches, that fuse at feature-level. However, a combination of linear metric learning and score-level fusion reaches even better results and slightly outperforms the currently best non-linear kernel-based metric learning approach. Furthermore, our approach is significantly faster in the execution phase. Score-level fusion is thus a powerful tool to fuse large feature sets, especially in combination with linear metric learning.

References

[1] A. Kolarow, K. Schenk, M. Eisenbach, M. Dose, M. Brauckmann, K. Debes, and H.-M. Gross, "Apfel: The intelligent video analysis and surveillance system for assisting human operators." in *Proc. of IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance (AVSS)*, 2013, pp. 195–201.

[2] H.-M. Gross, K. Debes, E. Einhorn, S. Mueller, A. Scheidig, C. Weinrich, A. Bley, and C. Martin, "Mobile robotic rehabilitation assistant for walking and orientation training of stroke patients: A report on work in progress." in *Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC)*, 2014, pp. 1880–1887.

[3] D. Figueira, L. Bazzani, H. Q. Minh, M. Cristani, A. Bernardino, and V. Murino, "Semi-supervised multi-feature learning for person re-identification," in *Proc. of 10th IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, 2013, pp. 111–116.

[4] C. Liu, S. Gong, C. C. Loy, and X. Lin, "Person re-identification: What features are important?" in *Proc. of Workshop of European Conference on Computer Vision (ECCV)*, 2012, pp. 391–401.

[5] C. Liu, S. Gong, C. C. Loy, and X. Lin, *Evaluating Feature Importance for Re-Identification*. Springer, 2014, pp. 205–230.

[6] N. T. Pham, K. Leman, R. Chang, J. Zhang, and H. L. Wang, "Fusing appearance and spatio-temporal features for multiple camera tracking," in *Proc. of MultiMedia Modeling*, 2014, pp. 365–374.

[7] F. Xiong, M. Gou, O. Camps, and M. Sznaier, "Person re-identification using kernel-based metric learning methods," in *Proc. of European Computer Vision Conference (ECCV)*, 2014, pp. 1–16.

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

[9] D. Maltoni, D. Maio, J. A.K., and S. Prabhakar, *Handbook of fingerprint recognition. Biometric Fusion*. Springer, 2009, pp. 303–339.

[10] A. Ross and K. Nandakumar, *Encyclopedia of Biometrics. Fusion, Score-Level.* Springer, 2009, pp. 611–616.

[11] B. Ulery, A. Hicklin, C. Watson, W. Fellner, and P. Hallinan, "Studies of biometric fusion," NISTIR 7346, Tech. Rep., 2006.

[12] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *Proc. of IEEE Int. Workshop on Performance Evaluation for Tracking and Surveillance (PETS)*, 2007.

[13] K. Nandakumar, A. Ross, and A. K. Jain, "Biometric fusion: Does modeling correlation really matter?" in *Proc. of IEEE 3rd Int. Conf. on Biometrics: Theory, Applications and Systems (BTAS)*, 2009, pp. 1–6.

[14] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. CRC Press, 1986.

[15] M. Eisenbach, A. Kolarow, K. Schenk, K. Debes, and H. Gross, "View invariant appearance-based person reidentification using fast online feature selection and score level fusion," in *AVSS*, 2012, pp. 184–190.

[16] R. Cappelli, D. Maio, and D. Maltoni, "Combining fingerprint classifiers," in *Proc. of First Int. Workshop on Multiple Classifier Systems*, 2000, pp. 351–361.

[17] F. Hampel, P. Rousseeuw, E. Ronchetti, and W. Stahel, *Robust Statistics: The Approach Based on Influence Functions*. Wiley, 1986.

[18] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pat. Rec.*, vol. 38, pp. 2270–2285, 2005.

[19] C. Chia, N. Sherkat, and L. Nolle, "Biometric fusion: Does modeling correlation really matter?" in *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, 2010, pp. 1176–1179.

[20] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino, "Custom pictorial structures for re-identification." in *Proc. of British Machine Vision Conference (BMVC)*, 2011.

[21] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *Proc. of European Cof. on Computer Vision (ECCV)*, 2008, pp. 262–275.

[22] B. Prosser, W.-S. Zheng, S. Gong, T. Xiang, and Q. Mary, "Person re-identification by support vector ranking." in *Proc. of British Machine Vision Conference (BMVC)*, 2010.

[23] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 886–893.

[24] A. Kolarow, M. Brauckmann, M. Eisenbach, K. Schenk, E. Einhorn, K. Debes, and H.-M. Gross, "Vision-based hyper-real-time object tracker for human-robot interaction," in *Proc. of Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[25] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 24, pp. 971–987, 2002.

[26] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *Int. Jour. of Computer Vision (IJCV)*, vol. 62, no. 1-2, pp. 61–81, 2005.

[27] M. Kostinger, M. Hirzer, P. Wohlhart, R. P.M., and H. Bischof, "Large scale metric learning from equivalence constraints." in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2288–2295.

[28] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3586–3593.

[29] I. Fogel and D. Sagi, "Gabor filters as texture discriminator," *Biological Cybernetics*, vol. 61, no. 2, pp. 103–113, 1989.

[30] C. Schmid, "Constructing models for content-based image retrieval," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. 39–45.