Contextual Constraints for Person Retrieval in Camera Networks

Martin Bäuml, Makarand Tapaswi, Arne Schumann, Rainer Stiefelhagen
Karlsruhe Institute of Technology
Institute for Anthropomatics, 76131 Karlsruhe
{baeuml, makarand.tapaswi, arne.schumann, rainer.stiefelhagen}@kit.edu

Abstract

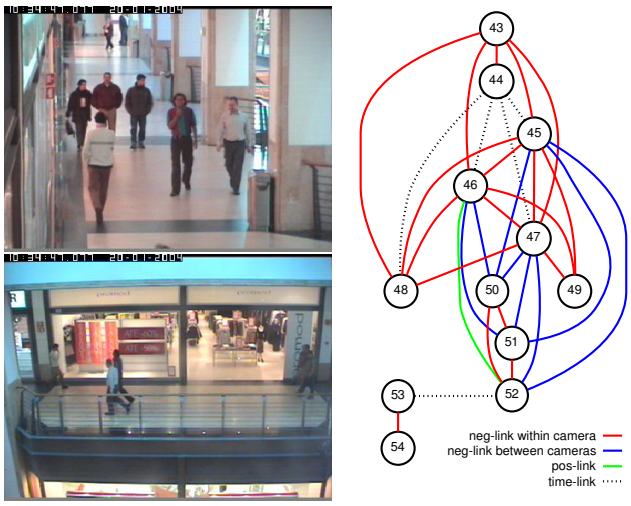
We use contextual constraints for person retrieval in camera networks. We start by formulating a set of general positive and negative constraints on the identities of person tracks in camera networks, such as a person cannot appear twice in the same frame. We then show how these constraints can be used to improve person retrieval. First, we use the constraints to obtain training data in an unsupervised way to learn a general metric that is better suited to discriminate between different people than the Euclidean distance. Second, starting from an initial query track, we enhance the query-set using the constraints to obtain additional positive and negative samples for the query. Third, we formulate the person retrieval task as an energy minimization problem, integrate track scores and constraints in a common framework and jointly optimize the retrieval over all interconnected tracks. We evaluate our approach on the CAVIAR dataset and achieve 22% relative performance improvement in terms of mean average precision over standard retrieval where each track is treated independently.

1. Introduction

Person retrieval and re-identification in camera networks is generally approached by treating each person track independently (*e.g.*, [2, 3, 5]). However, in reality, tracks are not fully independent of one another. In fact, a track carries much more information than just its appearance, and in this paper we leverage this additional information in order to improve person retrieval performance.

In order to keep operation of a camera network scalable, it is advantageous to perform person tracking and low-level feature extraction for each camera in the network independently, even in presence of overlapping cameras. We therefore do not assume that the person tracks are available in world coordinates.

Our main contributions are three-fold. (i) We formalize a set of contextual constraints in camera networks that can be leveraged to reduce the set of candidates during person retrieval. (ii) We use the constraints to learn a discriminative metric with a Euclidean prior. (iii) We formulate the retrieval task as an energy minimization problem which jointly optimizes the ranking over all tracks taking both track scores and constraints into account.



(a) (b)

Figure 1: (a) Camera views and sample frames from the CAVIAR sequence *OneShopOneWait2* and (b) the automatically determined links between tracks for the same sequence.

The following subsection gives a brief overview of related work. Sections 2 and 3 describe the proposed retrieval approach and use of contextual constraints while Sec. 4 outlines the person tracking approach and the features used for retrieval. We validate our approach on the CAVIAR dataset. Results on the retrieval performance are presented and discussed in Sec. 5.

1.1. Related Work

Person (re-)identification and retrieval are common and challenging problems in camera networks as well as multimedia data and personal image collections. Accordingly, they have received a fairly large amount of research inter-

est. Many different aspects of a person's appearance are used for identification. This includes for example facial features [7, 11], clothing appearance [6, 2] or semantic attributes [18]. A detailed survey on the topic can be found in [5]. However, only a small number of works make use of contextual cues for person identification.

Anguelov *et al*. [1] use contextual constraints for identity recognition in photo albums. They integrate clothing and facial cues in a Markov Random Field model and add a uniqueness constraint that prevents different people in a photo from receiving the same identity. Similarly, Song and Leung [17] include this constraint in their approach to person recognition in single images by clustering. O'Hare *et al*. [15] and Naaman *et al*. [14] use soft constraints in the form of occurrence frequency and co-occurrence to obtain identity priors for person identification in photo albums. Gallagher and Chen [6] employ a uniqueness constraint to prevent labeling two people in the same image with the same identity.

Constraints are more frequently – though mostly implicitly – used for person tracking. The uniqueness constraint for a single camera is often used when performing tracking and identification at the same time to prevent two co-occurring tracks to carry the same identity (*e.g*., [16]). Constraints based on the homography between multiple cameras can be used to improve tracking accuracy and robustness to occlusions (*e.g*., [10]). Uniqueness and temporal constraints are also the basis for learning association functions in tracklet-association-based tracking approaches (*e.g*., [9, 12, 19]).

Closest to our work is the work of Yu *et al*. [20] in which positive and negative constraints for person tracks are obtained from a set of cameras calibrated to a common world coordinate system. These constraints are then used to improve online identity tracking via spectral clustering. Our work generalizes the set of constraints to camera networks without the need for calibration to a common world coordinate system and makes use of the constraints on different levels for person retrieval.

2. Contextual Constraints on Person Tracks

In this work, we consider a person track t_i to be a set of consecutive location estimates, such as bounding boxes, of a person for an image sequence. We begin by postulating four properties of person tracks and their corresponding identities in camera networks (see Fig. 2 for a graphical depiction).

P1: Two temporally co-occurring tracks in spatially non-overlapping camera views cannot originate from the same person.

P2: Two temporally co-occurring tracks in the same camera cannot originate from the same person.

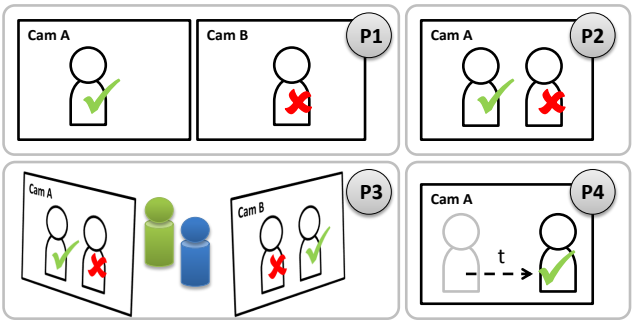


Figure 2: Graphical depiction of four properties of person tracks and their corresponding identities in camera networks.

P3: Two temporally co-occurring tracks in spatially overlapping camera views (with homography H), originate from the same person, if the position of the track t_A in camera A maps to the position of the track t_B in the camera B (**P3a**). Vice-versa, if t_A does not map to the position of t_B, they *cannot* originate from the same person (**P3b**).

P4: Two non-overlapping tracks within a specified duration of time have a likelihood of originating from the same person, if they are similar in appearance and the extrapolated trajectory of the previous track is close to the starting point of the next track.

Properties P1 to P3 are derivations from the obvious fact that a person can only be at one point in space at a given time. Nevertheless, they provide useful information, *e.g*. link tracks between different overlapping cameras or prohibit the assignment of a similar identity to co-occurring tracks.

P1 to P3 directly induce positive and negative constraints on the track identities, while P4 will mainly be used in the global retrieval optimization in Sec. 3.3. The set of positive constraints is

$$\mathcal{C}^+ = \{(t_i, t_j) \in P_{3a}\} \tag{1}$$

and states that two related tracks share the same identity, *i.e*. $(t_i, t_j) \in \mathcal{C}^+ \Rightarrow \mathrm{id}(t_i) = \mathrm{id}(t_j)$.

Similarly, P_1, P_2 and P_{3b} lead to a set of negative track pairs

$$\mathcal{C}^- = \{(t_i, t_j) \in P_1 \cup P_2 \cup P_{3b}\} \tag{2}$$

where two tracks *do not* share the same identity, *i.e*. $(t_i, t_j) \in \mathcal{C}^- \Rightarrow \mathrm{id}(t_i) \neq \mathrm{id}(t_j)$.

3. Using Constraints for Person Retrieval

The above constraints can be utilized on several levels. We first give a brief overview of the different ways in which we use the constraints and then discuss each of them in detail in the following sections.

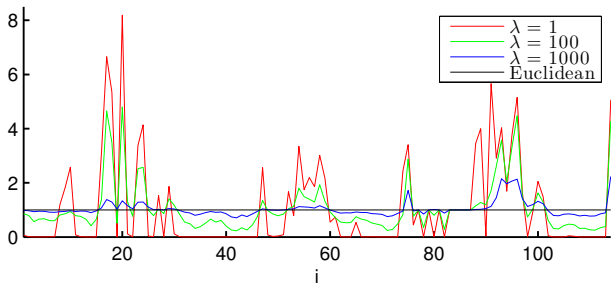


Figure 3: Learned metric parameters $\mathcal{M}_{ii}$ for different regularization parameters λ.

(L1) Metric learning (Sec. 3.1): We first learn a distance metric $\mathcal{M}$ optimized towards discriminating positive and negative feature pairs.

(L2) Query-set enhancement (Sec. 3.2): Starting from a specific query track, we automatically collect additional matching (positive) and non-matching (negative) tracks based on constraints to the query track. On the one hand this enhances the model of the query track for the retrieval, *i.e.* we obtain more features for a more accurate description. On the other hand, these tracks can be reported directly as positive and negative results.

(L3) Global retrieval optimization (Sec. 3.3): All above stages concern only the query track and its constraints. We can now take the constraints for all tracks into account during the actual retrieval. Instead of treating each track independently, groups of tracks that are interlinked by constraints are collectively scored by minimization of an appropriate energy function.

3.1. Metric Learning

We adapt Logistic Discriminant Metric Learning (LDML) [7] to learn a constrained distance between the features with a prior. Let $\mathbf{x}_i$ and $\mathbf{x}_j \in \mathbb{R}^d$ be feature vectors describing a single frame from person track t_i and t_j respectively. The Mahalanobis distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as

$$d_{\mathcal{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathcal{M} (\mathbf{x}_i - \mathbf{x}_j) \tag{3}$$

where $\mathcal{M}$ is a positive semi-definite matrix. Following [7], we can now model the probability that the two feature vectors describe the same person ($\mathrm{id}(t_i) = \mathrm{id}(t_j)$) as

$$p(\mathrm{id}(t_i) = \mathrm{id}(t_j) | \mathbf{x}_i, \mathbf{x}_j; \mathcal{M}, b) = \sigma(b - d_{\mathcal{M}}(\mathbf{x}_i, \mathbf{x}_j)) \tag{4}$$

where $\sigma(z) = (1+\exp(-z))^{-1}$ is the sigmoid function. As Guillaumin *et al.* [7] point out, this is the standard logistic regression model and parameters b and $\mathcal{M}$ can be learned from training data via gradient descent.

We modify LDML in two ways: (i) We constrain all non-diagonal entries of $\mathcal{M}$ to be 0 (*i.e.* $\mathcal{M}_{ij} = 0 \; \forall i \neq j$) and all diagonal entries to be non-negative (*i.e.* $\mathcal{M}_{ii} \geq 0$) which reduces the number of parameters from d^2 to d and thus is less prone to over-fitting. This essentially makes $d_{\mathcal{M}}$ a normalized Euclidean distance. (ii) We impose a normal prior on the parameters of $\mathcal{M}$ (*i.e.* $\mathcal{M}_{ii} \sim \mathcal{N}(\mu_i, \sigma_i)$) in order to leverage a previously learned model and adapt it to new training data.

In order to obtain the parameters of the model we minimize the negative log-posterior, taking the prior into account. Let the tuple $(y^{(k)}, \mathbf{d}^{(k)} = \mathbf{x}_1^{(k)} - \mathbf{x}_2^{(k)})$ denote a training sample, where $y^{(k)} \in \{0, 1\}$ denotes whether $\mathbf{d}^{(k)}$ stems from a negative or positive pair, respectively. The model predicts $h_{\mathcal{M}}^{(k)} = \sigma(b - d_{\mathcal{M}}(\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}))$. The negative log-posterior and its gradient with respect to the values of $\mathcal{M}$ can then be written as

$$J = \sum_k -y^{(k)} \ln(h_{\mathcal{M}}^{(k)}) - (1 - y^{(k)}) \ln(1 - h_{\mathcal{M}}^{(k)}) + \frac{\lambda}{2} \sum_i \sigma_i^{-2} (\mathcal{M}_{ii} - \mu_i)^2 \tag{5}$$

$$\frac{\partial J}{\partial \mathcal{M}_{ii}} = \sum_k (h_{\mathcal{M}}^{(k)} - y^{(k)}) (d_i^{(k)})^2 + \lambda \sigma_i^{-2} (\mathcal{M}_{ii} - \mu_i) \tag{6}$$

The parameter λ controls the influence of the prior. Fig. 3 shows the learned metric parameters $\mathcal{M}_{ii}$ for different regularization parameters λ. Note that the metric parameters for unimportant dimensions (for discriminating features) are pushed towards zero, while important dimensions receive higher weights. For higher λ the learned metric converges towards the prior, in this case the Euclidean with $\mathcal{M}_{ii} = 1 \, \forall i$.

Implementation details: We enforce the non-negativity of $\mathcal{M}_{ii}$ by introducing a helper parameter θ, where $\mathcal{M}_{ii} = \theta_i^2$. The optimization of J over θ is performed with L-BFGS [13]. λ is set to 5 in our experiments. We employ a Euclidean prior (*i.e.* $\mu_i = 1$) with $\sigma_i = 1$.

3.2. Query-set Enhancement

Let $\mathcal{Q} = \mathcal{Q}^+ \cup \mathcal{Q}^-$ denote the *query-set*, which consists of positive examples $\mathcal{Q}^+$ of a searched person, and negative examples $\mathcal{Q}^-$ of other people. We assume that a retrieval task is started by a user who selects an example person track t_q^+ from among previously recorded tracks. In that case, the initial query-set becomes $\mathcal{Q}^+ = \{t_q^+\}$ and $\mathcal{Q}^- = \{\}$. The constraints very easily allow us to enlarge this initial query-set by additional tracks: Positive constraints ($\mathcal{C}^+$) result in additional query tracks for the searched person

$$\mathcal{Q}^+ = \{t_q^+\} \cup \{t | (t_q^+, t) \in \mathcal{C}^+\} \tag{7}$$

while negative constraints enhance the query-set with tracks of persons that are not targets of the retrieval

$$\mathcal{Q}^- = \{t | (t_q^+, t) \in \mathcal{C}^-\}. \tag{8}$$

This has two advantages: On the one hand, the additional tracks from the enhanced query-set can be directly reported as positive and negative results. On the other hand, $\mathcal{Q}^+$ and $\mathcal{Q}^-$ allow to obtain positive *and* negative training samples for the current query in an unsupervised way. These can be used for training a discriminative classifier for the current query (*e.g.*, as done for faces in [3]). Another viable option would be to learn a query-specific metric $\mathcal{M}_2$ with the method presented in Sec. 3.1. In order to avoid avoid overfitting while learning $\mathcal{M}_2$, *e.g.* if the query-set is small, the previously learned metric $\mathcal{M}$ can be used as prior. This essentially adapts the more general metric $\mathcal{M}$ to the current query, leveraging the information about the specific query while not completely disregarding the the general information about the data set. We will explore this further in future work.

3.3. Retrieval as Energy Optimization

In this section we globally optimize the retrieval ranking. To this end we formulate the retrieval as an energy optimization problem. This effectively combines the individual scores for the tracks and the constraints.

Let N be the number of tracks, and $s_i \in [0,1]$ a similarity measure between track t_i and the query track(s) obtained by matching appearance (see Sec. 4.3 for details). Let $v_i \in [0,1]$ be a variable that denotes the retrieval importance of track i (*i.e.*, the higher v_i is, the higher it will be ranked in the retrieval). v_i can also be interpreted as a (pseudo-)probability that the identity of track t_i is the same as of the query track(s). In order to globally optimize the retrieval ranking we propose an energy function over v_i

$$E(\mathbf{v}) = \sum_{1 \le i \le N} U_i \; + \; \gamma_1 \sum_{\{i,j\} \in \mathcal{C}^+} V_{ij}^+ \; + \; \gamma_2 \sum_{\{i,j\} \in \mathcal{C}^-} V_{ij}^- \; + \; \gamma_3 \sum_{\{i,j\} \in \mathcal{T}} Z_{ij} \tag{9}$$

whose individual terms are described in the following.

The unary term U_i captures the conformance of v_i with the corresponding score s_i:

$$U_i = v_i \ln(s_i) + (1 - v_i) \ln(1 - s_i) + \gamma_4 (v_i - 0.5)^2$$

The intuition is that v_i should be close to 1 if s_i is high and vice versa (remember $s_i \in [0,1]$). At the same time $(v_i - 0.5)^2$ acts as a prior that – in the absence of additional information – enforces a neutral ranking, *i.e.* v_i should be close to 0.5. The prior is weighted by parameter γ_4.

A negative link between tracks t_i and t_j imposes a penalty if both v_i and v_j are high. This prevents two tracks which are connected by a negative link from both being ranked high:

$$V_{ij}^- = (|\mathcal{C}_i^-|^{-1} + |\mathcal{C}_j^-|^{-1}) \cdot v_i \cdot v_j \tag{10}$$

where $|\mathcal{C}_i^-|$ is the cardinality of the set of negative constraints that contain track t_i, and $(|\mathcal{C}_i^-|^{-1} + |\mathcal{C}_j^-|^{-1})$ acts as a normalization factor. Note, that there is no penalty if v_i and v_j are both low. It essentially means that both tracks are not similar to the *query* track which can also be the case if there is a negative link between the both tracks (*i.e.* they stem from different people, but not the query person).

A positive link imposes a penalty if the difference between v_i and v_j is high:

$$V_{ij}^+ = (|\mathcal{C}_i^+|^{-1} + |\mathcal{C}_j^+|^{-1}) \cdot (v_i - v_j)^2. \tag{11}$$

This encourages that two tracks linked by a positive link appear near each other in the final ranking.

All track pairs which lie within a specified duration (*e.g.*, 5 seconds) from one another, form the set of temporal constraints $\mathcal{T}$. We encourage two tracks $(t_i, t_j) \in \mathcal{T}$ to have a similar ranking if both the appearance distance d_{ij}^{app} between the two tracks and the distance d_{ij}^{sp} between the extrapolated trajectory of the former to the latter is small:

$$Z_{ij} = \exp(-d_{ij}^{app}/\sigma^{app}) \cdot \exp(-d_{ij}^{sp}/\sigma^{sp}) \cdot (v_i - v_j)^2$$

After minimization of Eq. 9 over $\mathbf{v}$, we obtain the final retrieval ranking by sorting the weighted combination $v_i \cdot s_i$ in descending order.

Implementation details: We minimize E over $\mathbf{v}$ on the continuous domain $[0,1]$ via gradient descent [13]. In order to enforce v_i to remain in the range $[0,1]$ we introduce a helper variable ξ_i. We set $v_i = \sigma(\xi_i) = (1 + exp(-\xi_i))^{-1}$ and optimize over ξ instead.

4. Tracking and Features

In this section we give a brief overview about the tracking procedure and features that we use in our experiments.

4.1. Person Tracking

We perform person tracking based on a HOG detector [4]. A particle filter is used to connect detections in successive frames and bridge gaps over missing detections. In order to speed up the detection process and make the tracking more robust, we estimate ground-plane models in an unsupervised way for each camera in the network from the strongest detections within the respective views. A simple color histogram-based appearance model is employed to deal with occlusions and avoid track switches.

4.2. Features

For each person track, we extract color and texture features from a feature region within the track's bounding box with $[x, y, w, h] = [0.33, 0.2, 0.33, 0.3]$ in coordinates relative to the track bounding box. Before feature extraction, we resize the feature region to a fixed size of 64×128 pixels.

Color Structure Descriptor (CSD) As color feature we compute the CSD with 40 bins. This feature has been shown to work quite well in a person recognition setting [8].

Sobel Filter Response (SFR) The filter response computed on the gray-scale image of the feature region to both vertical and horizontal Sobel filters forms our texture descriptor. The response is binned into 37 bins, to obtain a 74 dimension feature vector (vertical and horizontal).

Discrete Cosine Transform (DCT) The DCT is applied on the gray-scale image of the feature region. The top 120 coefficients, scanned in zig-zag order (inclusive of the mean (0, 0)), form the feature vector.

4.3. Computing Track Distances

For each track, we extract one feature vector for each frame, *i.e.* we describe the appearance of track t_i with a set of features $\mathbf{f}_k^i$ with $1 \leq k \leq \text{length}(t_i)$. In order to compare the appearance of two tracks t_i and t_j we compute the min-min-distance between the features of the tracks:

$$d(t_i, t_j) = \min_k \min_l d_f(\mathbf{f}_k^i, \mathbf{f}_l^j) \tag{12}$$

where we use the Euclidean distance $d_f(\mathbf{f}_k^i, \mathbf{f}_l^j) := \sqrt{\mathbf{f}_k^{i\,T} \mathbf{f}_l^j}$ as a baseline. To use the learned metric $\mathcal{M}$ we set $d_f(\cdot, \cdot) := 1 - \sigma(b - d_{\mathcal{M}}(\cdot, \cdot))$ (*c.f.* Eq. 3).

During retrieval, the appearance distance of each track t_i is computed against the positive query-set $\mathcal{Q}^+$. If $\mathcal{Q}^+$ contains more than one track, we compute the minimum distance over all tracks in $\mathcal{Q}^+$. This is equivalent to combining all features of tracks in $\mathcal{Q}^+$ to a common feature pool $\mathbf{f}^+$, owing to the dual min-operator in the distance function.

We obtain a similarity score s_i for track t_i to the query-set $\mathcal{Q}^+$ as $s_i = 1 - d(t_i, \mathcal{Q}^+)$. For baseline retrieval – treating each track independently – the ranking is obtained by sorting tracks according to s_i in descending order.

For retrieval based on energy optimization (Sec. 3.3), the similarity scores s_i are incorporated in the unary term U_i. U_i requires s_i to be in the range $[0, 1]$. Note that when we use the learned metric $\mathcal{M}$, the obtained similarity score is already in the range $[0, 1]$ due to the sigmoid function. For other distances (*e.g.* Euclidean), a mapping of the distance output $d \in \mathbb{R}_0^+$, to the range $[0, 1]$ can be realized by modeling $s_i = \sigma(a + bd)$ and using distances between tracks within $\mathcal{C}^+$ and $\mathcal{C}^-$ as training data for learning parameters a and b.

5. Experimental Validation

5.1. Dataset and Setup

We evaluate our approach on the CAVIARDATA2 data set[1]. The data set consists of 26 sequences with two camera views each recorded within a shopping mall. The sequences' frame size is 384×288 pixels. The two camera views are roughly perpendicular to each other and partly overlapping. The homography between the ground-planes in the two cameras is estimated based on 4 points in the overlapping area. Our tracker generated a total of 248 tracks of 65 different people. For further comparisons, we made the generated tracks, the computed features and our annotations publicly available[2].

5.2. Results and Discussion

We use the mean average precision (MAP) as performance measure. Let N be the number of tracks in the database, then the average precision (AP) of a ranked list of tracks for each query is defined as

$$AP = \frac{1}{\sum_{i=1}^N r_i} \sum_{i=1}^N r_i \left(\frac{\sum_{j=1}^i r_i}{i} \right) \tag{13}$$

where r_i is 1 if the track at rank i is of the searched person and 0 otherwise. The MAP is computed as the mean of APs over all possible queries, starting with each track from our database as initial query track. Note that the AP is quite a strict measure. To illustrate, lets assume we have two correct result tracks in our database for a given query. If they are retrieved at ranks 2 and 3, the AP is 0.58. However, if they are retrieved at ranks 9 and 10, the AP drops to 0.16, although, for a system with a human in the loop, this would be considered a good result if the person had to look only at 10 possible tracks instead of hundreds.

We present results in multiple steps of improvement. An overview of the results can be found in Table 1 and Fig. 5.

Features: We compare the performance obtained with different features as described in Sec. 4.2. The Color Structure Descriptor (CSD) outperforms both the Sobel Feature Response and Discrete Cosine Transform descriptors with a MAP of 0.286 over 0.247 and 0.203, respectively. CSD+SFR+DCT denotes a concatenation of the individual features which outperforms each individual features with a MAP of 0.325. This serves as our baseline.

Feedback: For each experiment we simulate multiple rounds of user relevance feedback by marking the top 5 results as either positive or negative and use it to enlarge $\mathcal{Q}^+$ and $\mathcal{Q}^-$ respectively. As expected, feedback increases overall performance significantly to a MAP of 0.569 after 5 rounds of feedback for the CSD+SFR+DCT feature combination.

Leveraging constraints: For the evaluation of the different levels using the constraints we use the feature concatenation CSD+SFR+DCT. Without feedback, query-set enhancement (L2) alone is able to boost performance to a MAP of 0.368. Combined with a custom metric (L1+L2)

[1] http://homepages.inf.ed.ac.uk/rbf/CAVIAR

[2] http://cvhci.anthropomatik.kit.edu/projects/pri

Feature	using constraints			with feedback (iteration)					
	L1	L2	L3	Initial	1	2	3	4	5
SFR				0.2025	0.2737	0.3310	0.3587	0.3944	0.4309
DCT				0.2474	0.2977	0.3366	0.3744	0.4093	0.4290
CSD				0.2863	0.3873	0.4505	0.5076	0.5396	0.5717
CSD+SFR+DCT				0.3254	0.4131	0.4726	0.5129	0.5428	0.5694
CSD+SFR+DCT		✓		0.3683	0.4787	0.5315	0.5709	0.6039	0.6343
CSD+SFR+DCT	✓	✓		**0.3846**	0.4842	0.5558	0.5949	0.6252	0.6749
CSD+SFR+DCT	✓	✓	✓	0.3786	**0.4900**	**0.5669**	**0.6077**	**0.6406**	**0.6937**

Table 1: We present results in several steps of improvement. The combination of features CSD+SFR+DCT outperforms each of the features alone. Using the constraints on several levels (L1: Metric Learning, L2: Query-set enhancement, L3: Global retrieval optimization) provides each an incremental performance improvement.

Figure 4: Example retrieval results for the same query track: (top) Without using constraints and (bottom) using constraints at levels L1+L2+L3.

we achieve a MAP of 0.384, which is a relative improvement of 18% over the baseline. Adding the global ranking optimization (L1+L2+L3), the MAP drops slightly to 0.378, which is however still better than L2 alone. The global ranking optimization plays to its strength when user feedback is incorporated. Already after one iteration of user feedback, the full combination of L1+L2+L3 achieves a MAP of 0.490, consistently outperforming all other methods. After 5 rounds of feedback, L1+L2+L3 achieves a MAP of 0.694 which is an 9% relative improvement over L2, and a 22% relative improvement over the baseline. See Fig. 5 for Cumulative Matching Curves that further show the advantage of L1+L2+L3 over the baseline given user feedback.

The results clearly show that leveraging the formulated properties and their induced constraints is beneficial on multiple levels. Fig. 4 shows retrieval results for the same query track with and without using constraints. Observe that tracks from other cameras, with large variation in appearance, can be retrieved correctly due to the constraints. Fig. 6 shows retrieval results for a few additional sample queries (no feedback).

We also present some difficult examples and failure cases (Fig. 6, last row). One of the reasons for the problems with these queries is that our features are only computed from a relatively small region within the person's upper body (*c.f.* Sec. 4.2), and thus the resulting appearance description does not suffice for a correct retrieval. Since our proposed approach is independent of the employed features, a better appearance representation can easily be integrated.

6. Conclusion

We have formulated a set of properties of person tracks in camera networks. These properties induce constraints on the associated identities of the person tracks and we have shown how these constraints can be used on multiple levels for improving person retrieval in camera networks. We proposed to use the constraints for learning a discriminative metric, and formulated the retrieval task as an energy minimization problem. The experiments validate the benefit of our proposed methods, achieving a relative improvement of 22% over the baseline method. In future work, we would like to explore the adaption of the learned metric to specific queries, and the incorporation of both clothing and facial features in a common framework. This promises to allow retrieval over the course of multiple days which is not possible with full-body appearance-based features only.

Acknowledgments We thank the anonymous reviewers for their valuable comments. This work was partially funded by the German Federal Ministry of Education and Research (BMBF) under contract no. 01ISO9052E, as well as by OSEO, French State agency for innovation, as part of the Quaero Programme. The views expressed herein are the authors' responsibility and do not necessarily reflect those of BMBF or OSEO. We acknowledge the CAVIAR data set, created in the EC funded project/IST 2001 37540.

References

[1] D. Anguelov, K.-C. Lee, S. B. Gokturk, and B. Sumengen. Contextual Identity Recognition in Personal Photo Albums. In *CVPR*, 2007. 2

[2] S. Bak, E. Corvee, F. Bremond, and M. Thonnat. Multiple-shot Human Re-identification by Mean Riemannian Covariance Grid. In *AVSS*, 2011. 1, 2

[3] M. Bäuml, K. Bernardin, M. Fischer, H. Ekenel, and R. Stiefelhagen. Multi-Pose Face Recognition for Person Retrieval in Camera Networks. In *AVSS*, 2010. 1, 4

[4] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, 2005. 4

[5] G. Doretto, T. Sebastian, P. Tu, and J. Rittscher. Appearance-based Person Reidentification in Camera Networks: Problem Overview and Current Approaches. *Journal of Ambient Intelligence and Humanized Computing*, 2:127–151, 2011. 1, 2

[6] A. C. Gallagher and T. Chen. Clothing Co-segmentation for Recognizing People. In *CVPR*, 2008. 2

[7] M. Guillaumin, J. Verbeek, C. Schmid, and L. J. Kuntzmann. Is that you? Metric Learning Approaches for Face Identification. In *ICCV*, 2009. 2, 3

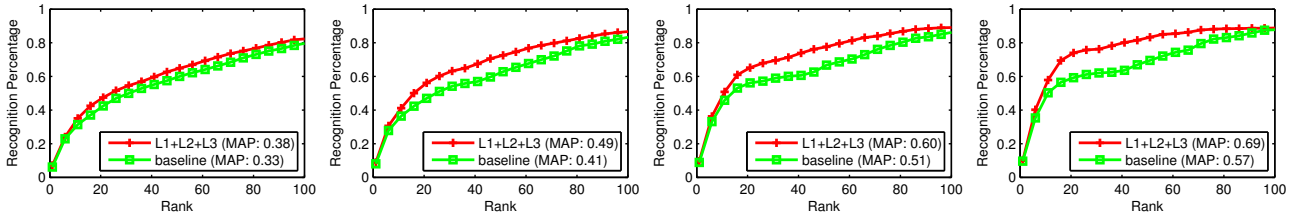


Figure 5: Cumulative Matching Curves for features CSD+SFR+DCT for increasing number of rounds of feedback (from left to right: 0, 1, 3 and 5 rounds of feedback). The retrieval with constraints can make better use of the feedback than the baseline.

Figure 6: Examples of retrieval results for sample queries (using constraints, but no feedback). Note that incorrect retrieval results often appear visually similar to the query track in texture or color. The last row shows some very difficult examples and failure cases.

[8] M. Hähnel, D. Klünder, and K.-f. Kraiss. Color and Texture Features for Person Recognition. In *International Joint Conference on Neural Networks*, 2004. 4

[9] C. Huang, B. Wu, and R. Nevatia. Robust Object Tracking by Hierarchical Association of Detection Responses. In *ECCV*, 2008. 2

[10] S. Khan and M. Shah. Tracking Multiple Occluding People by Localizing on Multiple Scene Planes. *PAMI*, 31(3):505–519, 2008. 2

[11] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Learning to Recognize Faces from Videos and Weakly Related Information Cues. In *AVSS*, 2011. 2

[12] C. Kuo and C. Huang. Multi-target tracking by online learned discriminative appearance models. In *CVPR*, 2010. 2

[13] D. C. Liu and J. Nocedal. On the limited memory BGFS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989. 3, 4

[14] M. Naaman, R. B. Yeh, H. Garcia-Molina, and A. Paepcke. Leveraging Context to Resolve Identity in Photo Albums. In *ACM Joint Conference on Digital Libraries*, 2005. 2

[15] N. O'Hare and A. F. Smeaton. Context-Aware Person Identification in Personal Photo Collections. *IEEE Transactions on Multimedia*, 11(2):220–228, 2009. 2

[16] J. Sivic, M. Everingham, and A. Zisserman. Person Spotting: Video Shot Retrieval for Face Sets. In *International Conference on Image and Video Retrieval*, 2005. 2

[17] Y. Song and T. Leung. Context-Aided Human Recognition-Clustering. In *ECCV*, 2006. 2

[18] D. a. Vaquero, R. S. Feris, D. Tran, L. Brown, A. Hampapur, and M. Turk. Attribute-based People Search in Surveillance Environments. In *WACV*, 2009. 2

[19] B. Yang, C. Huang, and R. Nevatia. Learning Affinities and Dependencies for Multi-target Tracking Using a CRF Model. In *CVPR*, 2011. 2

[20] T. Yu, Y. Yao, D. Gao, and P. Tu. Learning to Recognize People in a Smart Environment. In *AVSS*, 2011. 2