



## Learning Feature Aggregation in Temporal Domain for Re-Identification

Jakub Špaňhel<sup>a,\*\*</sup>, Jakub Sochor<sup>a,2</sup>, Roman Juránek<sup>a</sup>, Petr Doběš<sup>a</sup>, Vojtěch Bartl<sup>a</sup>, Adam Herout<sup>a</sup>

<sup>a</sup>*Graph@FIT, Brno University of Technology, Faculty of Information Technology, Centre of Excellence IT4Innovations*

### ABSTRACT

Person re-identification is a standard and established problem in the computer vision community. In recent years, vehicle re-identification is also getting more attention. In this paper, we focus on both these tasks and propose a method for aggregation of features in temporal domain as it is common to have multiple observations of the same object. The aggregation is based on weighting different elements of the feature vectors by different weights and it is trained in an end-to-end manner by a Siamese network. The experimental results show that our method outperforms other existing methods for feature aggregation in temporal domain on both vehicle and person re-identification tasks. Furthermore, to push research in vehicle re-identification further, we introduce a novel dataset CarsReId74k. The dataset is not limited to frontal/rear viewpoints. It contains 17,681 unique vehicles, 73,976 observed tracks, and 277,236 positive pairs. The dataset was captured by 66 cameras from various angles.

© 2019 Elsevier Ltd. All rights reserved.

### 1. Introduction

We consider the problem of re-identification of individuals observed by different cameras at different locations and times. Our work applies to the fairly standard person re-identification (Wang et al., 2014; Hirzer et al., 2011; Xu et al., 2017; Zhang et al., 2017b; Chen et al., 2017b; Zhou et al., 2017b), and to the rather emerging vehicle re-id (Liu et al., 2016a,c; Shen et al., 2017; Wang et al., 2017b; Yan et al., 2017; Zhang et al., 2017c), but it can be used for other similar tasks as well.

The re-id system is given a query track of images and a database of pre-stored tracks, one of which is assumed to share the same identity with the query. The system is supposed to output a small subset of the best matching database samples along with their similarity scores. Some solutions process the images in the tracks directly (comparing images in the query track versus images in the database – e.g. Zapletal and Herout (2016)). However, fast and real-time processing requires the system to extract a short feature vector for each of the database tracks and to match them to the feature vector extracted from the query track by computing a cheap pairwise metric. Our work is targeted on the second, generally more efficient, mode

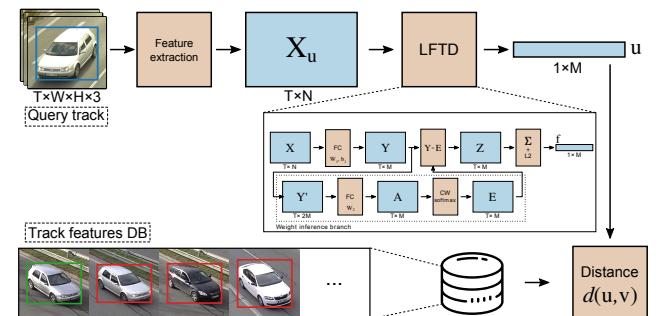


Fig. 1. We propose a new method LFTD for aggregation of features in temporal domain. The method generates one feature vector per track of observed objects (e.g. vehicles, persons). See Section 3.2 for details.

of processing, that is extraction of a single fixed-size feature vector for a track of variable length by aggregating the feature vectors extracted from individual observations (images).

We propose a new method for **feature aggregation in temporal domain** LFTD (Learning Features in Temporal Domain) which takes feature vectors extracted from the individual observations (images) as its input, and it results in a single relatively low-dimensional time-pooled feature vector usable by the re-id system. Unlike other methods which use either RNN (McLaughlin et al., 2016; Zhang et al., 2017b; Yan et al., 2016; Chen et al., 2017b; Xu et al., 2017; Zhou et al., 2017b; Zhang et al., 2017a) or produce weights for feature vectors as a whole

<sup>\*\*</sup>Corresponding author: Tel.: +420-54114-1298;  
e-mail: [ispanhel@fit.vutbr.cz](mailto:ispanhel@fit.vutbr.cz) (Jakub Špaňhel)

<sup>2</sup>Joint first author

(Yang et al., 2017; Zhou et al., 2017b; Xu et al., 2017), our method produces a different weight for every element of the feature vectors which leads to an improved performance as different parts of feature vectors are weighted differently. The weights are generated by a neural network for each set (track) of feature vectors. The final feature vector for the track is obtained by computing element-wise product between the track’s features and the weight matrix, and then reducing the matrix in temporal domain by summation. The results show that the proposed method outperforms other methods (Yan et al., 2016; McLaughlin et al., 2016; Gao et al., 2016; Xu et al., 2017; Zhang et al., 2017b; Chen et al., 2017b; Zhou et al., 2017b; Zhang et al., 2017a) in both vehicle and person re-identification tasks. See Figure 1 for the full re-id pipeline.

Furthermore, we propose to use a different metric for comparing the feature vectors. Previous works (Köstinger et al., 2012; Liao et al., 2015; Shi et al., 2016) showed that it is beneficial to use Mahalanobis distance for feature comparison rather than Euclidean (or cosine) distance. However, the Mahalanobis distance has significant limitations, mainly its time complexity which is quadratic with respect to feature vector dimensionality. Therefore, we propose to use **Weighted Euclidean** distance, constructed by constraining the Mahalanobis distance learning to diagonal matrix. The experiments show that it outperforms both Mahalanobis (Shi et al., 2016) and Euclidean distance, while it keeps linear time and memory complexity.

To improve the availability of datasets for vehicle re-identification, we collected and annotated a new vehicle re-identification dataset called **CarsReId74k**. As it is common in traffic surveillance to have whole tracks of vehicles and not individual images, the dataset includes multiple observations for each vehicle as it is passing in front of the cameras (*left, center, right*). We focus on **appearance-based** vehicle re-identification: vehicles’ license plates were only used for ground truth data acquisition (recorded by a *zoomed-in* camera). The images of vehicles taken by the other cameras are in most cases so small that it is not possible to recognize the license plates. The dataset contains 17,681 unique vehicles, 73,976 observed tracks, and 277,236 positive pairs, taken by 66 cameras from various angles in multiple sessions. We make the dataset publicly available<sup>3</sup> for future comparison and research.

## 2. Related Work

### 2.1. Image Feature Pooling in Temporal Domain

In this section, we provide an overview of existing methods for feature pooling (aggregation) in temporal domain. Such pooling is usually used in the context of person re-identification (with the exception of Yang et al. (2017) who used it for video face recognition). The methods are often trained by using a Siamese network (McLaughlin et al., 2016; Zhang et al., 2017b; Yan et al., 2016; Chen et al., 2017b; Xu et al., 2017; Yang et al., 2017) with contrastive loss and optionally identification loss as well.

McLaughlin et al. (2016) propose an approach for temporal domain pooling based on Recurrent Neural Networks (RNN). The authors extract features using a CNN and use a recurrent layer to compute the features for the whole track. The used RNN has an output for each time step and these outputs are averaged to obtain the final feature vector for re-identification. The authors further propose to use optical flow as an additional input to the network. A similar approach was proposed by Zhang et al. (2017b) with the exception that their method uses bi-directional RNN to get better re-id results. Also, the method proposed by Yan et al. (2016) is similar with the exception that the image level features are not trained and LBP and color features are used instead.

Chen et al. (2017b) also follow the work of McLaughlin et al. (2016). However, they propose to merge the features extracted by RNN together with CNN spatial features averaged over the time steps. The authors use three such networks for different body parts and fuse their output features (by a weighted sum).

Another approach based on the work by McLaughlin et al. (2016) is proposed by Xu et al. (2017), who introduce significant modifications to the method. First, image level features are extracted by spatial pyramid pooling; thus, spatial information is preserved in the feature vector. These features are then fed into a recurrent layer (similar to McLaughlin et al. (2016)). Finally, the recurrent features are pooled by an Attentive Temporal Pooling layer proposed by the authors. However, a significant drawback of the proposed method is that it requires both the query and the gallery raw feature vector sequences during distance computation, leading to more complex processing during the search in the database.

Zhang et al. (2017a) adds a feature pooling layer into the CNN architecture before the first fully connected layer. This layer aggregates key information from different views of the person’s trajectory (different time steps) in a single feature vector. They also incorporate two different learning distance metrics – minimum distance and average of minimum distance for comparing the query track with tracks in the database.

Unlike the other authors, Yang et al. (2017) focus on video face recognition. The authors propose an approach to temporal pooling based on weighting of feature vectors from different time steps. The weight for a feature vector is obtained as a dot product with a template, which is computed by a fully connected layer. The weights are then normalized to a probability distribution by softmax function. The weights for different time steps scale the contributions of images in the sequence according to their discriminative value.

Similarly, Zhou et al. (2017b) propose to use a temporal attention model and generate weights for feature vectors in the track. However, in contrast to Yang et al. (2017), the weights are generated at each time step for all the feature vectors in the sequence. Then, at all time steps, all feature vectors (from the given sequence) are weighted by a set of (different) weights; thus, at each time step, differently weighted input feature vectors are produced. The weights at each time step are obtained by a RNN layer. Furthermore, similarly to McLaughlin et al. (2016), the weighted feature vectors are fed into another RNN layer with output at each time step and then averaged to ob-

---

<sup>3</sup><https://medusa.fit.vutbr.cz/traffic>

tain the final track representation. The authors also use spatial RNNs to further improve the re-identification results.

Generally, for temporal pooling, the authors use either recurrent neural networks (McLaughlin et al., 2016; Zhang et al., 2017b; Yan et al., 2016; Chen et al., 2017b; Xu et al., 2017), learned weighting of feature vectors (Yang et al., 2017), or a combination of these approaches (Zhou et al., 2017b). In contrast to the described methods, the proposed method produces a different weight for every element of the feature vectors.

## 2.2. Person Re-Identification

Besides standard deep features learned by a Siamese network (McLaughlin et al., 2016; Zhang et al., 2017b; Yan et al., 2016; Chen et al., 2017b; Zhou et al., 2017b), other approaches to person re-identification have been proposed.

Several papers proposed to use body parts (Cheng et al., 2016; Khan and Brèmond, 2017; Li et al., 2017; Zhao et al., 2017). Other papers went beyond Siamese networks and proposed triplet loss (Cheng et al., 2016; Hermans et al., 2017) or quadruplet loss (Chen et al., 2017c). There were also attempts to learn a metric for the re-identification like KISSME (Köstinger et al., 2012), XQDA (Liao et al., 2015), You et al. (2016) learn Mahalanobis distance on LBP and HOG3D features, and finally Shi et al. (2016) learn Mahalanobis distance in an end-to-end manner. Sun et al. (2017) proposed to use SVD for weight matrix orthogonalization to de-correlate feature vectors for person re-id.

Other authors exploit different types of features. For example, Wu et al. (2016) propose to use deep features learned by a CNN together with hand-crafted features. The final representation for an image is obtained by fusing these features. Matsukawa et al. (2016) use a novel descriptor based on hierarchical gaussians computed for patches in image. Chen et al. (2017a) propose to use compact binary hash codes as features for fast person re-identification. There were also attempts (Liu et al., 2015; Gao et al., 2016) to recognize the walking cycle in image sequence and use the walking cycle to improve the accuracy of re-identification.

A group of works also propose to replace different parts of the re-identification pipeline by alternative solutions. Zhong et al. (2017) use re-ranking based on  $k$ -reciprocal nearest neighbors to improve the performance. Zhou et al. (2017a) propose to use point-to-set distance instead of standard point-to-point. Lin et al. (2017) take inter-camera consistencies of id assignment into account during training and inference to boost the results of re-identification. Xiao et al. (2016) propose to use domain guided dropout to improve re-identification performance when trained on multiple datasets. Wang et al. (2016) propose to add a network computing a cross-image representation for pairs of images. Cho and Yoon (2016) estimate persons' poses and compare images with each person in an as similar as possible pose. Su et al. (2016) use attributes (e.g "long sleeve") for person re-id. The attributes are first learned on a different dataset with attributes present and then fine-tuned for the target dataset. The attributes supervision for the target dataset comes from the assumption that same person has the same (unknown) attributes.

## 2.3. Vehicle Re-Identification

There are mainly two types of methods – methods based on automatic license plate recognition (Du et al., 2013; Kluwak et al., 2016; Wen et al., 2011), which are not anonymous and require zoomed-in cameras. The other type of methods is based on vehicles' visual appearance (Arth et al., 2007; Feris et al., 2012; Zapletal and Herout, 2016; Liu et al., 2016b) or on a combination of both approaches (Liu et al., 2016c).

Formerly, different types of *hand-crafted* features were used. For example authors used PCA-SIFT (Arth et al., 2007), HOG descriptors and color histograms (Zapletal and Herout, 2016), SIFT-BOW and Color Names model (Liu et al., 2016b) or just information about date, time, color, speed and vehicles' dimensions (Feris et al., 2012). Recently, *deep* features learned by CNNs (Liu et al., 2016a; Shen et al., 2017; Wang et al., 2017b; Yan et al., 2017; Zhang et al., 2017c) were used for this task. Liu et al. (2016c) combine the hand-crafted and deep features.

Improvements were also made by exploiting *spatio-temporal* (Liu et al., 2016c; Wang et al., 2017b) or *visual-spatio-temporal* (Shen et al., 2017) properties. Some of them benefit from Siamese CNNs for license plate verification (Liu et al., 2016c) or vehicle image similarities (Shen et al., 2017). Moreover, introduction of triplet loss (Zhang et al., 2017c) or Coupled Cluster Loss (CCL) (Liu et al., 2016a) led to accuracy improvements and faster convergence. Recently, Yan et al. (2017) propose to use Generalized Pairwise Ranking or Multi-Grain based List Ranking for retrieval of similar vehicles, which performs even better than CCL.

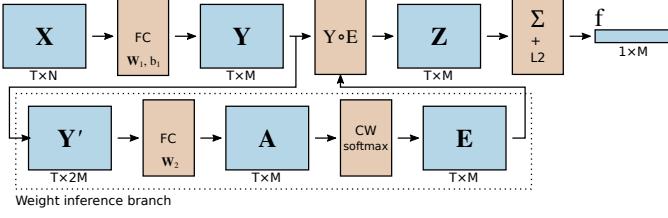
## 2.4. Vehicle Re-Identification Datasets

There are datasets of vehicles (Krause et al., 2013; Yang et al., 2015; Sochor et al., 2017), which are created for fine-grained recognition with annotations on several attributes such as type, make and color. However, the identities of the vehicles in the datasets are not known; thus, the datasets are not directly applicable for vehicle re-identification, especially for evaluation.

When it comes to genuine vehicle re-identification, Liu et al. (2016c) constructed a rather small VeRi-776 dataset containing 50,000 images of 776 vehicles. Liu et al. (2016a) collected VehicleID dataset containing 26,267 vehicles in 220k images taken from a frontal/rear viewpoint above road. Recently, Yan et al. (2017) published two datasets VD1 and VD2 for vehicle re-identification and fine-grained classification with over 220k of vehicles in total, with make, model, and year annotation. However, both datasets are limited to frontal viewpoints only.

## 3. Proposed Method for Learning Feature Aggregation in Temporal Domain

The standard baseline to aggregating features from multiple observations of the same object in temporal domain is to use averaging over time. However, existing literature (Yan et al., 2016; McLaughlin et al., 2016; Gao et al., 2016; Xu et al., 2017; Zhang et al., 2017b; Chen et al., 2017b; Zhou et al., 2017b) shows that the accuracy can be improved over the simple averaging by feature vector weighting or by using RNN. We propose a novel method for the aggregation in temporal domain,



**Fig. 2.** Schematic network design representing the proposed method for feature aggregation in temporal domain. See Section 3.2 for explanation of the symbols.

which is based on weighting different elements of the features vectors by different weights.

The proposed LFTD method aggregates arbitrary features from a sequence of images (of an arbitrary length), extracted by any feature extractor (it can be even some of newly presented spatial attention networks (Wang et al., 2017a; Su et al., 2017)) into a single fixed-sized feature vector. It allows to create a database of previously seen objects (with multiple observations) with such fixed-sized feature vectors and then quickly search the database for objects similar to query objects. LFTD expands the feature dimensions by concatenating the average feature vector to features extracted in every time step. It allows the network to propagate global information form the track to each individual observation. Feature vectors are weighted by column-wise softmax (i.e. along time axis) which forces the network to pick important observation for every feature in the vector instead of weighting observations as a whole. This network design performed the best during our preliminary experiments, compared with user-based vector normalization (subtracting or dividing features by average feature vector), or different types of feature expansion (e.g. by max-pooled feature vector, etc.).

The method is detailed in the following sections.

### 3.1. Image Feature Extraction

We are processing *the whole tracks* of objects of interest with labels corresponding to identities  $\{(\mathcal{T}_i, l_i)\}$ , where  $\mathcal{T}_i$  is a sequence of images  $(\mathbf{I}_1, \mathbf{I}_2, \dots, \mathbf{I}_{T_i})$ , i.e. observations of an object  $l_i$  in the track.

For each track (image sequence), features are extracted for each image independently by a feature extractor (a CNN-based or another one, the method is not limited by design to a particular type). The feature extractor yields a feature matrix  $\mathbf{X}_i \in \mathbb{R}^{T_i \times N}$  for each track  $\mathcal{T}_i$ .  $T_i$  is number of time samples (images) for each track  $\mathcal{T}_i$  and  $N$  is the length of an individual feature vector. In our experiments  $N = 2048$ , in case of ResNet50, and  $N = 1536$  for Inception-ResNet-v2.

To make the notation uncluttered, we will omit the lower index  $i$  from now on. Therefore, we will refer to a individual track as  $\mathcal{T}$ , the number of time samples of the track as  $T$ , and its features as  $\mathbf{X} \in \mathbb{R}^{T \times N}$ .

### 3.2. Processing of Features in Temporal Domain

The schematic design of the feature aggregation network is illustrated in Figure 2 and the description follows. Aggregation

of features  $\mathbf{X} \in \mathbb{R}^{T \times N}$  in temporal domain is essentially a mapping  $\varphi : \mathbb{R}^{T \times N} \mapsto \mathbb{R}^M$ , where  $M$  is the dimensionality of feature vector  $\mathbf{f}$  representing track  $\mathcal{T}$ .

First, the feature vector of each observation in the track is compressed from  $N$  to  $M$  dimensions ( $M < N$ ) by

$$\mathbf{y}_\tau = \tanh(\mathbf{W}_1 \mathbf{x}_\tau + \mathbf{b}_1), \quad 1 \leq \tau \leq T, \quad (1)$$

where  $\mathbf{W}_1 \in \mathbb{R}^{M \times N}$  are the parameters of the first fully connected layer (Figure 2), forming a compressed feature matrix  $\mathbf{Y} \in \mathbb{R}^{T \times M}$ .

In order to allow “communication” between the features across the track, we form a new feature matrix  $\mathbf{Y}' \in \mathbb{R}^{T \times 2M}$ , where each row contains the original feature vector in that row and an average feature vector for the whole track. Therefore  $\mathbf{Y}' = [\mathbf{y}'_1, \mathbf{y}'_2, \dots, \mathbf{y}'_T]^\top$ , where

$$\mathbf{y}'_\tau = \left[ \begin{array}{c} \mathbf{y}_\tau \\ \frac{1}{T} \sum_{i=1}^T \mathbf{y}_i \end{array} \right]. \quad (2)$$

From these feature vectors concatenated with the average feature vector, we generate activations by another fully connected layer  $\mathbf{a}_\tau = \mathbf{W}_2 \mathbf{y}'_\tau$ , forming matrix  $\mathbf{A} \in \mathbb{R}^{T \times M}$ . These activations are then normalized by softmax; however, the normalization is not done by rows (as usually), but by columns to normalize the activation for each component of the feature vector. Therefore, the normalization yields matrix  $\mathbf{E} \in \mathbb{R}^{T \times M}$ , where

$$e_{\tau j} = \frac{\exp(a_{\tau j})}{\sum_{i=1}^T \exp(a_{ij})}. \quad (3)$$

The weight matrix  $\mathbf{E}$  is then merged with the compressed feature matrix  $\mathbf{Y}$  by Hadamard (element-wise) product into matrix  $\mathbf{Z} = \mathbf{Y} \circ \mathbf{E}$ . The final feature vector  $\mathbf{f}$  is then obtained as a sum of feature vectors in rows of matrix  $\mathbf{Z}$ , normalized to a unit vector.

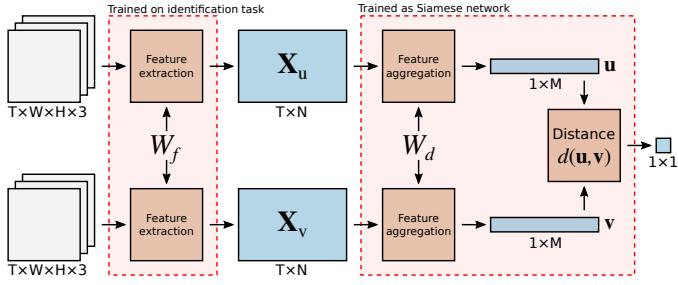
$$\mathbf{f} = \frac{\sum_{\tau=1}^T \mathbf{z}_\tau}{\|\sum_{\tau=1}^T \mathbf{z}_\tau\|_2} \quad (4)$$

Therefore, if matrix  $\mathbf{A}$  contained a single constant value, the aggregation would be reduced to one fully connected layer followed by average pooling. Instead, the weights  $\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1$  are trained by back-propagation and the network is thus able to produce better features.

### 3.3. Metrics for Distance Computation

The re-identification task is defined by a query sample (track) and a gallery of samples (tracks), where one sample from the gallery is supposed to have the same identity as the query sample. It is common to use Euclidean (or cosine for unit feature vectors) distance  $d_E(\mathbf{u}, \mathbf{v}) = \sqrt{\sum_i (u_i - v_i)^2}$  to rank the gallery samples by their distance from the query feature vector.

Previous works have shown that other distance metrics can outperform the Euclidean one, and the Mahalanobis distance seems to be powerful (Köstinger et al., 2012; Liao et al., 2015). Mahalanobis distance between vectors  $\mathbf{u}$  and  $\mathbf{v}$  is computed as  $\sqrt{(\mathbf{u} - \mathbf{v})^\top \mathbf{M} (\mathbf{u} - \mathbf{v})}$ , requiring that matrix  $\mathbf{M}$  is symmetric and positive semi-definite (Shi et al., 2016). They claim that such a constraint is hard to enforce and propose to decompose the



**Fig. 3.** Schematic design representing the full training and inference pipeline. In our approach, we train the image feature extractor NN on the identification task on the given dataset; however, the proposed method for feature aggregation can work with an arbitrary image feature extractor.  $W_f$  and  $W_d$  refer to the shared weights of feature extractor part and feature aggregation part, respectively.

matrix  $\mathbf{M} = \mathbf{W}\mathbf{W}^\top$  and learn  $\mathbf{W}$  instead. Then, the Mahalanobis distance is computed by the following equation:

$$d_M(\mathbf{u}, \mathbf{v}) = \sqrt{(\mathbf{u} - \mathbf{v})^\top \mathbf{W}\mathbf{W}^\top (\mathbf{u} - \mathbf{v})}. \quad (5)$$

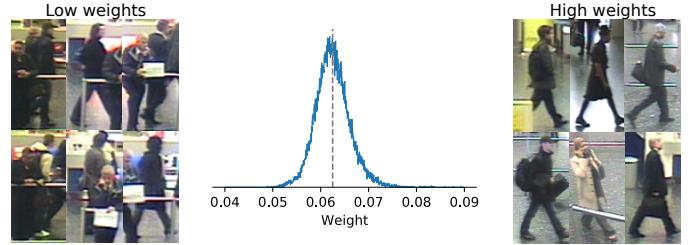
Using Mahalanobis distance as proposed by Shi et al. (2016) improves the re-identification accuracy, paying a high price in terms of its time complexity. Both time and memory asymptotic complexities are  $O(D^2)$  where  $D$  is the dimensionality of the feature vectors. This can cause significant problems in re-identification as the computational cost for quadratic time complexity is significantly larger even for  $D = 128$ . Therefore, we propose to learn suitable weights for Weighted Euclidean distance (equivalent to Mahalanobis distance when matrix  $\mathbf{M}$  is diagonal), instead. We express the Weighted Euclidean distance by

$$d_{WE}(\mathbf{u}, \mathbf{v}) = \sqrt{\sum_{i=1}^D w_i(u_i - v_i)^2}, \quad (6)$$

where  $\mathbf{w} = [w_1, w_2, \dots, w_D]$  are learned weights. It should be noted that if all the weights  $w_i$  are equal to 1, the metric is reduced to standard Euclidean distance. Before learning, we initialize the weights by randomly sampling from normal distribution with  $\mu = 1$  and  $\sigma = 0.1$ .

As the Weighted Euclidean distance can be interpreted as Mahalanobis distance with diagonal matrix  $\mathbf{M}$ , the same conditions must be kept. The symmetry is satisfied trivially as it is a diagonal matrix. However, to ensure the positive semi-definite property, we ensure that all the weights  $w_i$  are non-negative by clipping values below zero after each update of the weights during learning.

The Weighted Euclidean distance has benefits when compared to both standard Euclidean and Mahalanobis distances. Compared to the Euclidean distance, it has a higher expressive power thanks to learned weights  $\mathbf{w}$ . On the other hand, compared to full Mahalanobis distance, it is much faster as both time and memory complexity of the Weighted Euclidean distance is  $O(D)$ . At the same time, as the results in Section 5.1 show, our proposed Weighted Euclidean distance also outperforms both Euclidean and full Mahalanobis distance in terms of re-identification accuracy.



**Fig. 4.** Middle: Distribution of mean weights for test images in iLIDS-VID dataset (Wang et al., 2014). The dashed grey line denotes image weight for average pooling with  $T = 16$ . Sides: Images with the lowest and highest weights which show that low weight is usually assigned to images with occluding pedestrians.

### 3.4. Full Training and Inference Network

Both the feature aggregation network (Section 3.2) and the Weighted Euclidean metric (Section 3.3) are trained by a Siamese network (Hadsell et al., 2006), see Figure 3. For speeding up the training, we pre-train the feature extractor (Inception-ResNet-v1 (Szegedy et al., 2017) for vehicle re-id and ResNet50 (He et al., 2016) for person in our case) for the identification task using the dataset training data and then we cache all features for the tracks and train the feature aggregation and distance metric with the cached features. Training the network end-to-end did not improve the results further. We use a standard contrastive loss (Hadsell et al., 2006)

$$L(\mathbf{u}, \mathbf{v}, y) = y \cdot d(\mathbf{u}, \mathbf{v})^2 + (1 - y) \cdot [m - d(\mathbf{u}, \mathbf{v})]_+^2, \quad (7)$$

where  $\mathbf{u}$  and  $\mathbf{v}$  are feature vectors,  $m$  is the margin between negative samples,  $[\dots]_+$  denotes maximum value with zero, and  $y = 1$  if  $l_u = l_v$  or 0 otherwise ( $l_u$  and  $l_v$  are sample identities). Distance  $d$  is one of  $d_E$ ,  $d_M$ , or  $d_{WE}$  from the previous section.

### 3.5. Design Choices

We analyzed several design choices we made. During preliminary experiments we used ReLU nonlinearity in Equation (1) and found out that the results are significantly better with tanh nonlinearity.

Furthermore, on iLIDS-VID dataset (Wang et al., 2014), we tested how important different parts of the network are. In these experiments, 128 dimensional features were used (except the average pooling, where the features had 2048 dimensions). When only average pooling was used, we got Hit@1 46.3 % and with the full network Hit@1 is 61.4 %. However, if we use only the weighting mechanism (omit feature projection by (1)), the Hit@1 is 51.6 %. And finally, if we use average pooling (omit the weighting mechanism) with the feature projection (1), we receive Hit@1 56.7 %. This shows that both parts of the network contribute to the accuracy and the contributions can be merged to obtain better results. A graphical comparison of design choices evaluation can be found in Figure 5. Full results of design choices evaluation for different Hit@Rank can be found in Table 1.

Finally, we analyzed the mean weights for different images and the distribution of mean weights together with images with

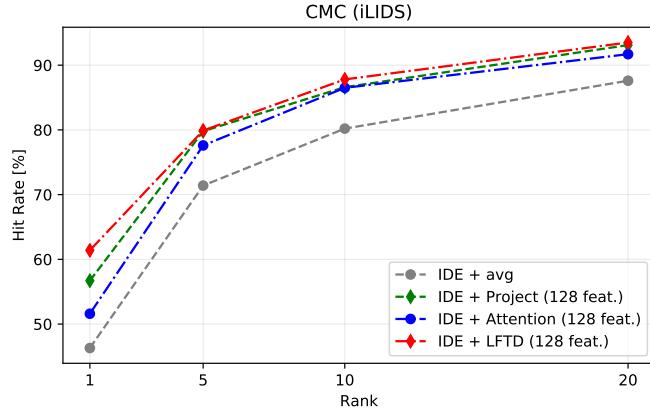


Fig. 5. CMC curve on the iLIDS dataset for individual parts of the proposed network.

Table 1. Evaluation of individual parts of the proposed network on the iLIDS dataset.

Method	Hit@R (iLIDS)			
	1	5	10	20
IDE + avg	46.3	71.4	80.2	87.6
IDE + Project (128 feat.)	56.7	79.8	86.6	93.1
IDE + Attention (128 feat.)	51.6	77.6	86.5	91.7
IDE + LFTD (128 feat.)	61.4	79.9	87.8	93.5

lowest and highest weights can be found in Figure 4. The results show that the weights are centered around  $1/T$  (i.e. average pooling weight) which was expected. Also, low weights are usually assigned to images with occluding objects or pedestrians.

Furthermore, we analyzed the homogeneity of the weights for individual observations (i.e. how much the weights differ within one observation). The mean relative standard deviation is 0.34; the weights therefore differ significantly.

#### 4. Novel Vehicle Re-Identification Dataset CarsReId74k

We focus on vehicle re-identification and we want to differentiate even vehicles with the same fine-grained type but different identities (different license plates). Therefore, we cannot use fine-grained vehicle recognition datasets (Sochor et al., 2016, 2017; Yang et al., 2015; Krause et al., 2013) for the task. As other existing vehicle re-identification datasets VeRi-776 (Liu et al., 2016c), VehicleID (Liu et al., 2016a) and PKU-VDS (Yan et al., 2017) are either small (VeRi-776) or limited to frontal/rear viewpoints (VehicleID, PKU-VDS). We collected a novel dataset **CarsReId74k** which does not have these limitations. The data were collected by 66 cameras from various angles and the dataset contains almost 74 k of vehicle tracks with precise identity annotation (acquired from license plates). More detailed comparison of different available vehicle re-identification datasets can be found in Table 2.

##### 4.1. Dataset Acquisition

The dataset was collected in multiple sessions. In each session, we placed four cameras on a bridge overlooking a freeway and four cameras on another bridge in vehicles' traveling

Table 2. The comparison of various vehicle re-id datasets.

\* – Tracks are not guaranteed for each unique vehicle.

† – Unique vehicles from each dataset part can overlap. The total number of unique vehicles is probably lower.

	CarsReId74k	VehicleID	VeRi-776	PKU-VDS
# unique vehicles	17,681	26,267	776	†221,519
# tracks	73,976	—	6,822	N/A
# images	3,242,713	221,763	51,035	1,354,876
viewpoints	various	front/rear	various	front
multiple images in track	yes	no	*yes	yes

direction. Figure 6 illustrates the recording setup and Figure 7 shows example frames from one such session. The videos were recorded for  $\sim 1$  hour and synchronized. One of the cameras was zoomed in enough to be able to read the license plates of all the passing vehicles (Figure 7 left). The other three cameras were placed so that they observed the road from left, center, and right position.

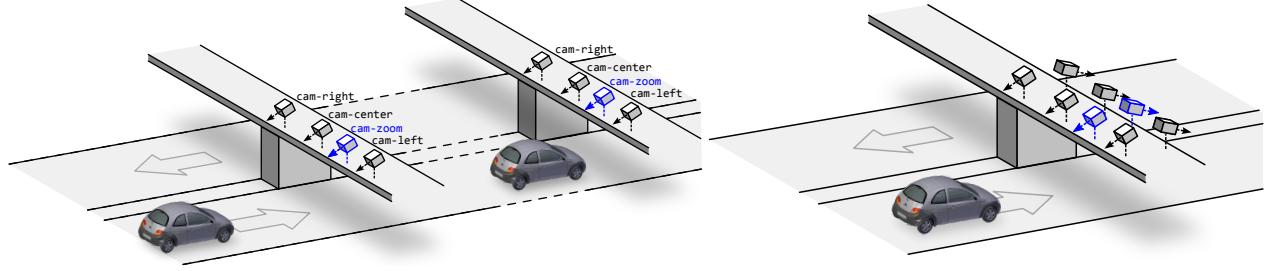
We used the zoomed-in videos to identify the passing vehicles. We detected the license plates by an ACF detector (Dollár et al., 2014), tracked them, recognized by a recent method (Špaňhel et al., 2017) and manually verified in order to eliminate any recognition errors. We also assigned a lane to each license plate track for easier matching. On all the other videos (left, center, right), we detected and tracked the vehicles. We also constructed 3D bounding boxes (Dubská et al., 2014) around the vehicles as Sochor et al. (2016) showed that the 3D bounding boxes were beneficial for fine-grained recognition. We also assigned the lane for each of these vehicles (Dubská et al., 2014). Finally, we matched the vehicles from the zoomed-in cameras (with known identities) to vehicles from the other cameras. We omitted all the vehicles which were not matched. It should be noted that the vehicles in the dataset from non-zoomed-in cameras have almost unreadable license plates; therefore, the dataset is suitable for **appearance-based** vehicle re-identification, preserving the anonymity of the vehicles.

##### 4.2. Dataset Statistics

The dataset was recorded in 11 sessions at different locations. We divided the dataset into the training, the testing and the validation part by sessions (five sessions for training, five sessions for testing and one validation). The total dataset statistics can be found in Table 3. The table shows that our dataset is significantly larger than VeRi-776 (Liu et al., 2016c) dataset with only 776 unique vehicles. And compared to VehicleID, VD1 and VD2 datasets (Liu et al., 2016a; Yan et al., 2017), our dataset is not limited to frontal/rear viewpoints. Compared to VehicleID dataset, CarsReId74k dataset has fewer unique vehicles (17,681 vs. 26,267), however far more image (3,242,713 vs. 221,763) as vehicles are seen from more viewpoints.

##### 4.3. Proposed Evaluation Protocol

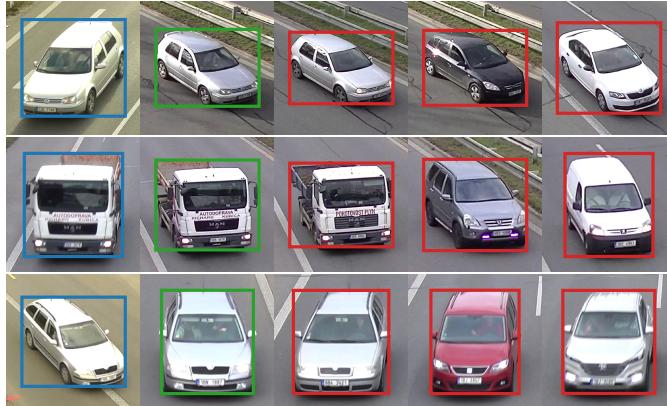
For each part (training, testing and validation), we collected all the pairs of tracks with the same vehicle identity (marked as *query, positive*). The query and positive tracks are always from different videos; however, they can come from the same



**Fig. 6.** Recording setup for acquisition of novel CarsReId74k dataset. We simultaneously recorded data on two bridges by multiple cameras. One camera on each bridge was zoomed in so that it is possible to automatically recognize license plates and use them for the construction of the ground truth labeling (left image). For part of dataset vehicles were captured from single bridge on both sides, which yields to capture observed vehicles from frontal and rear viewpoints (right image).



**Fig. 7.** Frames from all cameras in one session. The license plates acquired from the *zoom* camera (left) were used for ground truth re-identification (silver car). Each row shows frames from one location within the session.



**Fig. 8.** Examples of **queries**, **positive**, and **negative** samples. The negatives are sorted by difficulty from left to right (hard to easy) based on distances obtained from our re-identification feature vectors. It should be noted that the hardest negative sample has usually subtle differences (e.g. missing a small spoiler in the first row).

session and location (e.g. left – right), from the same session and different location, or (in rare cases) also from different sessions within the training (or testing) set. This yields a significant number of positive pairs (277,236 in total). As the negative pairs, we use all other vehicle tracks in the same video as the positive track with the exception of vehicle tracks with the same identity as the positive track (a vehicle could be observed multiple times in one video). This yields a mean number of 1283 negative vehicle tracks per positive pair. See Figure 8 for examples of positive and negative pairs.

Following other papers (Liu et al., 2016a; Yan et al., 2017; Liu et al., 2016c; Hirzer et al., 2011; Wang et al., 2014) on re-identification, we use **mAP** and **hit at rank** as the metrics for evaluation on the dataset. We encourage others to report

hit rates at ranks 1, 5, 10, and 20 together with Cumulative Matching Curve for ranks 1 to 20.

## 5. Experimental Results

We evaluate our method on the vehicle and person re-identification tasks on multiple public datasets to show that the aggregation performs well on various classes of data. Datasets for evaluation were chosen considering the availability of tracks (multiple observations) of each object’s identity in the dataset because this work proposes a method for aggregation of features in the time domain and variable camera viewpoints.

### 5.1. Vehicle Re-Identification

Currently available datasets does not fit conditions described before at least in one condition (see Sec. 4), thus vehicle re-identification task was evaluated on our novel CarsReId74k only.

For feature extraction from images we use **Inception-ResNet-v2** (Szegedy et al., 2017) with images resized to  $331 \times 331$  yielding feature vectors with length 1536 for each input image. Sochor et al. (2016, 2017) showed that unpacking the input vehicle by 3D bounding box and alternating the input image colors is beneficial for fine-grained recognition of vehicles; we use these modifications for re-identification of vehicles as well.

The feature extractor was fine-tuned on the identification task using the training part of the CarsReId74k dataset. The fine-tuning was done with Adam optimizer, learning rate 0.0001, batch size 4 for 300 epochs with standard augmentation techniques (random flip and shift of the bounding box).

When it comes to feature aggregation in temporal domain, we compare several methods with the following naming conventions:

- **avg** – standard average pooling of feature vectors,
- **RNN** – method proposed by McLaughlin et al. (2016) based on recurrent neural network,
- **NAN** – Neural Aggregation Network proposed by Yang et al. (2017),
- **LFTD** – our method (short for Learning Features in Temporal Domain).

**Table 3.** CarsReId74k dataset statistics. \*The total number of unique vehicles is lower than the sum of unique vehicles from training, test and validation set because a small number of vehicles appear in all sets (same car present at two or more recording sessions by accident). †Number of negative pairs = mean number of negative pairs per positive pair.

	training	test	validation	total
# cameras	30	30	6	66
# unique vehicles*	7,658	9,678	1,100	17,681
# tracks	32,163	36,535	5,278	73,976
# images	1,469,494	1,467,680	305,539	3,242,713
# positive pairs	125,086	129,774	22,376	277,236
# negative pairs†	1,149	1,459	881	1283

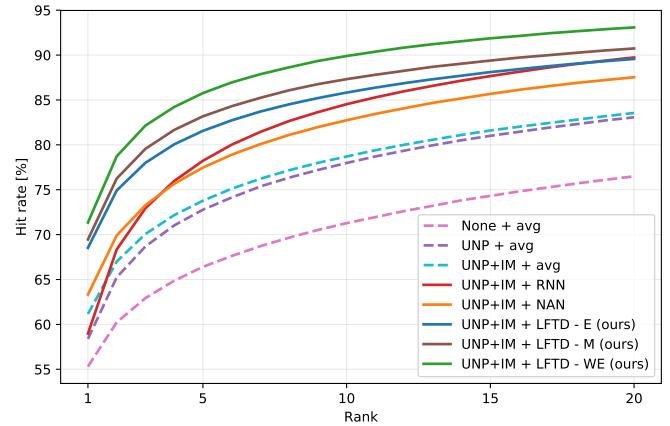
**Table 4.** Results for different methods for vehicle re-identification on CarsReId74k dataset. The methods use 128 dimensional feature vectors with the exception of avg which uses 1536 dimensional feature vectors. The methods use Euclidean distance with the exception of LFTD – M (full Mahalanobis (Shi et al., 2016)) and LFTD – WE (Weighted Euclidean as proposed in Section 3.3). Input modifiers – UNP, UNP+IM (Sochor et al., 2017). Aggregation methods – RNN (McLaughlin et al., 2016), NAN (Yang et al., 2017).

Input Modif.	Aggregation	mAP	Hit@Rank			
			1	5	10	20
None	avg	0.608	55.3	66.4	71.3	76.5
UNP	avg	0.652	58.4	72.8	78.0	83.1
UNP+IM	avg	0.672	61.2	73.8	78.7	83.5
UNP+IM	RNN	0.678	59.0	78.2	84.5	89.7
UNP+IM	NAN	0.700	63.3	77.5	82.7	87.5
UNP+IM	LFTD	0.746	68.5	81.6	85.8	89.6
UNP+IM	LFTD – M	0.757	69.5	83.2	87.3	90.7
UNP+IM	LFTD – WE	<b>0.779</b>	<b>71.3</b>	<b>85.8</b>	<b>89.9</b>	<b>93.1</b>

To make the comparison fair, we always compare the methods with features of the same length (128 dimensional features by default). The only exception is average pooling where the final features are always 1536 dimensional. As NAN (Yang et al., 2017) does not reduce the number of features, we added a trainable fully connected layer between the feature extractor and the aggregation network. As both RNN (McLaughlin et al., 2016) and NAN (Yan et al., 2017) use Euclidean distance in the original design, we evaluate the networks with the Euclidean distance. Following other previous works (McLaughlin et al., 2016; Zhang et al., 2017b; Chen et al., 2017b; Xu et al., 2017; Zhou et al., 2017b), we fix the number of time samples to  $T = 16$ .

We also compare different metrics for comparison of the feature vectors. The standard Euclidean distance is used as the baseline. We also use the full Mahalanobis distance (as proposed by Shi et al. (2016)) – shortened as **M**; and our Weighted Euclidean distance – shortened as **WE**. The full Mahalanobis distance was trained with regularization term  $0.5\lambda||\mathbf{WW}^\top - \mathbf{I}||_F^2$  as proposed by the authors (Shi et al., 2016) with  $\lambda = 0.01$ .

To increase the training speed, all the aggregation networks were trained on cached features extracted by the Inception-ResNet-v2 feature extractor. The networks were trained in Siamese settings for 30 epochs with batch size 32 on train and validation set. We employed hard negative mining during the training and all positive pairs and one hardest negative pair per positive pair were presented to the network in one epoch during the training. For the RNN (McLaughlin et al., 2016), we used original hyperparameters as proposed in the paper (SGD,



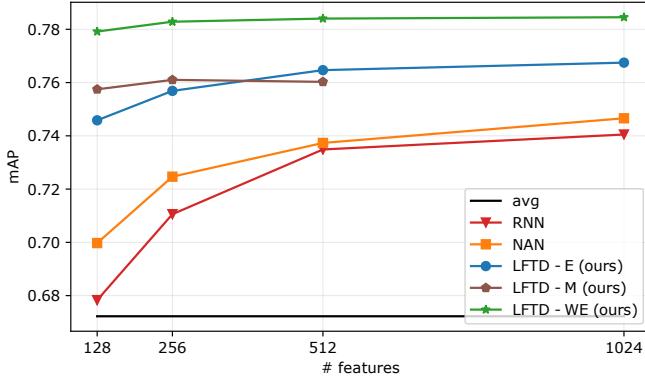
**Fig. 9.** Cumulative Matching Curve for different methods for vehicle re-identification on CarsReId74k dataset. The methods use 128 dimensional feature vectors with the exception of avg which uses 1536 dimensional feature vectors. The methods use Euclidean distance with the exception of LFTD – M (full Mahalanobis (Shi et al., 2016)) and LFTD – WE (Weighted Euclidean as proposed in Section 3.3).

lr: 0.001, margin: 2); changing them did not improve the accuracy further. We were forced to change the hyperparameters for NAN (Yang et al., 2017) to different values than used in the paper as the network did not converge with the original ones. We used RMSprop optimizer with learning rate 1e-6, and margin 1; different hyperparameters did not improve the accuracy further. Our method LFTD was trained by Adam optimizer with learning rate 1e-5 (1e-4.4 in the case of Mahalanobis and Weighted Euclidean distance) and margin 2.

The vehicle re-identification results can be found in Table 4 and Cumulative Matching Curve (CMC) is shown in Figure 9. The results show several things. First, both the Unpack (UNP) (Sochor et al., 2016) modification and image modifications (IM) (Sochor et al., 2017) improve the accuracy of vehicle re-identification. Second, all feature aggregation methods in the temporal domain (RNN (McLaughlin et al., 2016), NAN (Yang et al., 2017), LFTD) improve the accuracy when compared with the average pooling in the task of vehicle re-identification. Third, our method (LFTD) outperforms other methods for temporal aggregation (RNN (McLaughlin et al., 2016), NAN (Yang et al., 2017)). Finally, using other metrics than Euclidean also improves the accuracy. Our proposed Weighted Euclidean distance significantly outperforms the full Mahalanobis distance (as proposed by Shi et al. (2016)); and at the same time, our method has significantly lower time demands. It has time and memory complexity  $O(D)$  instead of  $O(D^2)$  for the full Mahalanobis distance, where  $D$  is the dimensionality of the feature vectors.

Our explanation of better performance of Weighted Euclidean distance instead of Mahalanobis distance is that there is not enough training data to train the full matrix **M**. This hypothesis is supported by Fig. 10 where the performance with Mahalanobis distance does not increase and by the fact that  $\frac{\text{tr}(\mathbf{M})}{\sum |\mathbf{M}|} = 0.997$ , i.e. almost all the information in the matrix is on its diagonal.

We were also curious how the accuracy changes with increas-



**Fig. 10.** Performance analysis of different methods for feature vector aggregation in temporal domain with changing number of features on CarsReId74k dataset. The avg pooling is shown only for visual comparison and uses 1536 dimensional feature vectors. We omitted LFTD - M with 1024 features from evaluation because of long evaluation time (months) and performance drop of version with 512 features. All the methods use Euclidean distance with the exception of LFTD - M (full Mahalanobis (Shi et al., 2016)) and LFTD - WE (Weighted Euclidean as proposed in Section 3.3).

ing the dimensionality of the feature vectors. As Figure 10 shows, all methods improve with increasing dimensionality; however, the results are still similar. Our method LFTD with our proposed Weighted Euclidean distance is outperforming all other methods for all of the tested feature vector dimensionalities.

## 5.2. Person Re-Identification

To show that our method is applicable also outside the scope of vehicle re-identification, we evaluated it on the person re-identification task. We use two common datasets: iLIDS-VID (Wang et al., 2014) and PRID-2011 (Hirzer et al., 2011) as they are usually used by other methods for feature aggregation in temporal domain (Yan et al., 2016; McLaughlin et al., 2016; Gao et al., 2016; Xu et al., 2017; Zhang et al., 2017b; Chen et al., 2017b; Zhou et al., 2017b). Furthermore, for fair comparison of proposed method, our work was also evaluated on the MARS dataset by Zheng et al. (2016).

It should be noted that the subject of our study is the aggregation of features extracted on images by an existing feature extractor. That is why we include in the comparison those methods that do the same, not methods which use a significantly different method of image **feature extraction**.

For the above reasons, we are not comparing our method with some of the currently published methods such as QAN (Liu et al., 2017) or SpaAttn (DRSTA) (Li et al., 2018) as they are focusing on *spatiotemporal* attention pooling, and because of that, they provide enhanced feature extraction. In our work, we target fusion of existing feature extractors. Besides that, their evaluation is not following the standard evaluation protocol used in the previous works and with the used datasets, as they are pre-training the networks on different types of image-based person re-identification tasks, thus their results are hardly comparable.

**Table 5.** Person re-identification results on PRID-2011 and iLIDS-VID dataset. The top-3 results are highlighted in the following way: **first**, **second**, and **third**. KISSME – Köstinger et al. (2012), XQDA – Liao et al. (2015). LFTD metric used for this experiment is the standard Euclidean distance because of insufficient amount of training data for the Weighted Euclidean.

Method	Hit@R (PRID)				Hit@R (iLIDS)			
	1	5	10	20	1	5	10	20
Yan et al. (2016)	58.2	85.8	93.4	97.9	49.3	76.8	85.3	90.0
McLaughlin et al. (2016)	70.0	90.0	95.0	97.0	58.0	84.0	91.0	96.0
Gao et al. (2016)	68.6	<b>94.6</b>	<b>97.4</b>	98.9	55.0	<b>87.5</b>	<b>93.8</b>	<b>97.2</b>
Xu et al. (2017)	77.0	<b>95.0</b>	<b>99.0</b>	<b>99.0</b>	62.0	<b>86.0</b>	<b>94.0</b>	<b>98.0</b>
Zhang et al. (2017b)	72.8	92.0	95.1	97.6	55.3	85.0	<b>91.7</b>	95.1
Chen et al. (2017b)	77.0	93.0	95.0	98.0	61.0	85.0	<b>94.0</b>	<b>97.0</b>
Zhou et al. (2017b)	<b>79.4</b>	<b>94.4</b>	—	<b>99.3</b>	55.2	<b>86.5</b>	—	<b>97.0</b>
Zhang et al. (2017a)	60.2	85.1	—	94.2	83.3	93.3	—	96.7
avg	69.4	90.5	95.0	97.6	46.3	71.4	80.2	87.6
avg + KISSME	70.5	91.0	95.1	97.7	56.1	79.0	87.9	93.9
avg + XQDA	75.6	94.3	<b>98.2</b>	<b>99.0</b>	59.5	83.7	90.3	96.2
LFTD (128)	79.2	92.4	95.8	98.4	61.4	79.9	87.8	93.5
LFTD (256)	<b>79.4</b>	93.7	96.8	98.6	<b>62.8</b>	82.1	88.1	94.1
LFTD (512)	<b>80.2</b>	<b>94.6</b>	97.3	98.9	<b>63.5</b>	83.3	89.5	94.9
LFTD (1024)	<b>80.0</b>	93.9	<b>97.4</b>	<b>99.2</b>	<b>63.7</b>	82.9	90.0	94.7

## iLIDS-VID and PRID-2011

We always used a half of the dataset for training and the other half for testing. Therefore, the evaluation is done on 100 tracks (150 tracks) with PRID-2011 (iLIDS-VID) dataset. We used 10 random splits in the case of the PRID-2011 dataset, and the 10 published splits in the case of iLIDS-VID.

We used ResNet50 (He et al., 2016) as the feature extractor from the images and trained it on the identification task by Adam optimizer with learning rate 0.0001 for 60 epochs with batch size 8, using standard augmentation techniques (random flip, rotation, and shift). We trained our method (LFTD) in a Siamese network by Adam optimizer with cross-validated learning rate for 150 epochs with batch size 8. We always used 16 time samples per track and contrastive loss margin 2. We also evaluate the average pooling with KISSME (Köstinger et al., 2012) and XQDA (Liao et al., 2015) with cross-validated hyperparameters (regularization, and PCA reduction dimensionality in the case of KISSME).

We used standard Euclidean distance as the metric for our algorithm, because the number of training data in the datasets is rather low and the accuracy did not improve further with other distances. This is caused mainly by insufficient amount of training data because the network was able to re-identify the training tracks without any error already with the standard Euclidean distance.

The results can be found in Table 5 and as the table shows, LFTD significantly increases the performance compared to average pooling or average pooling with other metric learning (KISSME, XQDA). The results also show that our method outperforms other methods for feature aggregation in temporal domain (Yan et al., 2016; McLaughlin et al., 2016; Gao et al., 2016; Xu et al., 2017; Zhang et al., 2017b; Chen et al., 2017b; Zhou et al., 2017b) in Hit@1.

Evaluation of KISSME or XQDA metrics together with the features produced by the proposed LFTD method is not included in the results because of lacking relevancy of such comparison. LFTD produces features dependent on the metric used

**Table 6. Person re-identification results on MARS dataset.** Baseline is the variant (*IDE, average pooling, Euclidean distance, single query*) reported by authors of the dataset (Zheng et al., 2016).

\* - RNN-CNN (McLaughlin et al., 2016) trained by Xu et al. (2017). [R2: - Experiments on Mars lack of recent baselines.]

Variant	mAP	Hit@Rank			
		1	5	10	
Baseline	0.424	60.0	77.9	-	87.9
RNN-CNN* (Xu et al., 2017)	-	40.0	64.0	70.0	77.0
ASTPN (Xu et al., 2017)	-	44.0	70.0	74.0	81.0
Zhang et al. (2017a)	-	55.5	70.2	-	80.2
LFTD - E (512)	0.481	65.5	80.3	85.5	89.4
LFTD - E (1024)	0.483	65.9	80.7	84.8	89.2
LFTD - WE (512)	0.488	66.1	81.0	85.4	89.8
LFTD - WE (1024)	0.489	66.4	81.5	85.9	89.8

during training and it generates different feature vector representations for different metrics involved (Euclidean, Weighted Euclidean, Mahalanobis).

#### MARS dataset

Features published by the authors of the dataset were used in our experiments. The network was trained on the training part of the published features in the Siamese setting for 30 epochs with batch size 32 with Contrastive Loss and Adam optimizer. Hard negative mining was employed during the training, and all positive pairs and 20 hardest negative pairs were presented to the network in one epoch during the training. Values of learning rate and loss margin were fine-tuned for each variant individually. All variants of our method evaluated on the dataset can be found in Table 6. It should be noted that *Baseline* is the variant (*IDE, average pooling, Euclidean distance, single query*) reported by the authors Zheng et al. (2016).

## 6. Conclusions

We proposed a new scheme for extracting feature vectors for the whole tracks of multiple observations of an object (vehicle, person) of interest in the re-identification task. Our method can work with arbitrary per-image features (e.g. feature vectors from ResNet50 or Inception-ResNet-v2). Based on such feature vectors we learn a considerably shorter (128 features) per-track feature vector by using the newly proposed LFTD (Learning Features in Temporal Domain). We also propose to use a different distance metric for comparing the feature vectors – WE (Weighted Euclidean). It is based on the Mahalanobis distance, whose learned matrix  $\mathbf{M}$  is made diagonal. This proposed distance metric is much cheaper in terms of computational and memory resources ( $O(D)$  instead of  $O(D^2)$  in the case of the full Mahalanobis metric), but at the same time, it is better at solving the re-identification task.

The results show that the increase of HIT@1 by using the LFTD was 7.3 percentage points for the vehicle re-identification task compared to average pooling, and 17.4 percentage points for the person re-identification with the iLIDS-VID dataset and up to 6.4 percentage points on the MARS dataset. The Weighted Euclidean metric further increased HIT@1 by other 2.8 percentage points in case of vehicle re-identification.

We collected and annotated a vehicle re-identification dataset CarsReId74k for development and evaluation of vehicle re-identification systems and we make it public. It contains 17,681 unique vehicles, 73,976 observed tracks, and 277,236 positive pairs, taken from various angles – not just from the front or rear.

## Acknowledgments

This work was supported by TACR project “SMARTCarPark”, TH03010529. Also, this work was supported by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II); project IT4Innovations excellence in science - LQ1602.

## Notice

This paper is under consideration at Computer Vision and Image Understanding

## References

- Arth, C., Leistner, C., Bischof, H., 2007. Object reacquisition and tracking in large-scale smart camera networks, in: 2007 First ACM/IEEE International Conference on Distributed Smart Cameras, IEEE. pp. 156–163.
- Chen, J., Wang, Y., Qin, J., Liu, L., Shao, L., 2017a. Fast person re-identification via cross-camera semantic binary transformation, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Chen, L., Yang, H., Zhu, J., Zhou, Q., Wu, S., Gao, Z., 2017b. Deep spatial-temporal fusion network for video-based person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.
- Chen, W., Chen, X., Zhang, J., Huang, K., 2017c. Beyond triplet loss: A deep quadruplet network for person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Cheng, D., Gong, Y., Zhou, S., Wang, J., Zheng, N., 2016. Person re-identification by multi-channel parts-based cnn with improved triplet loss function, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Cho, Y.J., Yoon, K.J., 2016. Improving person re-identification via pose-aware multi-shot matching, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Dollár, P., Appel, R., Belongie, S., Perona, P., 2014. Fast feature pyramids for object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 36, 1532–1545. doi:[10.1109/TPAMI.2014.2300479](https://doi.org/10.1109/TPAMI.2014.2300479).
- Du, S., Ibrahim, M., Shehata, M., Badawy, W., 2013. Automatic license plate recognition (ALPR): A state-of-the-art review. IEEE Trans. on Circuits and Systems for Video Technology 23, 311–325.
- Dubská, M., Sochor, J., Herout, A., 2014. Automatic camera calibration for traffic understanding, in: BMVC.
- Feris, R.S., Siddique, B., Petterson, J., Zhai, Y., Datta, A., Brown, L.M., Pankanti, S., 2012. Large-scale vehicle detection, indexing, and search in urban surveillance videos. IEEE Trans. on Multimedia 14, 28–42.
- Gao, C., Wang, J., Liu, L., Yu, J.G., Sang, N., 2016. Temporally aligned pooling representation for video-based person re-identification, in: 2016 IEEE International Conference on Image Processing (ICIP), pp. 4284–4288. doi:[10.1109/ICIP.2016.7533168](https://doi.org/10.1109/ICIP.2016.7533168).
- Hadsell, R., Chopra, S., LeCun, Y., 2006. Dimensionality reduction by learning an invariant mapping, in: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06), pp. 1735–1742. doi:[10.1109/CVPR.2006.100](https://doi.org/10.1109/CVPR.2006.100).
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Hermans, A., Beyer, L., Leibe, B., 2017. In defense of the triplet loss for person re-identification. arXiv:1703.07737. [arXiv:arXiv:1703.07737](https://arxiv.org/abs/1703.07737).

- Hirzer, M., Beleznai, C., Roth, P.M., Bischof, H., 2011. Person Re-Identification by Descriptive and Discriminative Classification, in: Proc. Scandinavian Conference on Image Analysis (SCIA).
- Khan, F.M., Brèmond, F., 2017. Multi-shot person re-identification using part appearance mixture, in: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 605–614. doi:[10.1109/WACV.2017.73](https://doi.org/10.1109/WACV.2017.73).
- Kluwak, K., Segen, J., Kulbacki, M., Drabik, A., Wojciechowski, K., 2016. ALPR - Extension to Traditional Plate Recognition Methods. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 755–764. URL: [http://dx.doi.org/10.1007/978-3-662-49390-8\\_73](http://dx.doi.org/10.1007/978-3-662-49390-8_73), doi:[10.1007/978-3-662-49390-8\\_73](https://doi.org/10.1007/978-3-662-49390-8_73).
- Krause, J., Stark, M., Deng, J., Fei-Fei, L., 2013. 3D object representations for fine-grained categorization, in: The IEEE International Conference on Computer Vision (ICCV) Workshops.
- Köstinger, M., Hirzer, M., Wohlhart, P., Roth, P.M., Bischof, H., 2012. Large scale metric learning from equivalence constraints, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2288–2295. doi:[10.1109/CVPR.2012.6247939](https://doi.org/10.1109/CVPR.2012.6247939).
- Li, D., Chen, X., Zhang, Z., Huang, K., 2017. Learning deep context-aware features over body and latent parts for person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Li, S., Bak, S., Carr, P., Wang, X., 2018. Diversity regularized spatiotemporal attention for video-based person re-identification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 369–378.
- Liao, S., Hu, Y., Zhu, X., Li, S.Z., 2015. Person re-identification by local maximal occurrence representation and metric learning, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Lin, J., Ren, L., Lu, J., Feng, J., Zhou, J., 2017. Consistent-aware deep learning for person re-identification in a camera network, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Liu, H., Tian, Y., Yang, Y., Pang, L., Huang, T., 2016a. Deep relative distance learning: Tell the difference between similar vehicles, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Liu, K., Ma, B., Zhang, W., Huang, R., 2015. A spatio-temporal appearance representation for video-based pedestrian re-identification, in: The IEEE International Conference on Computer Vision (ICCV).
- Liu, X., Liu, W., Ma, H., Fu, H., 2016b. Large-scale vehicle re-identification in urban surveillance videos, in: IEEE International Conference on Multimedia and Expo (ICME), IEEE, pp. 1–6.
- Liu, X., Liu, W., Mei, T., Ma, H., 2016c. A deep learning-based approach to progressive vehicle re-identification for urban surveillance, in: ECCV, Springer, pp. 869–884.
- Liu, Y., Yan, J., Ouyang, W., 2017. Quality aware network for set to set recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5790–5799.
- Matsukawa, T., Okabe, T., Suzuki, E., Sato, Y., 2016. Hierarchical gaussian descriptor for person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- McLaughlin, N., Martinez del Rincon, J., Miller, P., 2016. Recurrent convolutional network for video-based person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Shen, Y., Xiao, T., Li, H., Yi, S., Wang, X., 2017. Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals, in: The IEEE International Conference on Computer Vision (ICCV).
- Shi, H., Yang, Y., Zhu, X., Liao, S., Lei, Z., Zheng, W., Li, S.Z., 2016. Embedding Deep Metric for Person Re-identification: A Study Against Large Variations. Springer International Publishing, Cham, pp. 732–748. URL: [https://doi.org/10.1007/978-3-319-46448-0\\_44](https://doi.org/10.1007/978-3-319-46448-0_44), doi:[10.1007/978-3-319-46448-0\\_44](https://doi.org/10.1007/978-3-319-46448-0_44).
- Sochor, J., Herout, A., Havel, J., 2016. BoxCars: 3D boxes as CNN input for improved fine-grained vehicle recognition, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Sochor, J., Špaříhel, J., Herout, A., 2017. BoxCars: Improving fine-grained recognition of vehicles using 3D bounding boxes in traffic surveillance, arXiv:1703.00686. [arXiv:arXiv:1703.00686](https://arxiv.org/abs/1703.00686).
- Špaříhel, J., Sochor, J., Juránek, R., Herout, A., Maršík, L., Zemčík, P., 2017. Holistic recognition of low quality license plates by CNN using track annotated data, in: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE, pp. 1–6. doi:[10.1109/AVSS.2017.8078501](https://doi.org/10.1109/AVSS.2017.8078501).
- Su, C., Li, J., Zhang, S., Xing, J., Gao, W., Tian, Q., 2017. Pose-driven deep convolutional model for person re-identification, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 3960–3969.
- Su, C., Zhang, S., Xing, J., Gao, W., Tian, Q., 2016. Deep Attributes Driven Multi-camera Person Re-identification. Springer International Publishing, Cham, pp. 475–491. URL: [https://doi.org/10.1007/978-3-319-46475-6\\_30](https://doi.org/10.1007/978-3-319-46475-6_30), doi:[10.1007/978-3-319-46475-6\\_30](https://doi.org/10.1007/978-3-319-46475-6_30).
- Sun, Y., Zheng, L., Deng, W., Wang, S., 2017. Svdnet for pedestrian retrieval. The IEEE International Conference on Computer Vision (ICCV).
- Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A., 2017. Inception-v4, inception-resnet and the impact of residual connections on learning., in: AAAI, p. 12.
- Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., Wang, X., Tang, X., 2017a. Residual attention network for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164.
- Wang, F., Zuo, W., Lin, L., Zhang, D., Zhang, L., 2016. Joint learning of single-image and cross-image representations for person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Wang, T., Gong, S., Zhu, X., Wang, S., 2014. Person re-identification by video ranking, in: European Conference on Computer Vision, Springer International Publishing, pp. 688–703. URL: [https://doi.org/10.1007/978-3-319-10593-2\\_45](https://doi.org/10.1007/978-3-319-10593-2_45), doi:[10.1007/978-3-319-10593-2\\_45](https://doi.org/10.1007/978-3-319-10593-2_45).
- Wang, Z., Tang, L., Liu, X., Yao, Z., Yi, S., Shao, J., Yan, J., Wang, S., Li, H., Wang, X., 2017b. Orientation invariant feature embedding and spatial temporal regularization for vehicle re-identification, in: The IEEE International Conference on Computer Vision (ICCV).
- Wen, Y., Lu, Y., Yan, J., Zhou, Z., von Deneen, K.M., Shi, P., 2011. An algorithm for license plate recognition applied to intelligent transportation system. IEEE Trans. on Intelligent Transportation Systems 12, 830–845.
- Wu, S., Chen, Y.C., Li, X., Wu, A.C., You, J.J., Zheng, W.S., 2016. An enhanced deep feature representation for person re-identification, in: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1–8. doi:[10.1109/WACV.2016.7477681](https://doi.org/10.1109/WACV.2016.7477681).
- Xiao, T., Li, H., Ouyang, W., Wang, X., 2016. Learning deep feature representations with domain guided dropout for person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Xu, S., Cheng, Y., Gu, K., Yang, Y., Chang, S., Zhou, P., 2017. Jointly attentive spatial-temporal pooling networks for video-based person re-identification, in: International Conference on Computer Vision (ICCV).
- Yan, K., Tian, Y., Wang, Y., Zeng, W., Huang, T., 2017. Exploiting multi-grain ranking constraints for precisely searching visually-similar vehicles, in: The IEEE International Conference on Computer Vision (ICCV).
- Yan, Y., Ni, B., Song, Z., Ma, C., Yan, Y., Yang, X., 2016. Person Re-identification via Recurrent Feature Aggregation. Springer International Publishing, Cham, pp. 701–716. URL: [https://doi.org/10.1007/978-3-319-46466-4\\_42](https://doi.org/10.1007/978-3-319-46466-4_42), doi:[10.1007/978-3-319-46466-4\\_42](https://doi.org/10.1007/978-3-319-46466-4_42).
- Yang, J., Ren, P., Zhang, D., Chen, D., Wen, F., Li, H., Hua, G., 2017. Neural aggregation network for video face recognition, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Yang, L., Luo, P., Change Loy, C., Tang, X., 2015. A large-scale car dataset for fine-grained categorization and verification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- You, J., Wu, A., Li, X., Zheng, W.S., 2016. Top-push video-based person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Zapletal, D., Herout, A., 2016. Vehicle re-identification for automatic video traffic surveillance, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 25–31.
- Zhang, W., Hu, S., Liu, K., 2017a. Learning compact appearance representation for video-based person re-identification. arXiv preprint arXiv:1702.06294.
- Zhang, W., Yu, X., He, X., 2017b. Learning bidirectional temporal cues for video-based person re-identification. IEEE Transactions on Circuits and Systems for Video Technology PP, 1–1. doi:[10.1109/TCSVT.2017.2718188](https://doi.org/10.1109/TCSVT.2017.2718188).
- Zhang, Y., Liu, D., Zha, Z.J., 2017c. Improving triplet-wise training of convolutional neural network for vehicle re-identification, in: Multimedia and Expo (ICME), 2017 IEEE International Conference on, IEEE, pp. 1386–1391.
- Zhao, H., Tian, M., Sun, S., Shao, J., Yan, J., Yi, S., Wang, X., Tang, X., 2017. Spindle Net: Person re-identification with human body region guided

- feature decomposition and fusion, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., Tian, Q., 2016. Mars: A video benchmark for large-scale person re-identification, in: The IEEE European Conference on Computer Vision (ECCV), Springer. pp. 868–884.
- Zhong, Z., Zheng, L., Cao, D., Li, S., 2017. Re-ranking person re-identification with k-reciprocal encoding, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Zhou, S., Wang, J., Wang, J., Gong, Y., Zheng, N., 2017a. Point to set similarity based deep feature learning for person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Zhou, Z., Huang, Y., Wang, W., Wang, L., Tan, T., 2017b. See the forest for the trees: Joint spatial and temporal recurrent neural networks for video-based person re-identification, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).