Image and Vision Computing 31 (2013) 673–685

Contents lists available at SciVerse ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Vehicle matching in smart camera networks using image projection profiles at multiple instances[☆]

Vedran Jelača [*], Aleksandra Pižurica, Jorge Oswaldo Niño-Castañeda, Andrés Frías-Velázquez, Wilfried Philips

Ghent University, Department of Telecommunications and Information Processing, TELIN-IPI-IBBT, Sint-Pietersnieuwstraat 41, Ghent, Belgium

ARTICLE INFO

Article history:
Received 31 August 2012
Received in revised form 8 April 2013
Accepted 19 June 2013

Keywords:
Multi-camera tracking
Feature extraction
Appearance modeling
Object recognition
Traffic surveillance

ABSTRACT

Tracking vehicles using a network of cameras with non-overlapping views is a challenging problem of great importance in traffic surveillance. One of the main challenges is accurate vehicle matching across the cameras. Even if the cameras have similar views on vehicles, vehicle matching remains a difficult task due to changes of their appearance between observations, and inaccurate detections and occlusions, which often occur in real scenarios. To be executed on smart cameras the matching has also to be efficient in terms of needed data and computations. To address these challenges we present a low complexity method for vehicle matching robust against appearance changes and inaccuracies in vehicle detection. We efficiently represent vehicle appearances using signature vectors composed of Radon transform like projections of the vehicle images and compare them in a coarse-to-fine fashion using a simple combination of 1-D correlations. To deal with appearance changes we include multiple observations in each vehicle appearance model. These observations are automatically collected along the vehicle trajectory. The proposed signature vectors can be calculated in low-complexity smart cameras, by a simple scan-line algorithm of the camera software itself, and transmitted to the other smart cameras or to the central server. Extensive experiments based on real traffic surveillance videos recorded in a tunnel validate our approach.

1. Introduction

For the purpose of traffic management and fast reaction in cases of traffic accidents it is important to timely detect potential incidents or disturbances in the traffic flow. Therefore, surveillance cameras are typically mounted along roads, for commercial reasons often with non-overlapping fields of view. As an aid to human operators, computer vision algorithms can then be used for automatic detection and tracking of vehicles in the acquired videos. Such algorithms consist of three parts: vehicle detection, tracking of vehicles in a field of view of one camera (single-camera tracking) and vehicle matching, which is used for a “handover” of vehicles between cameras, i.e., for multi-camera tracking. Typically, results of vehicle detections and single-camera tracking are bounding boxes with vehicle images being regions of interest inside the bounding boxes. Such vehicle images are referred to as vehicle detections. Vehicle detections are input to the vehicle matching.

In traditional camera networks the cameras send all acquired data to the central server that performs video analysis. However, networks of smart cameras open a possibility to process the acquired video data by the cameras themselves and transfer only the obtained metadata to the other cameras and to the central server. In this context, this paper addresses the problem of matching vehicles as they are imaged by a network of stationary smart cameras with non-overlapping views. We focus on the problem of finding a computationally and data efficient, but still discriminative and robust representation of vehicle appearances that can be computed by cameras themselves and sent between cameras without sending the whole images. We also focus on finding a computationally efficient algorithm for matching such vehicle representations, suitable for execution on cameras themselves. Although the framework proposed in this paper is developed in the context of a vehicle tracking application in tunnels, where the cameras are placed to view the vehicles from relatively similar viewpoints, the basic idea and the associated techniques can be applied to vehicle tracking and matching in general, as well as to matching of other types of rigid objects.

In traffic surveillance such placement of cameras is achieved in many cases: in tunnels, along roads, or even at intersections of roads by placing more cameras at the crossroads. Still, vehicle matching remains challenging due to significant appearance changes in between cameras, observation differences and inaccurate and false vehicle detections, which are all common in real-world applications. The vehicle *appearance changes* are due to various reasons: illumination changes in the environment (e.g., a different lighting in different areas of the environment, shadows, light reflections), changes of the vehicle pose as it moves through the multi-camera environment and turning vehicle

[☆] This paper has been recommended for acceptance by Qiang Ji.
[*] Corresponding author. Tel.: +32 9 264 3410; fax: +32 9 264 4295.
E-mail address: vedran.jelaca@telin.ugent.be (V. Jelača).

0262-8856/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.imavis.2013.06.007

lights on or off. The *observation changes* result from differences in camera settings (e.g., a scale difference due to different zoom settings). *Inaccurate vehicle detections* are detections of vehicles or their parts together with a part of the background. They cause misalignment of vehicle images. *False detections*, i.e., detections of the background as vehicle, detections of multiple vehicles as one and multiple detections of one vehicle can cause significant problems to matching algorithms, if not discarded.

In our test application there are also some challenges due to a tunnel environment. Firstly, tunnels are often partly dark, artificially illuminated, which makes color unreliable information (the same holds for outdoor environments in general in cases of poor lighting conditions). Secondly, tunnels are tubular environments, so strong light reflections from the walls and ceiling can disturb cameras and "pollute" the images. Fig. 1 shows images of six vehicles, acquired in a tunnel by three cameras and automatically detected using the detector proposed by Rios Cabrera et al. [1]. We used this detector in our work because it is a state-of-the-art vehicle detector suitable for tunnels, as demonstrated in [1]. It uses rectangular Haar features (similar to those proposed by Viola and Jones for face detection [2]) and a cascade of strong classifiers, which are combinations of weak classifiers selected using the Ada-Boost algorithm. The detection accuracy is increased by introducing a confidence score accumulated and normalized over all cascade stages. However, even with these improved detections there is still a significant variety of vehicle appearance and observation changes across cameras, illustrated in Fig. 1. Moreover, if vehicle images are of low to medium resolution, which is common in video surveillance, the motion blur and noise in the images are also significant. This imposes an additional challenge for extraction of robust features from vehicle images.

Previous work on object appearance matching has mainly focused on extracting robust features from acquired images, so that those features remain invariant to appearance changes [3–13]. Many different features have been proposed, based on color, local features, edges, image eigenvectors or entropy, all with limited success in achieving the goal of invariance. Calculation and matching of such features is also often computationally demanding, so object comparison in real-time is typically done using only one image per camera for each object. Therefore, the accuracy of these approaches strongly depends on the quality of observations and the matching is much more challenging if observations contain disturbances like light reflections, strong shadows or occlusions.

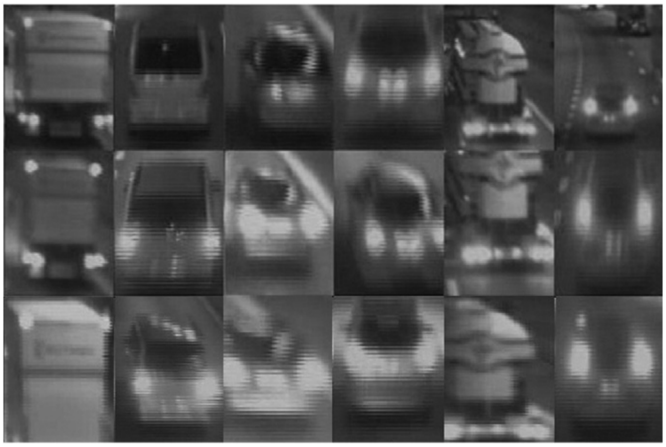

Fig. 1. Each column contains vehicle images (detections) of the same vehicle observed by three cameras along a tunnel pipe. The cameras are mounted roughly in the middle of the tunnel pipe ceiling and oriented in the direction of the traffic flow. From left to right the images illustrate a vehicle appearance change due to different levels of visible details when the detections are taken at different distances from the camera, turning on/off the rear lights, change of the scene illumination, change of pose as vehicle moves away from the camera or changes lane, and inaccurate detections.

In our work we try to overcome these problems by a conceptually different approach, based on two novelties in vehicle matching. Firstly, we use simple descriptors of vehicle appearances that are easy to compute and compare, yet highly informative in low resolution images. For this purpose we model vehicle appearances using signatures that are Radon transform like projection profiles of the acquired vehicle images. Matching of the appearance models is then obtained by a simple combination of 1-D correlations in a coarse-to-fine procedure. The signatures are also used to learn scale differences between the observations from different cameras, which is important for their alignment. The second novelty is to use signatures from multiple images for creating a multiple observation appearance model and for automatic selection of good observations for matching (i.e., informative observations with few disturbances), as shown in Fig. 2. Such an appearance model enables representation of vehicles from multiple views, collected online as they move through the multi-camera environment. This is especially beneficial when vehicles change pose (e.g., by changing lane or moving away from the camera). Finally, since each vehicle has one and only one corresponding vehicle in other cameras, we employ the Hungarian algorithm to resolve ambiguities and to optimize the matching.

The remainder of the paper is organized as follows. Section 2 gives related work on multi-camera tracking, object representation and matching. Section 3 briefly formulates the problem of vehicle matching. In Section 4 we propose a novel appearance model based on the vehicle signatures, together with the procedure for collection of good observations along the vehicle trajectory. Matching of vehicle appearances using the proposed appearance model is explained in Section 5. The complete matching algorithm that optimizes the association of vehicle correspondences is given in Section 6. In Section 7 we present and discuss the experimental results and finally, we conclude the paper in Section 8.

2. Related work

Most of the work on multi-camera tracking by cameras with non-overlapping views, e.g., [10–12,14–18], uses object appearance representations based on color information (e.g., mean, histogram or correlogram of the color). Color alone is, however, not reliable as a feature in many traffic surveillance applications, especially in tunnels. To address such a problem [11,12,19] present a method for matching object appearances by calculation of a brightness transfer function for every pair of non-overlapping cameras. They map an observed color value in one camera to the corresponding observation in the other camera. Once such mapping is known, the correspondence problem is reduced to matching of the transformed appearance models. However, real illumination often varies between frames and scenes depending on a large number of parameters, which is very difficult to model. Moreover, colors of artificial lights in tunnels or in artificially illuminated environments in general can supersede vehicle colors, which make the mapping of vehicle colors even more challenging (especially in the presence of variable road signs, rotating and emergency lights, etc.).

Appearance representations that do not need color information are often based on eigenimages (often used for face recognition) [3,4], local invariant features (e.g., SIFT [6], SURF [7] or ASIFT [8]) or edge maps [9,10,20].

Methods based on *eigenimages* require offline training and their accuracy highly depends on variations of objects and their appearances present in the training set. Therefore, adaptation of these methods to appearance changes is limited. These methods also require alignment of objects before matching, which is an additional challenge in real world scenarios.

The accuracy of methods based on *local features* depends on the number of corresponding key points found in images and on the dimension of the local descriptors calculated for each key point. In our experiments with vehicle images acquired by surveillance cameras, too few reliable and unique features were found and thus many features were

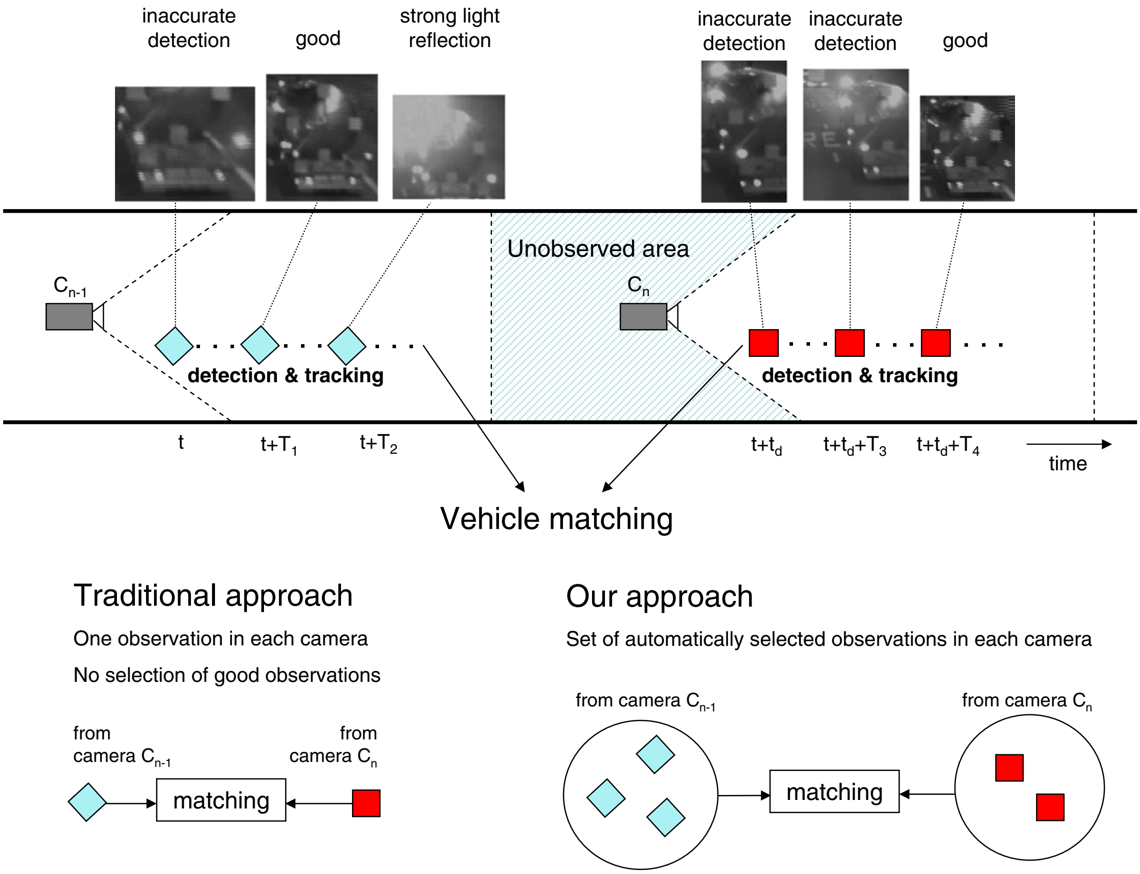

Fig. 2. Observations of a vehicle along the trajectory viewed by two consecutive cameras. The images can be very different due to inaccurate detections and significant appearance changes. The key idea of our approach is to reduce this problem by preselecting good observations before performing the inter-camera matching itself. We use multiple good observations from each camera to allow for difference in appearance.

wrongly matched. Similar findings have been reported in previous works [10,9]. Also, calculation of high dimensional local descriptors is computationally demanding, which creates additional difficulties to use local features for real-time vehicle matching.

In the context of *edge* based methods [9] has proposed a measurement vector and an unsupervised approach to learn edge measures for matching vehicle edge maps. The edge maps are compared after spatial alignment. A solution for the alignment has been proposed in [10]. The reported results show that the learned edge matching measures can be relatively invariant to changes in illumination and vehicle pose. This invariance is further increased [20], by matching embedded vehicle descriptors instead of direct vehicle matching between different camera views. The embedded descriptors were obtained by matching vehicles with exemplar vehicles from the same camera view. However, automatic selection of good exemplars has remained a problem. Also, the learned edge measure weights indicate the illumination and aspect differences between two scenes, but not the quality of compared observations themselves (some observations can be influenced more than others by certain changes, especially if changes are temporary and only in some parts of a scene).

Beside the mentioned features and approaches, recently there are proposals to use *Haar-like* features for vehicle matching [1]. These features are often successfully used for object detection [1,2], so reusing the same features for matching reduces the computational cost of the matching itself. Rios Cabrera et al. [1] recently introduced a whole framework for vehicle detection, tracking and matching, in which all these three blocks use rectangular Haar features. Integral images speed up the calculation of Haar features so it is possible to reach a near real-time performance throughout the framework. The most informative Haar features are selected by a supervised Ada-Boost training in several cascades. Binary vehicle fingerprints embedded from those same Haar features are used to match vehicles, as well as to track vehicles in a tracking-by-identification fashion. The matching in a real multi-camera setup is further optimized using the Hungarian algorithm and vehicle kinematics.

The work of [1] shows that reusing Haar-features for vehicle matching is possible and can be highly accurate if the training set of vehicle images is acquired under similar environmental conditions as the testing vehicle images. This condition is, however, difficult to meet in real world applications, which is a limitation of this approach, demonstrated also in the experiments of [1]. The training set needs to be large enough to include vehicle images from various environments (e.g., different tunnels, different cameras) and various environmental conditions (e.g., different lighting, wet/dry road, different aspect of vehicles, etc.). This further increases complexity of the training process. Therefore, in our work we are focused on finding a vehicle matching approach that does not require supervised training and has a built-in procedure for collecting various vehicle appearances to create more informative appearance models.

Our method for vehicle appearance matching is inspired by the work of [21], which used vertical and horizontal projections of a vehicle edge map for accurate positioning of the bounding box in tracking. Similar projections have also been used for human gait recognition [22]. Compared to these works we go a step further, showing that it is possible to use projections for vehicle appearance representation and matching. Instead of using projections of the vehicle edge map, we represent vehicle appearances by projections of the vehicle images themselves. Such projections we call vehicle signatures. Using the signatures for automatic selection of good observations for matching is another step forward of our method compared to previous works. Such an automatic selection could also be useful for selection of vehicle exemplars in [20,1].

3. Problem formulation

We define the vehicle matching problem as the problem of classifying pairs of vehicles observed by cameras with non-overlapping views in the categories “the same vehicle” or “different vehicle”. Without losing generality we can assume that these cameras are consecutive in a predefined sequence of cameras, so we denote two of them as C_n and C_{n-1} (camera C_{n-1} being the one that vehicles pass before reaching camera C_n). The match score between vehicle appearances, μ, is defined as

$$\mu = f\left(A_i^{n-1}, A_j^n\right), \tag{1}$$

where f is a similarity measure between two appearance models A_i^{n-1} and A_j^n corresponding to the i-th and j-th observations O_i^{n-1} and O_j^n in cameras C_{n-1} and C_n, respectively. We call the model A_j^n the template and the model A_i^{n-1} the candidate. In the context of online multi-camera tracking, vehicle observations are responses of vehicle detection and single-camera tracking. For each template a set of possible candidates (a temporal matching window) is defined according to road constrains, inter-camera distances and vehicle kinematics, see Fig. 3. A template–candidate association is then obtained according to the matching score μ, assuming that each template inside its matching window has one and only one corresponding candidate.

4. Signature based appearance model

4.1. Signatures and signature vectors

Let I be the vehicle image of size $M \times N$. We define the signatures of the image I as Radon transform like projection profiles along a certain direction. The vertical signature v_I consists of the arithmetic means of the pixel intensities in each image row,

$$v_I(m) = \frac{1}{N}\sum_{n=1}^{N} I(m,n), \quad m = 1, \ldots, M, \tag{2}$$

where $I(m,n)$ is an intensity value of the image pixel at the position (m,n). Analogously, the components of the horizontal signature $\mathbf{h}_I$ are the arithmetic means of the pixel intensities in each image column,

$$h_I(n) = \frac{1}{M}\sum_{m=1}^{M} I(m,n), \quad n = 1, \ldots, N. \tag{3}$$

In Fig. 4 we see three images of the same vehicle observed by two gray scale cameras. The images are represented by horizontal and vertical signatures. The vehicle's bright areas, captured by the signatures, are marked with arrows. The bright areas correspond to the local maxima in the signatures. Analogously, the dark patterns are represented by the local minima. If two horizontal signatures are plotted one over the other (the signatures at the bottom left of Fig. 4), we see that they have similar behavior (shape). The background, a road, has almost uniform brightness so it does not change significantly the behavior of the signatures. Also in the vertical direction, the signature parts that correspond to the vehicle are similar, both when the detection includes the background or only a part of the vehicle. An advantage of the proposed signature based representation compared to an edge based representation, which also has responses in the areas of brightness changes, is in the fact that there is no thresholding in the calculation of the signatures. The signatures are independent of the intensity gradient values within the image and thus are more robust against lighting changes.

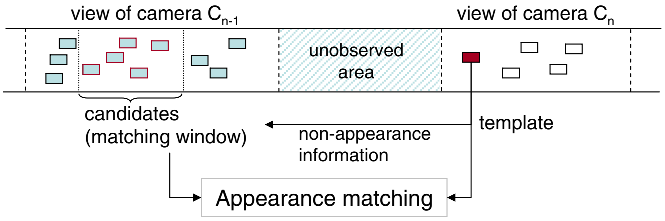

Fig. 3. The problem illustration. For each vehicle (a template) from camera C_n we find a set of possible matching candidates from camera C_{n-1} using non-appearance information (road constraints, inter-camera distances and vehicle kinematics). A template-candidate matching is then obtained based on similarity of vehicle appearances.

Since vehicles are rigid objects, the signatures are similar in different observations, except for translation and scale. If the vehicle appears rotated between observations, parts of the corresponding signatures shrink or stretch, but overall the shape of the signatures remains similar. Having noticed this, we propose using the signatures for the appearance modeling. Note that by using more precise vehicle masks instead of bounding boxes, as proposed in [10] for example, it is possible to reduce the influence of the background on vehicle signatures. However, since obtaining such masks in harsh tunnel conditions is a challenging problem on its own, in this paper we do not assume that such vehicle masks are obtained. This also increases generality of our approach.

Next to the vertical and horizontal signatures, defined by Eqs. (2) and (3), it is possible to use additional projections. We define the n-dimensional signature vector $\mathbf{s}_I$ calculated from the vehicle image I as an n-tuple of n projections (signatures) on different lines. In this paper we explicitly treat 2-dimensional (2-D) and 4-dimensional (4-D) signature vectors defined as the following. The 2-D signature vector is a pair of the vertical and horizontal signature,

$$\mathbf{s}_I = (\mathbf{v}_I, \mathbf{h}_I), \tag{4}$$

while the 4-D signature vector contains also two diagonal signatures (see Fig. 4, bottom right),

$$\mathbf{s}_I = (\mathbf{v}_I, \mathbf{h}_I, \mathbf{d}_I, \mathbf{a}_I), \tag{5}$$

where $\mathbf{d}_I$ and $\mathbf{a}_I$ are signatures on the main-diagonal and anti-diagonal, respectively. The signature vectors represent an image as multiple 1-D vectors, which significantly reduce the amount of the vehicle appearance representation data. In the experiments we measured the benefit of adding two diagonal projections for the appearance matching (see Section 7.5).

4.2. A robust multi-observation appearance model

During the movement through the multi-camera environment the appearance of vehicles between observations (detections) can change due to many reasons. For robust appearance matching it is essential to create an appearance model of each vehicle using a diversity of good observations. In this Section we analyze observations in tunnels and explain how signatures can be used for automatic selection of good observations.

The images in Fig. 5 are typical examples of vehicle observations in tunnels (the images are rescaled to the same size for easier visual comparison). Their signatures are also presented. Fig. 5a shows observations of the same vehicle, acquired by one camera. They illustrate the appearance change caused by a different camera viewing angle when the vehicle moves away from the camera. As the vehicle moves away from the camera it appears smaller and its lights and license plate appear bigger due to the light dissipation effect. Some parts of the vehicle are even not visible any more (e.g., the roof) and some other parts become more visible (e.g., the back window). As certain vehicle parts appear bigger or smaller, the corresponding signature

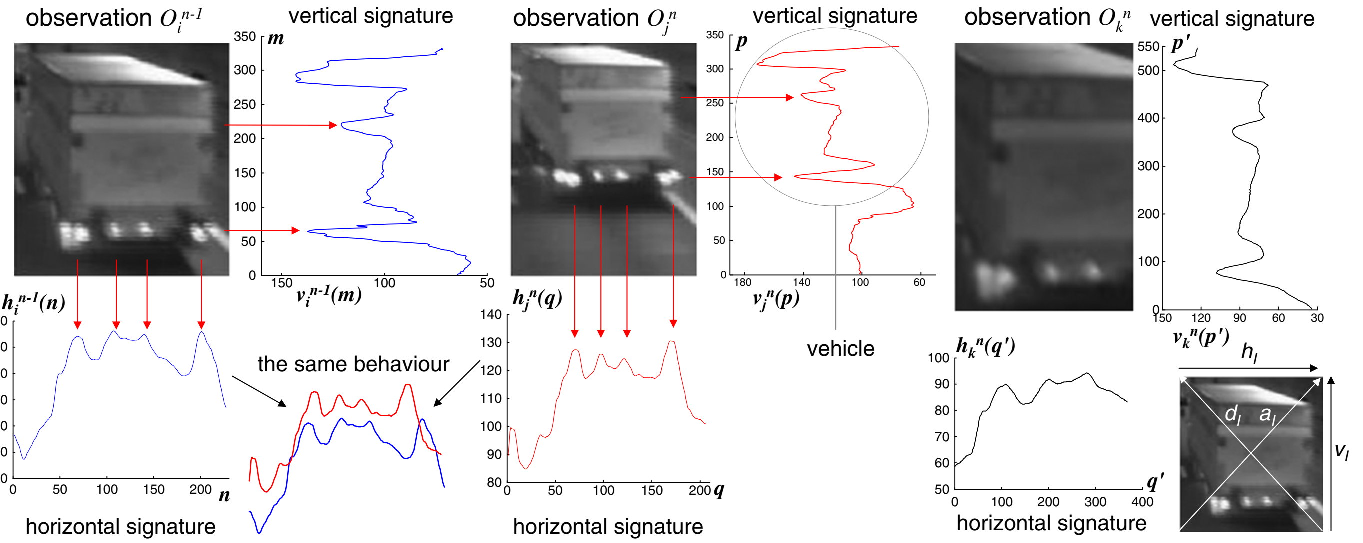

Fig. 4. Vehicle images from two different cameras along a tunnel pipe with the horizontal (below the vehicle image) and vertical signatures (on the right side of the vehicle image). Signature peaks correspond to the brightness changes of vehicle parts and patterns. There is a clear similarity in behavior of the signature parts that represent the vehicle. The horizontal, vertical, and diagonal signatures are defined as shown in the image in the bottom right corner.

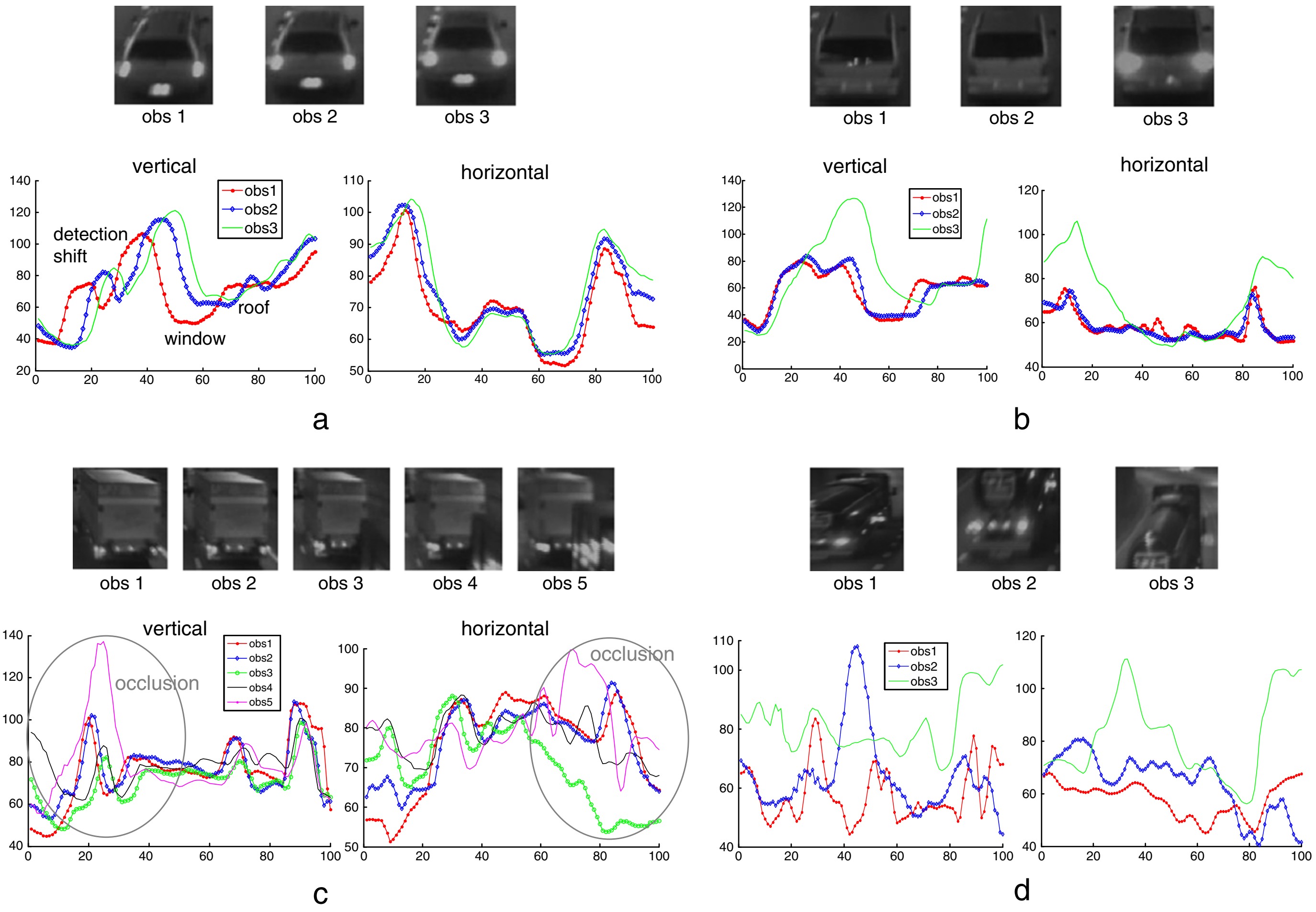

Fig. 5. Images of vehicles observed by one camera and their corresponding vertical and horizontal signatures: a) a typical case of the vehicle appearance change when vehicles are observed by a camera mounted on a tunnel ceiling: first, the vehicle is observed from above and then, as it moves away from the camera, it is viewed from behind, so some of its parts become more and some less visible; b) a vehicle appearance change when its lights turn on: signature parts that correspond to the lights gain higher value and the number of implied pixels when the lights are on; c) an occlusion from another object: there is a significant gradual change of the vehicle signatures in the parts affected by occlusion; d) inaccurate detections: the signatures change quickly, not gradually between consecutive frames.

parts stretch or shrink. Hence, these observations contain different, but complementary information and representing possible variations in appearance increases informativeness of the appearance model. This is especially important because vehicles are observed at various distances and from various angles.

A second example, in Fig. 5b, shows vehicle appearance change due to the actions taken by the driver, in this case turning on the rear lights. If we compare the signatures that represent the cases when the lights are off and on, we see that there is a corresponding change in values and behavior of the signatures. As in the first example, having the observations with lights off and on in the appearance model increases its robustness, because the vehicle can be captured in both states in other cameras.

Conversely, some observations should not be included in the appearance model, particularly false and inaccurate detections, clutters and occlusions. For detecting them we exploit the fact that the signatures in such observations change differently than in the two aforementioned situations. When occlusions occur, a new object in the image causes a significant change of the observations (see Fig. 5c). This change is gradual as the vehicle gets more or less occluded, until it reaches the unoccluded state again or the state in which the occlusion is constant. This behavior is present in the signatures as well. Also, strong illumination sources can blind the cameras or significantly disturb observations, e.g., when vehicles with rotating lights enter the scene those lights are periodically disturbing the camera and “polluting” the observations. In consecutive frames this effect is again visible as gradual change in vehicle images. Analogous phenomena outside of tunnels can be observed due to reflection of sunlight from vehicles or cast shadows from objects alongside the road.

False and inaccurate detections can also be detected by analyzing signatures. False and inaccurate detections occur in real scenarios regardless of a vehicle detector that is used, mostly due to lack of visible features, intensive illumination changes in some parts of the scene or light reflections on the road. Our experiments showed that such detections are typically unstable, i.e., they change quickly, capturing different vehicle parts in consecutive frames (see Fig. 5d). As a consequence, the signatures of false and inaccurate detections also change quickly and not gradually.

The stated signature characteristics allow us to use them for selection and representation of good appearance states that should be included in the appearance model. The selection procedure is presented in Fig. 6. Let A be the appearance model and $\mathbf{s}_t$ the signature vector of the appearance state observed at the time instance t. We consider that the appearance state is stable if the appearance remains similar enough in a predefined number of successive frames T. In terms of signatures this condition is satisfied if a similarity measure μ_s between the signature vectors of these T successive observations remains above some predefined similarity value M_{st},

$$\mu_s\left(\mathbf{s}_t, \mathbf{s}_{t+\tau}\right) > M_{st}, \forall \tau \in [1, T]. \tag{6}$$

We call this condition the stability criterion, with M_{st} and T being the stability parameters. A method for calculating the similarity μ_s between the signature vectors is given in detail in Section 5. If the appearance state at the time instant t is stable, its signature vector should be included in the appearance model $A \equiv \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$ only if it brings additional information to the model, i.e., if it is different enough from other, previously observed states already included in the model,

$$\mu_s\left(\mathbf{s}_t, \mathbf{s}_n\right) < M_{\text{var}}, \forall n \in [1, N]. \tag{7}$$

This condition represents variability criterion and M_{var} is the variability threshold.

By using the proposed appearance collection procedure, most occlusions, clutters and inaccurate and false detections can be excluded from the appearance model (due to the stability condition). However, the model can still include persistent inaccurate detections for which both a vehicle detector and a single-camera tracker constantly return a stable, positive response. Such persistent inaccurate detections typically occur when bigger vehicles (e.g., trucks or buses) are partly detected or when the detections of smaller vehicles (e.g., cars) contain parts of the background or other vehicles. We noticed also that for some vehicles all stable detections were inaccurate, so our intention in this work was to develop a signature matching procedure robust to such detection inaccuracies.

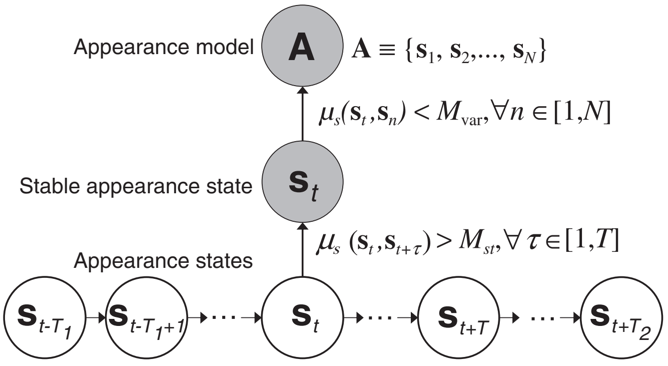
Appearance model
A
$\mathbf{A} \equiv \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$
$\mu_s(\mathbf{s}_t, \mathbf{s}_n) < M_{\text{var}}, \forall n \in [1,N]$
Stable appearance state
$\mathbf{s}_t$
$\mu_s\ (\mathbf{s}_t, \mathbf{s}_{t+\tau}) > M_{st}, \forall \tau \in [1,T]$
Appearance states
$\mathbf{s}_{t\text{-}T_1}$
$\mathbf{s}_{t\text{-}T_1+1}$
$\mathbf{s}_t$
$\mathbf{s}_{t+T}$
$\mathbf{s}_{t+T_2}$

Fig. 6. A procedure for collection of good observations for modeling the appearance of vehicles as they move in a multi-camera environment; the stability and variability conditions for including appearance states into an appearance model are given. All appearances are represented by signature vectors.

5. Vehicle appearance matching

5.1. Signature matching

The challenges for robust matching of the signatures come from scale, shift and rotation variations between the vehicle observations that are compared. *Scale differences* result from different camera zoom settings or different distances between the observed vehicles and the camera. *Shift* results from differences in the bounding box location with respect to vehicles. *Rotation* is caused by vehicle pose changes together with camera viewing angle changes. All these effects are present in the example of Fig. 7. Due to the scale difference the lengths of the corresponding parts of the signatures differ. A consequence of the bounding box shift is the signature shift along the shift direction while the vehicle rotation results in shrinking and stretching of the signature parts. Thus, we propose a coarse-to-fine signature matching procedure composed of four parts: signature rescaling, and global and local alignment, followed by calculation of the final similarity measure.

5.1.1. Learning of rescaling factors

To achieve the scale invariance necessary for signature matching, we rescale the signatures by estimated factors using cubic interpolation. In the following we present a method to estimate these rescaling factors. They depend on the camera zoom settings and the position of the vehicle in the scene (further from the camera the smaller the vehicle image, i.e., the shorter the vehicle signatures and vice versa). We represent the vehicle position by the y-coordinate of its bounding box bottom line (see Fig. 7a). Suppose we want to determine the rescaling factors between the vertical signatures of two vehicle images O_i^{n-1} and O_j^n, extracted at positions y_i^{n-1} and y_j^n. We define the rescaling factor for vertical signatures as the following:

$$r_v^{ji}\left(y_j^n, y_i^{n-1}\right) = \frac{l_{v_j}^n}{l_{v_i}^{n-1}}, \tag{8}$$

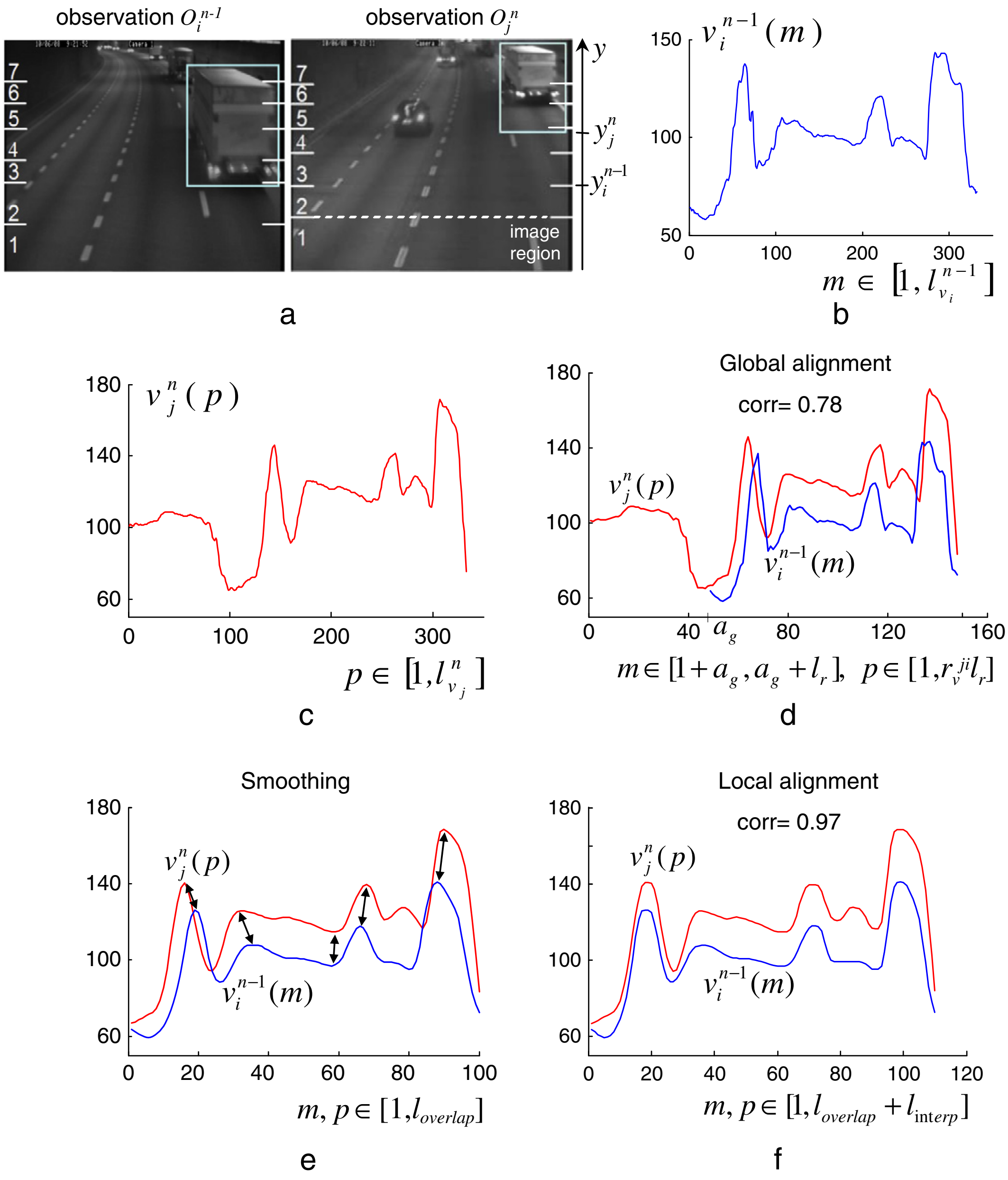

Fig. 7. The signature matching procedure: a) two observations of the same vehicle captured by two cameras (the same as in Fig. 4); y-coordinates of the bounding box bottom lines represent the positions of vehicles in the scenes; images are divided in regions (marked with numbers) so that vehicle scale difference within each region remains less than 15%; b) vertical signature of the observation O_i^{n-1} (good detection); c) vertical signature of the observation O_j^n (inaccurate detection); d) signatures are rescaled using the reference length $l_r = 100$ using the rescaling factor $r_v^{ji} = 1.48$, estimated between the positions y_j^n and y_i^{n-1} in images from cameras C_n and C_{n-1}, respectively; after rescaling, the global alignment is found in the position a_g; e–f) the local alignment: unstable extrema are removed by smoothing the signatures; the remaining local extrema (some of which are marked with arrows) are aligned by interpolating the signatures; after local alignment the final matching measure between the signatures is calculated.

where $l_{v_j}^n$ and $l_{v_i}^{n-1}$ are the lengths of the vertical signatures of images O_j^n and O_i^{n-1}, respectively. We rescale the signatures to the same reference length l_r and compare using 1-D correlation. If the obtained correlation coefficient is high enough, i.e., above a predefined threshold (0.9 in our experiments), the reference rescaling factors $r_v^j = l_{v_j}^n / l_r$ and $r_v^i = l_{v_i}^{n-1} / l_r$ properly estimate the scale difference between the vehicles at positions y_i^{n-1} and y_j^n. Then, the rescaling factor r_v^{ji} is calculated as $r_v^{ji} = r_v^j / r_v^i$ and we use it as an estimate of the scale difference between observations at positions y_i^{n-1} and y_j^n. If the obtained correlation coefficient is below the threshold, such a signature pair is considered unreliable, so it is not used for the rescaling factor learning.

In this way we estimate the rescaling factors for different pairs of positions in two fields of view. Taking into account that vehicles get detected at multiple positions along their trajectory in each camera view, the estimation of the rescaling factors can be done fairly quickly for many position pairs. If there is no rescaling factor for a certain position pair, we use the factor for the nearest position pair. If there are multiple factors for the same position pair, the arithmetic mean of the latest n is used ($n = 3$ in our experiments). This enables automatic adaptation to the change of the camera zoom parameters. The rescaling factors for horizontal and diagonal signatures are estimated analogously.

Note that by dividing the images in regions, see Fig. 7a, it is possible to group vehicle positions and to learn rescaling factors for pairs of image regions instead for position pairs. This enables faster learning, but the estimated rescaling factors are less precise. However, if the scale differences within the same regions are relatively small (in our experiments 15%), the global and local alignment can still be properly obtained.

5.1.2. Global alignment by correlation with shifting

Due to the possible signature shift, it is necessary to align the signatures before comparing them (see Figs. 4 and 7d). The alignment we perform is twofold. First, after rescaling the signatures by the estimated factor we align them globally. Then, a finer, local alignment is obtained. The global alignment is done by shifting one signature along the other one, finding the position with the highest correlation coefficient between the two signatures. Suppose $\mathbf{x}$ is the signature with M elements and $\mathbf{y}$ the signature with $N > M$ elements. The signature $\mathbf{x}$ is then shifted along $\mathbf{y}$ and the correlation coefficient ρ_s, obtained in each shift position $s \in [0, N-M]$ is defined as

$$\rho_s = \frac{\sum_{i=1}^{M} (x(i) - \overline{\mathbf{x}})(y(i+s) - \overline{\mathbf{y}}_s)}{\sqrt{\sum_{i=1}^{M} (x(i) - \overline{\mathbf{x}})^2 (y(i+s) - \overline{\mathbf{y}}_s)^2}}, \tag{9}$$

where $\mathbf{y}_s$ is the part of the signature $\mathbf{y}$, which in shift position s overlaps with the signature $\mathbf{x}$. The signatures are aligned in a position a_g, in which the correlation coefficient ρ_s has a maximal value ρ_g,

$$\rho_g = \max_s \rho_s. \tag{10}$$

This is a coarse, global matching measure of two signatures.

5.1.3. Local alignment and signature matching measure

Perspective changes of the vehicle observation, subtle appearance changes and imprecise rescaling cause shrinking and stretching of signature parts (see Figs. 4 and 7d). Hence, a local alignment of signatures is needed before calculating their correlation. For that purpose we propose a method similar to Iterative Closest Point (ICP) [23]. Our method aligns corresponding local extrema. Local extrema are robust features of the signatures, preserved even if vehicles change pose or if they are observed in different illumination conditions. This is because they correspond to different parts/patterns of the vehicle. As long as those parts/patterns remain visible in two observations, the local extrema remain present in the signatures (see Fig. 4). Therefore, we propose the following local alignment method.

Step 1 The signatures are iteratively smoothed until the same number of extrema is found in two consecutive iterations. Smoothing removes most of the extrema that originate from noise and camera interlacing. Fine appearance details can also be lost, but due to the low resolution of vehicle images they are mostly not present.

Step 2 The signatures are iteratively interpolated to align the closest local extrema of the same kind (maximum or minimum), see Fig. 7e–f. Suppose $\mathbf{x}$ and $\mathbf{y}$ are two signatures. For each local maximum $x(m)$ we find its closest maximum $y(n)$, i.e., the one for which the absolute difference in their position $|m - n|$ is minimal. In the same way the closest minimum is found for each local minimum of the signal $\mathbf{x}$.

Ambiguities occur if multiple extrema from the signature $\mathbf{y}$ have the same closest extrema in the signature $\mathbf{x}$. Therefore, the local alignment is performed in iterations. Firstly, the signatures are interpolated so all extrema with a unique correspondence are aligned. We used cubic interpolation for this purpose. After interpolation some of the ambiguities might be resolved. Then, the whole procedure of finding the closest extrema and aligning them repeats until all extrema with unique correspondence are aligned. Note that the requirement for extrema to be non-ambiguous before their alignment prevents local aligning of non-similar signatures.

Note also that other possible curve alignment approach is dynamic time warping (DTW), e.g., the method of [24]. DTW automatically handles both scale and translation effects globally and locally. It can be implemented in dynamic programming so it is also efficient. However, in real scenarios the signatures can be significantly misaligned so for a proper initialization of DTW (selection of the starting and ending point) a coarse global alignment is still an advantage. Moreover, the vehicle signatures taken at different lighting conditions can vary significantly in intensities and gradients, both globally and locally, which further can lead to inaccuracies of DTW if it aligns points with similar intensities or derivatives. DTW also does not intrinsically take into account whether the aligned extrema are non-ambiguous, as we do in our ICP-like approach. Therefore, we found that using the proposed iterative ICP-like approach is a better option in our matching method.

Finally, after the local alignment, 1-D correlation coefficient ρ_l between the signatures is calculated. We define the final matching measure between the signatures $\mathbf{x}$ and $\mathbf{y}$ as

$$\rho(\mathbf{x}, \mathbf{y}) = \begin{cases} \rho_l(\mathbf{x}, \mathbf{y}), & \rho_l(\mathbf{x}, \mathbf{y}) > 0 \\ 0 \quad , & \rho_l(\mathbf{x}, \mathbf{y}) \leq 0. \end{cases} \tag{11}$$

Negative values of the correlation coefficient ρ_l are set to zero in the signature similarity measure ρ. This is because vehicles are different both when significant parts of the signatures are mutually inverse as well as when the signatures do not correlate at all.

5.2. Matching of the appearance models

As explained in Section 4, the appearance of each vehicle is modeled by multiple appearance states of which each is represented by a signature vector. In this work we used signature vectors that consist of two and four signatures. Therefore, if A is the vehicle appearance model, it is a set of N signature vectors, $A \equiv \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\}$, of which each is represented by one vertical and one horizontal signature, $s_n = (\mathbf{v}_n, \mathbf{h}_n)$ or with diagonal signatures added, $\mathbf{s}_n = (\mathbf{v}_n, \mathbf{h}_n, \mathbf{d}_n, \mathbf{a}_n)$, $n \in [1, N]$. The correlation between signatures is calculated as presented in Section 5.1. Comparison of the appearance states requires then combining the correlation coefficients between signatures into one matching measure between signature vectors.

Fig. 8 shows 2-D and 3-D scatter plots of the correlation coefficients ρ_l for the pairs of vertical, horizontal and main diagonal signatures of the 300 vehicles in our database (each compared with 21 candidates). Red circles represent the correlation values between the signatures of the same vehicles (according to the manually annotated ground truth) and cyan crosses represent the values for different vehicles. As expected, the values for the same vehicles are clustered in the area with high correlation coefficients for the each signature pair, i.e., the area furthest from the zero correlation point (point (0,0) for 2-D plot or (0,0,0) for 3-D plot). Also, even if in some cases the correlation values of true matches are not in the cluster of red circles their distance from the zero correlation point is mostly still higher than the distance for false matches, as in the example at the bottom in Fig. 8. Therefore, we define a similarity measure μ_s between two signature vectors as the Euclidean norm of an n-D vector, where each dimension represents the similarity measure ρ between the signatures along the same projection direction, i.e.,

$$\mu_s(\mathbf{s}_n, \mathbf{s}_m) = \|(\rho(\mathbf{v}_n, \mathbf{v}_m), \rho(\mathbf{h}_n, \mathbf{h}_m))\| \tag{12}$$

for the appearance representation by 2-D signature vectors, or analogously the Euclidean norm of a 4-D vector when the appearance states are represented by 4-D signature vectors.

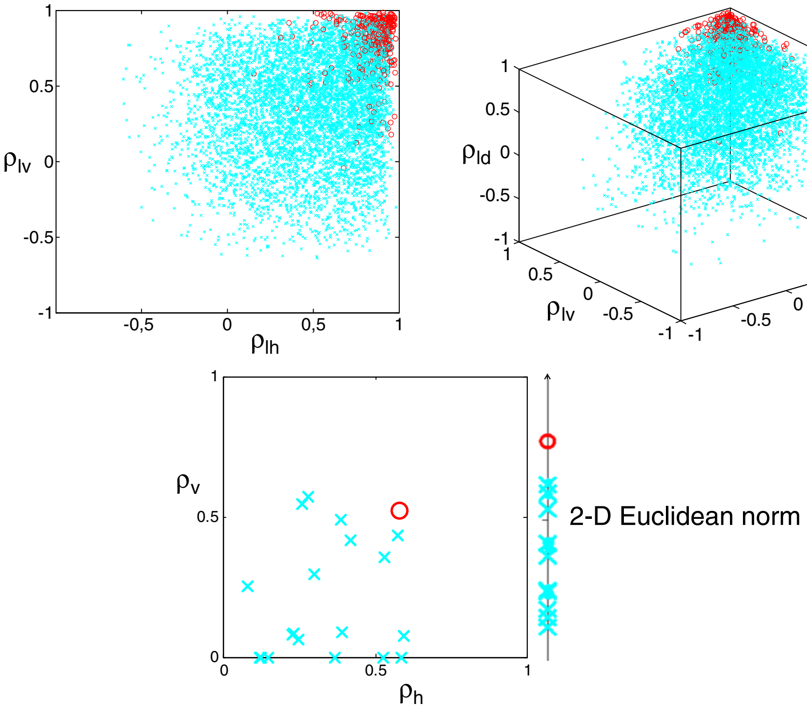

Fig. 8. Top: 2-D and 3-D scatter plots of the correlation coefficients ρ_l between the signatures of the vehicle images from our database; each vehicle from one camera (template) is compared with the same vehicle and 20 different vehicles viewed by the other camera; on x, y and z axes are correlation coefficients between pairs of horizontal, vertical and main diagonal signatures, ρ_{lh}, ρ_{lv} and ρ_{ld} respectively. Red circles represent values for the same vehicles while cyan crosses represent values for different vehicles. The correlation values for the same vehicles are clustered in the area with high values for each of the signature pairs. Bottom: example for one template and its candidates; 2-D scatter plot of the signature similarity measures ρ. Even when the similarity measures ρ_h and ρ_v are relatively low for a true match, a true match is still further from the zero correlation point than false matches.

Finally, the matching measure μ between two appearance models $A_p \equiv \{\mathbf{s}_1^p, \mathbf{s}_2^p, \ldots, \mathbf{s}_M^p\}$ and $A_q \equiv \{\mathbf{s}_1^q, \mathbf{s}_2^q, \ldots, \mathbf{s}_N^q\}$ is the maximal similarity measure obtained when comparing all their states,

$$\mu\left(A_p, A_q\right) = \max_{m,n} \mu_s\left(\mathbf{s}_m^p, \mathbf{s}_n^q\right), \quad m \in [1, M], n \in [1, N]. \tag{13}$$

This means that the vehicle matching is done according to the most similar appearances in the vehicle appearance models.

6. Vehicle matching algorithm

Given the problem of matching vehicles observed by two cameras with non-overlapping views, C_n and C_{n-1}, formulated in Section 3, our matching algorithm consists of the following four steps.

Step 1 During the movement of the j-th vehicle through the field of view of camera C_n its appearance model (template T_j) is created using the procedure explained in Section 4.2. The vehicle observations are responses of vehicle detection and single-camera tracking.

Step 2 The matching window (the set of candidates) for the template T_j is determined. It is done according to the distance $D_{n,n-1}$ between the cameras C_n and C_{n-1}, taking into account minimal and maximal possible velocities of the template vehicle. Let $v_{\min}$ and $v_{\max}$ be the minimal and maximal allowable vehicle velocities (taking into account possible over and down speeding). Then, all vehicles that disappeared from the field of view of camera C_{n-1} between time instances $t - \frac{D_{n,n-1}}{v_{\min}}$ and $t - \frac{D_{n,n-1}}{v_{\max}}$ are considered as matching candidates for the template T_j, if being in the same or adjacent lane as the template (this is how we determined the matching window in our experiments). The matching window could also be determined more precisely, using the estimated velocity and the trajectories of the template vehicle as observed by camera C_{n-1}. The velocity can be estimated according to the lane marks on the road, from responses of the single-camera tracking. The distance between the lane marks is known (complies with the known standards) so the velocity can be estimated by measuring the time vehicles need to move between the lane marks, while the lane marks in the images could be automatically detected or marked manually.

Step 3 A template-candidate association is computed using the Hungarian algorithm with voting, as proposed in [1].

Step 4 After the template-candidate assignments, we update all candidate appearance models with new states. These are the appearance states that are collected in the field of view of camera C_n and are different enough from the states collected in previous cameras, $C_1, \ldots, C_{n-1}$. The appearance states are different enough if they fulfill the condition in Eq. (7). This updating procedure enables learning of vehicle appearances online, along the multi-camera track.

7. Experimental evaluation

We composed two databases of vehicle images from three security cameras with non-overlapping views, mounted roughly in the center of a tunnel pipe ceiling and oriented in the direction of the traffic flow. The databases contain 300 different vehicles, manually annotated for evaluation purposes. Each vehicle is represented by 20 images per camera, extracted from successive video frames along their tracks, starting from the frame in which the vehicles are observed completely. For the first database, denoted as DBM, vehicle images were manually extracted from the videos resulting in similar detections between the frames and the cameras. The second database, denoted as DBA, contains real (automatic) vehicle detections, which are less accurate, thus less stable along the single-camera tracks and different between the cameras. Figs. 1, 2 and 5 show some examples of vehicle images in DBA database. The automatic detections are obtained using the detector of Rios Cabrera et al. [1].

Five major results are presented in this section. Firstly, we give the results of our matching method for two camera pairs (C_2,C_1) and (C_3,C_2), with and without using multiple templates (observations) of each vehicle. We demonstrate that our proposed method yields a better matching accuracy than the reference matching techniques (2-D image correlation, SIFT, eigenimages, and Haar feature based matching). Secondly, we prove that our method performs well if vehicles are visually distinctive, i.e., if there is enough information in the images based on which they can be recognized. For that purpose we present separately the matching results for big vehicles (trucks, buses, etc.) and cars. Our third result shows that the signatures can be downsampled without losing the essential information, which significantly increases the computational efficiency of our method. The fourth result illustrates the gain from adding two diagonal signatures to the appearance representation based on only horizontal and vertical signatures. Finally, the fifth result demonstrates the performance of the whole matching algorithm in a tunnel application, including non-visual information derived from physical constraints of vehicle motion.

7.1. Results for different camera pairs

We have compared the matching score for two different camera pairs, (C_2,C_1) and (C_3,C_2) using our method with and without collection of multiple observations (templates) along the vehicle trajectories, see Fig. 9. The matching rate is significantly higher when multiple templates are used for matching. Note that having multiple templates also reduces the difference in performance between different environments (if a single template is used, the matching rate drops in a more challenging environment of the camera pair (C_3,C_2) while this is not the case when multiple templates are used).

7.2. Comparison with other methods

We have compared the matching score computed between the vehicles in DBM and DBA databases using our matching algorithm with four other appearance matching methods based on 2-D image correlation, SIFT [6], eigenimages [3] and Haar features [1]. In our method we used 2-D signature vectors, which contain horizontal and vertical signatures. 2-D image correlation was obtained using vehicle images normalized to the same size. In the SIFT-based method, vehicle matching was done using the kd-tree and nearest neighbor search between SIFT features found in vehicle images (as in [6]). For the eigenimage method the datasets were divided in two disjunct parts, the training and testing subsets (in both databases 100 images were taken for training and tests were then performed on the other 200 images). Finally, for the comparison with the matching method of Rios Cabrera et al. we refer to their results reported in [1] since those results were obtained using the images from the same tunnel recordings we used in this paper. To evaluate how discriminative the signature based appearance model is, we did multiple experiments with different numbers of candidates in the matching window. The Hungarian algorithm with voting, as proposed by Rios Cabrera et al. [1], was used to optimize the assignment in all methods.

The results are given in Fig. 10a–b, separately for DBM and DBA dataset and the matching window size in the range from 3 to 101 with the step of 2, taken to include the corresponding vehicle and 2 to 100 other vehicles. The graphs show percentages of correct matches obtained using different methods. We see that selecting good observations suitable for matching increases the matching accuracy, especially when vehicle detections are done automatically. In our experiments the stability and variability parameters for our method (defined in Section 4.2) have been set to values $M_{st} = 0.9$, $T = 4$ and $M_{\mathrm{var}} = 0.75$, selecting on average 1.7 good observations per single-camera vehicle track in DBM set and 2.4 in DBA set. These numbers mean that the majority of vehicles change in appearance along the track and that many disturbing observations in DBA set get disqualified. This preselection of observations good for matching is a key advantage comparing to the other methods, which create the resulting difference in the matching accuracy. The methods without this functionality fail when the input from the detector and/or tracker is not accurate enough or the observations used for matching contain some disturbances. The methods that require registration of images before matching, like 2-D correlation and eigenimage based methods, are especially sensitive to inaccurate detections, which explain the rapid drop of their performance on DBA images (see Fig. 10b). In this sense, the results of 2-D correlation and eigenimage based methods are given here also to illustrate the difference between detection accuracies in DBM and DBA sets.

7.3. Results for different vehicle categories

On inspection we found that many of the wrong matches could be attributed to a visual similarity of vehicles. This especially holds for

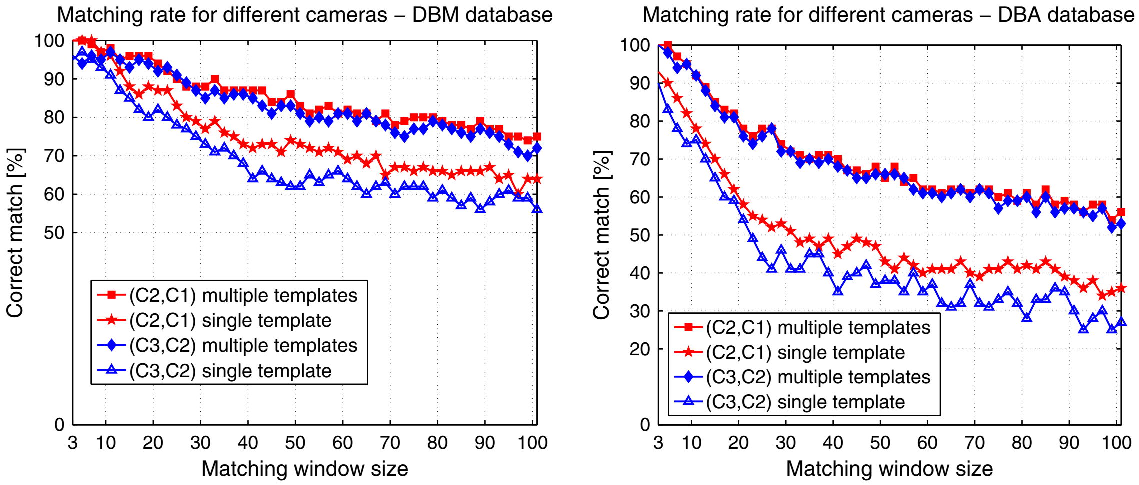

Fig. 9. Comparison of the results of our matching method with a single and multiple templates, for two camera pairs (C_2,C_1) and (C_3,C_2) and datasets of manual (DBM, left column) and automatic (DBA, right column) vehicle detections. Each curve depicts the percentage of correct matches on the y-axis in function of the number of matching candidates (the matching window size) on the x-axis.

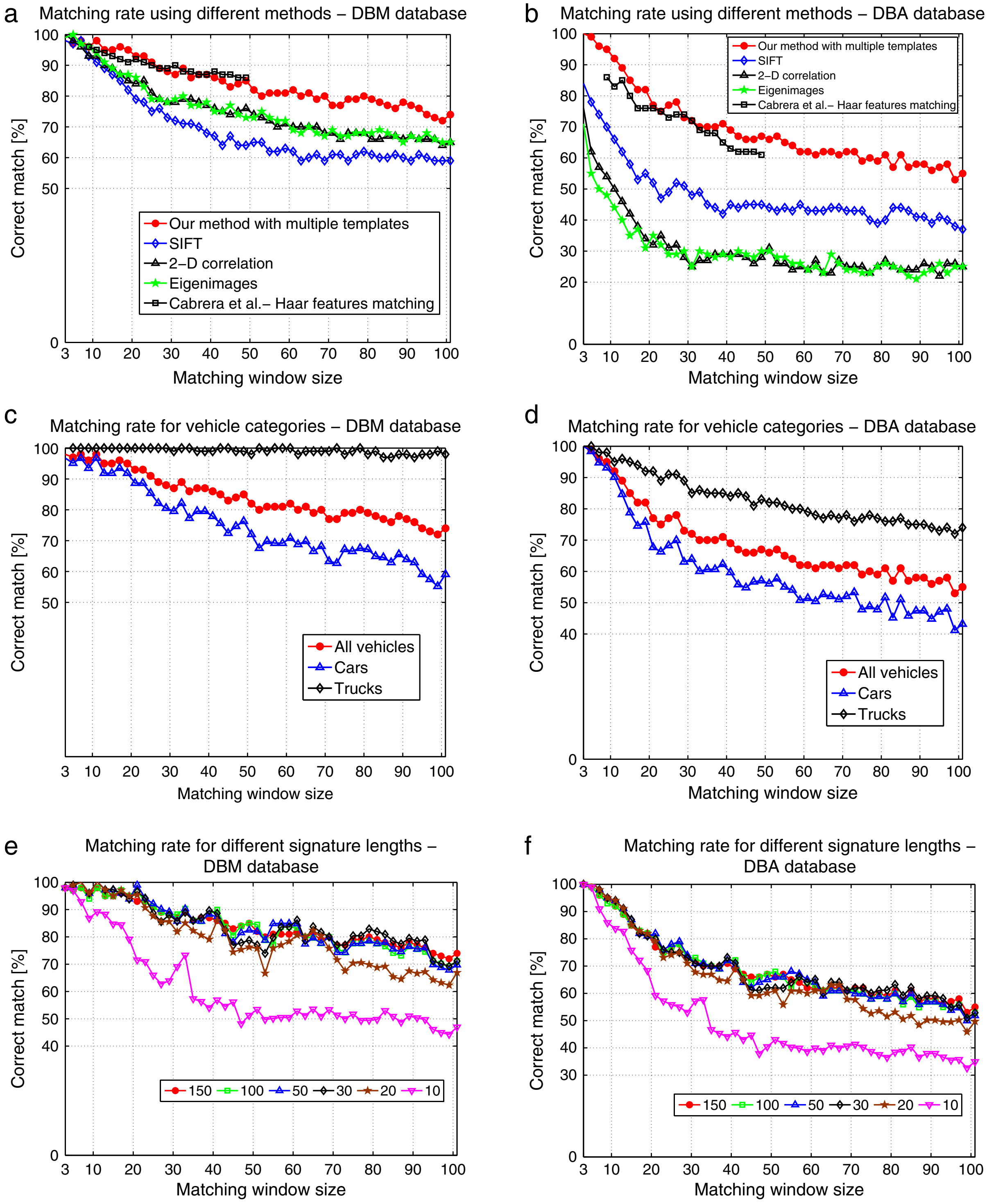

Fig. 10. The matching results of our method on manual (DBM, left column) and automatic vehicle detections (DBA, right column), averaged over two camera pairs ((C_2,C_1)) and ((C_3,C_2)). Each curve depicts the percentage of correct matches on the *y*-axis in function of the number of matching candidates (the matching window size) on the *x*-axis: a–b) comparison with other methods; c–d) results for different categories of vehicles; e–f) results using the signatures of different lengths.

smaller vehicles (cars), see Fig. 1. On the other hand, big vehicles like trucks, buses and vans usually have characteristic patterns (different design, company logos, commercials and so on), which are visually distinctive and big enough to be visible in low resolution videos. To evaluate the influence of this constraint on performance of our matching method, we have divided our database of vehicles in two categories, denoted as *cars* and *trucks*. Category *cars* contains 185 vehicles, while other 115 vehicles in the database are categorized as *trucks*. The matching results, obtained using our signature based method with collection of good observations, are in Fig. 10c–d presented separately per each category.

These results clearly show that the proposed method is highly accurate for matching vehicles from the *trucks* category. Even when the templates are compared with as much as 101 candidates, more than 96% of *trucks* in the DBM set and above 70% in the DBA set are correctly matched. This is beneficial for applications where tracking trucks is more important, e.g., for tracking of vehicles that transport dangerous goods. The accuracy in the *car* category is much lower and it shows that the success of appearance matching is highly limited by the quality of images from surveillance cameras and the distinctiveness of vehicles themselves. One way to increase vehicle distinctiveness could be using the color information in the environments where it is available.

7.4. Results for different reference lengths

As explained in Section 5.1, the signatures are rescaled using the reference length l_r before performing the matching operations. Thus, the reference length has a major impact on the amount of computations needed for signature matching. In Fig. 10e–f we present the matching results obtained using different reference lengths, to evaluate their influence on the performance of the method.

We see that similar results are obtained for reference lengths in the range from 30 to 150 and that the performance drop is noticeable when the lengths are below 30 points (the curves for 20 and 10 are shown). This shows that the signatures can be highly downsampled between the local extrema, without affecting the performance significantly. It is due to the fact that the local extrema of the signatures capture most of the information, so most of the points between the extrema can be discarded. However, discarding all points except the local extrema leads to a performance drop because the shape of the signature between the extrema captures subtle appearance differences, which are important to distinguish similar vehicles (signatures of the vehicles in our database have 12 local extrema on average while the performance drop is noticeable when the signatures contain less than 30 points).

The possibility of downsampling the signatures for the reference length $l_r = 30$ before their matching, enables very efficient performance of vehicle matching, both in terms of data and computations. In our implementation of the proposed algorithm, matching of two appearance states was achieved in 1.02 ms on a single-core 1.86 GHz CPU. Such efficiency allowed us to compare in 11.5 s all 300 vehicles viewed by two cameras in a period of 8 min. Also, each signature vector was computed in less than 1 ms, which enabled calculation of signatures and collection of good observations online, during tracking of vehicles in a single camera view.

7.5. Comparison of 2-D and 4-D signature vectors

In Section 4.1 we have defined the appearance representation using two and four signatures. The previous results are all obtained using the appearance model based on two signatures (vertical and horizontal), while in this Section we analyze the gain from adding two additional signatures. A comparison of the results obtained using the two appearance models is given in Fig. 11. We see that for manual detections there is a slight increase of accuracy (approx. 5 to 10%) when the diagonal signatures are added, but it is negligible for automatic detections. This suggests that most of vehicle parts and patterns can be distinguished in vertical or horizontal projections. Also, the diagonal signatures are more sensitive to the detection misalignment and the vehicle pose change. Taking into account that the diagonal signatures triple the amount of data needed for appearance representation and matching, we propose using the appearance model based on only vertical and horizontal signatures.

7.6. Results in a tunnel application

In the previous sections the results of the vehicle appearance matching have been shown for different sizes of the matching window. However, in most traffic environments it is possible to reduce the number of candidates for each template. For this purpose we use the information based on space-time consistency of vehicle motion, as explained in Section 6. In this way, taking 30 kmph for the minimal and 160 kmph for the maximal velocity of vehicles in a tunnel application with medium traffic density, we have been able to reduce the matching window size to on average 9 candidates for each template (each vehicle was compared with 9 other vehicles). In this setting the accuracy of our 2-signature based vehicle matching between two successive cameras along the tunnel was 97% on the DBM set and 95% on the DBA set, see Fig. 10a–b. Classified per vehicle categories, as shown in Fig. 10c–d, correct matching was achieved for 100% of trucks and 95% of cars on the DBM set, while 98% of trucks and 94% of cars on the DBA set. In Fig. 10a–b we also see that our method outperforms the state-of-the-art vehicle matching method of Rios Cabrera et al. [1] by 9% on the DBA set, while they perform similarly on the DBM set. In our method there is also no need for supervised training, which is an additional advantage compared to the method of [1].

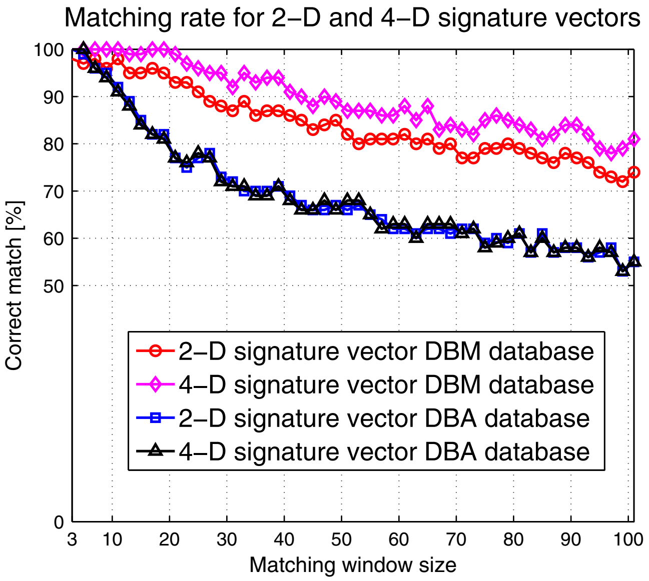

Fig. 11. Comparison of the results using the appearance models based on two and four signatures, averaged over two camera pairs (C_2,C_1) and (C_3,C_2). Each curve depicts the percentage of correct matches on the *y*-axis in function of the number of matching candidates (the matching window size) on the *x*-axis.

7.7. Discussion

In this Section we discuss reasons for possible failures of vehicle matching using the proposed framework and we propose some solutions to prevent these failures.

1. *If multiple vehicles from the same matching group (vehicles that appear at approximately the same time in the scene) appear in poses significantly different from the poses observed in previous cameras.* When a vehicle appears in a camera in a pose significantly different from the poses observed in previous cameras, the multi-observation appearance model of the vehicle would not contain a descriptor of the vehicle in such a pose, and this could likely lead to a failure in vehicle matching. To reduce the chance of such a failure, the proposed framework has a matching optimization step, which results in the optimal assignment for each vehicle in the corresponding matching group. Consequently, this means that the failure would possibly happen if multiple vehicles, not only one, from the same matching group change pose to a previously not observed pose, which is less likely. By extending the proposed framework with a multi-hypotheses alike approach it would be further possible to

recover from incorrect assignments over time, as there are more observations of each vehicle from multiple cameras.

2. *If a vehicle is very similar or the same in appearance with some other vehicle or vehicles from the same matching group.* There are a lot of vehicles, especially cars, of the same make or with similar appearance, which is a significant challenge for any appearance based vehicle matching method. Oftentimes, it is not possible to re-identify vehicles based on their appearance only. In the proposed framework it is, therefore, possible to add additional information to select matching candidates for each vehicle based on vehicle kinematics, and this is what we exploit in this paper. The proposed framework also supports a more precise selection of the matching candidates by using the contextual information such as vehicle constellations, and probabilities and evidences of changes in these constellations (e.g., probabilities or evidences that a vehicle overtook another vehicle, changed the lane, etc.), but in this paper we did not use this additional information.
3. *If a vehicle is occluded or inaccurately detected so that significant or distinctive parts of the vehicle are not captured.* To correctly match two vehicles using any appearance based matching method it is important that there is enough visual information to do the matching. If some vehicles are occluded or inaccurately detected, a significant amount of information might be lost. Therefore, in the proposed framework we introduced a method to automatically detect such cases and select good observations to perform matching. As shown in our experiments (see Fig. 10a–b) there is a significant improvement in the matching performance due to the selecting of good observations. However, if a vehicle is significantly occluded or detected with high inaccuracy in a whole scene, its re-identification would likely fail due to absence of good observations. This is in some cases solved by matching optimization in the proposed framework, but it could be further improved by the previously mentioned approaches of multi-hypotheses and contextual alike matching.

8. Conclusion

In this paper we proposed a novel method for vehicle appearance modeling and matching. We proposed using image projection profiles to obtain vehicle signatures that significantly reduce the amount of data needed for vehicle matching. We showed that in low resolution images such signatures capture well the spatial distribution of vehicle parts and patterns, which was used for their matching. We showed also that by selecting a set of good observations along the multi-camera track, it was possible to overcome many matching problems that occur due to inaccurate detections, intensive illumination changes, clutters and occlusions, as well as changes of the vehicle appearance. As a consequence, object matching itself was significantly simplified and yet outperformed more complex methods. The presented results also show that it is possible to highly downsample the signatures without affecting the performance significantly, which further reduces the amount of computations needed for their matching. Thus, the proposed appearance matching method can be used to obtain vehicle matching on embedded systems (e.g., smart cameras) or by a low-complexity central server without a need for sending the images between the cameras or to the server.

An interesting future direction is to extend this approach towards matching of vehicles in traffic environments in which camera views are significantly different, e.g., along city roads or crossroads. In this case it is important to add automatic detection of good appearance states for matching depending on the cameras' view. Also, in the environments where color information is available, it can be incorporated to further increase matching accuracy.

Acknowledgments

This research has been supported by the Flemish Interdisciplinary Institute for Broadband Technology (IBBT).

References

[1] R. Rios Cabrera, T. Tuytelaars, L. Van Gool, Efficient multi-camera vehicle detection, tracking, and identification in a tunnel surveillance application, Computer Vision and Image Understanding 116 (2012) 742–753.

[2] P. Viola, M.J. Jones, Robust real-time face detection, International Journal of Computer Vision 57 (2) (2004) 137–154.

[3] M. Turk, A. Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience 3 (1991) 71–86.

[4] H. Bischof, H. Wildenauer, A. Leonardis, Illumination insensitive recognition using eigenspaces, Computer Vision and Image Understanding 95 (2004) 86–104.

[5] O. Sidla, L. Paletta, Y. Lypetskyy, C. Janner, Vehicle recognition for highway lane survey, International Conference on Intelligent Transportation Systems, 2004, pp. 531–536.

[6] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[7] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, SURF: speeded up robust features, Computer Vision and Image Understanding 110 (3) (2008) 346–359.

[8] G. Yu, J. Morel, Asift: An algorithm for fully affine invariant comparison, Image Processing On Line, 2011.

[9] Y. Shan, H. Sawhney, R.T. Kumar, Unsupervised learning of discriminative edge measures for vehicle matching between non-overlapping cameras, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (4) (2008) 700–711.

[10] Y. Guo, S. Hsu, H.S. Sawhney, R. Kumar, Y. Shan, Robust object matching for persistent tracking with heterogeneous features, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (5) (2007) 824–839.

[11] F. Porikli, Inter-camera color calibration using cross-correlation model function, International Conference on Image Processing, vol. 2, 2003, pp. 133–136.

[12] O. Javed, S. K., M. Shah, Appearance modelling for tracking in multiple non-overlapping cameras, International Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE, 2005, pp. 26–33.

[13] T. Hou, S. Wang, H. Qin, Vehicle matching and recognition under large variations of pose and illumination, International Conference on Computer Vision and Pattern Recognition, 2009, pp. 290–295.

[14] T. Huang, S. Russell, Object identification in a Bayesian context, Proceedings of International Joint Conferences on Artificial Intelligence, 1997.

[15] V. Kettnaker, R. Zabih, Bayesian multi-camera surveillance, International Conference on Computer Vision and Pattern Recognition, 1999, pp. 253–259.

[16] R. Collins, A. Lipton, H. Fujiyoshi, T. Kanade, Algorithms for cooperative multi-sensor surveillance, Proceedings of the IEEE 89 (10) (2001) 7–12.

[17] O. Javed, K. Rasheed, M. Shafique, M. Shah, Tracking across multiple cameras with disjoint views, International Conference on Computer Vision, 2003, pp. 952–957.

[18] T. Choe, L. M., N. Hearing, Traffic analysis with low frame rate camera networks, First IEEE Workshop on Camera Networks, 2010, pp. 9–16.

[19] T. Hou, S. Wang, H. Qin, Active lighting learning for 3D model based vehicle tracking, International Conference on Computer Vision and Pattern Recognition, 2010.

[20] Y. Shan, H. Sawhney, R.T. Kumar, Vehicle identification between non-overlapping cameras without direct feature matching, IEEE International Conference on Computer Vision, 2005.

[21] M. Betke, E. Haritaoglu, L.S. Davis, Real-time multiple vehicle detection and tracking from a moving vehicle, Machine Vision and Applications 12 (2000) 69–83, (Springer-Verlag).

[22] S. Lee, Y. Liu, R. Collins, Shape variation-based frieze pattern for robust gait recognition, International Conference on Computer Vision and Pattern Recognition, 2007.

[23] S. Rusinkiewicz, M. Levoy, Efficient variants of the ICP algorithm, International Conference on 3D Digital Imaging and Modeling, 2001, pp. 1–8.

[24] T. Sebastian, P. Klein, B. Kimia, On aligning curves, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (1) (2003) 116–125.