Unsupervised Processing of Vehicle Appearance for Automatic Understanding in Traffic Surveillance

Jakub Sochor[1], Adam Herout
Graph@FIT, Brno University of Technology, Czech Republic
Email: isochor,herout@fit.vutbr.cz

***Abstract*—This paper deals with unsupervised collection of information from traffic surveillance video streams. Deployment of usable traffic surveillance systems requires minimizing of efforts per installed camera – our goal is to enroll a new view on the street without any human operator input. We propose a method of automatically collecting vehicle samples from surveillance cameras, analyze their appearance and fully automatically collect a fine-grained dataset. This dataset can be used in multiple ways, we are explicitly showcasing the following ones: fine-grained recognition of vehicles and camera calibration including the scale. The experiments show that based on the automatically collected data, make&model vehicle recognition in the wild can be done accurately: average precision 0.890. The camera scale calibration (directly enabling automatic speed and size measurement) is twice as precise as the previous existing method. Our work leads to automatic collection of traffic statistics without the costly need for manual calibration or make&model annotation of vehicle samples. Unlike most previous approaches, our method is not limited to a small range of viewpoints (such as eye-level cameras shots), which is crucial for surveillance applications.**

I. Introduction

We are dealing with automatic and accurate visual traffic surveillance. In this particular work, we aim at accurate and automatic calibration of camera including scene scale, and at fine-grained recognition of passing vehicles. We show that both these problems can be successfully approached by analyzing observed traffic in an unsupervised manner. Instead of providing the system with a large degree of supervision – performing camera calibration procedures and manually annotating sufficiently large datasets of vehicle samples – we let it collect as much information as possible by itself and provide small amount of annotation ex post, as the final bit needed for its understanding. Figure 1 shows examples of three groups of vehicles obtained fully automatically on different cameras. We show in this paper that once having these exact make&model examples, it is possible to calibrate the camera very accurately including scale, and to train specific car detectors. A crucial point is to distinguish between very subtle visual differences, and at the same time, we want the system to include in one class samples of an identical car model under varying lighting conditions, in different environments, different in their color, etc.

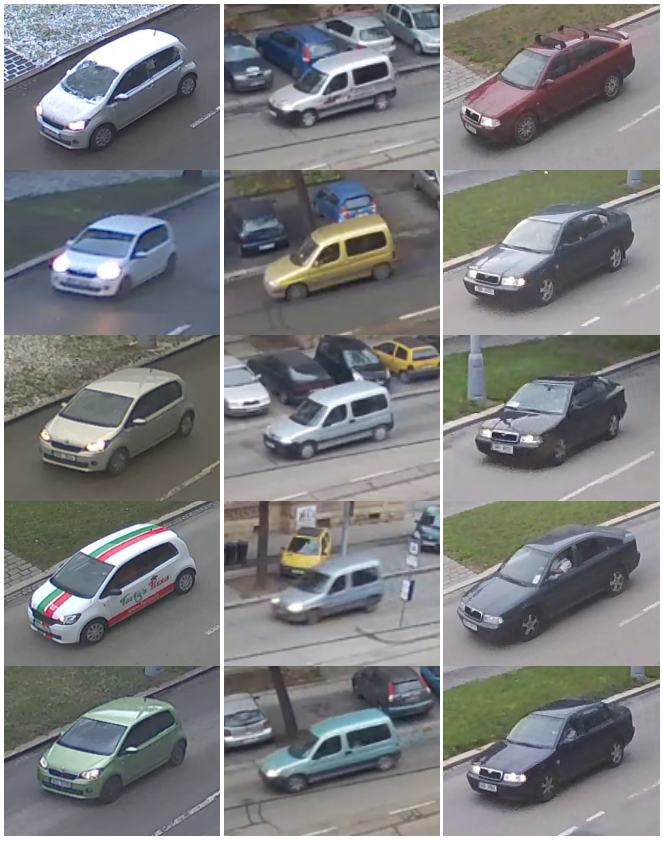

Fig. 1. Examples of groups of vehicles obtained from different cameras at different locations. The groups were obtained from acquired dataset of 472 k vehicles (3 images per vehicle) from 7 different camera positions at multiple locations in multiple countries.

Recognition of exact make and model of cars has attracted more attention recently. Most approaches are limited to a single viewpoint or a narrow range of viewpoints [1], [2], [3], [4], [5]. Only recently, Hsiao et al. [6] proposed a make and model recognizer operating from freely displaced viewpoints. They train on rendered data (still limited to eye-level views) of eight different Honda models. Similar achievements are still limited by training and evaluation data – the sizes of available datasets are rather modest. Collecting datasets of real-world samples for fine-grained car classification is a difficult task. Stark et al. [7] collected a dataset of 14 categories (1904 samples in total) and they comment: *"We note that the data set is heavily biased w.r.t. viewpoints, which reflects the availability of images we encountered during data collection. It proved almost impossible to collect more than a handful of images*

[1]Jakub Sochor is a Brno Ph.D. Talent Scholarship Holder — Funded by the Brno City Municipality. This work was supported by TACR project "RODOS", TE01020155 and by the research CEZMSMT project "IT4I", CZ 1.05/1.1.00/02.0070.

for certain combinations of car-type and viewpoint." Krause et al. [8] mention other difficulties they faced when collecting a larger dataset (16,185 images, 197 classes). They crawl popular car websites and analyze the car models. Based on the list of models they obtained images by using Flickr, Google, and Bing, and annotated/verified the set manually. Among the difficulties, they mention: *"The class list of cars changes on a yearly basis, the appearance of some models of cars remains constant from year to year, and typical car websites may even list cars which differ only in terms of non-visual features, e.g. their engine, as separate classes."* The viewpoints are also distributed very unevenly – labeled images of cars posted online by fans are typically taken from a limited range of viewpoints. Lin et al. [9] collected their own dataset of 300 images of 30 car models, because they missed landmark annotation in the previously mentioned sets.

When it comes to fine-grained vehicle classification, many approaches are limited to frontal or rear viewpoint and they are based on detection of the license plate for ROI extraction [1], [2], [3], [4]. Authors of these papers are using different schemes for extracting the feature vectors and for the classification itself. Stark et al. [7] use fine-grained categorization of cars in order to obtain metric information and obtain a rough estimate of depth information for single images (containing cars in usable poses). Krause et al. [10] reconstruct 3D models of cars by wide baseline stereo and combine the recognized 3D model with 2D representations (locality constrained linear coding LLC [11] and BubbleBank [12]) for classifying the vehicles. Another approach proposed by Prokaj and Medioni [13] is based on pose estimation and it is able to handle any viewpoint. The authors suggest to use 3D models of vehicles, fit them to the recognized pose, project them to 2D and use SIFT-like features for the comparison of the vehicles.

Existing systems for camera calibration including scale often involve physical measurements in the scene of interest [14], known length of lane marking [15], or marking dimensions [16], known camera position [15], [17] or average vehicle size [18]. A novel state-of-the art approach proposed by Dubská et al. [19] was published recently. This approach uses three-dimensional bounding boxes and mean dimensions of vehicles; therefore, all vehicles are treated the same way and assumed they will have identical dimensions. In contrast to this approach, we propose camera scale calibration which treats groups of vehicles separately and it assumes that only vehicles of the same type have the same dimensions.

We propose to collect fine-grained vehicle classes in a totally unsupervised manner. We process continuous video from a static surveillance camera and track the passing cars. From each tracked car we obtain several (3-4) representative images at predefined locations. These groups of images are analyzed pairwise by a dissimilarity cost function and clustered into clusters of common appearance. The clustering can be done in a conservative way, so that only samples with very low dissimilarity fall into one cluster. These clusters are annotated manually with only little human effort. Along with the make and model annotation, also the viewpoint direction and scale is automatically obtained for each of the samples. For static cameras, the viewpoint direction and scale can be derived from vanishing points that can be fully automatically extracted for a roadside surveillance camera [19].

Clusters obtained by our approach can be matched from camera to camera or between different time spans of observation. Matching clusters from a single camera at different times is important for processing over time. The ability to construct clusters from limited time spans and to join them together later allows for continuous processing of input cameras and for gradually growing the collected dataset. The between-camera matching is used for transferring manual annotations made for one camera to other views so that each make&model cluster contains samples from various viewpoints and of many individual instances of the given car at different locations. The between-camera matching can also transfer camera scales and our algorithm can thus be used for precise automatic camera calibration, including the scale. The accuracy of the clustering and camera scale inference is measured experimentally and reported in Section III.

The main contributions of this paper are the following:

- We propose a new unsupervised approach to traffic understanding and camera calibration. It is specific for traffic surveillance – dataset creation, camera calibration, size+speed measurement, etc.
- Our approach beats the state-of-the-art in traffic camera calibration including scale [19], leading to twice as accurate length and speed measurement.
- Our approach leads to unsupervised traffic dataset collection from all viewpoint angles imaginable in traffic surveillance. This contrasts to majority of available traffic datasets, focusing on eye-level viewpoints with almost constant Yaw angle [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31].
- We are making the collected datasets (472k of vehicles, 3 images each, various viewpoints relevant for traffic surveillance) and the source codes publicly available for future development and benchmarking[1].

The structure of the text is following: First, we describe into a detail the algorithms for automatic grouping and matching of vehicles in Section II. The used method for scene scale inference is also described in this section. Then, in Section III, we evaluate our algorithms in numerous scenarios. Finally, we conclude our work in Section IV.

II. Unsupervised Similarity-Based Vehicle Grouping

The images of vehicles for grouping are obtained automatically from traffic surveillance cameras. 3D bounding boxes are constructed around the cars [19], Figure 3. Samples of vehicles are acquired when the center of their base is close to manually predefined positions (see Figure 2). The set of such designated positions will be denoted as $\mathbb{P}$ and an individual position as $\alpha \in \mathbb{P}$. The dimensions and the mask of the vehicle foreground [32], [33] are also acquired for each of the samples. Only vehicles manifested in all of the designated positions α are further processed. In the following text, v denotes a vehicle from the set of vehicles V; the number of vehicles is denoted $N = |V|$.

Our goal of the following effort is to automatically obtain groups of vehicles with the same make&model (potentially

[1]The dataset and source codes are available at medusa.fit.vutbr.cz/traffic/

Fig. 2. Acquisition of vehicle samples. The surveillance camera tracks moving vehicles (green line represents one vehicle track). Several (typically 3) image locations are manually designated (red crosses) and vehicle locations corresponding to them are selected based on distance of the center of the 3D vehicle base location.

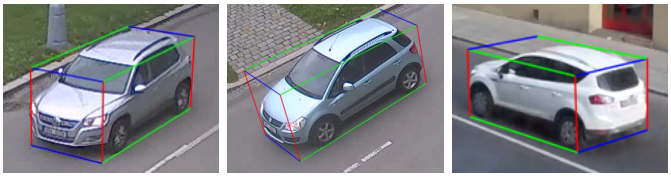

Fig. 3. Examples of three-dimensional bounding boxes constructed according to [19]. Lengths, widths, and heights are denoted by green, blue, and red color.

Fig. 4. A typical example of surveillance camera where vehicle identification or recognition based on SIFT matching fails, as totally different SIFTs are detected at different positions.

also submodel and model year). The computer vision approach of first choice is to extract keypoints and their descriptors and match the images based on such visual keywords. However, Figure 4 shows that in the given image quality and resolution, this approach would not lead to desired outcomes – the SIFTs (other available keypoint detectors behave similarly) are not stable and do not respond to the identical object stably. To deal with this problem, we propose to use edge-based similarity for vehicles as the edges are present on non-textured vehicles even in low resolutions. See Figure 11 for the resulting extracted vehicles.

For each vehicle v and position α, edge points $F_v^\alpha = \{(x_1, y_1), \dots (x_n, y_n)\}$ are detected by an edge detector [34], [35] and their orientation $O_v^\alpha : F_v^\alpha \mapsto \mathbb{R}$ is also computed. We define a dissimilarity cost between two vehicles $u, v \in V$ in the following way. First, edge points F_v^α are aligned to edge points F_u^α by ICP [36] in order to reduce small movements caused by taking the image at a slightly different position. The transformation aligning points F_v^α to F_u^α obtained by ICP is denoted as ϕ. We propose to use dissimilarity cost d_{uv}, Eq.

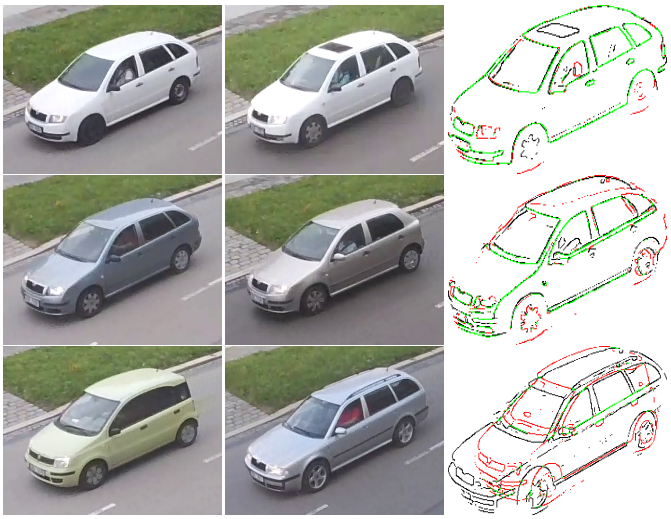

Fig. 5. Dissimilarity cost between vehicle samples. **left:** reference vehicle, **middle:** test vehicle, **right:** edge dissimilarity visualization (*black:* test vehicle, *green:* pixels of reference vehicle with a match, *red:* pixels of reference vehicle without a match).

(1), which is inspired by Chamfer Matching [37]. Threshold λ is (in our experiments) set to 1 and $\sphericalangle[\cdot,\cdot]$ denotes the unsigned difference between two unoriented angles from range $\langle 0, \pi\rangle$; therefore, the maximal angular difference is $\frac{\pi}{2}$.

$$d_{uv} = \sum_{\alpha \in \mathbb{P}} d_{uv}^\alpha \tag{1}$$

$$d_{uv}^\alpha = \frac{\sum\limits_{p_u^\alpha \in F_u^\alpha} d(p_u^\alpha, \arg\min\limits_{p_v^\alpha \in \phi(F_v^\alpha)} ||p_u^\alpha - p_v^\alpha||)}{|F_u^\alpha|}$$

$$d(p_u^\alpha, p_v^\alpha) = \begin{cases} \lambda & ||p_u^\alpha - p_v^\alpha|| > \lambda \\ \dfrac{2\sphericalangle[O_u^\alpha(p_u^\alpha), O_v^\alpha(p_v^\alpha)]}{\pi} & \text{otherwise} \end{cases}$$

Let us emphasize some properties of the dissimilarity cost. It is useful that it does not depend on the color of the vehicle nor exact location and small changes are allowed. Some examples are shown in Figure 5. Also, one can notice that the score is not inherently symmetric ($d_{uv} \neq d_{vu}$) and the symmetrization will be introduced later.

The dissimilarity cost should not be computed for every two pairs of vehicles as it would be computationally infeasible – there are $N(N-1)$ combinations. Such computation would be bearable up to thousands or small tens of thousands vehicles, but in our work, we are aiming at millions. Therefore, we use methods of pre-selection to decrease the number of comparisons. We take only first N_B vehicles which will be used for bootstrapping the computation. Hu [38] invariants $(m_1^\alpha, \dots, m_7^\alpha)$ are computed for shapes of these vehicles for each position α. Then, for every pair of vehicles (u, v) from N_B vehicles function ζ, Eq. (2), is evaluated.

$$\zeta({}_u m, {}_v m) = \sum_{\alpha \in \mathbb{P}} \sum_{i=1}^{7} |\operatorname{sgn}{}_u m_i^\alpha \cdot \log {}_u m_i^\alpha - \operatorname{sgn}{}_v m_i^\alpha \cdot \log {}_v m_i^\alpha| \tag{2}$$

The computation of the function is fast enough to perform $N_B(N_B-1)$ comparisons even for high N_B. Finally, for each

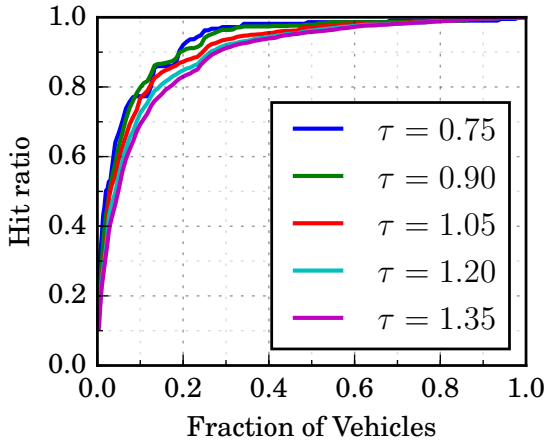

Fig. 6. Evaluation of inflection of pre-selection of pairs to compare by Hu invariants. The curves show how many values below certain threshold τ would be found for different N_H/N_B ratios, which is represented by X axis.

vehicle from the first N_B vehicles only N_H dissimilarity costs are computed with the vehicles with the lowest function ζ and for other pairs of vehicles (u, v) it is assumed that $d_{uv} = \infty$. We evaluated how much of values below threshold τ (threshold for graph construction) will be found for different ratio of N_H/N_B and it is shown in Figure 6. The $N - N_B$ vehicles will be processed in a different, classification-based approach whose description follows. Figure 6 indicates that N_H can be set around $10 - 20\,\%$ to reasonably cover most matches, but at the same time to reduce the computation considerably.

A. Grouping by Graph Analysis

Weighted graph $\mathcal{G}_\tau = (\mathcal{V}_\tau, \mathcal{E}_\tau, \mathcal{W}_\tau)$ based on the dissimilarity costs is then constructed. The graph is undirected as the costs are symmetrized (3), (5), and threshold τ defines the maximal value of dissimilarity score when the vehicles are considered to be similar (should fall into an identical group):

$$\mathcal{E}_\tau = \left\{(u, v) \in V^2 \,|\, d_{uv} \leq \tau \wedge d_{vu} \leq \tau \wedge u \neq v\right\} \quad (3)$$

$$\mathcal{V}_\tau = \{v \in V \,|\, \exists u \in V : (u, v) \in \mathcal{E}_\tau\} \quad (4)$$

$$\mathcal{W}_\tau = \{((u, v), w) \,|\, (u, v) \in \mathcal{E}_\tau \wedge w = \max(d_{uv}, d_{vu})\} \quad (5)$$

The graph is divided into connected components (candidates for the target groups) which are further examined so that the identified groups are "compact". The motivation is to avoid the cases when two groups of distinct car types with high connectivity within them are connected by just one or two links with high cost, while these groups should be separate. This happens from time to time and it is overcome by the following iterative algorithm. We propose to use Algorithm 1 to eliminate these cases. The algorithm recursively examines connected components in the graph and when a connected component has too small number of edges relative to the number of nodes, the edge with the highest weight is removed and the component is processed in recursive manner. The relative number of edges and nodes is expressed by function ω:

$$\omega(\mathcal{V}, \mathcal{E}) = \frac{2 \cdot |\mathcal{E}|}{|\mathcal{V}| \cdot (|\mathcal{V}| - 1)} \quad (6)$$

The rationalization behind the recursive removing of edges is the following. The graph is built in a way that it is possible

Algorithm 1 Grouping by recursive graph analysis

input: Graph $\mathcal{G}_\tau = (\mathcal{V}_\tau, \mathcal{E}_\tau, \mathcal{W}_\tau)$
output: Set of groups G

```
G = ∅
for G_cc ∈ connComps(V_τ, E_τ, W_τ) do
    GROUPING(V_cc, E_cc, W_cc)

function GROUPING(V_c, E_c, W_c)
    if ω(V_c, E_c) ≥ κ then
        G = G ∪ {V_c}
    else
        e_max ← arg max_{e∈E_c} W_c(e)
        E_c ← E_c \ {e_max}
        W_c ← W_c \ {(e_max, W_c(e_max))}
        for G_cc ∈ connComps(V_c, E_c, W_c) do
            GROUPING(V_cc, E_cc, W_cc)
```

that one false edge will connect two groups of distinct vehicles. However in our experiments these edges represented large dissimilarity cost relative to others and they can be removed to separate these two groups of vehicles.

The process described up till now is used for clustering an initial collected set of (tens to hundreds of thousands) of vehicle samples V. This process leads to stable and coherent clusters (see experiments below) which serve as seeds for further growing and collecting samples at different lighting and weather conditions in linear time.

For grouping further collected vehicles, classifiers are trained for the largest groups of vehicles (the significant groups) which is able to distinguish vehicles from different clusters and join the new samples into them. In our experiments, we used LBP features [39] fed to SVM. We used N_C classifiers for N_C groups of vehicles when for each classifier, a group is considered to be positive and all other vehicles in groups are considered to be negative examples.

B. Scene Scale Inference

Current state-of-the art for scale calibration proposed by Dubská et al. [19] is based on the mean vehicle estimation for every observed vehicle. We propose a different approach also based on vehicles' dimensions. We estimate the scale as linear least squares approximation of real vehicles dimensions and measured dimensions of vehicles in the image [19]. See Figure 12 for example of the scene scale inference, we used the dimensions of vehicles (length, width, height) to obtain transformation function from relative 3D image distance into real world distance measured in meters. The function in our case is linear and acquired by least squares approximation of the vehicles' dimensions. The approximation of vehicle dimensions improves the scale calibration as we have more measurements of vehicles' dimensions than just a single mean as the previous method [19].

III. Experimental Results

We acquired a large dataset consisting of 472k of vehicles with 3 images per vehicle (see Figure 2). The dataset contains 13 different lanes (one or more per camera) and each lane

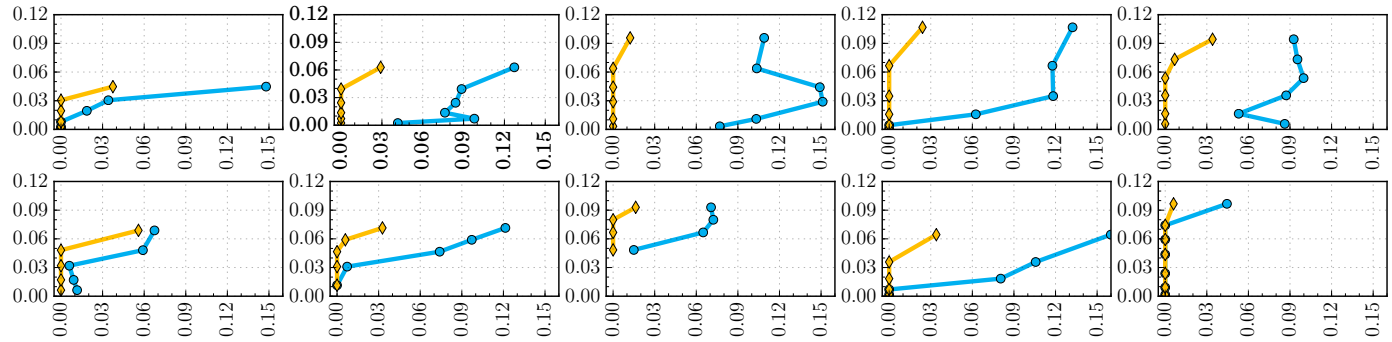

Fig. 7. Grouping algorithm manual evaluation. Each graph represent one camera and lane. The mean false positive rate across 15 largest groups, represented by the X axis, for each settings of thresholds. Y axis represents the fraction of vehicles in the dataset included in clusters with more than 5 vehicles. Yellow curves denote evaluation of make&model grouping. The blue curves represent evaluation when even different submodels of one make&model are considered as different types. The curves are not monotonically increasing as the number of vehicles in the groups may grow and the absolute number of false positives stay the same.

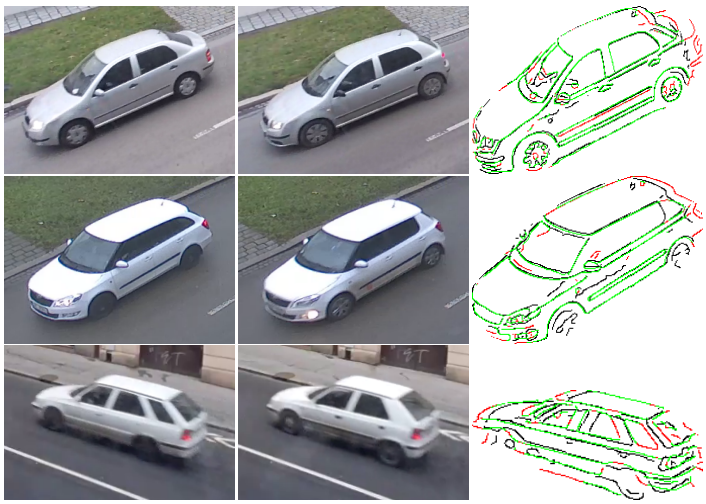

Fig. 8. Rows represent examples of false positives for different groups of vehicles. As it is seen the vehicles are very similar and differ only in the submodel of the vehicle (eg. long version and sedan version).

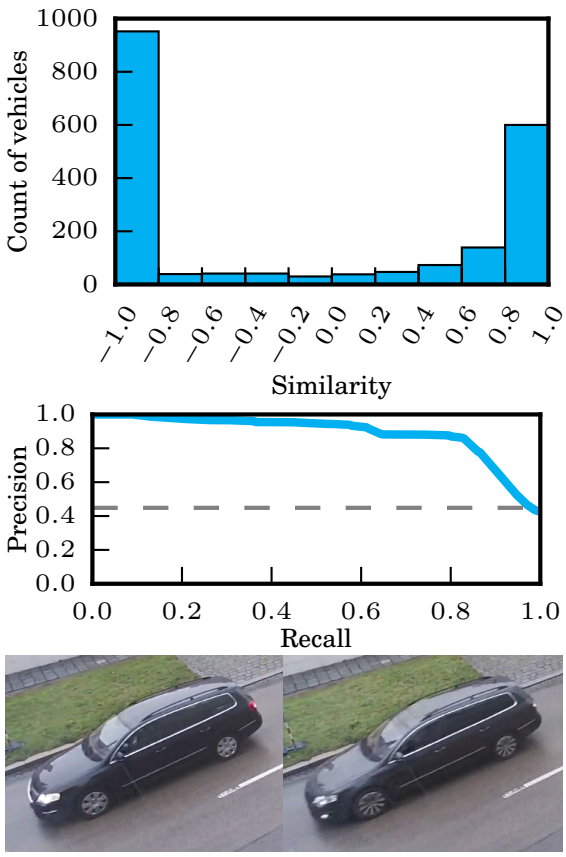

Fig. 9. Evaluation of accuracy of the grouping algorithm on humans annotated pairs of vehicles. People annotated 2,000 pairs of vehicles as "same type" and "different type" while identical types of vehicles should be the same in the make, model, submodel, and model year. **top:** Histogram of normalized annotation responses, where 1.0 means that every annotation for the pair is "same vehicle" and -1.0 denotes the opposite. The middle image shows Precision-Recall Curve for the grouping algorithm when the pairs annotated by human are considered as ground truth, gray dashed line is the baseline. Finally, the bottom image shows a pair of vehicles which are annotated (correctly) as "different type".

is processed separately, Fig. 2. The cameras were installed in multiple cities and countries and the types of observed roads range from local ones to large roads with intensive traffic.

The evaluation is focused mainly on automatic grouping and classification accuracy, precision of comparison of vehicles, and accuracy of scene scale inference. The comparison with other methods [1], [2], [3], [4], [5] on fine-grained recognition is not relevant as they usually require the frontal viewpoint.

A. *Automatic Grouping Quality*

Our first experiment to evaluate the quality of vehicle grouping is based on the quantity of vehicles in the found groups of vehicles and the mean false positive rate across these groups. Figure 7 demonstrates the result where the yellow lines represent the quality of groups. To obtain the evaluation, we performed the grouping on numerous cameras with varying thresholds and for each camera and lane we evaluated the mean false positive rate across the 15 largest (the significant ones) groups and the number of vehicles in the groups of vehicles with more than 5 vehicles. For the experiment, we used $N_B = 20,000$ and $N_H = 10,000$.

In order to examine our method further, we also evaluated

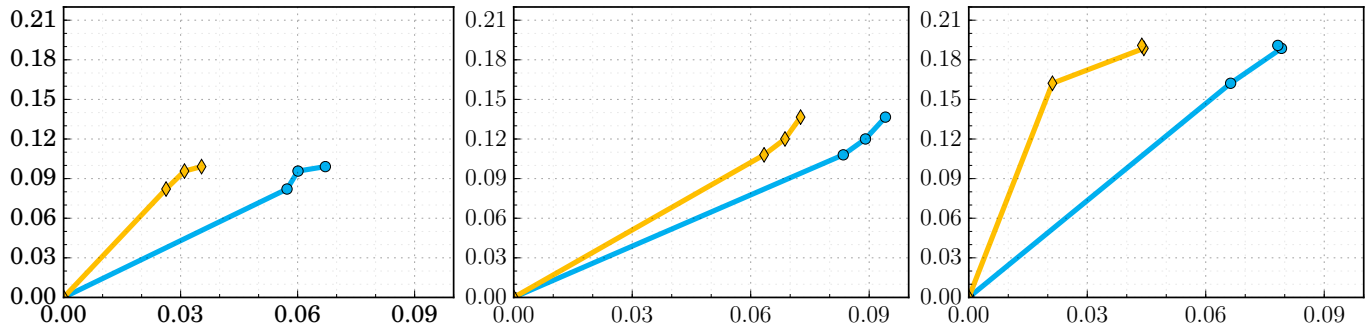

Fig. 10. Evaluation of the classifiers trained on the automatically obtained groups of vehicles: the mean false positive rate across 10 largest groups, represented by the X axis, for each settings of thresholds. The Y axis represents the fraction of vehicles added to the first 10 clusters from the whole dataset. Yellow curves denote evaluation of make&model grouping. Blue curves represent evaluation when even different submodels of one make&model are considered as different types.

Fig. 11. Examples of groups of vehicles. Yellow boxes contain representatives of groups obtained by the proposed unsupervised grouping. Vehicles added by the trained classifiers are in the green boxes. False positives for each group added by the classifiers are in the red boxes. As the images show, our method is robust to color (row 1) and markers variations (row 2), weather (rows 2,3) and lighting changes (rows 2,3) and partial occlusions (see row 4, where vehicles are partially occluded by a tram or by a plastic bag on a tree). Note that the vehicle recognizer works well despite high occlusion, scene clutter and low image quality.

how our algorithm behaves when we consider also different submodels (eg. long version and sedan version) of one make&model as different types. The results are included in Figure 7 as blue curves. This task is really challenging as the submodels are usually very similar and differ only in small subtle details. For examples of such situations see Figure 8.

B. Evaluation of Grouping on a Human-Annotated Dataset

In order to examine how our method will be able to distinguish pairs of identical and different types we used a dataset of 2,000 vehicle pairs annotated by a human. Volunteers were instructed to recognize same and different types. To be labeled as a match, the pair of vehicles had to match in make, model, submodel, and also model year. The difference of vehicles in different model years can be really small, almost unnoticeable as it can be for example only a small change in shape of lights. One pair was annotated by multiple humans and we collected more than 9 annotations for each pair.

The top graph in Figure 9 shows normalized responses for vehicle pairs while 1.0 denotes that all annotators agreed that the pair contains vehicles of the same type. Value -1.0 denotes that all annotators considered the vehicles' types as

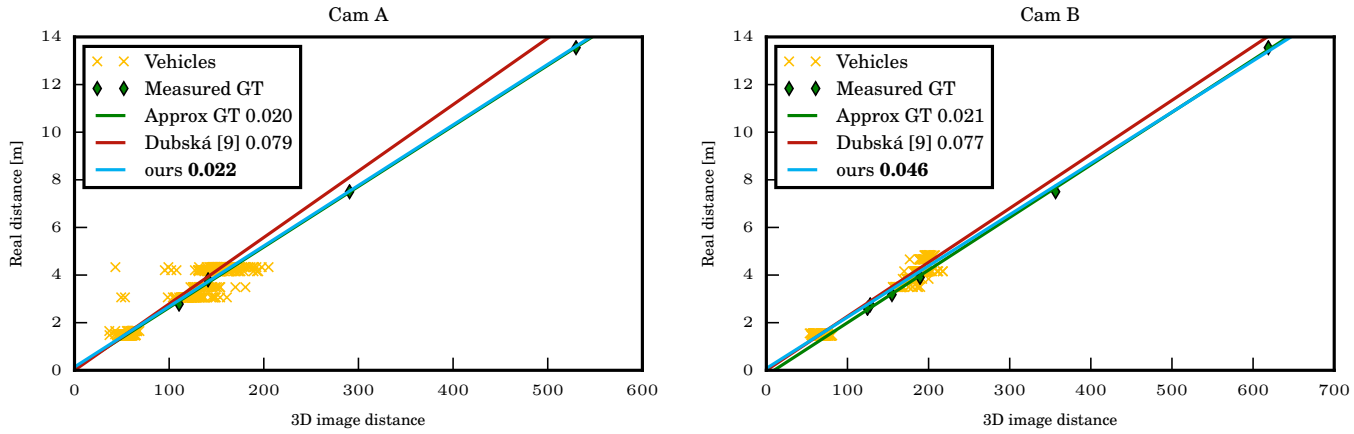

Fig. 12. Comparison of our camera scale calibration with the method proposed by Dubská et al. [19]. The green line represents the calibration obtained by linear least squares optimization for the measured distances in the scene (green diamonds). The numbers in the legend are mean relative errors. See Table I for details. The contributing vehicles are marked by yellow crosses, which are numerous (588 cam A, 542 cam B) and highly concentrated around the drawn lines representing the calibration.

	Ground Truth		**Dubská et al. [19]**		**ours**	
cam	mean	worst	mean	worst	mean	worst
A	0.020	0.043	0.079	0.108	**0.022**	**0.059**
B	0.021	0.047	0.077	0.104	**0.046**	**0.073**

TABLE I. EVALUATION OF SCALE CALIBRATION FOR DIFFERENT CAMERAS (A, B). WE MEASURED DISTANCES ON THE ROAD PLANE AND EVALUATED RELATIVE ERRORS OF THE DISTANCES OBTAINED BY THE SCALE CALIBRATION METHOD AND THE REAL DISTANCES. THE ERROR FOR ONE MEASURED DISTANCE l AND ITS GROUND TRUTH l_{gt} IS COMPUTED AS $|l - l_{gt}|/l_{gt}$.

different. We generated groups with different thresholds and evaluated our system by comparing its responses with the human annotated ground truth. The bottom graph in Figure 9 shows the Precision-Recall curve with average precision 0.890 while the baseline based on the fraction of positive samples is 0.449.

C. Fine-Grained Classification of Previously Unseen Vehicles into Groups

We used the obtained clusters for training fine-grained recognition of vehicles based on LBP and SVM and processed new vehicles which were not grouped at all and were not shown to the grouping algorithm. We evaluated the groups in the same way as in the Section III-A and the results are shown in Figure 10. Quantitative evaluation of the classification to the fine-grained vehicles' types shows that the mean false positive rate corresponds to the mean false positive rate of the groups which were used for training of the classifiers.

Qualitative evaluation of the vehicles is shown in Figure 11 and as the images show, our method is robust to color and markers variations, weather and lighting changes and partial occlusions.

D. Scene Scale Inference Precision and Speed Measurement

We compared our scale calibration method for traffic surveillance with a state-of-the art method by Dubská et al. [19]. We measured distances manually on the road in two different scenes, computed the scale calibration by mean of vehicles [19], and compared the result to our scale calibration. We adopted the evaluation protocol proposed by Dubská et al. [19] and used it to compare the methods. The results are displayed in Table I and as the table shows, our method outperforms the state-of-the art method for traffic surveillance scale calibration.

The method proposed by Dubská et al. [19] requires manual input in the form of mean vehicle dimensions and our method requires manual annotation in the form of few make&model names for the largest groups of vehicles. The method proposed by Dubská et al. [19] achieves worse results than in their original paper mainly because of our low-cost IP camera with high radial distortion, see Figure 2. Therefore, also the ground truth is imperfect and the mean error of linear least squares approximated ground truth is approximately 0.02. The accuracy of our calibration almost reaches the accuracy of the ground-truth measurements. Despite the camera's low quality, the scale measurement (directly leading to speed and size measurements) reaches usable numbers (2.2 %, 4.6 % mean accuracies, about twice as good as [19]), thanks to calibrating from multiple different but coherent clusters of vehicles automatically clustered. We also included graphs how our scale calibration methods work for different lengths in the image and it is shown in Figure 12.

IV. CONCLUSIONS AND FUTURE WORK

This paper shows algorithms forming a system for grouping vehicles based on their appearance – grouping vehicles of identical make&model (to a large extent also submodel and model year). The proposed method is capable of grouping vehicles despite their being texture-less and it is resistant to paintings on the cars and small modification (such as small rooftop equipment). We are showing experimentally that the groups can be successfully used for training classifiers capable of detecting the vehicles belonging to the same group reliably. Such detected cars can be used for inferring camera scale reliably (outperforming best known existing method twice in accuracy). We are making the collected dataset (472 k vehicle samples, 3 images each, 13 different lanes of traffic,

2,000 hand-annotated vehicle pairs) available together with the source codes.

We identify one important question open by our work: How to automatically match the clusters obtained from different cameras. Having three rather different views for every vehicle sample (Figure 2) should be enough to roughly capture the 3D shape of the car and we are experimenting with local descriptors not relying on object texture (which is hardly present on vehicles) [40] to form a model matchable between similar (but not identical) camera views. That way, user annotation and other inferred knowledge can be transferred from view to view transitively, leading to practically unlimited coverage of viewpoints.

REFERENCES

[1] V. Petrovic and T. F. Cootes, "Analysis of features for rigid structure vehicle type recognition," in *BMVC*, 2004, pp. 587–596.

[2] L. Dlagnekov and S. Belongie, "Recognizing cars," UCSD CSE Tech Report CS2005-0833, Tech. Rep., 2005.

[3] G. Pearce and N. Pears, "Automatic make and model recognition from frontal images of cars," in *IEEE AVSS*, 2011, pp. 373–378.

[4] P. Negri, X. Clady, M. Milgram, and R. Poulenard, "An oriented-contour point based voting algorithm for vehicle type classification," in *ICPR*, 2006, pp. 574–577.

[5] B. Zhang, "Classification and identification of vehicle type and make by cortex-like image descriptor HMAX," *IJCVR*, vol. 4, pp. 195–211, 2014.

[6] E. Hsiao, S. Sinha, K. Ramnath, S. Baker, L. Zitnick, and R. Szeliski, "Car make and model recognition using 3D curve alignment," in *IEEE WACV*, March 2014.

[7] M. Stark, J. Krause, B. Pepik, D. Meger, J. Little, B. Schiele, and D. Koller, "Fine-grained categorization for 3D scene understanding," in *BMVC*, 2012.

[8] J. Krause, J. Deng, M. Stark, and L. Fei-Fei, "Collecting a large-scale dataset of fine-grained cars," in *FGVC*^{2}, 2013.

[9] Y.-L. Lin, V. I. Morariu, W. Hsu, and L. S. Davis, "Jointly optimizing 3D model fitting and fine-grained classification," in *ECCV*, 2014.

[10] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *ICCV Workshop 3dRR-13*, 2013.

[11] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *CVPR*, June 2010, pp. 3360–3367.

[12] J. Deng, J. Krause, and L. Fei-Fei, "Fine-grained crowdsourcing for fine-grained recognition," in *CVPR*, June 2013, pp. 580–587.

[13] J. Prokaj and G. Medioni, "3-D model based vehicle recognition," in *IEEE WACV*, Dec 2009.

[14] L. Grammatikopoulos, G. Karras, and E. Petsa, "Automatic estimation of vehicle speed from uncalibrated video sequences," in *Proceedings of International Symposium on Modern Technologies, Education and Profeesional Practice in Geodesy and Related Fields*, 2005, pp. 332–338.

[15] K. Wang, H. Huang, Y. Li, and F.-Y. Wang, "Research on lane-marking line based camera calibration," in *International Conference on Vehicular Electronics and Safety, ICVES*, 2007.

[16] F. Cathey and D. Dailey, "A novel technique to dynamically measure vehicle speed using uncalibrated roadway cameras," in *Intelligent Vehicles Symposium*, 2005, pp. 777–782.

[17] T.-W. Pai, W.-J. Juang, and L.-J. Wang, "An adaptive windowing prediction algorithm for vehicle speed estimation," in *IEEE Intelligent Transportation Systems*, 2001.

[18] T. H. Thi, S. Lu, and J. Zhang, "Self-calibration of traffic surveillance camera using motion tracking," in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems*, 2008.

[19] M. Dubská, J. Sochor, and A. Herout, "Automatic camera calibration for traffic understanding," in *BMVC*, 2014.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *IJCV*, 2015.

[21] S. Agarwal, A. Awan, , and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE PAMI*, vol. 26, no. 11, pp. 1475–1490, Nov. 2004.

[22] C. Papageorgiou and T. Poggio, "A trainable object detection system: Car detection in static images," Tech. Rep. 1673, October 1999, (CBCL Memo 180).

[23] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[24] Y. Xiang and S. Savarese, "Estimating the aspect layout of object categories," in *CVPR*, 2012, pp. 3410–3417.

[25] C. Caraffi, T. Vojir, J. Trefny, J. Sochman, and J. Matas, "A System for Real-time Detection and Tracking of Vehicles from a Single Car-mounted Camera," in *ITS Conference*, Sep. 2012, pp. 975–982.

[26] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer, "Generic object recognition with boosting," EMT, TU Graz, Austria, Tech. Rep. TR-EMT-2004-01, 2004, submitted to the IEEE Tr. PAMI.

[27] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool, "Dynamic 3D scene analysis from a moving vehicle," in *CVPR*, June 2007, pp. 1–8.

[28] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich, "Viewpoint-aware object detection and continuous pose estimation," *Image&Vision Comp.*, 2012.

[29] S. Savarese and L. Fei-Fei, "3D generic object categorization, localization and pose estimation," in *ICCV*. IEEE, 2007.

[30] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *CVPR*, 2012.

[31] M. Özuysal, V. Lepetit, and P. Fua, "Pose estimation for category specific multiview object localization," in *IEEE CVPR*, June 2009, pp. 778–785.

[32] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *CVPR*, vol. 2, 1999, pp. 246–252.

[33] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *ICPR*, 2004, pp. 28–31.

[34] J. Canny, "A computational approach to edge detection," *IEEE PAMI*, vol. PAMI-8, no. 6, pp. 679–698, Nov 1986.

[35] P. Dollár and C. Zitnick, "Structured forests for fast edge detection," in *IEEE ICCV*, Dec 2013, pp. 1841–1848.

[36] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP Variants on Real-World Data Sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, Feb. 2013.

[37] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, and R. Chellappa, "Fast directional chamfer matching," in *CVPR*, June 2010, pp. 1696–1703.

[38] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Tr. Information Theory*, vol. 8, no. 2, pp. 179–187, February 1962.

[39] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Tr. PAMI*, vol. 24, no. 7, pp. 971–987, Jul 2002.

[40] D. Damen, P. Bunnun, A. Calway, and W. Mayol-Cuevas, "Real-time learning and detection of 3D texture-less objects: A scalable approach," in *BMVC*, 2012.