Traffic Scene Classification on a Representation Budget

Ivan Sikirić, Karla Brkić, Petra Bevandić, Ivan Krešo, Josip Krapac, and Siniša Šegvić, *Member, IEEE*

***Abstract*— Visual cues can be used alongside GPS positioning and digital maps to improve understanding of vehicle environment in fleet management systems. Such systems are limited both in terms of bandwidth and storage space, so minimizing the size of transmitted and stored visual data is a priority. In this paper, we present efficient strategies for computing very short image representations suitable for classifying various types of traffic scenes in fleet management systems. We anticipate that the set of interesting classes will change over time, so we consider image representations that can be trained without knowing the labels of the target dataset. We empirically evaluate and compare the presented methods on a contributed dataset of 11447 labeled traffic scenes. Our results indicate that excellent classification results can be achieved with very short image representations, and that fine-tuning on the target dataset image data is not mandatory. Image descriptors can be as short as 128 components while still offering good performance, even in presence of adverse weather or illumination conditions.**

***Index Terms*— Computer vision, intelligent vehicles, image classification.**

I. Introduction

In most vision-based systems for scene recognition, size of image representation is of no concern [1]. Very compact image representations are primarily utilized in image retrieval systems [2], but they are not designed to be used for image classification. However, limiting image representation size is critical in systems where images are acquired by thin clients with low processing power and limited bandwidths, as is often the case in intelligent transportation. Many of these systems assume a centralized server which retrieves and stores visual information from a number of thin clients at regular time intervals, as illustrated in Figure 1. In these systems, the number of clients can be very large, while the record keeping time can be very long (several months, or even years). Thus, transferring raw image data from the clients to the server may be prohibitively expensive both in terms of storage and in terms of data transfer. The latter is especially relevant when thin clients communicate via a paid mobile network, as is often the case in fleet management systems.

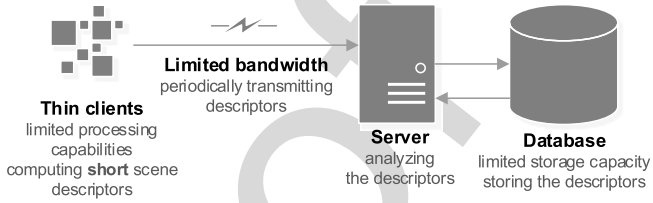

Fig. 1. The proposed target application framework: a number of thin clients send visual information about their surroundings to a server via limited bandwidth. Figure reproduced from [3].

Our primary motivation are fleet management systems, where a server tracks the locations of a fleet of vehicles in real time. In most fleet management systems, vehicles are equipped with a GPS sensor and a range of simple supplementary sensors (e.g. ignition sensor, fuel gauge, thermometer). We additionally propose equipping each vehicle with a low cost dashboard camera. The addition of a camera, while not the industry standard in fleet management, would enable the recognition of the type of traffic environment the vehicle is currently in. The traffic environment classification could also be achieved using other sensors, such as radar and lidar [4]. However, in this paper we focus exclusively on visual data for traffic scene classification.

There are several ways to make use of traffic scene classification in fleet management systems. One is to aid route reconstruction algorithms, by differentiating between various types of roads. Another is to detect scenarios in which a degradation of GPS accuracy is expected, such as driving through tunnels, under overpasses, toll booths, gas stations, etc. Finally, many companies have their own set of characteristic locations and traffic scenarios which they would like to detect. These might be specific to their particular business needs, and difficult to guess ahead of time, which is a major obstacle to doing the image classification on the clients themselves. If the classification were to be performed on the client, then client software would have to be updated each time the set of classes is changed. Additionally, it would be impossible to re-classify the historical visual data stored on the server. If the clients simply compute the *image descriptors* and transmit them to the server, then changes in the business logic

Manuscript received October 16, 2017; revised June 5, 2018 and October 30, 2018; accepted December 14, 2018. This work was supported in part by the Croatian Science Foundation under Grant I-2433-2014 and in part by the European Regional Development Fund (DATACROSS) under Grant KK.01.1.1.01.0009. The Associate Editor for this paper was Q. Ji. *(Corresponding author: Ivan Sikirić.)*

I. Sikirić is with Mireo d.d., 10000 Zagreb, Croatia (e-mail: ivan.sikiric@mireo.hr).

K. Brkić, P. Bevandić, I. Krešo, and S. Šegvić are with the Faculty of Electrical Engineering and Computing, University of Zagreb, 10000 Zagreb, Croatia (e-mail: karla.brkic@fer.hr; petra.bevandic@fer.hr; ivan.kreso@fer.hr; sinisa.segvic@fer.hr).

J. Krapac is with Mobius Labs GmbH, 10997 Berlin, Germany (e-mail: josip@mobius.ml).

Digital Object Identifier 10.1109/TITS.2019.2891995

are localized, cheaper and easier to maintain. Re-examination of historical visual data becomes possible, and the clients are significantly simplified.

In this paper we study image classification on a tight representation budget in the context of a thin client - central server scenario. Our focus is on *very short* image descriptors which stand a fair chance to perform well on classes which have not been seen during training. We consolidate best approaches from our previous study [5] and compare them to recent approaches based on deep convolutional architectures. The performance of all presented approaches is comprehensively evaluated on an extended and improved version of our image classification dataset for fleet management applications.

Section II overviews prior research related to this paper. Section III summarizes our approach and categorizes the methods we use for image classification. Section IV details the experimental setup, including the presentation of our dataset, and presents the results. The conclusions and outlook are given in Section V.

II. Related Work

Our work is related to three interconnected research areas: (i) image classification in general, (ii) traffic scene classification, and (iii) short image descriptors.

A. Image Classification in General

Active research in image classification rarely considers descriptor length as a research topic. Rather, the majority of efforts are focused on building reliable methods for classifying a wide range of image categories [6]–[8].

Before the advent of deep learning methods, most successfull approaches [9] were based on the seminal bag-of-visual words method [10]. Some extensions improve the performance by augmenting the representation with the spatial layout of visual words [11]. State of the art performance has been obtained with spatial Fisher vectors (SFV) [12] which aggregate locations of visual words by leveraging Fisher vectors with respect to a spatial generative model.

After the success of the AlexNet [13], deep learning methods started developing rapidly. Increasing the width and depth of the network further improved classification accuracy, as demonstrated by the architectures VGG-E [14] (19 layers) and Inception [15] (22 layers). The 100-layer barrier was surpassed by highway [16] and residual [17] (ResNet) architectures.

ResNet reduces the problem of vanishing gradient by introducing *shortcut connections* between every few stacked layers, thus forming residual building blocks. This architecture substantially improves the convergence of the optimization algorithm. Although this network can be over a hundred layers deep, further research showed that most gradient comes from shallow paths [18].

DenseNet [19] architecture uses dense building blocks in which outputs of every layer are connected to inputs of every subsequent layer in the block. The features are combined via concatenation. The dense connectivity enables feature reuse, simplifies the information flow between layers, requires less parameters per layer than previous approaches, and thus enables training of even greater number of layers.

Some recent approaches have used ImageNet classification as a proxy task for learning high-quality convolutional feature extractors [20]. Cimpoi *et al.* [21] obtain the FV-CNN descriptor by extracting features from deep convolutional layers and aggregating them with a Fisher vector framework. Their descriptor has achieved state-of-the-art performance in texture recognition and image classification. Garcia-Gasulla *et al.* [22] measure the differences of distributions of network features in different classes, showing that features of deeper layers are more specialized than features in lower levels. Nanni *et al.* [23] demonstrate that hand-crafted features and features learned in deep networks extract different information from input images, and that best results may be achieved if both types of features are used, and dimensionality reduced via PCA.

The work in this area of research was not limited to supervised learning approaches. Radford *et al.* [24] extend the ideas of Goodfellow *et al.* [25] and present an unsupervised learning framework called DCGAN which succeeds to learn image representations from unlabeled data. Arjovsky *et al.* [26] introduce the WGAN framework, an improvement of DCGAN which results in simplified learning process and avoiding the mode collapse problem.

B. Traffic Scene Classification

Use of machine vision in vehicles is on the rise [27], so the problem of traffic scene classification is receiving increased attention recently. Oeljeklaus *et al.* [28] present a deep neural network capable of simultaneous traffic scene recognition and segmentation. The network has less than six million parameters (not counting the FC layer), two separate output paths, and a shared encoder used by both tasks. Di *et al.* [29], [30] research understanding of traffic scenes from images taken from the same location, but under different weather or illumination conditions. Their approach consists of extracting fine-tuned CNN features and transferring the annotations from the retrieved best matching image based on cross-domain dense correspondences. Hussain *et al.* [31] demonstrate that adequate vehicle type classification can be done using CNN even if vehicle regions are as small as 90×90 pixels.

Traffic scene classification is not limited to visual data only. Hsu *et al.* [32] determine whether a vehicle is driving on roads or viaducts by applying a dynamic Bayesian network on map data, satellite visibility data and position calculated by GPS. Seeger *et al.* [4] demonstrate that four types of traffic scenes can be successfully classified by relying on occupancy grid data. Additionally, they find that using the hand-crafted features with an SVM classifier shows results on par with those achieved by deep learning networks such as VGG16.

C. Short Image Descriptors

Most methods for producing short image descriptors are motivated by reducing memory consumption in large-scale image retrieval. They also aim to speed up the retrieval

process, which is often done via an approximate nearest neighbor search [2]. A common approach is to produce a very short similarity-preserving image hash, such as DSH introduced by Liu *et al.* [33]. They use supervised information of image pair similarity to train a CNN structure to produce a small number of discrete values on output, ranging in size from 12 to 48 bits. Improved results are achieved by approaches that take different image modalities into consideration, such as CSDH [34] and DCH [35].

An alternative to crafting very short descriptors is starting with long descriptors and applying dimensionality reduction methods such as principal component analysis (PCA) [36] or product quantization (PQ) [37], [38]. Both approaches have a potential to detect parts of image representation which are not activated for a particular target dataset [22]. However, this requires training on images from the target dataset, which can be regarded as both an advantage and a disadvantage, as we will discuss in the next section.

In our own work [3], [5], we investigate how limiting the size of state-of-the-art image descriptors impacts the performance of traffic scene classification. We propose a short image descriptor that combines compacted spatial Fisher vectors and GIST descriptor in a lossy encoding scheme. Classification performance is retained for the descriptor size as low as 48 bytes per image. The main shortcoming of this method is inadequate performance on some classes of traffic scenes (e.g. dense traffic). This became more apparent as we expanded our dataset to improve variability of under-represented classes.

To the best of our knowledge, no previous work proposes nor evaluates very short image descriptors suitable for classification of traffic scenes, which is the main motivation for this paper.

III. Our Approach

We consider short image descriptors (less than 2000 components) which are suitable for multi-label classification. We are motivated by the application scenario where many mobile agents acquire images and transmit image representation over a tight data-channel to a remote server which classifies the representation into an open set of attributes. We explore three different approaches to training of descriptors: (i) unsupervised training on target imagery (addressed in III-A), (ii) supervised transfer learning (addressed in III-B) and (iii) unsupervised transfer learning (addressed in III-C). Descriptors in category (i) are trained on images of traffic scenes, but are unaware of their class labels. Descriptors in category (ii) are trained on an unrelated dataset of labeled images, and then applied to traffic images without any additional training. Descriptors in category (iii) are trained on unrelated dataset of images without class labels, and as such are fully unsupervised. Note that none of the three descriptor categories uses the class labels of traffic scene images during training, which ensures that the set of traffic scene classes remains open. Additionally, the descriptor categories (ii) and (iii) are not trained on the traffic scene images, which means they are equally likely to perform well on different geographical locations. The descriptors from category (i) might potentially perform better on the geographical location on which they were trained, which would be undesirable. In this paper, all descriptors in categories (ii) and (iii) are trained on the ILSVRC subset of ImageNet [7].

A. Descriptors Based on Unsupervised Training on Target Imagery

Both descriptors in this category are based on the traditional bag-of-words framework [10]. The two descriptors use different local features (hand-crafted vs convolutional) and the common aggregation mechanism. The aggregation is performed with spatial Fisher vector embeddings [12], which encode both the appearance and the spatial layout through the use of Fisher vectors. Both approaches require access to target images, since they rely on a visual vocabulary of local features found in the dataset.

The first descriptor is a concatenation of GIST scene descriptor [39], [40] and spatial Fisher vectors (SFV) [12] embeddings of SIFT [41] local feature descriptors. The GIST scene descriptor is completely handcrafted and has no training requirements whatsoever. By itself, it performs poorly on our dataset. However, concatenating it to the SFV descriptor improves the performance at very low descriptor lengths. Short image representations can be obtained through the choice of hyper-parameters (such as number of appearance or spatial components), or by producing a large representation first, and subsequently reducing it with PCA. We call this concatenation a SIFT/SFV+GIST descriptor. Of all the descriptors considered in this paper, this is the only one that does not make use of ImageNet dataset during training.

Next, we replace all hand-crafted features with features obtained with end-to-end learning. More precisely, we drop the GIST part and we replace the SIFT features in spatial Fisher vector framework with responses of a deep convolutional neural network. We use responses of `conv5_4` layer of VGG-19 [14] convolutional neural network, trained on the ImageNet dataset. We do not fine-tune the network on our dataset, but the spatial Fisher vector framework does build a visual vocabulary on features found in target images. We refer to the resulting descriptor as VGG/SFV.

B. Descriptors Based on Supervised Transfer Learning

Approaches presented in III-A are based on the VGG architecture which aggregates convolutional features with two fully connected layers (`fc5`, `fc6`). However, these two layers require over 100 million parameters which makes them very prone to overfitting. Hence, it did not come as a surprise when later research showed that `conv5` features aggregated with Fisher vectors outperform `fc6` features in various visual tasks [21]. Recent convolutional architectures [17], [19] replace fully connected layers with simple global average pooling. As a consequence, they succeed to overperform VGG despite using much less parameters.

In this category, we consider pooled features extracted by two recent architectures: ResNet [17] and DenseNet [19]. As in the case of VGG, we use public parameterization trained

(a) (b) (c) (d)

Fig. 2. Examples of visually degraded images from the FM3a set. (a) Falling rain. (b) Falling snow. (c) Fog. (d) Low sun.

on ImageNet. The resulting features are a good match for our task since their dimensionality is rather low while offering a high classification potential due to state-of-the-art ImageNet performance.

The performance of these features might further increase by performing the aggregation with Fisher vectors since that would present opportunities to leverage spatial layout and/or the knowledge of what is unusual in the target dataset. Nevertheless, we refrain from doing that in order to avoid the descriptors having any knowledge of the target dataset. Consequently, approaches from this category stand a very good chance to generalize well to any geographical region in the world.

C. Fully Unsupervised Descriptors

In this category we consider a convolutional generative adversarial network [24]. More precisely, we work with convolutional features extracted by the discriminator, whose original task is to detect whether the input image is real or crafted by the generator. In order to perform the classification task, the discriminator network has to learn a sophisticated image representation which we leverage to classify the scene. However, the highest levels of this representation have a much lower semantical quality than in the case of strongly supervised approaches which output a distribution over 1000 classes. We therefore form the image descriptor by concatenating max-pooling evidence from all 6 convolutional layers and regulate its size by adjusting the max-pooling grid.

As in section III-B, we use a public parameterization learned on ImageNet [24]. Unlike previously presented supervised models, here the training proceeds without knowing the labels of the training images. In theory, this removes any limitations due to limited availability of labeled images, and provides opportunity to train on billions of web-scraped images. Unfortunately, as we shall see in the experiments, practical performance of fully unsupervised approaches still lags behind the approaches presented in sections III-A and III-B.

IV. Experiments and Results

Our experiments evaluate classification performance of the presented image descriptors, with emphasis on very short image representations. The experiments are performed on the novel FM3 dataset[1] which contains labeled traffic scenes of interest for fleet management systems (cf. subsection IV-A). We partition our dataset into fixed train, validation and test subsets, and employ the common classification setup throughout all experiments (cf. subsection IV-B). We consider approaches from the three descriptor categories presented in Section III and evaluate their performance on our dataset (cf. subsection IV-C-IV-E). We explore the dependency of classification performance on the descriptor length (cf. subsection IV-F). We briefly discuss the results (cf. subsection IV-G), and then proceed to detailed analysis of one of the more successful approaches (cf. subsection IV-H).

A. The FM3 Dataset

In this work we contribute an improved version of the Fleet Management dataset presented in [5]. The dataset contains 11448 images of various traffic scenes in Croatia, as seen from the perspective of a driver. All images are of resolution 640×480, and all were taken during day (from late morning to dusk). The dataset is split into two separate subsets: the main part, dubbed FM3m, which contains 6413 images with good visibility and illumination conditions, and the appendix, dubbed FM3a, which contains 5035 images with various types of visual degradation. Visual degradation in the FM3a set was caused by bad weather conditions (falling rain, falling snow, fog) and/or by very low sun angles (bad illumination due to camera pointing towards sun, and/or low amount of sunlight). Windshield wipers obstruct the scene in 785 of images. Examples of visually degraded images from this set are shown in Figure 2.

Most images were captured in the same fashion in which they would be captured in a fleet management system: by an in-car camera, at regular intervals while driving on Croatian roads. Since some interesting types of traffic scenes occur very rarely, we have increased the number of samples for such scenes in the FM3m set by downloading images from Mapillary.com. We have hand-picked a batch of images for each rarely occurring class and subsequently manually centered and cropped them to the appropriate point of view, as well as resized them to the resolution of 640×480 pixels.

As was discussed in the Introduction, it is not possible to anticipate every type of traffic scene that might be of interest in a fleet management system. The eight classes of FM3 cover a variety of plausible use cases. Class *highway* represents a fast, wide and open road. It must have either two lanes and a shoulder lane, or at least three lanes per direction. This class can be used to help with GPS matching disambiguation, and to verify that fast vehicle traffic is allowed and expected. The class *road* is similar to class *highway*, and represents an open non-highway road outside of settlement. It too can be

[1] http://www.zemris.fer.hr/~ssegvic/datasets/unizg-fer-fm3am.tar.gz

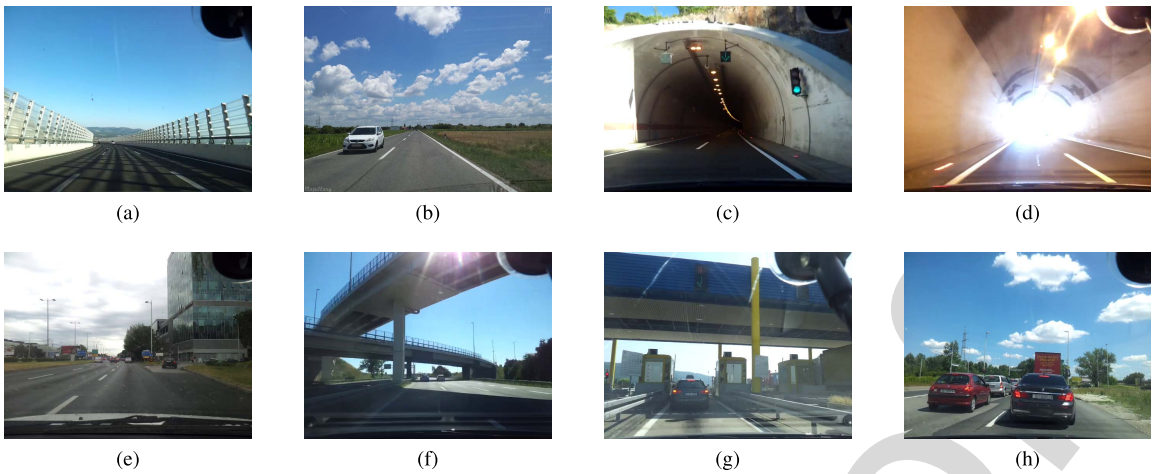

Fig. 3. Examples of classes from the FM3 dataset. (a) Highway. (b) Road. (c) Tunnel. (d) Exit. (e) Settlement. (f) Overpass. (g) Booth. (h) Traffic.

used to augment GPS matching. The class *tunnel* represents the entrance to, or inside of a tunnel (but not the very exit). It signals that the loss of GPS signal is expected. The class *exit* represents the exit of a tunnel, which signals that the GPS signal is expected to return soon. A traffic scene image is labeled as *settlement* if there is visible evidence of vehicle being inside a settlement (e.g. buildings, playgrounds, etc.). This may be interesting for a variety of reasons. For example, GPS precision is often low in settlements. The speed limit is likely to be lower and frequent vehicle stops are expected as well. The class *overpass* signals that the vehicle is directly in front of, or under an overpass. This may help to explain observed loss of GPS precision and to predict that a complete loss of GPS signal is unlikely. Class *booth* represents a scene directly in front of, or at a toll booth. It also signals a possible loss of GPS precision, and explains the reason of vehicle stopping. This may also be useful for keeping track of travel expenses. Even though most toll booths are present in the map data, the system might easily miss them without the use of visual cues due to temporary loss of GPS precision. Finally the class *traffic* represents the scenes of very dense traffic, or a major occlusion of a scene by a large vehicle. In either case, it explains the low speed and stopping of a vehicle, and as such might be interesting to e.g. delivery companies. Example instances of each class are shown in Figure 3.

All images were labeled according to these class descriptions by a single annotator, without the use of visual pre-processing of any kind. Some images have multiple labels assigned. We define classes *highway*, *road*, *tunnel*, *exit*, *settlement* and *booth* to be mutually exclusive, as they represent a location, but classes *overpass* and *traffic* represent an attribute of a scene, and can co-occur with other classes. In some countries it might be logical to allow simultaneous assignment of labels *settlement* and *highway*, to detect highways going through large cities, but we had no use of such a scheme in our dataset. The distribution of classes across the FM3m and FM3a sets, and across types of visual degradation is shown in Table I. Note that classes *tunnel*, *exit* and *booth* have very few samples in the FM3a set. This is because rain and snow do not occur inside tunnels (only on entry and exit), and because class *booth* is a rarely occurring event.

TABLE I
THE DISTRIBUTION OF CLASS LABELS IN THE FM3M AND FM3A DATASETS

class	FM3m total	FM3a total	FM3a (low sun)	FM3a (rain)	FM3a (snow)	FM3a (fog)
highway	4646	1813	1375	312	12	114
road	390	630	79	401	150	0
tunnel	616	5	2	3	0	0
exit	73	6	4	1	0	1
settlem.	583	2572	525	1707	352	0
overpass	152	90	42	32	14	2
booth	101	5	5	0	0	0
traffic	132	351	56	257	38	0
total	6413	5035	1990	2428	514	115

B. The Classification Setup

In all experiments classification was performed using an SVM classifier, as it was shown to be a good choice in [3], [21], and [42]. We used LibSVM implementation [43], which was trained in a one-vs-rest fashion. For each of 8 binary classifiers, the average precision (AP) is calculated, and mean average precision (mAP) of all 8 classes was used as the measure of classification performance. For each class, 25% of instances were used for training, and another 25% were used for validating the optimal SVM parameters. To counter the bias in the dataset, we weighted the SVM scores based on the inverse proportion of class samples in the training set. Finally, the classifier was trained using these parameters on union of training and validation subsets, and the performance was evaluated on the remaining 50% instances. We experimented with linear and RBF kernel, and in all cases RBF performed better. We do not consider that as a surprise since our image representations are rather small.

The FM3a set only contains enough samples for five out of eight classes of interest, and is poorly balanced across types of visual degradation. For that reason, in most experiments we only train and evaluate the methods on images from the FM3m

TABLE II
AVERAGE PRECISION (%) OF SIFT/SFV+GIST DESCRIPTOR ON FM3M DATASET (SVM WITH RBF KERNEL)

	descriptor length						
class	1024	512	256	128	64	32	16
highway	99.73	99.81	99.83	99.85	99.81	99.7	99.34
road	92.10	92.73	92.51	92.02	90.92	87.77	83.00
tunnel	99.92	99.90	99.84	99.65	99.60	99.54	99.83
exit	95.58	95.65	95.04	95.13	92.81	92.09	91.67
settlem.	96.10	96.64	96.38	96.44	95.79	94.24	92.39
overpass	96.49	95.99	96.04	94.70	92.48	86.72	87.05
booth	86.43	87.04	85.76	86.51	84.69	78.56	69.67
traffic	82.69	82.42	82.55	81.56	77.53	75.41	63.15
mean	93.63	93.77	93.49	93.23	91.70	89.25	85.76

TABLE III
AVERAGE PRECISION (%) OF VGG/SFV DESCRIPTOR ON FM3M DATASET (SVM WITH RBF KERNEL)

	descriptor length						
class	1024	512	256	128	64	32	16
highway	99.78	99.76	99.75	99.75	99.69	99.6	99.52
road	91.99	91.30	91.44	90.73	89.28	87.31	85.96
tunnel	99.93	99.94	99.95	99.95	99.96	99.92	99.91
exit	96.74	96.85	97.00	96.41	96.46	95.44	94.17
settlem.	97.49	97.46	97.48	97.38	96.87	96.31	95.28
overpass	97.55	96.95	96.11	95.96	96.49	96.03	90.22
booth	99.58	99.31	98.68	99.17	98.60	97.10	89.77
traffic	85.03	84.63	85.60	84.47	82.87	77.97	74.22
mean	96.01	95.78	95.75	95.48	95.03	93.71	91.13

set. We use the FM3a set to test resilience of our approach to visual degradation of images in subsection IV-H.

C. Descriptors With Knowledge of FM3 Image Data

In this category we evaluate descriptors presented in III-A. The SIFT/SFV+GIST descriptor is a concatenation of SIFT/SFV and GIST descriptors. The dimensionality of dense SIFT features d was projected down from $d = 128$ to $d = 80$ via PCA. In the SFV framework we used $K = 16$ appearance components and $C = 1$ spatial component, which produces a vector of length $K(1 + 2d + 5) = 2656$. For GIST we used the implementation provided by Oliva and Torralba [40] without any modifications, which produces a vector of length 512. The final size of the descriptor is thus $2656 + 512 = 3168$. We have used PCA to further compress the descriptor to lengths 1024, 512 and every other power of two, down to 16. The compressed descriptors were classified with an SVM classifier with the RBF kernel, and the results are shown in Table II.

The VGG/SFV descriptor uses convolutional features instead of handcrafted SIFT features. More precisely, it uses the responses of `conv5_4` layer of the VGG-19 [14] convolutional neural network which was trained on the ImageNet dataset. The resolution of our dataset is 640×480, so the shape of local features is $30 \times 40 \times 512$, i.e. 1200 features of size $d = 512$. We did not use PCA to reduce dimensionality of these features. As in SIFT/SFV+GIST, we used $K = 16$ appearance components and $C = 1$ spatial component, thus obtaining a vector of length 16480, which was then reduced via PCA to lengths from 1024 to 16. The compressed descriptors were classified with an SVM classifier with the RBF kernel, and the results are shown in Table III.

Even though the hand-crafted SIFT/SFV+GIST approach achieves decent results, the deep learning based approach VGG/SFV is clearly better. With only 64 components it outperforms the hand-crafted descriptor of length 1024 (95.9% vs 94.94%).

D. Descriptors Trained on ImageNet With Class Labels

In this category we evaluate descriptors presented in III-B. We extract activations from the second-to-last layers obtained with publicly available parameterizations of ResNet-50[2] [17] and DenseNet-121[3] [19] architectures. Therefore, both models were trained on labeled ImageNet data, and were not fine-tuned on FM3 dataset. No cropping or resizing of the input images was done. Both feature extractors are fully convolutional and applicable to input images of any resolution. Spatial dimensions of all activation tensors are automatically adjusted to the resolution of the input image. As the second-to-last layer in both networks performs global pooling, the two resulting descriptors have length equal to the number of feature maps in that layer, which is 2048 for ResNet-50, and 1024 for DenseNet-121.

TABLE IV
AVERAGE PRECISION (%) OF RESNET-50 AND DENSENET-121 DESCRIPTORS ON FM3M DATASET (SVM WITH RBF KERNEL)

	ResNet-50 (length 2048)		DenseNet-121 (length 1024)	
class	full FM3m	balanced	full FM3m	balanced
highway	99.84	86.73	93.68	93.60
road	91.13	94.29	99.97	92.76
tunnel	99.94	99.02	97.96	99.41
exit	97.70	98.64	98.33	99.14
settlement	98.33	89.46	97.86	87.58
overpass	97.15	97.54	98.81	98.63
booth	98.75	97.91	87.85	99.57
traffic	86.75	96.34	96.80	97.74
mean	96.20	94.99	96.41	96.05

The obtained classification results are shown in Table IV. We include results obtained on entire FM3m, and a balanced subset containing around 30 training and 30 test images per class. The results indicate that DenseNet-121 descriptor is slightly better, while also being shorter. We try to explain these results as follows. ResNet-50 has three times more parameters than DenseNet-121 (25 vs. 8 million) and performs slightly better on ImageNet. Due to less parameters DenseNet-121 is less prone to overfitting than ResNet-50, and therefore achieves better knowledge transfer from ImageNet towards FM3m. To verify this, we have fine-tuned both networks on FM3. Indeed the fine-tuned ResNet-50 achieved better performance than fine-tuned DenseNet-121 (98.0% vs 97.1% AP).

E. Descriptors Trained on ImageNet Without Class Labels

In this category we evaluate the descriptor presented in III-C by applying the public DCGAN parameterization to input images downsampled to 32×32 pixels [24]. We first extract

[2]https://github.com/KaimingHe/deep-residual-networks

[3]https://github.com/liuzhuang13/DenseNet

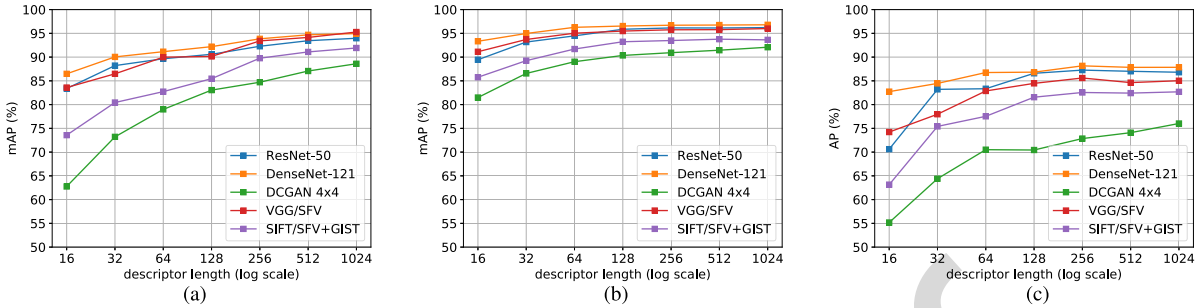

Fig. 4. Average precision (%) of selected descriptors on FM3m dataset with respect to representation budget. Particular representations are obtained via PCA as necessary. Mean AP values using linear SVM classifier are shown on the left. Mean AP values using SVM with RBF kernel are shown in the middle. AP values for the class *traffic* using SVM with RBF kernel are shown on the right. Best viewed in color. (a) Linear SVM, all classes. (b) SVM with RBF kernel, all classes. (c) SVM with RBF kernel, class traffic.

TABLE V
AVERAGE PRECISION (%) OF DESCRIPTORS BASED ON DCGAN DISCRIMINATOR ON FM3M DATASET (SVM CLASSIFIER)

trained on	ImageNet	ImageNet	ImageNet	untrained
kernel	linear	RBF	linear	linear
max-pool grid	4×4	1×1	1×1	1×1
length	28672	1792	1792	1792
highway	99.71	99.86	99.37	99.47±0.11
road	93.63	92.10	81.32	84.79±1.72
tunnel	99.07	99.33	99.48	99.20±0.11
exit	98.53	93.67	98.14	97.73±0.61
settlement	96.74	95.43	92.91	89.31±1.33
overpass	81.67	83.85	83.08	78.44±3.56
booth	88.59	86.53	86.24	71.42±3.23
traffic	78.53	66.47	65.28	62.92±3.29
mean	92.06	89.66	88.23	85.41±0.81

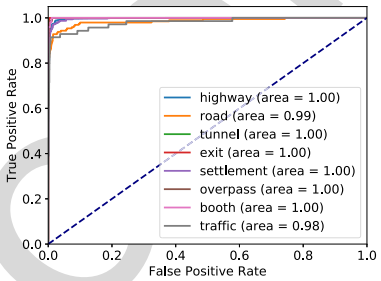

Fig. 5. ROC curves for the SVM classifier with RBF kernel trained on the DenseNet-121 descriptor of length 128. Measured on the FM3m dataset. Best viewed in color.

discriminator features by 4×4 max-pooling on each layer, and concatenating the results into a descriptor of length 28672. In the subsequent experiments, we shorten the descriptor by reducing the size of the max-pooling grid to 1×1. This results in descriptors of length 1792. We also include the result of untrained discriminator initialized with random weights. This was done to test how much knowledge transfer happens from ImageNet to FM3, and how much descriptiveness is due to network structure itself. We repeat this with 1000 different random initializations, and report the mean and the standard deviation. All these results are shown in Table V.

Much of the descriptiveness seems to come from the convolutional structure itself, since random-initialized discriminator achieves mAP of 85%. Some performance is lost when 4×4 max-pooling is replaced with 1×1 max-pooling, but 16-fold reduction of descriptor size makes this drop in performance an acceptable trade-off.

F. Impact of Descriptor Length

To measure the impact of descriptor length reduction on classification performance, we use PCA to compress the size of select descriptors to a series of different lengths. Starting from 1024, we consider all lengths that are a power of two, down to 16. The SIFT/SFV+GIST and VGG/SFV descriptors already use PCA as a part of their framework, and their full results are shown in Tables II and III. We now apply the PCA to ResNet-50, DenseNet-121 and DCGAN 4×4 descriptors as well. Since PCA is used on vectors calculated on FM3 dataset, all compressed descriptors fall into the category of methods with knowledge of FM3 image data. Note that full length of DenseNet-121 descriptor is 1024, so PCA was only used to produce its representations of length 512 and lower.

We classify the descriptors both with the linear SVM classifier and the SVM with RBF kernel. The summary of the results is best seen in graph form, in Figure 4 (a, b), which shows the mean average precision of all classes for each of the methods, with respect to the length of the representation, separated for linear and RBF kernels. The RBF kernel is strictly better here, but it is not as noticeable for large representations as it is for very small ones. We conclude that RBF is very advantageous when the representations are very small, although the linear SVM may prove the only applicable solution for large training datasets.

One thing to note is that reducing the size of the representation does not change the relative order of methods. The DenseNet-121 is a clear winner, followed by ResNet-50 and VGG/SFV, followed by SIFT/SFV+GIST (which relies on hand-crafted features) and finally the DCGAN 4×4 (which is trained in an unsupervised manner). Another thing to note is that reducing the size from 1024 down to 128 barely shows any changes in the mAP for most methods if RBF kernel is used. The drop is noticeable for lengths 64 and 32, and very noticeable for length 16.

We are not only interested in mean average precision. Ideally, every class in the dataset should be classified at acceptable levels of error. All descriptors show the worst performance on the *traffic* class, so the results for that class are shown in Figure 4 (c). Note that AP axis starts from 50% and that a considerable drop with respect to the mean performance can be observed for all representation budgets.

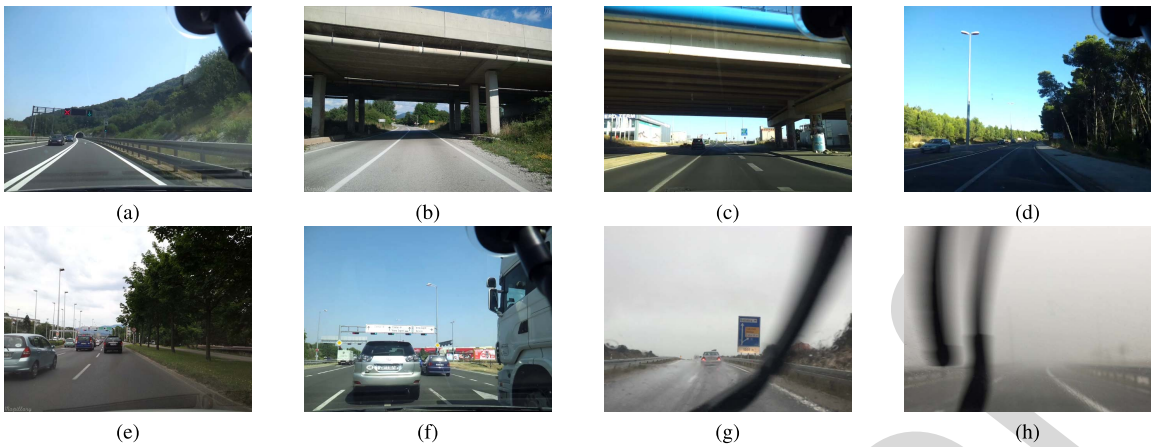

Fig. 6. Examples of mispredictions for the SVM classifier with RBF kernel on DenseNet-121 descriptors of length 128. Examples (a) to (f) are from the FM3m testing set, while examples (g) and (h) are from the FM3a set. In all examples the SVM was trained on the FM3m training set. (a) Road as highway. (b) Road as highway. (c) Settlement as highway. (d) Road as settlement. (e) Traffic false positive. (f) Traffic false negative. (g) Settlement as road (rain). (h) Highway as road (fog).

G. Discussion

Some classes in the FM3 dataset are clearly much easier to classify than others. In fact, the relative order of classes with respect to their achieved average precision is very similar across all presented methods. The best results are obtained on the *highway* class which is the most represented class in the dataset. The *tunnel* class is easy to classify, likely because tunnel scenes are almost always dominated by black or orange colors. The *road* class is most often confused with *highway* class. It is also less represented, and has greater variability of visual appearance. By far the hardest class, across all methods, seems to be the *traffic* class, which includes both the scenes of very dense traffic (which can happen in any kind of location), and also images with major occlusions of scene by other vehicles. One other thing to note is that the class *booth* is only hard to SIFT/SFV+GIST and DCGAN descriptors, while other methods have excellent performance on this class.

We can conclude that the best results are achieved by supervised deep convolutional models, even the ones that have no knowledge of the FM3 dataset. In fact, the best performing method is DenseNet-121, with descriptor length of 1024 and no knowledge of the target dataset. Fine-tuning these approaches to the FM3 dataset would further improve those results, but we avoid that in order to prevent overfitting.

DCGAN is at a severe disadvantage compared to supervised approaches. However, it could be improved by training on images acquired world-wide. Such training would not depend on the class labels, which is very desirable in our proposed framework.

H. Analysis of the DenseNet-121 Descriptor

We now analyze one of our best performing short descriptors: the DenseNet-121 descriptor of length 128. It achieves classification performance almost equal to that of full-length descriptor while being much shorter. We consider it as a strong candidate for our proposed fleet management setup. Its confusion matrix is shown in Table VI. Note that we only include the six classes that are defined as mutually exclusive.

TABLE VI
CONFUSION MATRIX FOR DENSENET-121 DESCRIPTOR OF LENGTH 128, SVM CLASSIFIER WITH RBF KERNEL

	predicted class					
actual class	highway	road	tunnel	exit	settl.	booth
highway	2298	2	0	0	6	0
road	28	153	0	0	5	0
tunnel	0	0	302	3	0	0
exit	2	0	1	37	0	0
settlement	16	4	0	0	276	0
booth	1	0	0	0	0	44

The ROC curves for all classes are shown in Figure 5. Some examples of mispredicted classes are shown in Figure 6.

I. Resilience of DenseNet-121 to Visual Degradation

We now use the FM3a set of images to verify the resilience of DenseNet-121 descriptor of length 128 to various types of visual degradation. First we evaluate how the SVM classifier trained on the FM3m training set performs on the FM3a set of degraded images. Next, we add some of images from the FM3a to the training set, and test the performance on the rest of images from FM3a. We do this in two steps, first adding only 10% of images from FM3a, then adding 25% of images to the training set. The average precision results are shown in Table VII. Note that we only list the results for the five classes that are adequately represented in FM3a, excluding classes *tunnel*, *exit* and *booth*. Even without any training on the degraded images, the system was able to classify the degraded images surprisingly well, with the exception of class *road*, which was most commonly confused with classes *settlement* (93 times) and *highway* (82 times). By adding some degraded images to the training set, the quality of classification rises quickly up to 94.46%, which is very close to mAP value for the same classes on non-degraded images: 95.22%. The ROC curves for the five evaluated classes are shown in Figure 7.

V. CONCLUSION AND OUTLOOK

In this work we have considered image classification of traffic scenes under requirements specific to fleet management applications, namely i) the bandwidth should be used sparingly

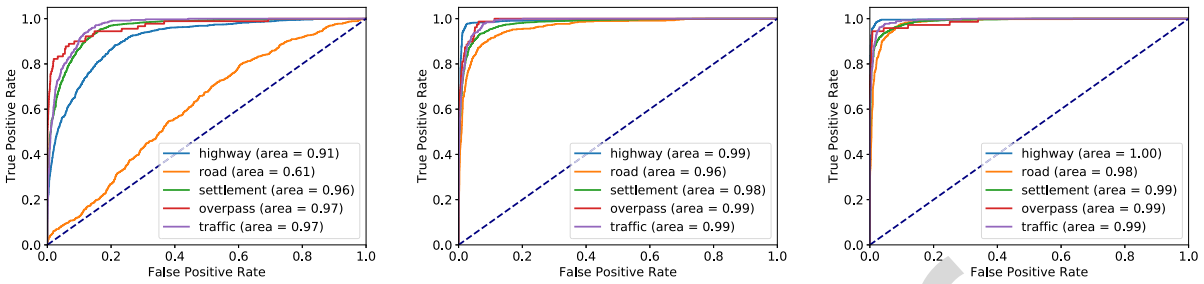

Fig. 7. ROC curves for the SVM classifier with RBF kernel trained on the DenseNet-121 descriptor of length 128. Training data contains 50% of FM3m samples, and incrementally increasing amounts of FM3a samples. Tested on the rest of FM3a images. Best viewed in color. (a) Training set contains no FM3a samples. (b) Training set contains 10% of FM3a samples. (c) Training set contains 25% of FM3a samples.

TABLE VII
AVERAGE PRECISION (%) OF DENSENET-121 DESCRIPTORS OF LENGTH 128 ON FM3A DATASET (SVM WITH RBF KERNEL)

class	not trained on FM3a images	trained on 10% of FM3a images	trained on 25% of FM3a images
highway	85.25	98.42	99.55
road	18.87	83.52	90.30
settlement	95.73	98.04	98.85
overpass	72.08	76.97	90.07
traffic	71.70	86.96	93.53
mean	68.73	88.78	94.46

to avoid excessive costs, and ii) the number of image classes has to be open. To satisfy these requirements, we studied image descriptors with a restricted representation budget and with no knowledge of the target dataset labels. We achieved resistance to overfitting by considering the following descriptor training requirements: i) no knowledge of the target dataset labels, ii) no knowledge of the target dataset image data, and iii) no use of supervised training whatsoever.

Our experiments have empirically compared a range of classification approaches. We have considered handcrafted image descriptors (GIST, SIFT), non-linear embeddings with respect to placement and appearance distribution of image patches (spatial Fisher vectors), and state-of-the-art convolutional representations (VGG, DenseNet, ResNet and DCGAN) trained on ImageNet. Additionally, we have used PCA to reduce image representations down to as low as 16 components, and investigated the drop of classification performance with respect to representation length.

Best performance is achieved by deep convolutional models trained in a supervised manner, followed by handcrafted models, and finally by a completely unsupervised descriptor based on DCGAN. We have shown that excellent performance can be achieved even with methods that have no knowledge of our target dataset. With an adequate classifier, image representations can be extremely reduced via PCA (down to as few as 128 components), while sacrificing negligible classification performance. In some cases, even representations with as few as 32 components provide useful results. Finally, we have shown that the best performing method, the DenseNet descriptor, performs well even on images with visual degradation caused by bad weather and low sun angles, provided some of the degraded images are added to the training data.

Future work will include further compression of convolutional descriptors by fine-tuning on ImageNet, advanced generative adversarial models, and exploring the generalization across datasets taken in different countries.

REFERENCES

[1] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the fisher vector: Theory and practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, 2013.
[2] J. Wang, T. Zhang, J. Song, N. Sebe, and H. T. Shen, "A survey on learning to hash," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 769–790, Apr. 2018.
[3] I. Sikirić, K. Brkić, J. Krapac, and S. Šegvić, "Robust traffic scene recognition with a limited descriptor length," in *Proc. CVPR Workshop VPRCE*, 2015, pp. 1–9.
[4] C. Seeger, A. Müller, L. Schwarz, and M. Manz, "Towards road type classification with occupancy grids," in *Proc. IEEE Intell. Vehicles Symp. Workshop*, Jun. 2016, pp. 1–4.
[5] I. Sikirić, K. Brkić, J. Krapac, and S. Šegvić, "Image representations on a budget: Traffic scene classification in a restricted bandwidth scenario," in *Proc. IEEE Intell. Vehicles Symp. Workshop*, Jun. 2014, pp. 845–852.
[6] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
[7] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
[8] A. Krizhevsky, "Learning multiple layers of features from tiny images," Canadian Inst. Adv. Res., Toronto, ON, Canada, Tech. Rep., 2009.
[9] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: An evaluation of recent feature encoding methods," in *Proc. BMVC*, 2011, pp. 1–8.
[10] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Workshop Stat. Learn. Comput. Vis. (ECCV)*, 2004, pp. 1–2.
[11] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. CVPR*, Jun. 2006, pp. 2169–2178.
[12] J. Krapac, J. J. Verbeek, and F. Jurie, "Modeling spatial layout with Fisher vectors for image categorization," in *Proc. ICCV*, Nov. 2011, pp. 1487–1494.
[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.
[14] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556
[15] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, Jun. 2015, pp. 1–9.
[16] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 2377–2385.
[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Jun. 2016, pp. 770–778.
[18] A. Veit, M. J. Wilber, and S. J. Belongie, "Residual networks behave like ensembles of relatively shallow networks," in *Proc. 29th Annu. Conf. Neural Inf. Process. Syst.*, 2016, pp. 550–558.
[19] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, Jul. 2017, pp. 1–3.
[20] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proc. CVPR*, Jun. 2014, pp. 1717–1724.

[21] M. Cimpoi, S. Maji, I. Kokkinos, and A. Vedaldi, "Deep filter banks for texture recognition, description, and segmentation," *Int. J. Comput. Vis.*, vol. 118, no. 1, pp. 65–94, 2016.
[22] D. Garcia-Gasulla *et al.* (2017). "On the behavior of convolutional nets for feature extraction." [Online]. Available: https://arxiv.org/abs/1703.01127
[23] L. Nanni, S. Ghidoni, and S. Brahnam, "Handcrafted vs. non-handcrafted features for computer vision classification," *Pattern Recognit.*, vol. 71, pp. 158–172, Nov. 2017.
[24] A. Radford, L. Metz, and S. Chintala. (2015). "Unsupervised representation learning with deep convolutional generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1511.06434
[25] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 2672–2680.
[26] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Mach. Learn. Res. (ICML)*, 2017, pp. 214–233.
[27] B. Ranft and C. Stiller, "The role of machine vision for intelligent vehicles," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 1, pp. 8–19, Mar. 2016.
[28] M. Oeljeklaus, F. Hoffmann, and T. Bertram, "A combined recognition and segmentation model for urban traffic scene understanding," in *Proc. ITSC*, Oct. 2017, pp. 1–6.
[29] S. Di, H. Zhang, X. Mei, D. Prokhorov, and H. Ling, "A benchmark for cross-weather traffic scene understanding," in *Proc. ITSC*, Nov. 2016, pp. 2150–2156.
[30] S. Di, H. Zhang, C.-G. Li, X. Mei, D. Prokhorov, and H. Ling, "Cross-domain traffic scene understanding: A dense correspondence-based transfer learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 745–757, Mar. 2018.
[31] K. F. Hussain, M. Afifi, and G. Moussa, "A comprehensive study of the effect of spatial resolution and color of digital images on vehicle classification," *IEEE Trans. Intell. Transp. Syst.*, to be published.
[32] L.-T. Hsu, Y. Gu, and S. Kamijo, "Intelligent viaduct recognition and driving altitude determination using gps data," *IEEE Trans. Intell. Vehicles*, vol. 2, no. 3, pp. 175–184, Sep. 2017.
[33] H. Liu, R. Wang, S. Shan, and X. Chen, "Deep supervised hashing for fast image retrieval," in *Proc. CVPR*, Jun. 2016, pp. 2064–2072.
[34] L. Liu, Z. Lin, L. Shao, F. Shen, G. Ding, and J. Han, "Sequential discrete hashing for scalable cross-modality similarity retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 107–118, Jan. 2017.
[35] X. Xu, F. Shen, Y. Yang, H. T. Shen, and X. Li, "Learning discriminative binary codes for large-scale cross-modal retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2494–2507, May 2017.
[36] I. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer-Verlag, 1986.
[37] H. Jegou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, Jan. 2011.
[38] T. Ge, K. He, Q. Ke, and J. Sun, "Optimized product quantization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 744–755, Apr. 2014.
[39] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.
[40] A. Oliva and A. B. Torralba, "Scene-centered description from spatial envelope properties," in *Proc. BMCV*, 2002, pp. 263–272.
[41] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[42] A. Mahmood, M. Bennamoun, S. An, and F. Sohel. (2016). "ResFeats: Residual network based features for image classification." [Online]. Available: https://arxiv.org/abs/1611.06656
[43] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 27, 2011.

Ivan Sikirić received the M.S. degree in computer science from the Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia, where he is currently pursuing the Ph.D. degree. Since 2008, he has been with Mireo d.d. His professional and scientific interests include computer vision, fleet management, and navigation and intelligent transportation systems.

Karla Brkić received the M.S. and Ph.D. degrees in computer science from the University of Zagreb, Croatia. She has been with the Faculty of Electrical Engineering and Computing, University of Zagreb, since 2008, where she is currently a Post-Doctoral Researcher. Her scientific interests include traffic sign detection and recognition, spatio-temporal data representations, action recognition, and machine learning.

Petra Bevandić received the B.Sc. and M.Sc. degrees from the Faculty of Electrical Engineering and Computing, Zagreb, Croatia, in 2012 and 2014, respectively. She is currently a Teaching Assistant with the Faculty of Electrical Engineering and Computing, Department of Electronics, Microelectronics, Computer and Intelligent Systems, University of Zagreb.

Ivan Krešo received the B.Sc. and M.Sc. degrees in computer science from the Faculty of Electrical Engineering and Computing, Zagreb, in 2011 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Department of Electronics, Microelectronics, Computer and Intelligent Systems, University of Zagreb. He is also with the Faculty of Electrical Engineering and Computing, University of Zagreb, as a Research Engineer with the Department of Electronics, Microelectronics, Computer and Intelligent Systems. His research interests are in the areas of computer vision and machine learning.

Josip Krapac received the B.S. and M.S. degrees in computer science from the University of Zagreb, Croatia, and the Ph.D. degree from the Université de Caen Basse-Normandie, France. He spent a year and a half as a Post-Doctoral Researcher with INRIA, Rennes, France, from 2011 to 2013, and three and a half years, as a Post-Doctoral Researcher with the University of Zagreb. He is currently a Senior Computer Vision Scientist with Mobius Labs GmbH. His research interests include image representations for object classification, detection and segmentation, and learning models from minimal amount of supervision.

Siniša Šegvić received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from the University of Zagreb, Croatia. He spent one year as a Post-Doctoral Researcher with IRISA/INRIA, Rennes, France, and also with TU Graz, Austria. He is currently a Full Professor with the Faculty of Electrical Engineering and Computing, University of Zagreb. His research and professional interests focus on deep convolutional architectures for classification, semantic segmentation, and stereo reconstruction.