Multi-Label Classification of Traffic Scenes

Ivan Sikirić
Mireo d.d.
Buzinski prilaz 32, 10000 Zagreb
e-mail: ivan.sikiric@mireo.hr

Karla Brkić, Ivan Horvatin, Siniša Šegvić
University of Zagreb
Faculty of Electrical Engineering and Computing
e-mail: karla.brkic@fer.hr,
ivan.horvatin@fer.hr, sinisa.segvic@fer.hr

***Abstract*—This work deals with multi-label classification of traffic scene images. We introduce a novel labeling scheme for the traffic scene dataset FM2. Each image in the dataset is assigned up to five labels: settlement, road, tunnel, traffic and overpass. We propose representing the images with (i) bag-of-words and (ii) GIST descriptors. The bag-of-words model detects SIFT features in training images, clusters them to form visual words, and then represents each image as a histogram of visual words. On the other hand, the GIST descriptor represents an image by capturing perceptual features meaningful to a human observer, such as naturalness, openness, roughness, etc. We compare the two representations by measuring classification performance of Support Vector Machine and Random Forest classifiers. Labels are assigned by applying binary one-vs-all classifiers trained separately for each class. Categorization success is evaluated over multiple labels using a variety of parameters. We report good classification results for easier class labels (*road*, $F1 = 98\%$ and *tunnel*, $F1 = 94\%$), and discuss weaker results (*overpass*, $F1 < 50\%$) that call for use of more advanced methods.**

I. Introduction and related work

Traffic scene classification is an emerging topic with considerable importance in the field of intelligent transportation systems. With the increased availability of cameras in vehicles (either on mobile devices or as embedded hardware in luxurious car models), there are more and more possibilities for simplifying common intelligent transportation tasks. We are especially interested in improving fleet management systems. Fleet management systems are used to track the status of fleets of vehicles belonging to various kinds of companies (e.g. taxi, delivery, cargo transport etc.). They use GPS sensors to track the location of the vehicle, but have little information about the vehicle's environment. Some useful information about the vehicle's surroundings can be inferred by using a camera to record images from the driver's perspective, and then solving a classification problem to detect interesting types of traffic scenes and scenarios. For example, this approach can be used to identify traffic jams, or to differentiate open road environments from urban/rural roads or tunnels.

Image classification in general is a common topic in computer vision, extensively researched in great number of papers. Active research focuses mainly on recognizing images in a large number of diverse classes [1]. The performance of new image classification techniques is usually evaluated on one or more of many publicly available benchmark datasets (e.g. Pascal VOC, Caltech 101, LabelMe etc). This enables a simple and meaningful comparison of state-of-the-art methods applied on various domains.

A common approach to image classification is to first reduce the dimensionality of the image representation using an image descriptor, and then use a general-purpose classifier to perform the classification. Commonly used classifiers are Support Vector Machine (SVM) [2] and Random Forest [3]. Among the best performing general-purpose image descriptors are the bag-of-words model [4], [5], [6], and its derivatives: Locality-constrained Linear Coding (LLC) [7], Fisher vectors (FV) [8] and Spatial Fisher vectors (SFV) [9]. The basis of these methods is finding local image features (e.g. SIFT [10]) and expressing their distribution and relative spatial relations, thus producing a short code that represents the image. Another successful image descriptor is GIST [11], [12], which is not general-purpose, but is designed specifically for scene classification purpose. It captures a set of semantic properties of an image (e.g. naturalness or openness) by measuring responses from several orientation filter over a fixed grid.

The volume of work focused on classifying traffic scenes is considerably smaller than generic image classification research. A small number of works apply general-purpose methods on the problem [13]. Most works present methods that are crafted specifically for classification and understanding of traffic scenes. For instance, Tang and Breckon [14] identify three regions of interest in a traffic scene image: (i) a rectangular region near the center of the image, (ii) a tall rectangular region on the left side of the image and (iii) a wide rectangular region at the bottom of the image. Each of the three regions of interest is represented by a predefined set of local features, as specific features are expected to respond to specific structures which occur in a traffic scene image (e.g. road, or road edge). They introduce a new dataset with four classes: motorway, offroad, trunkroad and urban road. Mioulet et al. [15] build on the ideas of Tang and Breckon [14], retaining the three predefined regions of interest, but representing them with different types of local features and using dedicated hardware.

In our previous work [16], [13], we evaluated classification of traffic scenes in a single-label setup. The main focus was not on the selected labeling approach, but instead on minimizing the image representation size, and on discussing implementation issues specific to fleet management systems. In this paper we evaluate the multi-label classification performance of general purpose image classification methods on traffic scene images. We use the bag-of-words and GIST descriptors combined with SVM and Random Forest classifiers. The performance is evaluated on the FM2 dataset[1] [13] of traffic scene images. Publicly available labeling assigns a single label to each image, even in cases where an image clearly belongs to two or more classes. We introduce a novel labeling scheme for this dataset, in which each image is assigned up to five

[1] http://www.zemris.fer.hr/~ssegvic/datasets/unizg-fer-fm2.zip

labels: settlement, road, tunnel, traffic and overpass.

II. The FM2 dataset

The FM2 dataset contains 6237 images of traffic scenes captured on Croatian roads from the driver's perspective, mostly on highways. The resolution of the images is 640x480. Most of the images were taken on a clear and sunny day. No images were taken during nighttime.

The publicly available labeling of the FM2 dataset assigns a single label per image. In reality, many traffic scenes belong to more than one class (for example, classes *settlement* and *overpass* are not mutually exclusive). Using a single-label classifier in such cases results in an unnecessary loss of information. For that reason, a multi-label approach, in which a set of class labels can be assigned to a single image is a more appropriate solution. In our novel labeling scheme each image is assigned a set of class labels.

We selected five class labels: *settlement*, *road*, *tunnel*, *traffic* and *overpass*. The overview and brief description of the classes is given in Table I. Classes *settlement*, *open road* and *tunnel* describe the location of the vehicle, and their labels are mutually exclusive. Classes *overpass* and *traffic* were chosen because they are interesting for fleet management systems, as described in [16], [13]. The overpass class label usually coexists with the road label, but it can also occur in settlements. The traffic label can occur with any other label. It is also possible that it will be the only label assigned to an image (if a large truck directly in front of camera completely obstructs the view). Some examples of labeled images are shown in Figure 1.

III. Methods

In this paper, we compare two different image representations in a multi-label classification setting. The first considered representation is the bag-of-words model [17], and the second considered representation is the GIST descriptor [11], [12]. For each of these representations, we trained two different classifiers: Support Vector Machine (SVM) [2] and Random Forest [3].

A. Multi-label classification methods

Existing methods for multi-label classification fall into two main categories [18]: (i) problem transformation methods and (ii) algorithm adaptation methods. The problem transformation methods transform the original problem into one or more single-label classification or regression problems. The algorithm adaptation methods do not transform the problem, but rather they adapt the learning algorithms themselves to handle multi-label data. Since we want to evaluate (among other things) the performance of standard SVM algorithm on this problem, we focus on the problem transformation methods. Two most commonly used problem transformation methods [19] are *label power-set method* [20] and *binary relevance* [21].

Label power-set method works by assigning each distinct subset of labels that occurs in the data its own unique label, thus transforming multi-label problem into a single-label one. This method will capture any existing dependence between labels (e.g. in FM2 dataset label *overpass* must coexist with either *road* or *settlement* label, while it cannot coexist with *tunnel* label). One major problem with this approach is having a large number of classes: in case of K labels, the number of resulting classes can be up to 2^K. This usually leads to some classes being represented with very few examples. Since number of examples per class is already low in the FM2 dataset, we chose not to use this method. Instead, we used the binary relevance method.

Binary relevance method works by creating K datasets from the original dataset, where K is the number of classes, and training a separate classifier for each of them. Each of the K datasets contains the same samples as the original dataset, but the labels are different, as they indicate whether the given sample belongs to the class k. Once the transformed datasets are obtained, it is a simple matter to train a binary classifier on each of them. The output for each sample is the union of the outputs for all K classifiers. Even though this method is unable to learn the dependence between labels, it has other advantages. It is suited for applications where label relationships may change over datasets (e.g. it might be able to properly classify scenes with both labels *settlement* and *overpass*, even if no such examples were present in the original training dataset). Its main advantage, however, is its low computational complexity, which scales linearly with the number of classes.

B. The bag-of-words model

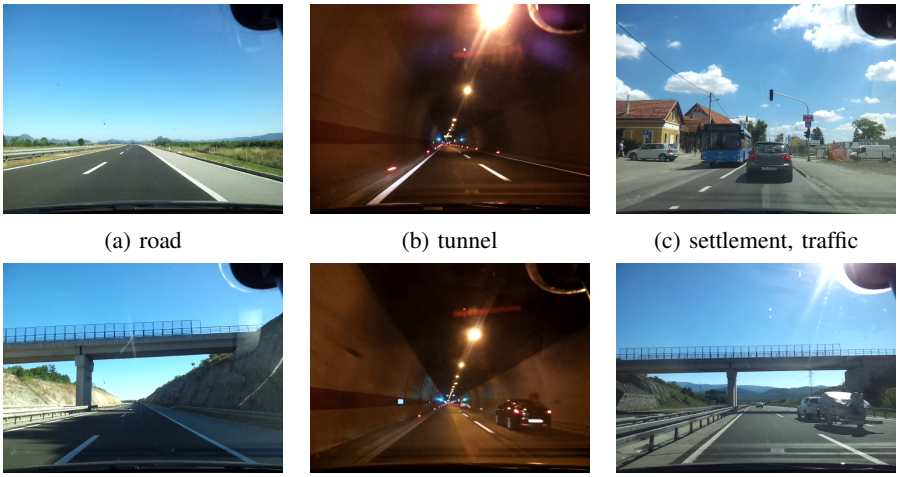
The bag-of-words image representation was adopted into computer vision from the field of text mining. In text mining, a bag-of-words model represents a textual document by a histogram of occurrences of words from a dictionary (thus disregarding their ordering in the document). Similarly, an image can be represented by a histogram of visual words. Local image features can be used as visual words, but the number of all possible local features is too large to represent a dictionary. For this reason, a dictionary of visual words is obtained by sampling local image features from each image in a dataset, and then clustering them into a set of more manageable size. Each cluster center represents a single visual word, and any local feature is considered to be the same visual word as its nearest cluster center. In this work we used SIFT (Scale Invariant Feature Transform) [10] algorithm to extract local features, and *k-means* clustering [22] to produce a dictionary of visual words.

Extraction of SIFT features was done using the implementation from the VLFeat library [23]. Local SIFT features are considered to be stable if they are invariant to changes in scale, orientation, illumination and noise. In the VLFeat library, the amount of features extracted from an image is regulated by two parameters of the extraction algorithm: *peak-thresh* and *edge-thresh*. The *peak-thresh* parameter represents the threshold on the contrast of the extracted features and is used to discard the low-contrast features. The *edge-thresh* parameter represents the threshold on the curvature of the extracted features, and is used to discard edge responses in favor of corner responses. The effect of varying these parameters can be seen on Figures 2, 3, 4 and 5.

A dictionary of visual words was obtained by organizing sampled local features into clusters. We used *k-means* cluster-

class label	class description	number of occurrences
settlement	vehicle is in a settlement	412
road	an open road scene	5239
tunnel	vehicle is in a tunnel, or directly in front of it	681
traffic	other vehicles are visible	2411
overpass	vehicle will soon be, or is already under an overpass	194

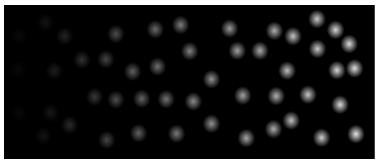
TABLE I: Selected class labels



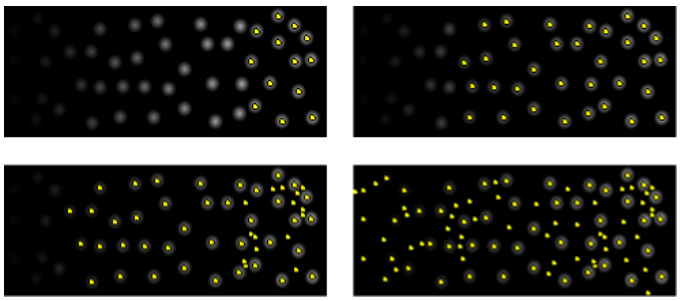
(a) road (b) tunnel (c) settlement, traffic

(d) road, overpass (e) tunnel, traffic (f) road, overpass, traffic

Fig. 1: Examples of labeled images

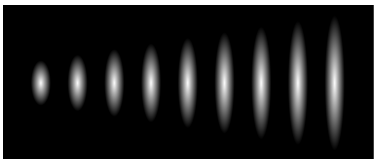


(a) Test image for setting the *peak-thresh* parameter

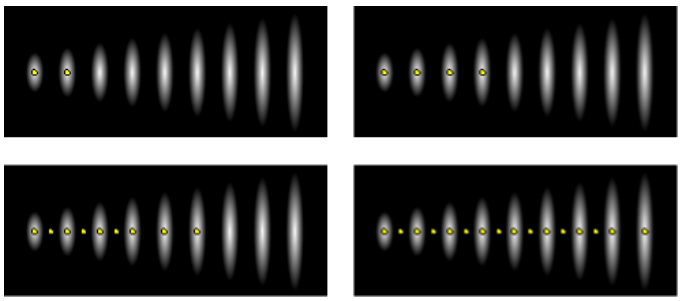


(b) Detected features for varying values of parameter *peak-thresh* (starting from top left: 20, 10, 5, 0)

Fig. 2: Effects of varying the *peak-thresh* parameter



(a) Test image for setting the *edge-thresh* parameter



(b) Detected features for varying values of parameter *edge-thresh* (starting from top left: 7, 10, 15, 25)

Fig. 3: Effects of varying the *edge-thresh* parameter

ing algorithm [22]. It is an iterative algorithm that minimizes the error term:

$$J = \sum_{k=1}^{K} \sum_{\mathbf{x}_i \in S_k} ||\mathbf{x}_i - \mu_k||^2 \quad (1)$$

where K is the desired number of clusters, μ_k is the centroid of cluster k, and S_k is the set of all feature vectors $\mathbf{x}_i$ in cluster k. The initialization of centroids is random, so this algorithm is run several times to increase the chance of finding the global optimum.

C. The GIST image descriptor

While the bag-of-words model can be applied to images of any kind, the GIST descriptor [11], [12] has been developed specifically for scene recognition. It is a low dimensional representation of the scene that captures perceptual features

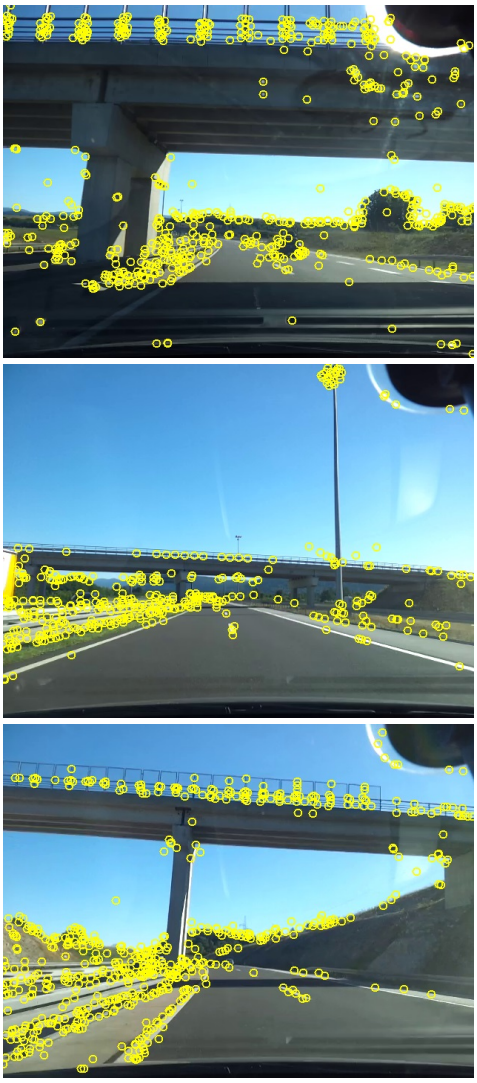


Fig. 4: SIFT features extracted on overpass images for extraction parameters of $edge{-}thresh = 10$ and $peak{-}thresh = 5$

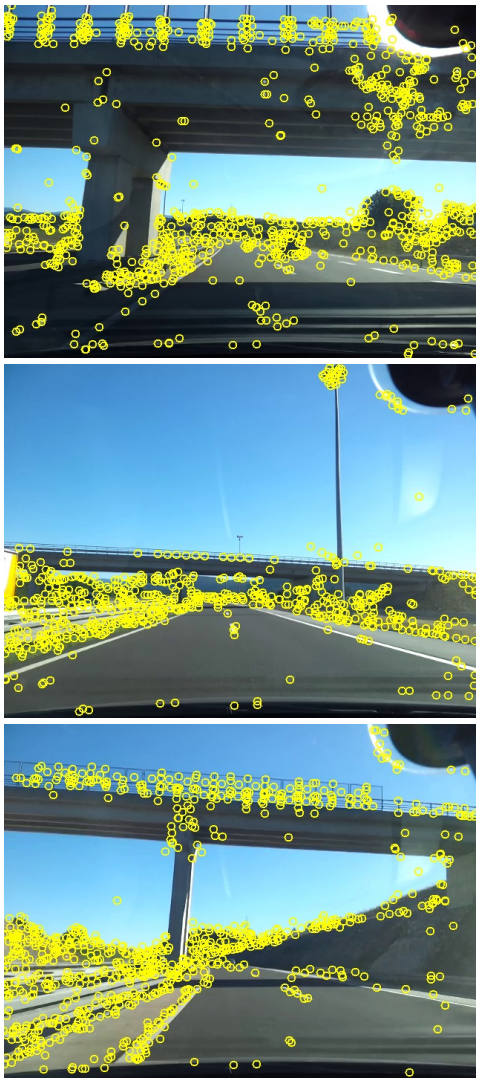


Fig. 5: SIFT features extracted on overpass images for extraction parameters of $edge{-}thresh = 10$ and $peak{-}thresh = 2$

of the scene that are meaningful to a human observer, such as naturalness, openness, roughness, etc. To calculate the GIST descriptor, one first subdivides the image into 16 regions (a 4×4 grid), and then concatenates the average energies of 32 orientation filter responses (8 orientations on 4 scales) for each cell. Therefore the length of the feature vector is $16 \cdot 32 = 512$. Since GIST is designed to ignore accidental presence of small objects in the scene, we expect it to perform better on class labels *road*, *settlement* and *tunnel* than on *traffic* and *overpass* (depending on how much the other vehicles / overpass are dominant in the scene).

D. Support Vector Machine

Support Vector Machine (SVM) [2] is a binary classifier which constructs a maximum-margin hyperplane that divides two sets of vectors. The construction of the hyperplane is done in the learning stage using labeled vectors. SVM is expected to generalize well because of maximizing the margin between sets. To allow for outliers in the learning dataset, we chose to use a variant of the algorithm called *soft-margin* SVM. It introduces an error term ξ_i that allows for misclassified instances, thus sacrificing linear separability in favor of stability:

$$\begin{aligned} &\arg\min_{\mathbf{w},\xi,b} \left(\frac{1}{2}||\mathbf{w}||^2 + C \sum \xi_i \right) \\ &y_i(\mathbf{x}_i \cdot \mathbf{w} - b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \end{aligned} \tag{2}$$

where y_i is the class label of point $\mathbf{x}_i$, and $\mathbf{w}$ and b are the parameters of the hyperplane. Parameter C can be used to choose how much error is to be allowed in the classification process. The lower it is, the more outliers will be tolerated. The higher it is, the closer we get to regular SVM algorithm. Figure 6 illustrates the effects of varying the parameter C. Big circles represent the vectors that are taken into consideration when maximizing the margin of the hyperplane.

E. Random Forest classifier

Random Forest classifier was developed by Breiman and Cutler [3]. The basic idea of the algorithm is to combine

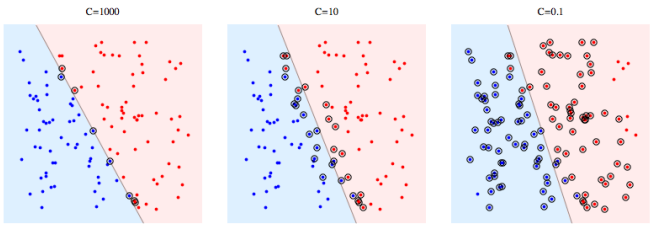


Fig. 6: Examples of margins for various values of C parameter in *soft-margin* SVM, illustrated on a toy problem in 2D space. Points belong to two classes, red and blue. Support vectors are circled. Figure courtesy of Yaroslav Bulatov.

many simple classifiers (decision trees) into a complex one. A decision tree is a classifier in which leaf nodes represent the outcome of the classification (class labels), and inner nodes (called decision nodes) direct the traversal of the tree by thresholding a specific subset of m attributes of the input feature vector. The m attributes evaluated at a given node are selected in a way that maximizes the information gain in the current subset of training data. Hundreds of samples are required to build a decision tree with good classification performance.

A Random Forest consists of many decision trees, where each tree is different because of randomized initialization. The final outcome of the classification is decided by voting of all decision trees. The error of a Random Forest classifier depends on the errors of individual decision trees, as well as on the level of correlation between trees (high correlation results in greater error). Parameter m directly affects the level of correlation and the error of individual trees. The higher the parameter m is, the greater the correlation, but the lower the error of trees becomes.

IV. Experiments and results

In this section we describe the performed multi-label classification experiments. Each image in the dataset was represented using both bag-of-words and GIST image descriptors. Since we used *binary relevance* multi-label classification method on the dataset with $K = 5$ classes, the labels of the dataset were separated into five distinct sets, one for each class. Subsequently, we trained five separate binary classifiers in a one-vs-all fashion. We used 70% of each set for training, while the rest was used for evaluation. The output for each sample is the union of the outputs for all K classifiers. The classifiers we evaluated were Support Vector Machine (SVM) and Random Forest. Two types of classifiers in combination with two types of image descriptors yield a total of four different classification setups.

For the GIST descriptor we used an implementation provided by its authors [11]. For the bag-of-words descriptor we used the solution developed in [24], which uses the VLFeat library [23] implementation of the SIFT algorithm. For the SVM and Random Forest classifiers we used the *scikit-learn* Python library [25]. The same library provides an implementation of *k-means* clustering algorithm, which was used to produce the dictionary of visual words in bag-of-words model. A simple grid search optimization was used to tune the parameters C and m of the classifiers, but the results were nearly identical for a wide range of parameter values.

The performance measure we chose to use if the F1 measure, which is the harmonic mean of precision and recall measures, and is calculated as:

$$F1 = \frac{2T_p}{2T_p + F_n + F_p} \tag{3}$$

where T_p, F_n and F_p are the number of true positives, false negatives and false positives, respectively.

The detailed per-class results are shown in Table II. All combinations of classifiers and descriptors have shown similar performance for every class. Very good performance was achieved on *road* and *tunnel* classes ($F1 \geq 0.94$). Moderate performance was achieved on *settlement* and *traffic* classes ($0.64 \leq F1 \leq 0.86$). Very poor performance was shown on the class *overpass* ($F1 \leq 0.51$). For successful classification of *overpass* and *traffic* images, in many cases it is necessary to consider some small detail of the scene (the overpass and vehicles are often in the distance, and rarely dominate the scene). Since GIST is designed for scene recognition, rather than being a general-purpose descriptor, it is not surprising that it often fails to capture such details. Similarly, our implementation of bag-of-words model is expected to have problems with the same type of images. Since the implementation we used extracts only stable SIFT features, it is likely that in many cases very few local features were extracted in the regions of important, but small details in the scene. It is important to note that the dataset contains several thousand images with the *traffic* label, but only a couple hundred with *overpass* label, which explains the difference in performance for those classes. The class *tunnel* is easy to classify because all the tunnel images are very similar to each other, and very different from images of other classes. On the other hand, the appearance of *settlement* images varies greatly, which makes their classification a more difficult task. To improve the classification performance of *settlement* class, we should include much more training examples. The best performance is achieved for *road* images, which are the most occurring image type in in the dataset, are not defined by small details in the scene, and are similar to each other in appearance.

V. Conclusion and future work

The proposed methods have shown remarkably good results for some class labels (*road* and *tunnel*), while performing rather poorly on some other class labels (*overpass*). Both classifiers (SVM and Random Forest), and both descriptors (bag-of-words) showed similar level of performance (both overall, and per-class). Therefore, we conclude that classes *settlement* and *traffic* are moderately hard to classify, and the class *overpass* is very hard to classify. This calls for use of more advanced methods, and expansion of the dataset to include much more instances of those classes (especially for the case of settlement, which is the class with the greatest variability in appearance). The GIST descriptor is designed for scene classification, and is expected to perform poorly in capturing small details in a scene (such as occasional vehicle and overpass in the distance). Performance of bag-of-words model is expected to be improved by using dense SIFT extractor, instead of the keypoint-driven one, because

SVM ($C = 1$) on bag-of-words

	precision	recall	F1
settlement	0.76	0.78	0.77
tunnel	0.94	0.94	0.94
road	0.98	0.97	0.98
traffic	0.62	0.70	0.66
overpass	0.47	0.39	0.42
average	0.86	0.88	0.87

Random Forest (500 trees) on bag-of-words

	precision	recall	F1
settlement	0.96	0.56	0.71
tunnel	0.99	0.89	0.94
road	0.97	1.00	0.98
traffic	0.79	0.53	0.64
overpass	1.00	0.06	0.11
average	0.92	0.83	0.86

SVM ($C = 1$) on GIST descriptor

	precision	recall	F1
settlement	0.59	0.97	0.73
tunnel	0.92	0.96	0.94
road	0.99	0.98	0.99
traffic	0.71	0.79	0.75
overpass	0.36	0.87	0.51
average	0.88	0.92	0.90

Random Forest (500 trees) on GIST descriptor

	precision	recall	F1
settlement	0.90	0.82	0.86
tunnel	0.99	0.91	0.95
road	0.99	0.99	0.99
traffic	0.89	0.75	0.81
overpass	1.00	0.30	0.46
average	0.96	0.90	0.92

TABLE II: results for linear SVM ($C = 1$) and Random Forest classifiers (500 trees) on the bag-of-words and GIST descriptors

in many occasions there were too few local features captured on some important part of the scene (such as a vehicle or an overpass in the distance). Other possible improvements include using RootSIFT method for comparing SIFT descriptors [26], adding spatial coding to bag-of-words (SPM or 1+4+3), and using RBF or other kernel in case of SVM. For our future work, we plan to expand the scope of multi-label classification experiments to the same extent as single-label experiments in our previous work [13]. This includes evaluating other types of image descriptors (Locality-constrained Linear Coding and Spatial Fisher vectors) as well as considering very small image representations. That will give us a strong basis for comparison of single-label vs multi-label classification performance from the user's standpoint.

ACKNOWLEDGMENTS

This research has been supported by the research project Research Centre for Advanced Cooperative Systems (EU FP7 #285939).

REFERENCES

[1] A. Pinz, "Object categorization," *Foundations and Trends in Computer Graphics and Vision*, vol. 1, no. 4, 2005.

[2] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[3] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[4] F.-F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *CVPR*, (Washington, DC, USA), pp. 524–531, IEEE Computer Society, 2005.

[5] A. Bosch, A. Zisserman, and X. Muñoz, "Scene classification via pLSA," in *ECCV*, (Berlin, Heidelberg), pp. 517–530, Springer-Verlag, 2006.

[6] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, (Washington, DC, USA), pp. 2169–2178, IEEE Computer Society, 2006.

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

[8] F. Perronnin and C. R. Dance, "Fisher kernels on visual vocabularies for image categorization," in *CVPR*, 2007.

[9] J. Krapac, J. J. Verbeek, and F. Jurie, "Modeling spatial layout with Fisher vectors for image categorization," in *ICCV*, 2011.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, pp. 91–110, 2004.

[11] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vision*, vol. 42, pp. 145–175, May 2001.

[12] A. Oliva and A. B. Torralba, "Scene-centered description from spatial envelope properties," in *BMCV*, (London, UK, UK), pp. 263–272, Springer-Verlag, 2002.

[13] I. Sikirić, K. Brkić, J. Krapac, and S. Šegvić, "Image representations on a budget: Traffic scene classification in a restricted bandwidth scenario," *IEEE Intelligent Vehicles Symposium*, 2014.

[14] I. Tang and T. Breckon, "Automatic road environment classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, pp. 476–484, June 2011.

[15] L. Mioulet, T. Breckon, A. Mouton, H. Liang, and T. Morie, "Gabor features for real-time road environment classification," in *ICIT*, pp. 1117–1121, IEEE, February 2013.

[16] I. Sikirić, K. Brkić, and S. Šegvić, "Classifying traffic scenes using the GIST image descriptor," in *CCVW 2013 Proceedings of the Croatian Computer Vision Workshop*, pp. 1–6, September 2013.

[17] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on Statistical Learning in Computer Vision, ECCV*, 2004.

[18] G. Tsoumakas and I. Katakis, "Multi label classification: An overview," *International Journal of Data Warehouse and Mining*, vol. 3, no. 3, pp. 1–13, 2007.

[19] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine Learning and Knowledge Discovery in Databases*, vol. 5782, pp. 254–269, 2009.

[20] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *Proceedings of the 18th European Conference on Machine Learning*, ECML '07, (Berlin, Heidelberg), pp. 406–417, Springer-Verlag, 2007.

[21] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," in *In Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 22–30, Springer, 2004.

[22] J. MacQueen, "Some methods for classification and analysis of multivariate observations.." Proc. 5th Berkeley Symp. Math. Stat. Probab., Univ. Calif. 1965/66, 1, 281-297 (1967)., 1967.

[23] A. Vedaldi and B. Fulkerson, "VLFeat - an open and portable library of computer vision algorithms," in *ACM International Conference on Multimedia*, 2010.

[24] I. Horvatin, "Multi-label classification of traffic scenes," Master's thesis, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, 2014.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[26] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.