Image-Based Road Type Classification

Viktor Slavkovikj, Steven Verstockt, Wesley De Neve, Sofie Van Hoecke, and Rik Van de Walle

In: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR 2014), 2014.

Image-Based Road Type Classification

Viktor Slavkovikj*, Steven Verstockt*,
Wesley De Neve*†, Sofie Van Hoecke*, and Rik Van de Walle*

* Multimedia Lab, Department of Electronics and Information Systems,
Ghent University-iMinds, Ghent, Belgium

† Image and Video Systems Lab, Korea Advanced Institute of Science and Technology (KAIST),
Yuseong-gu, Daejeon, 305-732, Republic of Korea

{viktor.slavkovikj, steven.verstockt, wesley.deneve, sofie.vanhoecke, rik.vandewalle}@ugent.be

***Abstract*—The ability to automatically determine the road type from sensor data is of great significance for automatic annotation of routes and autonomous navigation of robots and vehicles. In this paper, we present a novel algorithm for content-based road type classification from images. The proposed method learns discriminative features from training data in an unsupervised manner, thus not requiring domain-specific feature engineering. This is an advantage over related road surface classification algorithms which are only able to make a distinction between pre-specified uniform terrains. In order to evaluate the proposed approach, we have constructed a challenging road image dataset of 20,000 samples from real-world road images in the paved and unpaved road classes. Experimental results on this dataset show that the proposed algorithm can achieve state-of-the-art performance in road type classification.**

I. Introduction

The advance of sensor technology, coupled with increasing on-board processing capabilities of current smartphone devices, has enabled users to efficiently create, capture, and share information about their activities. At the same time, the abundance of user-generated sensor information has prompted the creation of web-based systems which provide different services from analyses of the aggregated user data. Online geographic information systems such as OpenStreetMap [1], RouteYou [1], and Bikemap [2] rely heavily on user-contributed sensor data to offer location oriented services.

Two common goals of this kind of systems are to provide querying of locations on interactive maps, and discovery of routes for recreational GPS-users such as cyclists and hikers. The latter makes use of pre-created GPS trajectories submitted by the users, while the former utilizes user annotations of objects and infrastructure. For route finding, it has been shown [2] that the road type or terrain characteristics, have an important influence on route ranking. Therefore, it is not surprising that people try to annotate the type of the route they are submitting to allow for an effective search of good routes for fellow users. As opposed to route recording, annotating the different parts of a route requires active user involvement, and is both laborious and error prone. In this paper, we propose a method for automatic content-based road type classification from images. The proposed method does not require user intervention and is suitable to operate on image data of road surfaces. Such images can be obtained from mobile sensors (e.g., from a smartphone camera setup [3]), for which our proposed method can be applied directly. Images from online geographic services like Google Street View in combination with a road detection method [4], [5], for easy extraction of road surface sub-images, can also be used. In this work, however, we focus only on the problem of learning road type categories from images.

The remainder of this paper is organized as follows. In Section II, we discuss related work in road and terrain classification. Subsequently, Section III contains a description of the proposed method for road type classification by unsupervised learning of image features. In order to make a meaningful comparison, two other algorithms for road type learning are also discussed. Next, Section IV details the road image dataset which we use to test the methods given in Section III. In Section V, we present experimental results. Section VI concludes the paper.

II. Related Work

Content-based road or terrain classification plays an important role in the domain of autonomous robot/vehicle navigation. Related works [6]–[8] in this domain make use of data from vibration sensors (on-board accelerometers or inertial measurement units (IMUs)) to classify the terrain type which the robot/vehicle is traversing. Visual terrain classification can be used when on-board accelerometer sensors or IMUs are not available. In road type classification from visual data, Popescu et al. [9] classify road surfaces based on texture features obtained from statistical properties of medium co-occurrence matrices of road images. Tang and Breckon [10] use a feature set of color, texture, and edge features from constrained sub-regions of driver's perspective images to train a neural network classifier of road types. For the color features, they derive histogram distributions and pixel statistics (mean,

[1] http://www.openstreetmap.org
[2] http://www.bikemap.net/en/

standard deviation, and entropy) from selected channels of different color space representations of the images. The texture features are based on gray-level co-occurrence matrix statistics and Gabor filters, while the edge features are based on Hough line fitting and contour tracking of the Canny edge output of an image. Khan et al. [11] calculate SURF features, over intersections of a regular grid, from terrain images captured by a mobile robot. The extracted features are used to train a Random Forest classifier to discriminate between terrain surfaces.

Unlike our proposed method, all of the previous visual content-based terrain classification approaches [9]–[11] make use of engineered color and/or texture features. In [9] and [11], the images used give a close-up view of uniform terrain surfaces. The approach in [10] is not suitable when only a limited area of the terrain surface is available (as in the case of robot navigation). By contrast, our road image dataset contains road surface images taken from real-world Google Street View photos, which contain artifacts such as motion blur, illumination changes, and overexposed areas.

III. Feature Extraction and Classification of Road Images

In this section, we describe three algorithms for content-based road type classification: our proposed algorithm for learning road image features from unlabeled samples, an algorithm which uses specifically engineered features for discrimination of road types, and a baseline method.

A. Unsupervised Learning of Road Image Features

Our proposed approach similarly to other convolutional learning methods, such as the one of Lee et al. [12], learns features from unlabeled images. In particular, we implement a single-layer processing pipeline as the one described by Coates et al. [13]. The processing pipeline consists of two stages: unsupervised feature learning, and feature extraction and classification.

1) Unsupervised Feature Learning: In the first stage, random patches of size $r \times r$ pixels are extracted from the unlabeled road images, where r is the receptive field size. Each of the extracted patches is reshaped as a vector of pixel values in $\mathbb{R}^M, M = r^2 \cdot c$, where c is the number of image channels. Normally, the input images are represented in three-channel RGB color space. However, due to the characteristics of the employed feature learning algorithm, and based on our empirical observations, we introduce a conversion of the input images from RGB to CIELAB [14] color space assuming neutral day illuminant (D65). The transform to a perceptually more uniform color space, such as CIELAB, enables more accurate distance calculations in algorithms for learning feature mappings from color images. In this way, we construct a dataset $X = \{x^{(1)}, \ldots, x^{(m)}\}$ of randomly sampled patches. Each of the vectors $x^{(j)} \in \mathbb{R}^M$ is locally normalized to zero mean and unit variance. Also, the entire dataset of random patches X is whitened [15]. The pre-processed dataset is then used for unsupervised learning of road image features.

K-means learning: The goal of the unsupervised learning algorithm is to learn a feature mapping function $g : \mathbb{R}^M \rightarrow \mathbb{R}^K$ from the dataset X, so that an input vector x can be mapped to a new feature vector $g(x)$. Experimental results [13]

procedure KMEANS(k, b, t, X)
  Input: k, mini-batch size b, iterations t, dataset X
  Return: centroids C
  Initialize each $c \in C$ with $k-means++$ initialization
  $v \leftarrow 0$ ▷ Per-centroid counts
  for $i \leftarrow 1, t$ **do**
    $M \leftarrow b$ examples picked randomly from X
    $m \leftarrow 0$ ▷ Batch centers
    $u \leftarrow 0$ ▷ Batch per-center counts
    for all $x \in M$ **do**
      $d \leftarrow f(C, x)$ ▷ Cache centroid nearest to x
      $D \leftarrow D \cup d$
      $u[d] \leftarrow u[d] + 1$
      $m[d] \leftarrow m[d] + x$
    end for
    for all $c \in D$ **do**
      $\mu \leftarrow \frac{m[c]}{u[c]}$ ▷ Mean sample
      $v[c] \leftarrow v[c] + u[c]$ ▷ Update counts
      $\eta \leftarrow \frac{1}{v[c]}$ ▷ Learning rate
      $c \leftarrow (1 - \eta)c + \eta\mu$ ▷ Take gradient step
    end for
  end for
  return C ▷ Return the centroids
end procedure

Fig. 1. K-means algorithm with mini-batch stochastic gradient descent cost minimization.

have demonstrated that an over-complete dictionary for feature mapping can be learned effectively with fast unsupervised learning algorithms such as k-means learning. Here, we implement a modified version of an efficient stochastic gradient descent k-means algorithm proposed by Sculley [16]. Because the k-means algorithm is only guaranteed to converge to a local optimum of its cost function, the resultant clustering is dependent on the manner of initialization. Therefore, we use an initialization procedure developed by Arthur and Vassilvitskii [17], where each of the k centroids are chosen one at a time, at random, from the dataset with probability proportional to the distance from the centroids already chosen (see [17] for more details).

In order to make the algorithm more adaptable for parallelization, the gradient update is performed with a larger step. That is, instead of performing the gradient update step on each of the random samples in the batch, we calculate an update step once for each of the unique centroids to which the samples in the batch are closest to. The modified algorithm is given in Figure 1.

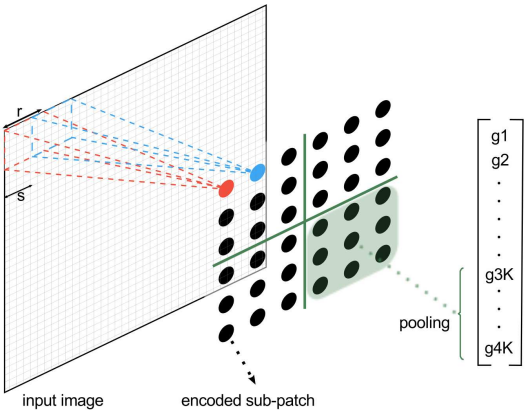


Fig. 2. Illustration of feature extraction from an input image. First patches of size $r \times r$ are sampled from the image. Each of the patches are sampled s pixels apart. Then, the reshaped and pre-processed vector representing each patch is mapped to a new K dimensional vector (depicted as filled circles) by using the learned dictionary. Finally, the encoded vectors are pooled over a two-dimensional grid and concatenated to form the final feature vector of the input image.

2) Feature Extraction and Classification: Once the dictionary C of basis functions $c^{(k)}$ has been learned from the unlabeled training set, it is used to map novel input samples to features. The mapping is done by using an encoding transform. We employ one of the sparse non-linear encodings given by Coates et al. [13], [18], which performs a soft assignment for each feature k of the feature vector $g(x)$:

$$g_k(x) = \max(0, \text{mean}(z) - z_k), \tag{1}$$

where $z_k = \left\| x - c^{(k)} \right\|_2$. The function in Equation 1 produces non-zero values only for the features k where the distance of x to $c^{(k)}$ is below the average of the distances of x to c, $\forall c \in C$.

The learned feature mapping function $g : \mathbb{R}^M \rightarrow \mathbb{R}^K$ allows for feature extraction from a single $r \times r$ patch. To extract features from a road surface image, we apply the feature extraction over the entire input image. The sampling of the input is convolutional (as shown in Figure 2), but it can also be performed with a step-size s between two consecutive patches.

Each of the extracted patches is represented by a vector in $\mathbb{R}^K$ after encoding. Grid regions in the $\mathbb{R}^K$ feature space are averaged to reduce the dimensionality of the feature representation of the input image, and to improve the robustness of the averaged feature vector to small spatial changes in the image. The averaged, or pooled, vectors are then concatenated into the final feature vector.

For each of the labeled images in the training set, we apply the previously described feature extraction process. The resultant feature vectors and training labels are then used for classification. Because of the large amount of features obtained through unsupervised feature learning, we can make use of a linear classification algorithm. A linear L2 Support Vector Machine (SVM) [19] compared favorably to other classification methods. Hence, we trained a linear L2 SVM for classification using cross-validation to determine the regularization parameter of the linear model.

B. Domain Engineered Features

For this method, we use a set of visual features similar to the one that has been used in previous work [3], which has achieved state-of-the-art results in terrain classification. Each of the features, described hereafter, are designed to discriminate a certain type or types of road surfaces. We use in total nine features, as follows:

- **Color:** four features quantifying the percentage of blue, green, white, and low saturated orange/red pixels in the road image. The features, respectively, give high output for cobblestones and asphalt, grass, asphalt road markings, and dirt roads and gravel.
- **Gray:** percentage of pixels that satisfy the RGB color equality R $\approx$ G $\approx$ B. This feature has higher value for asphalt and cobblestones than for unpaved roads.
- **Energy:** the Fourier transform energy spread of the road image. The energy is large for road surfaces which contain a lot of edges (such as cobblestones).
- **Hough:** number of distinct edge directions in the Hough transform of the road image. Road surfaces with structured texture (such as tiles) result in high number of edges.
- **EOH:** MPEG-7 Edge Orientation Histogram spread of edges [20]. EOH has large values for road surfaces with random edge distribution (such as gravel).
- **GLCM:** product of gray-level co-occurrence matrix statistics of local binary pattern filtered road image [21], [22]. High feature values for cobblestones and some unpaved road surfaces.

Using the features described above, we extract feature vectors from the set of training images. As in [3], a Random Forest classifier [23] is used for the classification task.

C. Baseline Method

Our third method is a simple baseline for road type classification to which we compare results obtained from the applied learning algorithms. That is, for each road image in

the training set, we extract a patch of size $r \times r$ from the center of the image. Then, the pixels of each color plane of the patch are concatenated to form a feature vector. Extracted feature vectors, together with the corresponding labels, are fed to a linear SVM classifier. The same classification method was used as the one in Section III-A. The proposed baseline makes use only of the color information of the road images. Because very simple content-based features are used, it also provides an insight into the separability of the samples in the dataset.

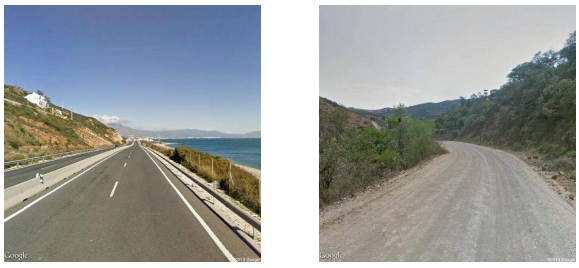
IV. Road Image Dataset

For the purpose of testing the different road type classification methods, we have built a dataset of small road surface images (see Figure 10a, and Figure 10b). The dataset is constructed using the geographical information from trajectories traversed by recreational cyclists in combination with the Google Street View web service.

In order to sample different road surfaces, we extract geo coordinates (latitude, and longitude) from points along a GPS trajectory. Duplicate trajectory points are removed and are not considered for further processing. To prevent redundant samples of road images in the final dataset, the trajectory points are filtered so that each point is at least 50 meters apart from the previous point. The distance between trajectory points, given their respective latitudes φ and longitudes λ, is calculated using the Haversine equation for the shortest distance d between two points over the Earth's surface:

$$\begin{aligned} a &= \sin\left(\Delta\varphi/2\right)^2 + \cos\left(\varphi_1\right)\cos\left(\varphi_2\right)\sin\left(\Delta\lambda/2\right)^2 \\ d &= 2R\arcsin\left(\sqrt{a}\right), \end{aligned} \tag{2}$$

where R denotes the radius of the Earth. Once we obtain the



(a) (b)

Fig. 3. Google Street View images from the paved (Figure 3a) and unpaved (Figure 3b) road classes.

filtered subset of geo coordinates from a given trajectory, we use the Google Street View API [3] to query images from the selected locations.

One issue of the proposed approach for road image querying is how to obtain a good view of the road surface. The Google Street View web service allows for optional parameters in the image query, such as pitch, which specifies the angle of the camera (up or down) relative to the Street View vehicle. A pitch of -90 degrees gives a camera view perpendicular

[3]https://developers.google.com/maps/documentation/streetview/

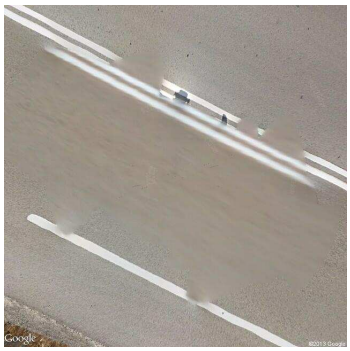


Fig. 4. Zoomed out Google Street View image perpendicular to the road surface (camera pitch $-90°$). The image contains blurred areas (under the vehicle) where the image content was interpolated.

to the road surface. However, with the camera in a straight down position, the quality of the image obtained is limited (see Figure 4). The reason is due to the way the camera is mounted on the Street View vehicle, i.e. the image from the road perpendicular camera view has to be interpolated from images taken from different angles of the camera relative to the vehicle. We use instead a different approach to obtain images with a clear view of the road, such as the images in Figure 3. Keeping the pitch to $0°$, we calculate for each position the compass heading θ of the camera with regard to the next position on the trajectory (as shown in Figure 5). The heading is calculated from the latitudes φ and longitudes λ of the two coordinate points:

$$\begin{aligned} a &= \sin\left(\Delta\lambda\right)\cos\left(\varphi_2\right) \\ b &= \cos\left(\varphi_1\right)\sin\left(\varphi_2\right) - \sin\left(\varphi_1\right)\cos\left(\varphi_2\right)\cos\left(\Delta\lambda\right) \\ \theta &= \arctan\left(\frac{a}{b}\right). \end{aligned} \tag{3}$$

The images obtained in this way are suitable for content-based analysis of road surfaces.

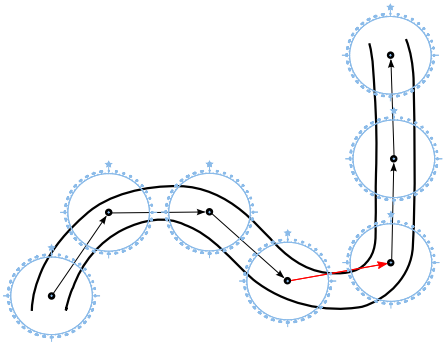


Fig. 5. Illustration of camera view placement along a trajectory based on compass heading (forward azimuth) calculation between points. The road is in the center of the acquired images. This is not the case (depicted by the red arrow) only in a small number of the acquired Google Street View images, where there is a sharp turn in trajectory direction.

We manually extract 32×32 sub-images from the acquired road images to build our dataset (see Figure 6). Because only images from roads traversable by a motor vehicle can be

obtained, we create 2 classes of road types: **paved roads**, and **unpaved roads**. There are in total 20,000 road images in the dataset, where the two classes are proportionally represented by half of the samples. Each of the two classes are comprehensive, i.e. they include samples from different subclasses of road types within the super class. For example, the paved roads class contains sample images from asphalt roads, but also other images of road surfaces with different texture and color, such as cobble stones, tiles, bicycle lanes, pedestrian crossings etc. In the unpaved roads class there are sample images of different dirt and gravel roads. By not dividing the dataset samples into further subclasses, we obtain a more challenging set which can be used to evaluate the inference capabilities of the proposed unsupervised learning method to the two higher level road categories.

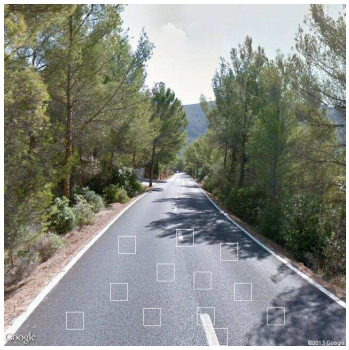


Fig. 6. Extraction of sub-images from the road surface. We extract 32×32 pixel sub-images of different road surfaces to form the road image dataset.

V. Experimental Results

For our experiments, we used the road image dataset presented above. The dataset was partitioned into a training set of 16,000 images (8,000 images per class), and a test set of 4,000 images (2,000 images for each class). For each

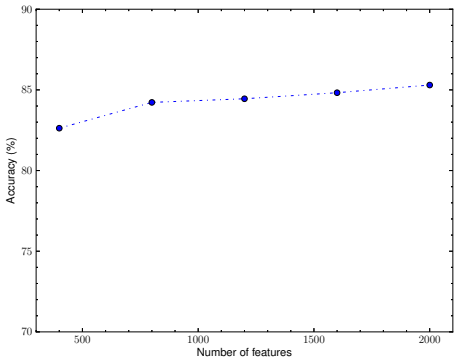


Fig. 7. Effect of number of features on test classification accuracy.

of the compared methods, we used 5-fold cross validation to optimize the model parameters. The optimal cross validation parameters were then used to train the model on the whole training set. Finally, the learned model was tested on the held out test set. For the unsupervised road image feature learning algorithm, we tested different values for the number of features, the step size s, and the receptive field r. Because the computational costs prohibit a full grid search over all parameters, we varied one parameter while keeping the rest fixed. Afterwards, we used the parameter values that achieved the optimal performance for the final test set results (given in Table I).

For the unsupervised feature learning algorithm, when varying the number of features used, better results were obtained when using a higher number of features (see Figure 7). As it can be seen in Figure 8, convolutional sampling of the input image with a step size $s = 1$ produced significantly better results than non-overlapping sampling. For the receptive field, smaller receptive field sizes gave better results (see Figure 9).

From the experiments, it can be inferred that, except for

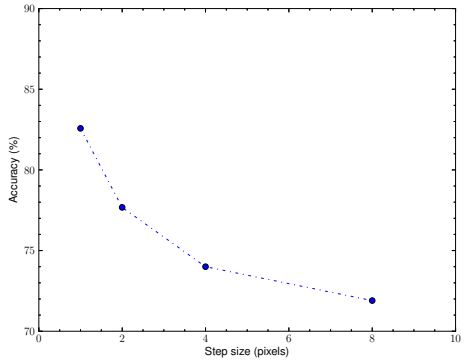


Fig. 8. Effect of step size on test classification accuracy.

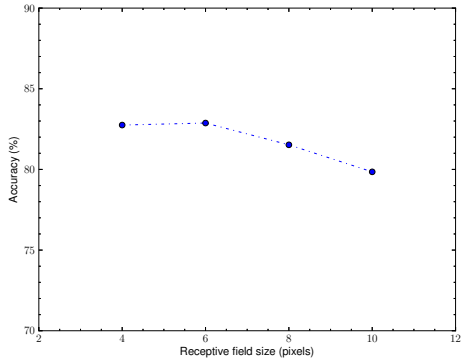


Fig. 9. Effect of receptive field size on test classification accuracy.

the step size parameter, the method is not very sensitive to parameter tuning.

TABLE I
Test classification accuracy on the road image dataset.

Algorithm	Test Set Accuracy
Baseline	74.23%
Engineered Features	84.25%
Unsupervised Features	**85.30%**

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

Fig. 10. Samples of road surface images from the paved (Figure 10a) and unpaved (Figure 10b) road classes. Within a class, there are samples with very different color and texture characteristics (compare surfaces from asphalt roads and the red bicycle lanes in Figure 10a). There are also very similar samples between classes (see patch on third row, second column from Figure 10a, and patch on second row, first column from Figure 10b). Some unpaved road samples misclassified as paved road (Figure 10c). Paved road samples incorrectly assigned to the unpaved road class (Figure 10d).

VI. Conclusion

In this paper, we have proposed a novel content-based method for road type classification by unsupervised learning of image features. We conducted experiments on a road image dataset of 20,000 samples partitioned into 2 comprehensive road classes. The experimental results show that the proposed approach is on par with a state-of-the-art method for road surface classification which makes use of domain engineered features. However, unlike other road surface classification algorithms, it can successfully learn discriminative features from unlabeled data. Therefore, the presented method is suitable for use in content-adaptive computer vision systems, such as systems for robot/vehicle navigation, and in systems for automatic route annotation.

Acknowledgments

The research activities as described in this paper were funded by Ghent University and the Independent Research Institute iMinds.

References

[1] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *Pervasive Computing, IEEE*, vol. 7, no. 4, pp. 12–18, 2008.

[2] S. Reddy, K. Shilton, G. Denisov, C. Cenizal, D. Estrin, and M. Srivastava, "Biketastic: sensing and mapping for better biking," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10. New York, NY, USA: ACM, 2010, pp. 1817–1820.

[3] S. Verstockt, V. Slavkovikj, P. D. Potter, J. Slowack, and R. V. de Walle, "Multi-modal bike sensing for automatic geo-annotation - geo-annotation of road/terrain type by participatory bike-sensing," in *SIGMAP*. SciTePress, 2013, pp. 39–49.

[4] H. Kong, J.-Y. Audibert, and J. Ponce, "General road detection from a single image." *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 2211–2220, 2010.

[5] J. Alvarez and A. Lopez, "Road detection based on illuminant invariance," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 1, pp. 184–193, 2011.

[6] F. G. Bermudez, R. C. Julian, D. W. Haldane, P. Abbeel, and R. S. Fearing, "Performance analysis and terrain classification for a legged robot over rough terrain." in *IROS*. IEEE, 2012, pp. 513–519.

[7] C. Weiss, H. Frhlich, and A. Zell, "Vibration-based terrain classification using support vector machines." in *IROS*. IEEE, 2006, pp. 4429–4434.

[8] C. C. Ward and K. Iagnemma, "Speed-independent vibration-based terrain classification for passenger vehicles," *Vehicle System Dynamics*, vol. 47, no. 9, pp. 1095–1113, 2009.

[9] D. Popescu, R. Dobrescu, and D. Merezeanu, "Road analysis based on texture similarity evaluation," in *Proceedings of the 7th WSEAS International Conference on Signal Processing*, ser. SIP'08. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2008, pp. 47–51.

[10] I. Tang and T. Breckon, "Automatic road environment classification," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 2, pp. 476–484, 2011.

[11] Y. Khan, P. Komma, and A. Zell, "High resolution visual terrain classification for outdoor robots," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, 2011, pp. 1014–1021.

[12] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09. New York, NY, USA: ACM, 2009, pp. 609–616.

[13] A. Coates, A. Y. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," *Journal of Machine Learning Research - Proceedings Track*, vol. 15, pp. 215–223, 2011.

[14] CIE, "Colorimetry," CIE Publication No. CIE 15.2, Central Bureau of the CIE, Vienna, 1986.

[15] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[16] D. Sculley, "Web-scale k-means clustering," in *Proceedings of the 19th International Conference on World Wide Web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 1177–1178.

[17] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '07. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[18] A. Coates and A. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ser. ICML '11, L. Getoor and T. Scheffer, Eds. New York, NY, USA: ACM, June 2011, pp. 921–928.

[19] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[20] A. Pinheiro, "Image descriptors based on the edge orientation," in *Semantic Media Adaptation and Personalization, 2009. SMAP '09. 4th International Workshop on*, 2009, pp. 73–78.

[21] S. F. Ershad, "Texture classification approach based on combination of edge & co-occurrence and local binary pattern," *CoRR*, vol. abs/1203.4855, 2012.

[22] M. Pietikäinen, G. Zhao, A. Hadid, and T. Ahonen, *Computer Vision Using Local Binary Patterns*, ser. Computational Imaging and Vision. Springer, 2011, no. 40.

[23] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, no. 1, p. 532, Oct. 2001.