

# Towards Road Type Classification with Occupancy Grids

Christoph Seeger, André Müller, Loren Schwarz, and Michael Manz<sup>1</sup>

**Abstract**—Occupancy grid mapping is a popular approach for robust obstacle fusion and is already used in series production in the automotive industry. In this work, local occupancy grids are used in the situation interpretation layer of the environment model for a context classification task. We present several approaches for recognizing, without the use of navigation map or image data, whether the vehicle is driving on a freeway, a highway, in a parking area or on an urban road. Inspired from the success of deep learning approaches, we compare an end-to-end Convolutional Neural Network classifier to a Support Vector Machine trained on hand-crafted features. All approaches were tested on a dataset containing about 700 local occupancy grids per class for training and 150 for testing. The best methods achieve a test accuracy of 94%. We see the proposed work as a first step towards classification of high-level information from occupancy grids and will extend the approach to other situations.

## I. INTRODUCTION

Environment perception is a key aspect in advanced driver assistance systems. In particular, autonomous driving requires increasingly diverse sensors and sophisticated fusion algorithms. Because the complexity of rule based-approaches is beyond a manageable level, machine learning approaches play an increasingly important role in many parts of the environment model. When it comes to urban environments, situation interpretation becomes a key ingredient. In this sense, we address the problem of road type classification. Being able to tell the type of road the vehicle is driving on can be beneficial for various purposes. First, environment perception and sensor fusion algorithms can be parameterized accordingly [1]. Second, the vehicle's user interface can be adapted or specific assistance functions can be enabled or disabled. In particular, autonomous driving functionalities can be limited to roads with a structural separation to oncoming traffic or pedestrians.

We propose to perform this situation interpretation task based on local occupancy grids, which are otherwise mainly used for fusion and detection of obstacles. One advantage of using occupancy grids instead of camera images as a source for road type classification is that they are largely illumination invariant. Furthermore, occupancy grids can be built from any type of sensor and thus offer a powerful level of abstraction to sensor raw data. In this work, we aim to distinguish the four road types freeway, highway, parking area and urban environment. We define a freeway, in contrast to a highway, as a road with a structural separation to oncoming traffic and without vulnerable road users. Several

variants of a classification algorithm are presented, where the four classes are solely recognized from fused occupancy grids, without the use of video or navigation map data. In particular, we treat the grids like images and are thus able to directly apply recent Convolutional Neural Network (CNN) topologies. With the expectation that CNNs would adapt to occupancy grids much better than state-of-the-art image based features, we compare them to a Support Vector Machine (SVM) classifier that is trained on hand-crafted features. The proposed approach is a first step towards occupancy grid based situation interpretation and will be extended with more situations in the future.

Prior work mainly focused on the classification of road types from camera image data. Sikiric et al. compared several image descriptors to classify typical traffic scenes [2]. Mioulet et al. used a Gabor filter bank to classify the current road type [3]. To classify the road environment, Tang et al. [4] extracted a set of color and edge based texture features in specific regions of the image. Contrary to the image based approaches, Taylor et al. [5] used data mining of vehicle signals to classify the road type. Hellbach et al. published an approach for indoor semantic labeling based on occupancy grids [6].

Section II briefly outlines occupancy grid mapping which is used as input for both, SVM and CNN classification approaches that are presented in Section III and Section IV, respectively. Classification results of all methods are presented in Section V which is followed by a conclusion and outlook in Section VI.

## II. OCCUPANCY GRID MAPPING

*Occupancy grid mapping* [7] is a well-established method for static obstacle fusion. The idea is to divide the environment into 2D cells, each containing the probability of being occupied. An inverse beam sensor model is used to model the uncertainty of obstacle measurements and to derive free space. This model assigns an occupancy probability to every cell that intersects the ray of a range measurement. In our processing pipeline the data of the resulting scan grids is then integrated over time into an accumulated grid for each sensor type and finally fused into a single grid. In the Dempster-Shafer-based grid fusion proposed by Pagac et al. [8], each grid cell can be occupied ( $O$ ) or free ( $F$ ). Formally, this frame of discernment  $\Theta$  is defined as  $\Theta = \{F, O\}$ . According to Dempster-Shafer theory, a mass of belief is assigned to every element of  $2^\Theta$  instead of just to singletons of  $\Theta$ , as in Bayesian occupancy grid fusion. Details about the fusion algorithm can be found in previous work [9].

<sup>1</sup>The authors are with the BMW Group, 80788 Munich, Germany {christoph.seeger, andre.a.mueller, loren.schwarz, michael.manz}@bmwgroup.com

### III. CLASSIFICATION WITH SVMs

To differentiate between occupancy grids representing freeways, highways, parking areas and urban environments, we compared several state-of-the-art global image features to train an SVM.

#### A. Feature Extraction

Because the classification task we consider is closely related to image categorization, we selected features that were successfully applied in this domain. Only the color channel with the occupancy probability mass was used as input of all feature extraction methods.

1) *PCA: Principal Component Analysis* aims to transform the data into linearly uncorrelated components and reduce the dimensionality of a feature vector. In our case the PCA was directly applied to the grid images in the training set.

2) *CENTRIST: Census Transform Histogram* features mainly represent the structural properties of an image by computing edge patterns of local image patches [10].

3) *Gist*: The *Gist* descriptor was designed to capture the *Spatial Envelope* of a scene that the authors described with naturalness, openness, roughness, expansion, and ruggedness [11].

4) *PHOG: Pyramid Histogram of Gradients* features [12] are an extension of the well-known histogram of gradients (HOG) features with a spatial encoding of the feature position using an image pyramid.

#### B. Classification

We use binary SVM classifiers [13] that were extended to the multi-class setting using error correcting codes [14]. A one-vs-one scheme that builds separate pairwise classifiers was used to encode the classes [15]. In total, this results in  $K \cdot (K - 1) / 2$  binary classifiers, where  $K = 4$  is the number of classes. The final result is then the class that minimizes the total loss of all binary classifiers.

### IV. CLASSIFICATION WITH CNNs

In contrast to the two-stage approach of the previous section, CNNs [16] are able to provide both, automatic feature extraction and the ability to discern between multiple classes. This property has made CNNs a popular choice for image classification and speech recognition tasks. As previously, we interpret fused occupancy grids as images, with the color values encoding the occupancy and free space probability of each cell.

Employing a CNN-based classifier is motivated by the observation that occupancy grids contain information that hand-crafted feature extractors may fail to capture. Furthermore, a CNN can hierarchically represent the interrelation of simple features, which may result in higher-order concepts, such as *tree* or *reflector post* that, in turn, facilitate the discrimination of road types.

We evaluated various network topologies from literature (*AlexNet* [16], *GoogLeNet* [17], *VGG16* [18]), as well as a self-designed network. Each network features an output layer with four neurons (for the four road types). In contrast to the

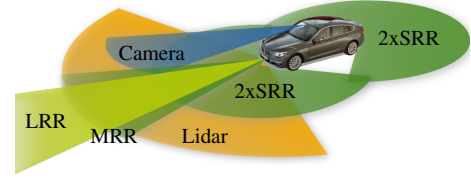


Fig. 1. Sensor setup of our test car. The lidar and the full-range front radar, with long-range (LRR) and mid-range (MRR) mode, are mounted in the front bumper. The stereo camera is mounted behind the windshield and the four short-range radars (SRR) are mounted in the corner of the cars, behind the bumper.

network topologies from literature, our network structure was designed to model small features, such as single occupied cells, and simultaneously to provide a means to describe the connection of features at a very high level. To this end, we chose a small kernel size of  $8 \times 8$  at the first convolutional layer, combined with a network depth of nine layers. The first six layers contain a combination of convolutions, rectangular linear units (ReLU), pooling and response normalization, each. The first layer comprises 48 kernels and layer two to six 256 kernels. The final two layers are fully connected and contain 4092 and 2048 neurons, respectively. We obtain a comparatively small number of neurons per layer and thus a relatively low computational cost that matches the four-class classification task at hand.

### V. RESULTS

#### A. Occupancy Grid Data Acquisition

The sensor setup of our test car is shown in Fig. 1. For all experiments, the grid size is  $100 \times 100$  m with 0.1 m cell resolution. The ego vehicle is centered in the grid and all grids were rotated, such that the car points upwards. All grid images were resized to  $256 \times 256$  pixels to speed up computation. The data was mostly acquired at daytime with sunny, cloudy, rainy and snowy weather. Some of the data was also acquired at night. To maximize the variation within the dataset, the original stream of grid data was subsampled every 10 m (parking area), 20 m (urban), and 40 m (freeway and highway). In total,  $\approx 700$  manually labelled grids of each category were used for training and  $\approx 150$  for testing the classifiers. Example occupancy grids and reference images are shown in Fig. 2.

#### B. Implementation and Parameters

Feature extraction and SVM classification was implemented in Matlab with the built-in PCA and SVM functions. Furthermore, open source implementations of the feature extraction methods *Gist*<sup>1</sup>, *PHOG*<sup>2</sup> and *CENTRIST*<sup>3</sup> were used. All features were normalized to have zero mean and unit variance. Sequential Minimal Optimization with a soft margin parameter of 0.01 was used to train the SVM. The parameters of the feature extraction methods are shown in Table I.

<sup>1</sup><https://github.com/adikhosla/feature-extraction>

<sup>2</sup><http://www.robots.ox.ac.uk/~vgg/research/caltech/phog.html>

<sup>3</sup><https://github.com/sometimesfood/spact-matlab>

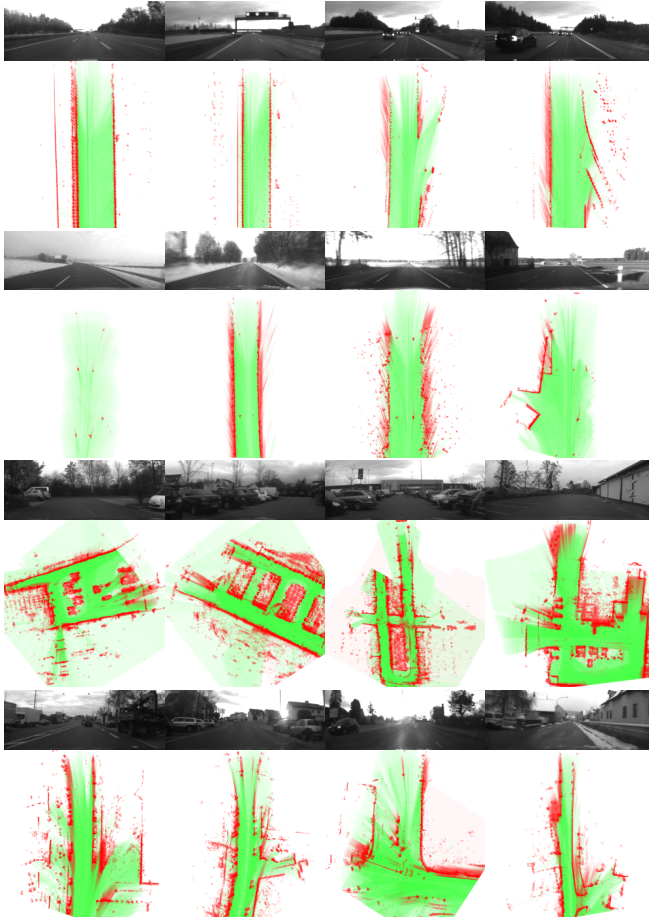


Fig. 2. Example occupancy grids of the four categories freeway (first row), highway (second row), parking area (third row), and urban (fourth row). Note the different weather conditions during data acquisition.

Method	Dimension	Remarks
PCA	100	PCA subspace computed on training set, test set only projected.
CENTRIST	256	Histogram of 8 bit census transform.
Gist	512	Gabor filter bank with 4 scales and 8 orientations, $4 \times 4$ equally sized blocks.
PHOG	680	3 pyramid levels, 8 angular bins in the range $[0, 360^\circ]$ .

TABLE I  
PARAMETERS OF THE FEATURE EXTRACTION METHODS.

The CNNs were modelled and trained in Caffe [19]. The training data was fed to each network to either train it from scratch with randomly initialized weights or fine-tune the upper layers of a network that was pre-trained on ImageNet data. Due to the rather small training data set ( $\approx 700$  per class), training was carried out using the AdaDelta optimizer [20], which tends to cope better with small training sets, as compared to Stochastic Gradient Descent. The learning rate  $\epsilon$  was chosen as 0.01 in case of a random initialized network and  $\epsilon = 0.001$  in case of fine-tuning. In both cases, a batch size of 50 was used and momentum  $\rho$  was set to 0.9.

Method	Accuracy
PCA	0.80
CENTRIST	0.88
Gist	0.90
PHOG	0.91
PHOG + Gist	0.94
AlexNet (fine-tuning)	0.94
GoogLeNet (fine-tuning)	0.93
VGG16 (fine-tuning)	0.93
AlexNet (from scratch)	0.88
Our topology (from scratch)	0.89

TABLE II  
COMPARISON OF THE TEST ACCURACY.









				
	0.94	0.013	0.0	0.047
	0.006	0.983	0.0	0.012
	0.0	0.0	0.919	0.081
	0.0	0.046	0.028	0.926

TABLE III  
CONFUSION MATRIX FOR GIST + PHOG WITH SVM CLASSIFIER.

### C. Quantitative Evaluation

For a general comparison of the proposed methods, the test accuracy of the learned classifiers was evaluated. Results are given in Table II. Both, SVM-based as well as CNN-based variants were able to classify 94% of the test data correctly. The results show that a feature-level fusion of Gist and PHOG features yields the best results of the SVM-based approaches. Other feature-level fusion approaches did not improve classification accuracy and are thus not included in the table. Fine-tuned CNN-topologies, even though they were pre-trained on the unrelated ImageNet data, outperformed randomly initialized networks, which can be largely attributed to the comparatively small set of training examples. Typically, a training set with  $\approx 10,000$  samples per class is required for randomly initialized networks. In the set of networks that were trained from scratch, the self-designed network performed better, by a small margin, than its competitors. However, due to the relatively small amount of training images, the increased performance cannot clearly be attributed to the altered network structure.

While the general performance of the best SVM-based and CNN-based classification methods seems on par, the confusion matrices of Tables III and IV reveal slight differences in their per-class performance. The classes *freeway*, *highway*, and *parking area* have a higher detection accuracy with the CNN-based approach than with the SVM-based approach. In contrast, the SVM-based method provides a higher accuracy with respect to class *urban*. Interestingly enough, the CNN was able to provide such a high accuracy for class *highway*, which implies that our test set might still be too small. Fig. 3 shows that for some occupancy grids it is even hard for humans to distinguish between the four road









				
	0.98	0.013	0.0	0.007
	0.0	1.0	0.0	0.0
	0.0	0.0	0.957	0.043
	0.04	0.091	0.034	0.835

TABLE IV  
CONFUSION MATRIX FOR PRE-TRAINED ALEXNET.

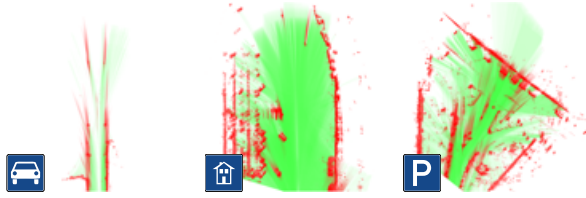


Fig. 3. Examples of false classifications with pre-trained AlexNet. Ground truth labels are urban (left), parking (middle) and urban (right).

types solely based on occupancy grids which emphasizes the good performance of the trained classifiers.

## VI. CONCLUSION AND OUTLOOK

We have presented different methods for road type classification based on fused occupancy grids. Our guiding idea was to approach the problem both, from a machine learning perspective with clearly separated feature extraction and classification and from the deep learning side. While our expectation was that the CNN-based classifier would outperform its SVM-based counterpart, it turned out that a reasonable choice of hand-crafted features can bring an SVM classifier on par with a deep neural network. Apparently, typical image features can be successfully used directly on occupancy grids. However, as occupancy grids do not have exactly the same properties as photographic images, a lot of experiments with feature extractors were necessary to achieve competitive results. Of course, the feature extraction step becomes superfluous when using a CNN, but finding suitable network topologies and optimization parameters also does not come for free. We found that using pre-trained CNNs improved classification rates significantly when compared to training from scratch, even though these networks had undergone pre-training on photographic images from ImageNet. Very likely, the size of our training data set was a limiting factor for the CNNs to exert their potential. Increasing the number of training samples and also the number of considered classes is the immediate next step on our agenda. Candidate situations to recognize from grid images include construction sites, freeway entrances and exits, traffic jam ends, accidents and road blockage.

Future work will furthermore focus on exploiting the time-series properties of a stream of occupancy grids. Clearly, information is lost when classifying each subsequent grid independently and ignoring previous classifications. This can be overcome by modeling class transition probabilities with

a Hidden Markov Model. Another promising improvement in this direction would be to build upon a Recurrent Neural Network. We also plan to add the video camera images to the classification and compare feature-level and classifier-level fusion. Furthermore, the topology of the CNNs needs to be further pruned to better match the sparsity of occupancy grids and, in consequence, also to reduce computational costs.

## REFERENCES

- [1] O. Marques, E. Barenholtz, and V. Charvillat, "Context modeling in computer vision: techniques, implications, and applications," *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 303–339, 2011.
- [2] I. Sikirić, K. Brkić, J. Krapac, and S. Šegvić, "Image representations on a budget: Traffic scene classification in a restricted bandwidth scenario," in *Intelligent Vehicles Symposium*. IEEE, June 2014, pp. 845–852.
- [3] L. Mioulet, T. P. Breckon, A. Mouton, H. Liang, and T. Morie, "Gabor features for real-time road environment classification," in *International Conference on Industrial Technology*. IEEE, February 2013, pp. 1117–1121.
- [4] I. Tang and T. P. Breckon, "Automatic road environment classification," *Transactions on Intelligent Transportation Systems*, vol. 12, no. 2, pp. 476–484, June 2011.
- [5] P. Taylor, S. S. Anand, N. Griffiths, F. Adamu-Fika, A. Dunoyer, and T. Popham, "Road type classification through data mining," in *International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. ACM, 2012, pp. 233–240.
- [6] S. Hellbach, M. Himstedt, F. Bahrmann, M. Riedel, T. Villmann, and H.-J. Böhme, "Find rooms for improvement: Towards semi-automatic labeling of occupancy grid maps," in *International Conference on Neural Information Processing*. Springer, November 2014, pp. 543–552.
- [7] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.
- [8] D. Pagac, E. Nebot, and H. Durrant-Whyte, "An evidential approach to map-building for autonomous vehicles," *Transactions on Robotics and Automation*, vol. 14, no. 4, pp. 623–629, 1998.
- [9] C. Seeger, M. Manz, P. Matters, and J. Hornegger, "Locally adaptive discounting in multi sensor occupancy grid fusion," in *Intelligent Vehicles Symposium*. IEEE, June 2016, pp. 266–271.
- [10] J. Wu and J. M. Rehg, "Centrist: A visual descriptor for scene categorization," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489–1501, August 2011.
- [11] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [12] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *International Conference on Image and Video Retrieval*, ser. CIVR. ACM, 2007, pp. 401–408.
- [13] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [14] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of artificial intelligence research*, pp. 263–286, 1995.
- [15] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, March 2002.
- [16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc., 2012, pp. 1097–1105.
- [17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.
- [18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Conference on Computer Vision and Pattern Recognition*. IEEE, June 2015, pp. 1–9.
- [19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *International Conference on Multimedia*. ACM, November 2014, pp. 675–678.
- [20] M. D. Zeiler, "Adadelta: An adaptive learning rate method," *CoRR*, vol. abs/1212.5701, December 2012.