Resource Efficient Classification of Road Conditions through CNN Pruning [⋆]

Daniel Fink [*] **Alexander Busch** [*] **Mark Wielitzka** [*]
Tobias Ortmaier [*]

[*] *Institute of Mechatronic Systems*
Leibniz University Hanover, Germany
(e-mail: daniel.fink@imes.uni-hannover.de).

Abstract: Towards autonomous driving, advanced driver assistance systems increasingly undertake basic driving tasks by replacing human assessment and interactions, when controlling the vehicle. The performance of these systems is directly related to knowledge of the vehicle's state and influential parameters. In this respect, the road condition has a major influence on the tires' traction and thus significantly affects the behavior of the vehicle. Therefore, a prediction of the upcoming road condition can improve the performance of the assistance systems which leads to an increased driving safety and comfort. The presented work aims to classify the road surface as well as its weather-related condition, based on images of the front camera view, using deep convolutional neural networks. In order to take computational limitations of vehicle control units into account, a pruning approach is investigated to reduce the network complexity.

Keywords: computer vision, road condition, classification, neural networks, pruning.

1. INTRODUCTION

Ensuring a high level of driving safety and comfort, driving systems in highly automated vehicles prospectively need to undertake human assessment abilities, besides their fundamental task of vehicle guidance. Regarding the driving safety, a key human ability is to assess the current road condition, in order to adapt the way of driving. This behavior needs to be transferred and applied to autonomous vehicles. Therefore, in upcoming driving systems the current road condition must be detected. In addition to this future application, information about the road condition can be utilized in already existing vehicles e.g. to warn a human driver of slippery road sections.

Furthermore, information about the road condition can be utilized to improve the performance of contemporary advanced driver assistance systems (ADAS). Most of these systems rely on knowledge of the vehicle's state and influential parameters. In this regard, the friction coefficient is of particular importance, as it mainly affects the interaction of tires and road, which significantly influences the vehicle's behavior. Based on the friction coefficient the control parameters of ADAS can be adapted. For example, when the friction is low, to set the collision avoidance system to break earlier and to increase the distance to a vehicle ahead, if using adaptive cruise control. The friction coefficient is affected by type and condition of tires as well as by the road surface including any intermediate medium. Therefore, information about the road condition contribute to increase the accuracy, when estimating the friction coefficient, which improves the performance of the ADAS.

A common approach to estimate the friction coefficient is to fuse information gained from heterogeneous sources, such as driving dynamics, vehicle front camera, and weather information (see Jarisa (2016)). Under sufficient excitation, driving dynamics provide well suited data to estimate the physical value of the friction coefficient (see Wielitzka et al. (2017)). However, this estimation only provides information for the current road section. In order to predict the friction coefficient for the upcoming road section, even for insufficient excitation, images of the vehicle's front camera can be processed additionally to detect the road condition. Based on this information, a value range for the friction coefficient can be determined. For example, when a predicting a snowy road, Raste et al. (2019) assume the friction coefficient to lie within the range of 0.2 and 0.45.

Regarding image based predictions of the road condition, Roychowdhury et al. (2018) compare a feature extraction method with a classification approach relying on a convolutional neural network (CNN). Distinguishing between a dry, wet, snowy, and icy asphalt road, it is shown that the CNN-based approach achieves a higher prediction accuracy. Besides the weather-related condition, the underlying road surface has a major impact on the wheel's traction. In addition to snowy roads and wet or dry asphalt surfaces, Nolte et al. (2018) consider cobblestone roads in their CNN-based classification of the road condition. However, the weather condition on cobblestone roads is not detected. Busch et al. (2019) compare CNN-structures of different sizes and architectures regarding their ability to classify asphalt and cobblestone road surfaces as well as their current weather-related condition, such as dry, wet, and snowy. The results show that even one of the smallest common pre-trained network structures, *SqueezeNet* (see Iandola

[⋆] The authors would like to thank the German Research Foundation (DFG) for founding this project.

et al. (2016)), is able to achieve a prediction accuracy of 92.8%, when providing input images by rectifying a trapezoidal section extracted from the front camera view. However, it is concluded that a classification approach, based on a state of the art CNN-structure, is still suboptimal in terms of computational effort.

Since computing resources are strongly limited in automotive applications, the computational costs of the classification are to be considered, to provide the ADAS with beneficial information in real-time. In this paper a pruning method is investigated to reduce the complexity of a *SqueezeNet*-based road condition classification approach. The aim is to minimize the number of computing operations without causing a significant decrease of prediction accuracy.

The paper is organized as follows. In section 2 the used road condition image dataset is presented. In section 3 the *SqueezeNet*-training is described before the pruning procedure is introduced and applied in section 4. Section 5 evaluates the resulting network structures and finally a conclusion is given in section 6.

2. DATASET

Extracting environmental information from the vehicle's front camera is a key challenge regarding automated driving systems. Hence, there are several computer vision datasets showing the front window view of a vehicle in various traffic situations. As there are no specific road condition datasets available, an individual dataset, suitable for this task, is created.

2.1 Composition of image data

One part of the dataset consists of images extracted from public vision benchmark datasets for automotive tasks, such as *BDD100K* (by Yu et al. (2018)), *KITTI* (by Geiger et al. (2013)), *Oxford Robotcar* (by Maddern et al. (2017)), and *Cityscapes* (by Cordts et al. (2016)). Since these datasets are mainly recorded under dry conditions, royalty free as well as self recorded front camera image material is added, to extend underrepresented classes, such as wet cobblestone and snow. This allows to generate a dataset consisting of 26,109 images that are divided into the five road condition classes: asphalt dry (AD), asphalt wet (AW), cobblestone dry (CD), cobblestone wet (CW), and snow (S). When collecting the image data, a wide variety of light conditions is considered for each class. Table 1 indicates the number of images used from each data source as well as their subdivision into the classes. The used image data sources mainly provide image sequences from video recordings that are kept together when splitting the dataset into 70% training, 20% validation, and 10% test images. Due to different frame rates in the available data sources, an individual amount of images is taken into account for each video sequence.

Table 1. Sources and distribution of dataset

Data source	**AD**	**AW**	**CD**	**CW**	**S**
KITTI	1450	-	37	-	-
Oxford Robotcar	1005	1103	-	-	-
Cityscapes	5492	29	331	-	-
BDD100K	2476	1241	47	1	88
Open source data	-	9	-	-	1393
Self recorded	1502	2229	3004	4457	215
Total	11925	4611	3419	4458	1696

2.2 Region of Interest

Vehicle front window view images show a wide area of environment besides the road. The weather condition might be detected better in this environmental area. However, a classifier could mistakenly learn to recognize similar looking areas without considering the road itself. Therefore, only a rectified, trapezoidal region of interest (ROI), showing the road surface, is used for the dataset. Busch et al. (2019) established this type of ROI to be particularly suitable for the road classification task. Fig. 1 illustrates the extraction process and shows an example image for each road condition class. Due to varying camera positions in the used data sources, the trapezoidal ROI have individual locations and shapes.

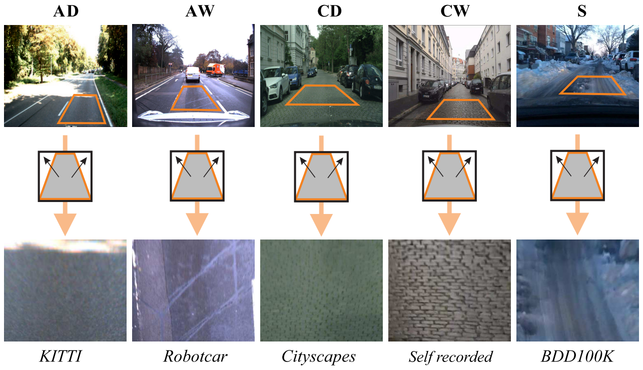

Fig. 1. Example images and extraction of the ROI

3. TRAINING PROCESS

Comparing many state-of-the-art CNNs, Busch et al. (2019) determine a *SqueezeNet*-based network structure to be well suited, in terms of accuracy, for road condition classification tasks. This section introduces the training process of a *SqueezeNet*-structure based on the road condition dataset described above.

The network structure is initialized with pre-trained weights from the *ImageNet* dataset (see Russakovsky et al. (2015)), while the last convolutional layer is replaced for the new classification task. Considering unbalanced class sizes, a weighted cross-entropy loss, given by

$$\mathcal{L} = -\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{K} w_i Y_{j,i}\log(\hat{Y}_{j,i})\,, \tag{1}$$

is used and minimized by stochastic gradient decent during the network training. Here the vector $\hat{\boldsymbol{Y}}_j$ represents the predicted classes, while the actual class of an image j is given trough $\boldsymbol{Y}_j$. N is the number of images and K the number of classes. The weighting vector $\boldsymbol{w}$, of length K, is calculated depending on the number of images in the corresponding classes.

Within layers with pre-trained weights the learning rate is set to 10^{-3}, while the new substituted classification unit is trained using a learning rate of 10^{-2}. Hence, untrained weights can be adapted faster, while the pre-trained layers are more protected. After each training epoch the learning rates are reduced by 10%. The batch size is set to 16 images that are randomly rotated by ±10°, horizontally flipped, and scaled by a random factor between 1 and 1.3, in order to augment the camera perspectives occurring in the dataset. Additionally, batch normalization is applied. Fig. 2 visualizes the training process. Using the F_1-score to evaluate the achieved accuracy for the training, validation, and test images, allows to consider the precision as well as the recall of the image classification. As the F_1-score is weighted according to the class sizes, misclassifying an image of an underrepresented class causes a higher decrease of accuracy (see Goutte and Gaussier (2005)). Afters 10 epochs of training a minimum validation loss is detected, while the validation F_1-score equals 89.5%. Representing the training result, this *SqueezeNet*-structure achieves a F_1-score of 92.84% when classifying the test data.

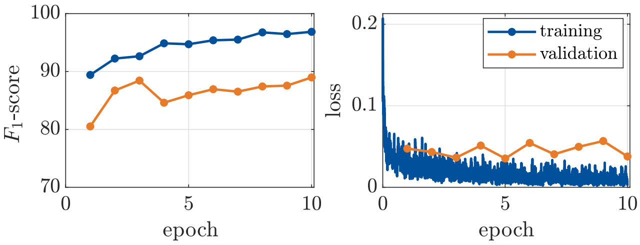

Fig. 2. Training process of the *SqueezeNet*-structure

4. NETWORK PRUNING

As shown above the trained *SqueezeNet* is well suited to classify the road condition. However, since this type of network structure is primary used for classification tasks with much higher complexities, it may not be optimal regarding the required computational resources. The aim is to reduce the computational complexity of this network structure without significantly decreasing its prediction accuracy. For this purpose, a network pruning approach, presented by Molchanov et al. (2016), is modified and applied to the trained *SqueezeNet*-structure.

4.1 Pruning procedure

The prediction runtime of image processing network structures is dominated by convolutional operations that result in feature maps. Aiming to speed up the prediction, a pruning approach is used that relies on reducing the number of generated feature maps by removing corresponding filter kernels. In order to maintain a high prediction accuracy, only feature maps with low influences on the classification result are removed. A single pruning step is completed by retraining the remaining network structure to compensate for resulting decreases in the prediction accuracy. This procedure is iterated up to a certain stopping criterion, such as dropping below a required prediction accuracy or achieving the targeted computational complexity. Fig. 3 illustrates the overall process of the pruning algorithm.

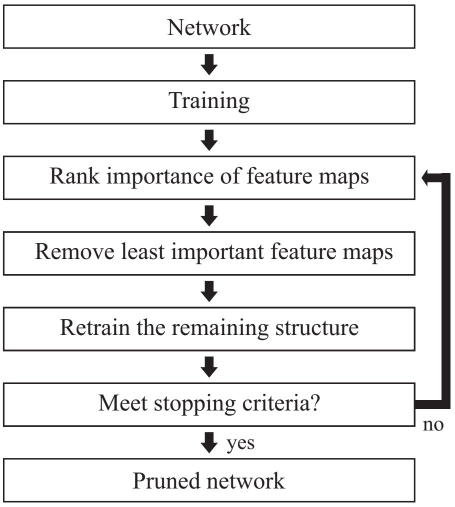
Network
Training
Rank importance of feature maps
Remove least important feature maps
Retrain the remaining structure
Meet stopping criteria?
no
yes
Pruned network

Fig. 3. Pruning procedure

4.2 Evaluation of feature maps

The key part of the pruning algorithm is to evaluate the importance of the network's feature maps. A common approach is to rank the importance of feature maps based on their influence on the loss value. In this regard, a deviation from the initial loss value $\mathcal{L}(\mathcal{D})$, due to the exclusion of a single feature map h_k, is defined as

$$\Delta\mathcal{L}(\mathcal{D}, h_k) = |\mathcal{L}(\mathcal{D}, h_k = 0) - \mathcal{L}(\mathcal{D})| \,, \tag{2}$$

where $\mathcal{L}(\mathcal{D}, h_k = 0)$ represents the loss value assuming h_k is excluded when classifying the training data $\mathcal{D}$. In order to rank all feature maps of a network structure, the difference equation (2) is to be set up for the exclusion of every single feature map. Since within the *SqueezeNet*-structure a total of 2624 feature maps are generated, this causes to much computational effort for a single pruning iteration. Therefore, an evaluation approach, introduced by Molchanov et al. (2016), is used that is based on the following Taylor expansion of $\mathcal{L}(\mathcal{D}, h_k = 0)$:

$$\mathcal{L}(\mathcal{D}, h_k = 0) \approx \mathcal{L}(\mathcal{D}) - \frac{\partial \mathcal{L}}{\partial h_k} h_k \,. \tag{3}$$

Substituting (3) in (2) enables a direct approximation of $\Delta\mathcal{L}(\mathcal{D}, h_k)$ as follows:

$$\Delta\mathcal{L}(\mathcal{D}, h_k) \approx \left| \frac{\partial \mathcal{L}}{\partial h_k} h_k \right| . \tag{4}$$

When using a gradient based optimizer, in every layer a gradient of the back propagated loss is taken once with respect to every single filter, in order to adapt its weights. Based on a filter relying gradient, the gradient of $\mathcal{L}$ with respect to the corresponding feature map can be derived. Therefore, approximating the loss-influence of every single feature map requires only one forward and one backward propagation of the training data.

Pruning an *AlexNet*- as well as a *VGG-16*-structure, Molchanov et al. (2016) compare different pruning criteria and find the Taylor expansion approach to cause a minimum decrease in prediction accuracy, when removing only the least important feature map per pruning iteration.

4.3 Adaption of the pruning step width

Since every pruning step is followed by a training process as described above, pruning the majority of a network's feature maps still requires high computational effort. In order to save computing time, the presented pruning procedure is adapted by removing multiple feature maps within a single iteration step. Regarding the step width, two approaches are compared as described below.

An obvious adaption of the pruning step width is to remove a fixed absolute number of the least important feature maps within each pruning iteration. However, deleting a constant number of feature maps increases the ratio of removed feature maps in relation to the total number of feature maps within the remaining structure. Over the course of the pruning procedure, this may causes a decrease in the prediction accuracy that can not be compensated during the retraining process, due to a relatively high number of removed feature maps. Therefore, additionally a relative pruning step width is investigated, that relates to the total number of feature maps in the remaining network structure.

4.4 Pruning of the SqueezeNet-structure

The presented pruning procedure is applied using the complete training data (18,626 images) to rank the feature maps as well as to retrain the reduced structure in every iteration. There are several options of selecting absolute and relative step width values. As a first introduction of step size adaptation, pruning is applied exemplary with an absolute step size of 32 and a relative step size of 5% pruned feature maps per iteration. Both step size variants allow to reduce the *SqueezeNet*-structure to a single feature map per convolutional layer within less than 24 hours using only a single GPU (NVIDIA GeForce GTX 780 Ti).

5. RESULTS

In this section the pruning results are presented and the two approaches of adapting the step sizes are compared. Finally, the road condition classification results using the initial, unpruned *SqueezeNet*-structure are opposed.

Fig. 4 illustrates the pruning processes for both types of step size adaption. It is demonstrated how removing feature maps influences the prediction accuracy regarding the training and validation data. Additionally, the computational requirements of the reduced network structures are displayed. These requirements are determined by measuring the number of floating point operations (FLOPs) executed for a single image classification. Regardless of the selected step size, it can be demonstrated that the validation F_1-score increases within the first pruning iteration steps. This suggests that not only the weights, but also the networks structure itself is optimized for the classification task. As a consequence, there are two possible types of network structures to be obtained from a pruning process. First, the initially targeted structure that requires minimal computational effort, while providing a similar prediction accuracy (①,②). Second, a pruned structure that profits from the increasing prediction accuracy due to the removal of unimportant feature maps (③,④). The structure with optimal computation time is defined to be the last one achieving a validation F_1-score nearly as high as the unpruned network. The structure, additionally meeting optimal accuracy requirements, is selected to be the last one differing not more than 1% from the maximum validation F_1-score achieved during the pruning procedure.

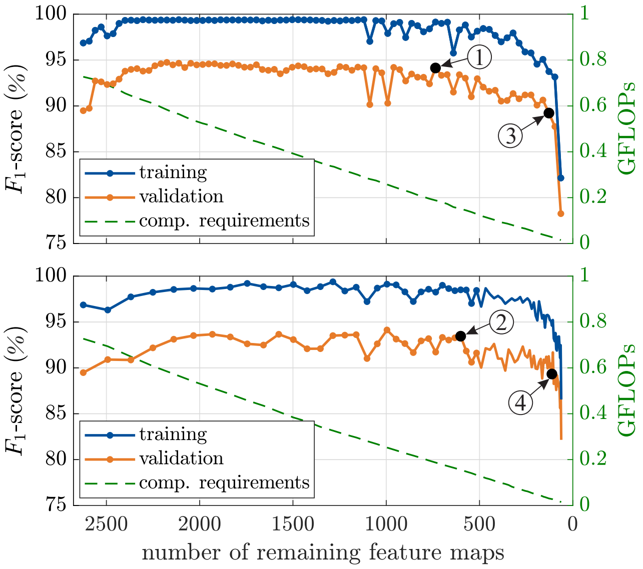

Fig. 4. Pruning results for an absolute step width of 32 (top) and a relative step width of 5% (bottom) removed filter maps per iteration

Comparing the variations of step size adaption, the absolute step width shows a steeper increase of the F_1-score and builds a plateau at about 95%. However, using the relative step width leads to similar accuracy achievements, but requires 30 fewer iteration steps. Crucial characteristics of the pruned network structures are summarized in Table 2 and contrasted to the initial *SqueezeNet*. Here, the number of remaining feature maps (fm), the achieved validation F_1-score ($F_{1,\mathrm{val}}$), and the number of executed pruning iterations (p) are compared. In addition to the required number of computing operations per image classification (in GFLOPs), the reduction of computational requirements is given related to the initial *SqueezeNet* in percent.

Table 2. Characteristics of pruned structures

Structure	**p**	**fm**	$\mathbf{F_{1,val}}$	**comp. requirements**	
			(%)	GFLOPs	reduced (%)
SqueezeNet	0	2624	89.5	0.727	0
①	78	128	89.2	0.032	95.6
②	64	112	89.2	0.028	96.1
③	60	736	94.0	0.188	74.1
④	30	601	93.5	0.155	78.7

Regarding the validation F_1-score, it is demonstrated that pruning can increase the prediction accuracy by about 4%, while simultaneously up to 78.7% of the required computational effort can be saved. Accepting a slight accuracy decrease of 0.3%, pruning even allows to reduce more than 95% of the initial computational effort.

The assessment of the pruning results relies on the F_1-score achieved classifying the validation dataset. In order to evaluate a generalized performances of the pruned network structures, regarding the road condition classification task, the test dataset is finally classified with each structure. Based on that, confusion matrices are set up as illustrated in Fig. 5.

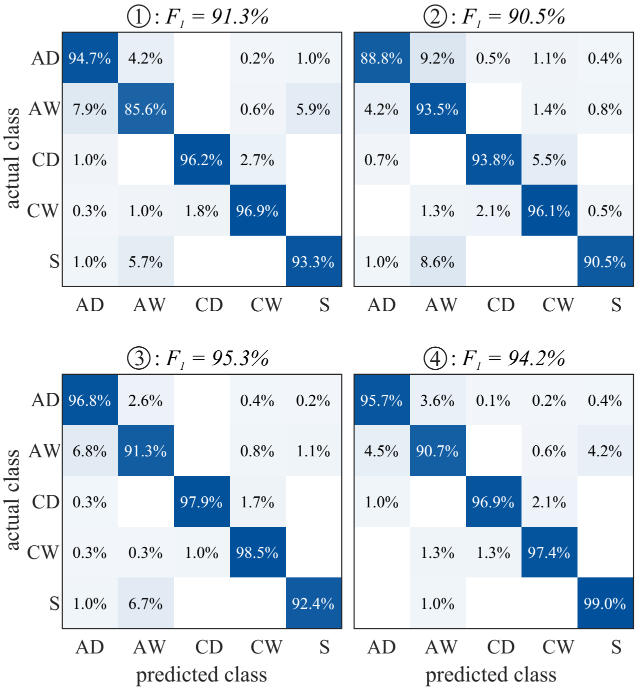

Fig. 5. Confusion matrices on test dataset for the pruned network structures ① − ④

To contrast the performances of the pruned structures to the initial road condition classification, Fig. 6 shows the confusion matrix of a test data classification using the unpruned *SqueezeNet*-structure.

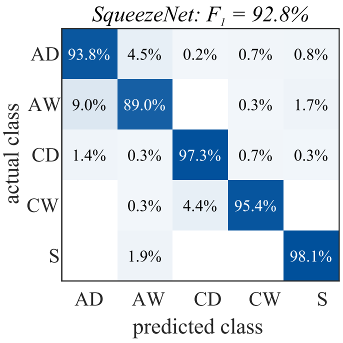

Fig. 6. Confusion matrix on test dataset using the initial *SqueezeNet*-structure

A Comparison of the test classification results confirms that the pruned network structures are still well suited to predict the road condition. However, the pruned structures ①, ②, and ③ have slight deficits in distinguishing between snow and wet asphalt.

6. CONCLUSION

In this paper a CNN-based, resource efficient classification approach is presented, that purposes to provide the ADAS with predictive information about the upcoming road condition. Proceeding from a pre-trained *SqueezeNet*, a pruning method is applied to reduce the computational requirements of the network structure. This method relies on removing filter maps causing a minor impact on the classification result. In order to speed up the pruning procedure, two variants to adapt the step size are investigated.

As a result of pruning the network structure, it is shown that the prediction accuracy can be increased, while the number of required FLOPs for a single image classification decreases. Compared to the initial *SqueezeNet*, the pruned network structure performs with a 1.4% higher prediction accuracy in the road condition classification task. Moreover, with 155 million FLOPs it requires nearly five times less computational effort.

In addition, pruning is continued until dropping below the initial validation accuracy. Thus, a structure is obtained requiring only 32 million FLOPs, which is nearly 23 times less, compared to the initial *SqueezeNet*-structure. However, this structure underperforms the initial prediction accuracy of 92.8% by 1.5% on test data.

Regarding available networks, pre-trained on *ImageNet*-data, *SqueezeNet* belongs already to the smallest ones. Therefore, pruning is concluded to be a valuable method to find a network that meets both, optimal prediction accuracy and computational requirements. Regarding the road condition classification task, a resource efficient network structure can be obtained, that still profits from a enormously time-consuming pre-training process.

However, the pruning procedure of the *SqueezeNet*-structure is to be further assessed. As a next step, it will be repeated several times, in order to consider stochastic influences in this process and to cross-validate the pruning results. Additionally, alternative parameter setting and variants of step size adaption for the pruning procedure are investigated. Furthermore, future work addresses training and pruning of similar sized, pre-trained networks based on the same road condition dataset.

REFERENCES

Busch, A., Fink, D., Laves, M.H., Ziaukas, Z., Wielitzka, W., and Ortmaier, T. (2019). Classification of road surface and weather-related condition using deep convolutional neural networks. In *Proceedings of the 26th IAVSD Symposium on Dynamics of Vehicles on Roads and Tracks*. Accepted.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding.

Geiger, A., Lenz, P., and Stiller, C. (2013). Vision meets robotics: The kitti dataset. 32(11), 1231–1237.

Goutte, C. and Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *Proceedings of the 27th European Conference on Advances in Information Retrieval Research*, 345–359. Springer-Verlag, Berlin, Heidelberg.

Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., and Keutzer, K. (2016). Squeezenet: Alexnet-level accuracy with 50x fewer parameters and.

Jarisa, W. (2016). Future technology – road condition classification using information fusion. In *7th International Munich Chassis Symposium 2016*, 939–957. Springer Vieweg, 1st ed. edition.

Maddern, W., Pascoe, G., and Linegar, C. (2017). 1 year, 1000 km: The oxford robotcar dataset. 36(1), 3–15.

Molchanov, P., Tyree, S., Karras, T., Aila, T., and Kautz, J. (2016). Pruning convolutional neural networks for resource efficient inference.

Nolte, M., Kister, N., and Maurer, M. (2018). Assessment of deep convolutional neural networks for road surface classification.

Raste, T., Lauer, P., and Hartmann, B. (2019). Development of a road condition observer for the “vehicle motion control” project. In *XXXVII. Internationales Symposium 2018 Bremsen-Fachtagung*, volume 5 of *Proceedings*, 127–141. Springer Berlin Heidelberg.

Roychowdhury, S., Zhao, M., Wallin, A., Ohlsson, N., and Jonasson, M. (2018). Machine learning models for road surface and friction estimation using front-camera images. In *International Joint Conference on Neural Networks (IJCNN)*, 1–8.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., and Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. 115(3), 211–252.

Wielitzka, M., Dagen, M., and Ortmaier, T. (2017). State and maximum friction coefficient estimation in vehicle dynamics using ukf. In *American Control Conference (ACC)*, 4322–4327.

Yu, F., Xian, W., Chen, Y., Liu, F., Liao, M., Madhavan, V., and Darrell, T. (2018). Bdd100k: A diverse driving video database with scalable annotation tooling.