Using Mixture of Expert Models to Gain Insights into Semantic Segmentation

Svetlana Pavlitskaya*[1], Christian Hubschneider[1], Michael Weber[1], Ruby Moritz[2], Fabian Hüger[2], Peter Schlicht[2] and J. Marius Zöllner[1]

[1]FZI Research Center for Information Technology
[2]Volkswagen Group Automation

Abstract

Not only correct scene understanding, but also ability to understand the decision making process of neural networks is essential for safe autonomous driving. Current work mainly focuses on uncertainty measures, often based on Monte Carlo dropout, to gain at least some insight into a models confidence. We investigate a mixture of experts architecture to achieve additional interpretability while retaining comparable result quality.

By being able to use both the overall model output as well as retaining the possibility to take into account individual expert outputs, the agreement or disagreement between those individual outputs can be used to gain insights into the decision process. Expert networks are trained by splitting the input data into semantic subsets, e.g. *corresponding to different driving scenarios, to become experts in those domains. An additional gating network that is also trained on the same input data is consequently used to weight the output of individual experts. We evaluate this mixture of expert setup on the A2D2 dataset and achieve similar results to a baseline FRRN network trained on all available data, while getting additional information.*

1. Introduction

Redundancy concepts as well as explainability are essential to ensure reliability of machine learning models and to use those models in practice. This is particularly true for autonomous driving tasks, where reliable and traceable or explainable outputs are of utter importance. Indeed, there exist strict requirements within the approval process to be able to use machine learning models in a production vehicle, mainly being able to trace individual decisions and being able to reason about them. A first step towards being able to get reliable conclusions was to intrinsically measure uncertainties of machine learning models and deep neural networks in particular [10], both epistemic (or model) and aleatoric (sensor or data noise) uncertainties. Those uncertainty models usually rely on either Bayesian deep learning approximations (as in [10]), or rely on having neural networks output parameters of mixture models [2].

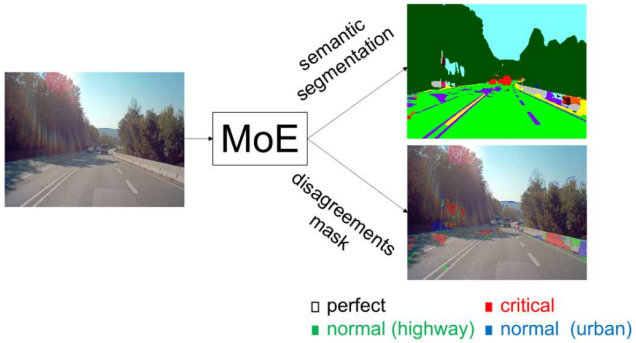

Figure 1. Overview of the mixture of expert (MoE) output and a visualization of different cases of agreement between experts and overall MoE output.

An alternative approach is to use combination techniques and architectures, that combine several (possibly redundant) submodules or -models. An exemplary approach to capture uncertainties has been proposed by [13] using deep ensembles to estimate epistemic uncertainties. One advantage of those combination architectures is the possibility to develop subfunctions and tasks independently and individually, and to combine them later in order to improve on security for individual functional components. Ensembling techniques and fusion architectures are currently already being used to combine outputs of different subcomponents or even several separate version of the same model, either trained on

*Corresponding author: pavlitskaya@fzi.de

different subsets of the whole dataset, or just with differing initialization. However, those techniques were mainly used to boost benchmark results or to combine several input modalities within one, larger model architecture.

In this work, we examine a further class of models, that can be seen as a parameterized generalization of ensembling techniques, so called *Mixture of Experts (MoE)* models. Furthermore, we try to derive the possibility to reuse the different outputs to get estimates about the certainty (or uncertainty) of an output, while also retaining the possibility to have access to alternative outputs (also cf. Figure 1).

Mixtures of experts were first proposed by Jacobs et al. in [9]. A MoE comprises several specialized models (*experts*), where each individual expert tries to approximate the target function on some subset of the input space. Possibilities to instead use subsets of the available class or label space for individual experts are discussed in the conclusion. A further component of a MoE architecture is the *gate*, a trainable component that selects which expert's output is best suited a particular input. In contrast to other approaches that combine several models (like ensembles, fusion, stacking, etc.), the decision-taking logic is explicitly realized within the gate, which itself has parameters that can be optimized during training. In fact, this gate can also be represented as a neural network and trained either separately or in combination with the experts.

Classical MoE approaches assume passing inputs in parallel through each of the experts as well as through the gate. In case both, the experts and the gate, are realized as neural networks, it is reasonable to use a shared feature extraction in early layers for each of the experts and the gate.

In this work, we used semantic image segmentation as an exemplary task, using deep neural networks (DNNs) both to implement the experts as well as the gate.

2. Related Work

Most current work on explainability and uncertainty estimation is based on the work by Yarin Gal on estimating Bayesian neural networks [5], using a technique commonly referred to as Monte Carlo dropout. [10] extended this work by applying Monte Carlo dropout to semantic segmentation tasks. Uncertainty measures as obtained by this Bayesian neural network approximation was consecutively further also used to weight individual task losses during training in multi-task settings [11].

One challenge for all approaches for uncertainty estimation is to adjust the estimated uncertainties to the statistical variance as measured on a test set. [8] is thus looking into different architectures and how well various uncertainty modeling approaches are calibrated, but on the task of end-to-end vehicle control. Other approaches are able to generate multiple plausible segmentation outputs to deal with multimodal outputs or ambiguous situations [12]. Such outputs could be further extended to deduce explainability from multiple output hypotheses. Finally, [3] are even evaluating different decision rules and their individual implications to draw conclusion from neural network output values (as given by a softmax layer) other than maximum a-posteriori.

Previous work on mixture of expert models mostly focuses on fusing inputs from different modalities. In this particular case an individual expert is trained per modality or input type. In [15] a CNN expert is chosen for each of the three modalities: appearance (RGB image), depth and motion (optical flow). The gate weights feature maps as extracted from all the experts, and weights them in order to get the best combination of these modalities with an overall goal of compensating for sensor noise and adapting to environment changes.

In [18], a mixture of experts for robust semantic segmentation is defined. The experts are trained separately on inputs from different modalities. They use the UpNet architecture (as defined in [19], an architecture similar to the U-Net architecture for semantic segmentation [17]) as a base architecture for their experts. Feature maps are extracted from the contracting part of each expert and are then passed through a single gate, which computes weights for each of the experts. The output of each expert is then weighted according to the gate's weights. Finally, a convolutional layer and a softmax layer follow. In their use case, they demonstrate that the mixture of experts achieves higher performance when compared to various other fusion approaches. For multi-sensorial input, the MoE can thus provide additional robustness: in case of sensor fusion if one sensor fails, then the gating network assigns higher weights to other sensors. The MoE in this case, however, is not used to achieve additional explainability and is not used as a real confidence measure.

A further popular usage of MoE is for part-whole relationships. In this case, each expert is assigned a subset of the problem space. [1] and [6] use mixture of expert architectures for fine-grained classification, where each expert is learned on a sub-category of objects. Eigen et al. take the concept of gated mixtures of experts another step further and introduce stacked MoE models to what they call *Deep Mixture of Experts* [4]. Further examples of MoE application include multi-task and multi-domain recognition [14].

3. Concept

For now, we focus on the evaluation of mixture of expert architectures applied to semantic image segmentation of traffic scenes. As a simplification and a first attempt, we utilize two experts trained on disjoint splits of the available input data according to some semantic concept, but otherwise with identical architectures. The same concept could also be extended to more expert networks, if required and if a further data split is appropriate.

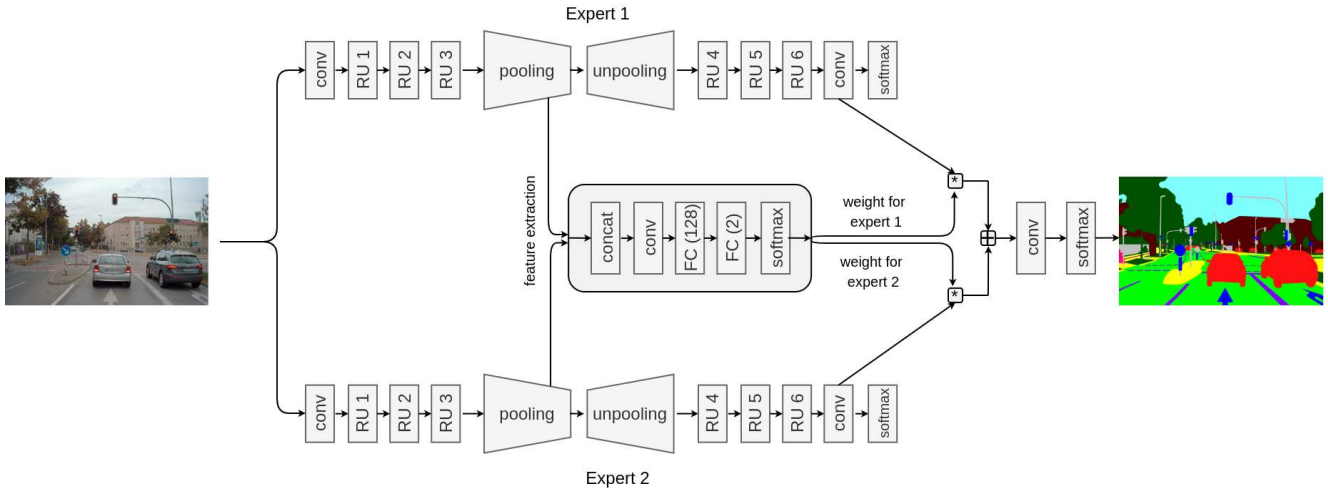

Figure 2. MoE Architecture for two experts with a simple gate. The rightmost convolutional layer is optional and we also evaluated MoE architectures without.

3.1. Experts

The experts used in this work build upon a ResNet-like architecture from [16], namely Full-Resolution Residual Network (FRRN). From the two proposed network designs (FRRN A and FRRN B), which differ in input image resolution and the size of intermediate feature volumes, we decided to use FRRN A as a basis, which is a shallower architecture. FRRN A (see Figure 3) consists of two streams: the pooling stream and the residual stream, whereas both of them are tightly connected to keep the information on the full image resolution via residuals. The pooling stream follows the encoder/decoder formulation and contains a number of full-resolution residual units (FRRUs). In the encoder part of the stream multiple max pooling operations are applied to shrink the feature maps, which are then gradually scaled up using unpooling architectures in the decoder part. For feature extraction, we consider FRRUs in the contracting part of the network: *FRRU_31* and *FRRU_42*. Since we extract features from the pooling layers of the corresponding FRRUs, we refer to the layers as *pool_31* and *pool_42*.

Each expert is capable of processing the whole input image and returning a label, as it outputs class values for each pixel individually. These outputs of both experts are then weighted by the weights as predicted by the gate, as is described in the following subsection.

3.2. Mixture of Experts Architecture

Our MoE architecture is inspired by the works of Valada et al. In [18], a single gating network is used, the combined feature maps are passed through an additional convolutional layer. In a further work [20] a class-wise gate is introduced, s.t. a separate gate is defined for each class. This way expert weights are predicted not for the whole image, but for each of the classes. Figure 2 shows the proposed MoE architecture. We have experimented with different gate architectures and it turned out that adding an additional convolutional layer after the combination of the weighted expert outputs, as done in the works by Valada et al., led to better performance only in case of a simple (*i.e.* not class-wise) gate. Our evaluation further shows that a single gate works better than a set of class-wise gates.

Furthermore, we have tried extracting feature maps from different layers of the experts as an input for the gate. The best results were achieved for the two layers from the contracting part of the network: *pool_31* with feature volume of size $60 \times 80 \times 16$ and *pool_42* with feature volume of size $30 \times 40 \times 16$. These layers have led to comparatively

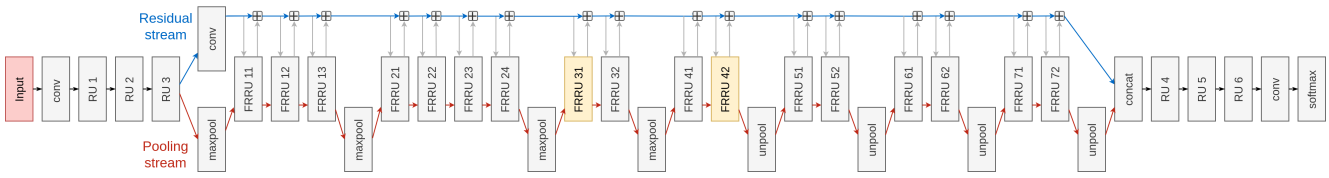

Figure 3. FRRN A architecture used as base for the expert networks. FRRN layers 31 and 42 are further evaluated as input feature layers for the gate networks.

Gate Architecture	Feature Layer	Highway	Ambiguous	Urban	Highway+Urban	Mixed
Simple gate	pool_31	**0.769**	0.633	0.710	0.763	0.753
	pool_42	0.768	0.633	0.712	0.764	0.754
Simple gate + conv	pool_31	0.687	0.583	0.632	0.675	0.665
	pool_42	0.763	0.640	**0.714**	**0.765**	**0.756**
Class-wise gate	pool_31	0.719	**0.649**	0.712	0.751	0.746
	pool_42	0.732	0.65	0. 712	0.758	0.732
Class-wise gate + conv	pool_31	0.742	0.645	0.707	0.754	0.747
	pool_42	0.762	0.641	0.658	0.710	0.703

Table 1. Mean IoU for different gate architectures and feature extraction layers (results for the manual data split by road type).

similar mean IoU values (see Table 1). Using raw images or feature maps from earlier layers (*e.g*. from *FRRU_11*) led to significantly worse results. Also, with the growing size of the feature volume, the gate training becomes more computationally expensive. In the following we report the results for the best architecture: features extracted from the *pool_42* layer, simple gate and an additional convolutional layer after the weighted expert outputs are combined.

3.3. Disagreements

In order to gain insights into decision process of the MoE, we defined the following classes of disagreements by comparing pixel-wise decisions of the experts and of the whole MoE:

- Perfect case: MoE agrees with all experts
- Normal case: MoE agrees with one expert
  - Normal case 1: expert 1 and MoE agree
  - Normal case 2: expert 2 and MoE agree
- Critical case: MoE does not agree with any expert, *i.e*. the class chosen was not proposed by any expert

4. Experiments

Experiments are performed on the semantic segmentation part of the A2D2 dataset [7]. These data splits (by road type and by sky-to-drivable ratio) are rather artificial, to create disjoint subsets of the available data, and the primary purpose was to evaluate the general concept of mixtures of experts. Real world applications of potential use cases for the described MoE architecture would instead be, for example, splits according to different countries with different road signs and differences in road layout, as well as in weather or lighting conditions.

4.1. Data Subsets

From a total of 31448 images from the front central camera, we have excluded 164 images with label "Blurred area" and 1 image with label "Rain dirt". Thus, a total of 31283 images were available for further evaluation.

First experiments used the original label set as specified. However, learning 38 labels turned out to be challenging and error prone for the MoE evaluation, so a mapping to 11 new labels was defined (see Table 2). Further experiments in this work rely on this new set of 11 labels.

Split by Road Type

To be able to learn experts on data subsets, we have manually labeled the records as belonging to one of the three classes: (1) urban with 15651 images, (2) highway with 8461 images and (3) ambiguous with 7171 images.

Based on this data split we train two experts: an urban and a highway expert. To ensure that each expert is trained and evaluated on the same number of images, each expert is trained on 6132 and validated on 876 images. Also, a separate set of 1421 images for each of the three road types is kept for testing purposes.

Split by Sky-to-Drivable Ratio

The second data split aims to distribute images into urban and highway subsets not manually, but automatically by calculating the ratio of the number of pixels labeled as sky to those labeled as drivable. To be able to get results comparable to the data split for road type, we decided to use the same number of training and test samples as in the previous case. We thus split the data according to the sky-to-drivable ratio as follows: (1) 8461 images with low sky-to-drivable ratio values (*i.e*. values from the interval $[0.0, 0.811]$), (2) 8481 images with high sky-to-drivable ratio values (*i.e*. from the interval $[1.487, 31.098]$) (3) 14341 images with ratio values ranging between these intervals (*i.e*. with ratio in $(0.811, 1.487)$). Figure 4 shows the distribution of the sky-to-drivable ratio values as well the boundaries of the data splits.

For this data split, we also train two experts: a high ratio and a low ratio expert. We also define three test sets for low, medium and high sky-to-drivable ratio.

Our Label	A2D2 Labels
person	animals, bicycle, pedestrian
car	ego car, car, small vehicles
truck	tractor, truck, utility vehicle
drivable	drivable cobblestone, parking area, RD normal street
nondrivable	curbstone, non-drivable street, sidewalk, slow drive area
blocker	irrelevant signs, grid structure, obstacles / trash, poles, RD restricted area, road blocks, signal corpus
info	traffic signal, painted driving instructions, sidebars, speed bumper, traffic guide object, electronic traffic, traffic sign, zebra crossing
sky	sky
buildings	buildings
nature	nature object
lanes	dashed line, solid line
-	blurred area, rain dirt

Table 2. Mapping from our label set with 11 classes to the 38 classes in the A2D2 dataset.

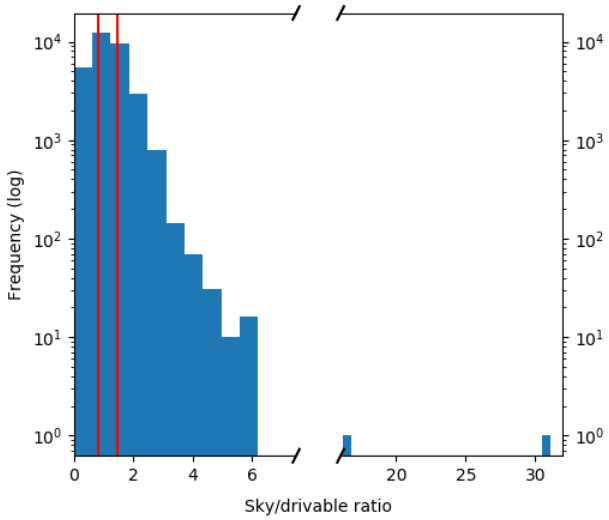

Figure 4. Distribution of the sky-to-drivable ratio values. Red lines show the boundaries of the data splits (high - medium - low ratio).

4.2. Baseline and Mixture of Experts

For the comparison, we defined a baseline model for each data split, which is trained on the combined data from both data subsets. This way, the baseline model sees twice as much data as each expert and is exposed to both cases. Similar to the experts, the baseline model also uses the FRRN A architecture (*c.f*. Figure 3). All models are trained for 100 epochs. Interestingly, we found that longer training intervals of the experts do not contribute significantly to their individual accuracy, but helps to improve the accuracy of the resulting MoE model.

Table 3 compares the mean IoU of the mixture of experts to individual experts as well as to a baseline for the manual data split by road type as well as for the split by the sky-to-drivable ratio. The results confirm that the single experts are only specialized on their own domain and can hardly generalize to the other scenario. The MoE, however, almost matches the accuracy of the baseline model.

4.3. Disagreements

To generate disagreement masks, we compare pixel-wise decisions of the experts and the overall MoE architecture. Table 4 shows the percentage of pixels belonging to each disagreement case as defined in Section 3.3. For the overwhelming majority of pixels, MoE and both experts agree. Also, in case of ambiguous and urban data, the MoE relies on the urban expert more often - this is also consistent with the mean IoU results (see Table 3) showing that the urban expert is more accurate. The same also holds for the sky-to-drivable data split, where the MoE agrees with the low-ratio-expert more often.

Additionally, these results show that in case of a better gate the portion of the perfect cases remains the same, but the MoE tends to agree with one of the experts rather than disagree with both - *i.e*. the number of the normal cases grows, whereas the number of the critical cases decreases. In particular, critical cases for the manual split by road type occur only in two images in the highway subset, 39 images in the ambiguous subset and two images in the urban subset. For the class-wise gate, however, only a single image in the ambiguous test set has no pixels belonging to the critical case.

For the qualitative results, we highlight the pixels belonging to each case thus obtaining a disagreement mask. Since perfect case dominates, we leave the corresponding pixels transparent. Figure 5 shows examples of the disagreement masks for images from different test subsets for the data split by road type. Figure 6 shows several randomly chosen images that have a large number of pixels belonging to the critical case when evaluated with a class-wise gate. Interestingly, critical cases correspond to image regions challenging for the semantic segmentation, such as blurred or overexposed areas.

5. Conclusion

In this work we have examined a first approach to use mixture of expert models for semantic segmentation with the added benefit of gaining additional interpretability. The current version focuses on experts that are trained on two different data subsets, a semantic urban/highway split and

Model \Dataset	Highway	Ambiguous	Urban	Highway+Urban	Mixed
Baseline	0.776	0.697	0.726	0.772	0.771
Highway Expert	0.769	0.6	0.367	0.456	0.476
Urban Expert	0.617	0.652	0.713	0.726	0.727
MoE	0.763	0.640	0.714	0.765	0.756

Model \Dataset	High Ratio	Medium Ratio	Low Ratio	High+Low Ratio	Mixed
Baseline	0.851	0.799	0.804	0.83	0.822
High Ratio Expert	0.833	0.729	0.627	0.721	0.725
Low Ratio Expert	0.673	0.734	0.799	0.743	0.743
MoE	0.824	0.759	0.8	0.818	0.801

Table 3. Mean IoU for for both data splits: manual split by road type (top) and split by the sky-to-drivable ratio (bottom). For MoE, the results are for the simple gate with convolutional layer and *pool_42* as a feature extraction layer.

Gate	Case	Highway	Ambiguous	Urban
Simple	Perfect Case	96.25%	95.29%	72.1%
	Normal case 1	3.74%	2.004%	0.16%
	Normal case 2	0.01%	2.704%	27.74%
	Critical Case	0.00002%	0.00017%	0.000001%
Class-wise	Perfect Case	96.2%	95.23%	71.97%
	Normal case 1	3.33%	1.38%	0.41%
	Normal case 2	0.27%	2.87%	27.13%
	Critical Case	0.2%	0.52%	0.49%

Gate	Case	High Ratio	Medium Ratio	Low Ratio
Simple	Perfect Case	93.25%	94.51%	91.44%
	Normal case 1	0.27%	0.33%	0.76%
	Normal case 2	6.17%	4.90%	7.55%
	Critical Case	0.31%	0.26%	0.25%
Class-wise	Perfect Case	92.31%	94.48%	91.05%
	Normal case 1	6.37%	0.36%	0.40%
	Normal case 2	1.0%	4.84%	8.30%
	Critical Case	0.32%	0.33%	0.25%

Table 4. Percentage of pixels, belonging to each disagreement case for both data splits: manual split by road type (top) and split by the sky-to-drivable ratio (bottom). Normal case 1 means MoE agrees with the first (highway resp. high ratio) expert. Normal case 2 means MoE agrees with the second (urban resp. low ratio) expert.

a more artificially created split according to the ratio between sky and drivable area in each input image. For both experts as well as a baseline trained on all available data, we use the FRRN A architecture for semantic segmentation, while the gate is a simpler network containing a single convolutional and two fully connected layers. Additionally, the gate shares common feature extraction layers with the experts, fed from different layers of the encoder part of the expert networks. The gating network predicts weights for the expert outputs which are in turn combined via a weighted linear combination of the outputs followed by an optional convolutional layer. We provided experimental evaluations of two gate architectures (a simple and a class-wise implementation) and also assessed the necessity of an additional convolutional layer as was proposed in the literature.

Our experiments using the semantic segmentation subset of the A2D2 dataset demonstrate that the mixture of experts architecture described is indeed able to reach baseline accuracy, while providing extra interpretability via the comparison of pixel-wise decisions of the experts and of the whole MoE architecture. Qualitative results via disagreement masks can help to identify areas of an image, for which the overall MoE architecture exhibits a higher uncertainty, such as in ambiguous, blurred or overexposed

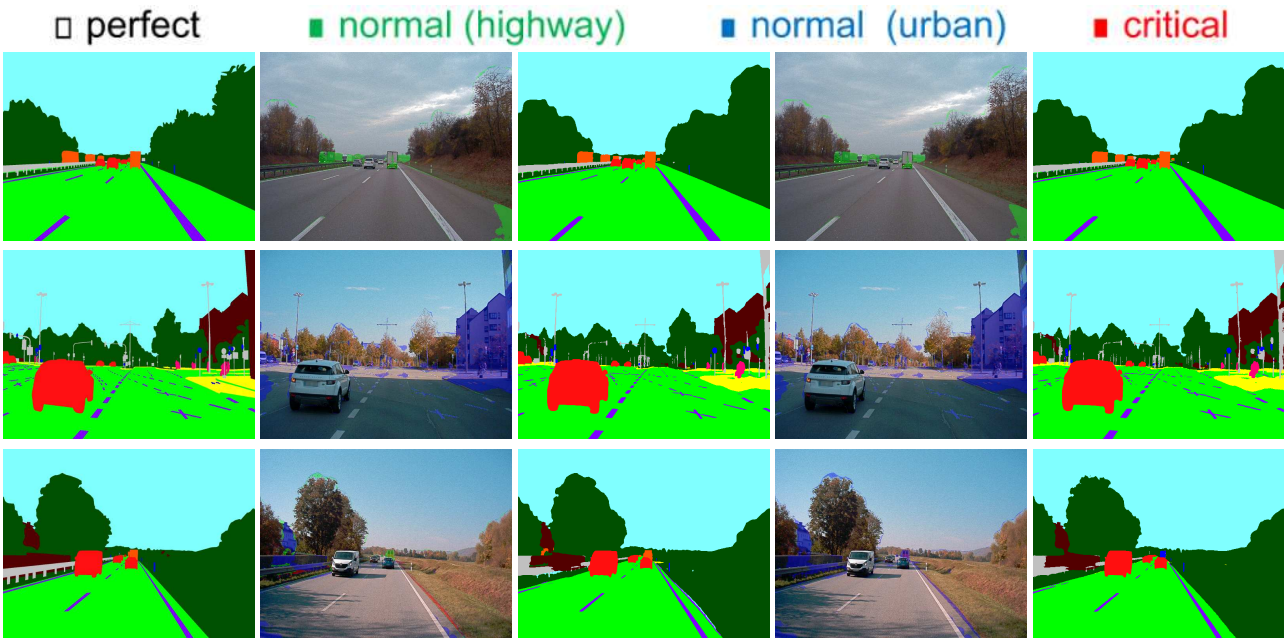

Figure 5. Disagreement masks and predictions for the manual data split by road type. Top to bottom: highway, urban, ambiguous. Left to right: ground truth, disagreement mask and prediction for the class-wise gate, disagreement mask and prediction for the simple gate.

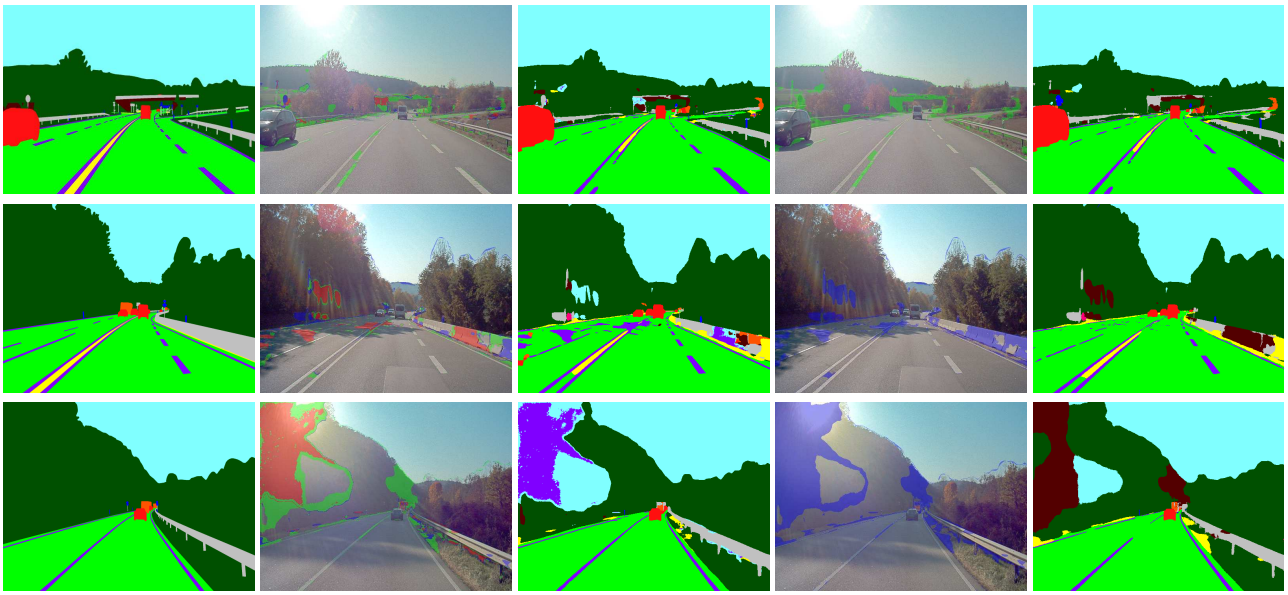

Figure 6. Examples of disagreement masks with critical case from the ambiguous test set for the manual data split by road type. Left to right: ground truth, disagreement mask and prediction for the class-wise gate, disagreement mask and prediction for the simple gate.

regions. However, this potential connection has to be investigated more in depth in future work. Current limitations include the necessity to identify relevant data splits manually as well as the complex training setup, where each expert and the overall MoE architecture have to be trained separately. Furthermore, the decision provided by the gate is only available at the end of the inference and requires execution of both experts and the gate itself in this current implementation of utilizing the expert's feature extraction within the gate.

Note that we did not perform an extensive architecture search to find the best possible expert architecture, but confine this work to utilizing FRRN as a basis, both for the experts as well as the baseline. In future work, an analysis of the effect of different expert architectures on overall quality, as well as the effect the added gating network within the

whole MoE construct has on expert network architectures should be thoroughly performed. This might yield further interesting properties.

Further experiments are planned on using multiple experts trained using disjoint combinations of two labels each (out of all available label classes), while all other classes are combined into a *unknown* class. This could be a potential path to have experts that are very good at distinguishing hard and ambiguous classes combined with a gate that learns to chose among all experts. A possible advantage would be that this could also yield multiple likely labels and also *backup* guesses in ambiguous situations that can be handled in subsequent modules in the data processing chain.

The advantages of using MoE architectures include explicit decision modeling and thus potential explainability via a gating network. It is still to be evaluated how this explainability compares to Bayesian methods (*e.g*. Monte Carlo dropout) or other attempts at estimating uncertainty.

An intriguing possibility of mixture of expert models is the fact that experts and the gating network may, but do not have to be implemented as neural networks. It is indeed also possible to replace both parts with different (*e.g*. probabilistic or simpler machine learning) models, that also utilize completely different or new inputs. Another interesting property of the mixture of expert model is the possibility, depending on implementation details, to parallelize the computation of all experts as well as the gating component or network. Decisions or weights of the gate can even be computed in advance to be able to skip execution of particular expert networks if their output is only marginal or not necessary at all, resulting in further runtime optimization possibilities. Furthermore, it might be possible to introduce redundancy into the whole model via several separate gates.

References

[1] K. Ahmed, M. H. Baig, and L. Torresani. Network of experts for large-scale image categorization. In *European Conference on Computer Vision (ECCV)*. Springer, 2016.

[2] C. M. Bishop. *Mixture Density Networks*. Aston University, 1994.

[3] R. Chan, M. Rottmann, R. Dardashti, F. Hüger, P. Schlicht, and H. Gottschalk. The Ethical Dilemma when (not) Setting up Cost-based Decision Rules in Semantic Segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR) - Workshops*. IEEE, 2019.

[4] D. Eigen, M. Ranzato, and I. Sutskever. Learning Factored Representations in a Deep Mixture of Experts. In *International Conference on Learning Representations (ICLR) - Workshops*, 2014.

[5] Y. Gal. *Uncertainty in Deep Learning*. Dissertation, University of Cambridge, 2017.

[6] Z. Ge, A. Bewley, C. McCool, P. Corke, B. Upcroft, and C. Sanderson. Fine-grained classification via mixture of deep convolutional neural networks. In *Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2016.

[7] J. Geyer, Y. Kassahun, M. Mahmudi, X. Ricou, R. Durgesh, A. S. Chung, et al. A2D2: AEV Autonomous Driving Dataset. `http://www.a2d2.audi`, 2019.

[8] C. Hubschneider, R. Hutmacher, and J. M. Zöllner. Calibrating Uncertainty Models for Steering Angle Estimation. In *International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2019.

[9] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[10] A. Kendall and Y. Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[11] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

[12] S. Kohl, B. Romera-Paredes, C. Meyer, J. De Fauw, J. R. Ledsam, K. Maier-Hein, S. M. A. Eslami, D. Jimenez Rezende, and O. Ronneberger. A Probabilistic U-Net for Segmentation of Ambiguous Images. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[13] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[14] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *International Conference on Knowledge Discovery & Data Mining*, 2018.

[15] O. Mees, A. Eitel, and W. Burgard. Choosing smartly: Adaptive multimodal fusion for object detection in changing environments. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016.

[16] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[17] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer, 2015.

[18] A. Valada, A. Dhall, and W. Burgard. Convoluted mixture of deep experts for robust semantic segmentation. In *International Conference on Intelligent Robots and Systems (IROS) - Workshops*, 2016.

[19] A. Valada, G. L. Oliveira, T. Brox, and W. Burgard. Deep multispectral semantic scene understanding of forested environments using multimodal fusion. In *International Symposium on Experimental Robotics*. Springer, 2016.

[20] A. Valada, J. Vertens, A. Dhall, and W. Burgard. Adapnet: Adaptive semantic segmentation in adverse environmental conditions. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 2017.