Please cite this paper as:

```
@inproceedings{blum2018fusion,
  title     = "Modular Sensor Fusion for Semantic Segmentation",
  author    = "Blum, Hermann and Gawel, Abel and Siegwart, Roland and Cadena, Cesar",
  booktitle = "2018 {IEEE/RSJ} International Conference on Intelligent Robots
               and Systems ({IROS})",
  year      = 2018;
}
```

Modular Sensor Fusion for Semantic Segmentation

Hermann Blum, Abel Gawel, Roland Siegwart and Cesar Cadena

Abstract— Sensor fusion is a fundamental process in robotic systems as it extends the perceptual range and increases robustness in real-world operations. Current multi-sensor deep learning based semantic segmentation approaches do not provide robustness to under-performing classes in one modality, or require a specific architecture with access to the full aligned multi-sensor training data. In this work, we analyze statistical fusion approaches for semantic segmentation that overcome these drawbacks while keeping a competitive performance. The studied approaches are modular by construction, allowing to have different training sets per modality and only a much smaller subset is needed to calibrate the statistical models. We evaluate a range of statistical fusion approaches and report their performance against state-of-the-art baselines on both real-world and simulated data. In our experiments, the approach improves performance in IoU over the best single modality segmentation results by up to 5%. We make all implementations and configurations publicly available.

I. INTRODUCTION

Semantic segmentation has become a popular discipline in recent years [1]. It most commonly deals with the pixel-wise categorical classification of image data, but can be employed for various sensor data, e.g., 3D data [2], [3]. In robotics, semantic segmentation is relevant for scene understanding in autonomous driving [4], localization tasks [5], or natural human-machine interaction [6]. While architectures for single modalities, e.g., RGB or 3D data are becoming increasingly accurate, there remain perceptually difficult cases that single sensors are unable to reliably classify. Lane-markings and Pictures on a wall are invisible to depth sensors. RGB cameras, on the other hand, are much more sensitive to weather and lighting conditions. Using multiple sensors can increase performance, compensating for other sensors' weaknesses or failures. Recently, several approaches were proposed to address the challenge of fusing multiple sensor inputs for semantic segmentation. One avenue of research leads towards training one specific network for all modalities together [7]–[9]. Another avenue is to leverage single-modality *expert* networks and a trained fusion network [10]–[12]. One major pitfall of these solutions is the requirement of training for any sensor combinations and therewith aligned multi-modal training data.

In contrast, we wish to design segmentation systems that remain modular, and can fuse different *expert* networks in-
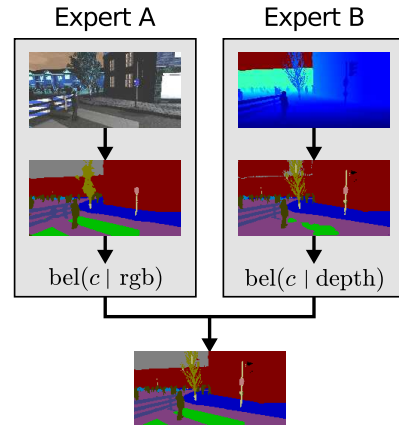
Fig. 1: Individual semantic segmentation classifiers are combined modularly using different statistical methods, resulting in improved semantic segmentation without additional training.

dependently trained on single modalities. Apart from neural networks, fusion of classifers has been a part of Machine Learning literature for decades already [13]. Dependent on well-known statistics like confusion matrices, classifier outputs are combined using statistical methods.

In this work, we propose a novel, scalable semantic segmentation fusion architecture based on separately trained expert networks that are statistically fused using Bayesian fusion or Dirichlet Fusion. The individual *experts* can thus be trained on different datasets, and no additional training is required to fuse their outputs. In addition, due to its modularity, input modalities can be added and removed on-the-fly. Hence, failures of single *experts* can be compensated if detected.

This paper presents the following contribution:

- A novel statistical fusion method for multi-modal semantic segmentation based on the Dirichlet distribution.
- The first fusion network alternative that does not require training on aligned data.
- Evaluation and analysis of different statistical fusion methods on real-world and synthetic datasets.

II. RELATED WORK

A. Semantic Segmentation

Semantic Segmentation is commonly understood as the task of pixel-wise classification of image data [1]. While being an important discipline in the Computer Vision community, it has relevance for various robotics applications, especially scene understanding and interaction [4]–[6]. Presently, the best performing algorithms for semantic segmentation on popular benchmarks, e.g., PascalVOC [14] are based on

Convolutional Neural Networks (CNNs). Here, a common architecture is the encoder-decoder structure [15]–[17]. The encoder aims to capture features on multiple abstractions of the images, as commonly used in image classification tasks. The decoder then unfolds the feature data back to the pixel-level.

In this work, we base our semantic segmentation structure on the Fully Convolutional Network (FCN) [17] and Adap-Net [11].

B. Fusion

While semantic segmentation is a maturing discipline for RGB-based systems, a fusion with complementary sensor data can further improve the segmentation results [7], [11], especially in challenging classes for the visual sensor. Based on CNN architectures, different mechanisms for fusion have been developed. In principle, we distinguish between two architectures, i.e., networks fully designed for fusion [7]–[9], and unimodal *expert* networks with an added fusion network [10], [11], as originally proposed by Jacobs et al. [18]. FuseNet [7] fuses features from RGB and Depth images gathered from two VGG16-encoders [19] and decodes the fused features into a semantic segmentation. A similar approach is used in [8], where features are extracted from different layers' outputs of the encoder in an adapted FCN structure. A multimodal autoencoder is proposed in [9] where the possible deficiencies in one modality have to be foreseen and introduced in training time. The works by [11] and [10] explore different mechanisms that fuse classifier outputs at a later stage, they find that a gating network learning factors for a weighted sum of the individual segmentation outputs works best for their network.

In contrast, our approach is inspired by work on fusion of classifiers [13]. While the original work deals with uni-modal handwriting recognition, we extend it towards multi-modal semantic segmentation by replacing the simple classification output with a full output score vector. In addition, we omit the possibility of rejecting classifications present in the original work.

Further general fusion techniques have been developed to deal with ensembles of neural networks. These include averaging over a range of experts or voting principles. In such a setup, classifiers are trained equally to specialize on different aspects of a problem, e.g., by employing techniques like bagging [20] and boosting [21]. However, as we deal with architectures that train on different input modalities, we do not have the same control over the specific strengths and weaknesses of the different classifiers, which is necessary for simple fusion techniques like voting to work. Moreover, these techniques require a multitude of expert networks, which is impracticable with big CNNs.

III. METHOD

A. Semantic Segmentation

As baseline systems for semantic segmentation we use two different neural network architectures. The FCN structure was first introduced by Long et al. in 2015 [17] and used by Xiang et al. to fuse different modalities [8]. The FCN uses the VGG-16 encoder [19], which consists of iterating 3×3 convolutions and 2×2 max-pooling layers that maps an input image onto a lower dimensional feature map. This feature map is then scaled up again with deconvolutions and mapped onto the output class using 1×1 convolutions. When we refer to the FCN in experiments, we mean the version shown in Figure 5 as used in [8], which reduces complexity by replacing the trainable deconvolutions with simple bilinear interpolation. While not achieving the performance of more complex networks, we used this simpler model for testing many different fusion methods and report the comparison.

AdapNet [11] is a structure designed to improve some of the shortcomings of the FCN. It uses the ResNet encoder [22] where every dimensionality-reducing stage is split into convolutions applied on different scales of the input to make the feature map more independent against different scales of an input object. Similar to the FCN above, different scales of this feature-map are then processed independently and stacked together before using 1×1 convolutions and deconvolutions to map the features onto the output classes. While the training is slower and complex, Valada et al. found the evaluation time of the AdapNet to be faster than that of the FCN and other methods [11].

B. Statistical Classifier Combination

For modality fusion, we train individual baseline networks for every input modality completely independent of each other. The fusion is then applied based on the outputs of all these different baselines that were evaluated on the same scene.

All of the approaches introduced in II-B have in common that the fusion process is part of the network structure, i.e. the fusion is not adaptable to different sensor combinations, and requires retraining. Moreover, with the exception of the gating network [11], none of these systems can deal with crashing sensors by exchanging or disabling input experts.

As a more modular approach, we propose a fusion technique that is based on statistically merging the outputs of individual classifiers. In this sense, this work is heavily inspired by techniques described by Xu et al. in 1992 [13].

Bayes Categorical Fusion: For every pixel we want to produce a probability $p(k|\text{all expert outputs})$ for every possible class $k \in 1, ..., K$, given the outputs of all uni-modal experts. From this, we can then choose for every pixel the class with the highest probability. This is sometimes called the believe of a class $\text{bel}(k)$. Based on Xu et al. [13] and Bayes' formula, we find:

$$p(k|\text{all expert outputs}) \propto p(\text{all expert outputs}|k)\, p(k)$$
$$\propto p(k) \prod_{i \in \text{modalities}} p(\text{out}_i|k)$$

with out_i the classification output of expert i. $p(\text{out}_i|k)$ is a categorical distribution over the expert's classification output, which we will know at inference time. We use that the conditionals $p(\text{out}_i|k)$ are independent of each other, as the
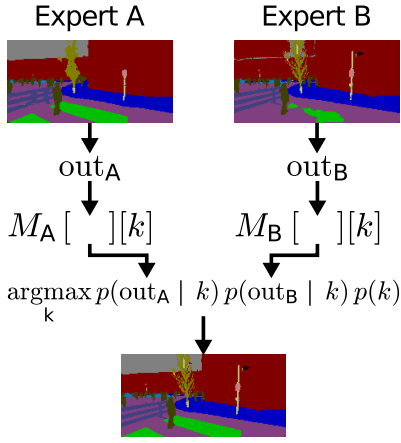
Fig. 2: Example of the Bayesian categorical fusion with 2 modalities. The classification output is used as index to the confusion matrix in order to produce the conditional class likelihoods. For fusion, the class with the biggest joint likelihood is choosen.

interdependency between the different modalities is exactly given by the ground truth class k.

Each of the individual conditional categorical distributions is given by the confusion matrix M_i of the corresponding classifier. If the first dimension of the matrix is the actual expert output out_i and the second dimension is the ground-truth class k, it follows:

$$p(\text{out}_i|k) = \frac{M_i[\text{out}_i][k]}{\sum_{j=1}^{K} M_i[j][k]}$$

The prior $p(k)$ can also be set on basis of the class-occurrence in these confusion matrices.

We find the fused classification by choosing the class that has the highest log-probability:

$$\text{output class} = \underset{k}{\text{argmax}}\, p(k|\text{all expert outputs})$$

$$= \underset{k}{\text{argmax}} \left[\log p(k) + \sum_{i \in \text{experts}} \log p(\text{out}_i|k) \right] \tag{1}$$

A schematic of this technique is shown in Figure 2.

Dirichlet Fusion: In the above approach, we fuse the different experts on basis of their classification output. However, this disregards important information, especially in such cases of special interest for fusion where two or more classes are equally likely. To take these cases into account, we have to produce $p(\mathbf{y}|k)$ that is dependent on the full softmax output vector $\mathbf{y}$ and therefore contains score values for all possible classes. This distribution is given by the Dirichlet distribution, the conjugate prior of the categorical distribution.

$$\mathbf{y} \sim \text{Dirichlet}(\boldsymbol{\alpha})$$

$$\text{pdf}(\mathbf{y}) = \frac{\Gamma\left(\sum_{j=1}^{K} \alpha_j\right)}{\prod_{j=1}^{K} \Gamma(\alpha_j)} \prod_{j=1}^{K} y_j^{\alpha_j - 1}$$

The concentration parameters $\boldsymbol{\alpha}$ of this distribution are constrained by $\alpha_j > 0\ \forall \alpha_j \in \boldsymbol{\alpha}$. The higher $\alpha_l/\sum_{j=1}^{K} \alpha_j$, the higher is the probability of a vector close to $y_j = \mathbb{1}_{j=l}$.

In order to find the correct concentration parameters for every conditional class and every expert, we make use of the fast Expectation-Maximization (EM) algorithm developed by Minka [23] and improved and translated to python by Sklar [24]. Both make use of the fact that there is a sufficient statistic for the EM fitting of the Dirichlet distribution $\mathbb{s} = \frac{1}{N} \sum_{n=1}^{N} \log \mathbf{y}^{(n)}$. Without the sufficient statistic over a set of N pixels, the M-Step would have to compute over all the images in every iteration. Instead, $\mathbb{s}$ can be produced from measurements over the data before starting the EM algorithm, which then can validate the likelihood of the produced parameters at every iteration against the sufficient statistic.

The standard loss function of the EM algorithm is the likelihood of the parameters $\boldsymbol{\alpha}$ against the data:

$$\mathscr{L}(\boldsymbol{\alpha}, \mathbb{s}) = N \left[\log \Gamma\left(\sum_j \alpha_j\right) - \sum_j \log \Gamma(\alpha_j) \right.$$
$$\left. + \sum_j (\alpha_j - 1)\log \mathbb{s}_j \right] \tag{2}$$
$$\mathbb{s}_j = \frac{1}{N} \sum_{n=1}^{N} \log y_j^{(n)}$$

We found that this standard method does produce reasonable parameters $\boldsymbol{\alpha}$, but the resulting conditional likelihoods do not work well in our decision function from Equation (1). The main problems are edge cases where two classes get similar scores in $\mathbf{y}$. In this case, conditionals found with Equation (2) assign equally low likelihoods to all classes. Therefore, we introduced 2 regularization terms: We added the norm $\sum \alpha_j^2$ as a regularization term to prevent the concentration parameters to grow too large, which in turn would produce low likelihoods for any $\mathbf{y}$ that is not close to a one-hot vector. In addition, we do not aim for parameters $\boldsymbol{\alpha}$ that explain our classifier output best, but we want to distinguish between different conditionals. Therefore, we build a sufficient statistic $\bar{\mathbb{s}}$ of all classifier outputs for different ground-truth than the conditional class we search for and add $-\beta \mathscr{L}(\boldsymbol{\alpha}, \bar{\mathbb{s}})$ into the Loss. With these 2 additions, and omitting the constant factor N, as proposed by Sklar [24], we arrive at the following loss function for the EM algorithm:

$$\mathscr{L}'(\boldsymbol{\alpha}, \mathbb{s}, \bar{\mathbb{s}}) = (1-\beta) \left[\log \Gamma\left(\sum_j \alpha_j\right) - \sum_j \log \Gamma(\alpha_j) \right]$$
$$- \beta \sum_j (\alpha_j - 1)\log \bar{\mathbb{s}}$$
$$+ \sum_j (\alpha_j - 1)\log \mathbb{s} - \delta \sum_j \alpha_j^2$$

Once the concentration parameters for all ground-truth classes $\boldsymbol{\alpha}^{(k)}$ are found, fusion can be performed with the following decision function:

$$\text{output class} = \underset{k}{\text{argmax}} \left[\log p(k) + \sum_{i \in \text{experts}} \log f(\mathbf{y}_i|k) \right]$$
$$f(\mathbf{y}_i|k) = \text{pdf}(\mathbf{y}_i, \boldsymbol{\alpha}_i^{(k)})$$

IV. EXPERIMENTS

We conduct experiments with implementations in tensorflowand train with RMS Prop [25] using default configurations.

We test our methods on the Synthia-Rand-Cityscapes [12] and the Cityscapes datasets [26], both showing urban street scenes. The Synthia dataset comes from simulation and features alongside RGB also very precise depth images, and pixel-wise semantic segmentation into 13 classes. The real-world Cityscapes dataset contains RGB, noisy disparity images from stereo matching, and pixel-wise semantic segmentation into 30 different classes. For both datasets, we use a common set of 12 classes that was also used in [11]: *void, sky, building, road, sidewalk, fence, vegetation, pole, car/bus/truck, traffic sign, pedestrian, bicycle/motorcycle/rider.* Furthermore, we resized input images to 768x384 following the experiments of [11]. As there is no given split between train- and test-set in the Synthia-Rand-Cityscapes, we take a random 10% sample as test-set, and another 10% sample as a validation and development set. The images in the dataset are produced from random viewpoints, which makes this simple split feasible. The development set is used to compute the confusion matrices and conditional Dirichlet distributions. For Cityscapes, we take a random 5% sample out of the given training set as development set. The parameters β and δ for the Dirichlet fusion are found with a grid search on the development set before evaluating the method with the found parameters on the separate test set. We run the EM algorithm with a maximum of 1000 iterations.

During training, we employ cropping and flipping as augmentation methods, after which we finetune the baselines on small batches of full images. The semantic segmentation performance is always measured in Intersection over Union (IoU) [14]. For overall performance, we take the mean over all available classes. In cases where we do not report the per-class IoU, we additionally report Average Precision (AP).

All implementations and configurations of the experiments are available at `https://github.com/ethz-asl/modular_semantic_segmentation`.

A. Fusion on Synthetic Data

We design 2 experimental setups. First, we use the previously listed 12 classes and later we add lane-markings as an additional class. In this context, lane-markings are a very interesting example for fusion as they are indistinguishable from roads on a depth image and therefore only visible to an RGB expert. Following the evaluation of [11] for semantic segmentation of RGB and Depth images, we compare Averaging, and our Bayes Categorical and Dirichlet fusion. Averaging fuses the experts by taking the mean over all softmax outputs and choosing the class with the highest mean score. Tables I and II show the results of our evaluation, Figure 3 shows qualitative examples.

In general, we find that the statistical fusion significantly improves the semantic segmentation with respect to the two uni-modal baselines. We also find that the inclusion of

TABLE I: Fusion of AdapNet Baselines on Synthia Rand Cityscapes

Lane-markings		Dirichlet	Bayes	Average	RGB	Depth
no	IoU	77.27	78.62	**78.70**	73.39	72.70
no	AP	83.31	**86.12**	84.04	80.87	79.54
yes	IoU	**80.19**	79.91	79.05	75.92	63.76
yes	AP	**87.43**	86.92	83.65	83.34	70.20

TABLE II: Per-Class IoU of AdapNet Baselines on Synthia Rand Cityscapes with Lanemarkings

Class	Dirichlet	Bayes	Average	RGB	Depth
Mean	**80.19**	79.91	79.05	75.92	63.76
Sky	97.39	**97.41**	95.57	95.54	9.47
Building	**96.85**	96.76	96.20	93.70	79.86
Road	**94.09**	92.78	93.11	91.58	88.92
Sidewalk	**95.07**	94.04	94.70	91.71	93.60
Fence	**74.76**	72.97	74.54	68.30	71.89
Vegetation	89.89	**90.62**	90.02	81.42	90.32
Pole	65.53	**66.42**	64.79	56.17	60.19
Car	**93.19**	91.39	92.81	87.81	90.56
Traffic Sign	41.75	**51.54**	48.31	45.77	37.76
Pedestrian	**79.94**	74.73	79.72	73.70	72.77
Bicycle	**65.47**	63.11	64.42	58.57	59.40
Lane-marking	**68.29**	67.17	54.39	66.73	10.44

lane-markings changes the comparison between the different fusion methods. Without lane-markings, we observe only a minor difference between averaging and the Bayes Categorical fusion. However, when including lane-markings, we see that the training of the Depth expert is much harder, resulting in a worse individual performance of this modality alone.

One effect of this is that the Bayes categorical fusion improves over the averaging due to its ability to 'pick' the better performing expert based on the classification outputs and mirror its performance.

The second effect is that the Dirichlet fusion is producing the best results in most individual classes, and the overall mean performance. We observe that in the classes where the Dirichlet fusion produces the best results, it usually also improves the classification with respect to both individual experts, instead of mirroring the performance of the better one. We offer the conclusion that this is possible through the more detailed input into the fusion mechanism.

B. Fusion on Real-World Data

To validate the results from the synthetic data, we test the same methods on the Cityscapes [26] dataset. The results of

TABLE III: Fusion of AdapNet Baselines on Cityscapes

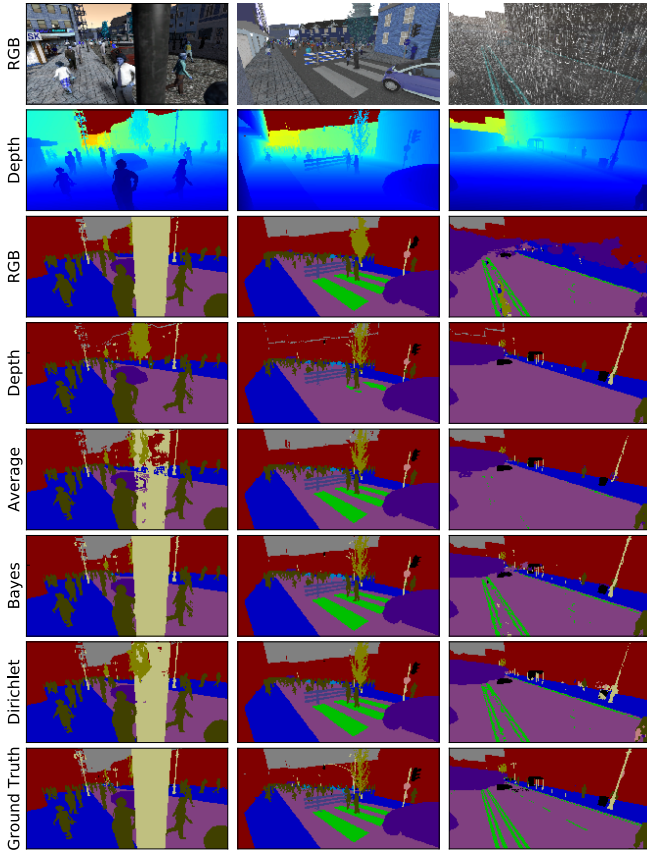| Class	Dirichlet	Bayes	Average	RGB	Depth
Mean	**69.22**	68.77	68.47	69.20	54.12
Sky	86.97	90.45	88.93	**90.54**	78.60
Building	**84.60**	83.59	84.32	84.09	72.83
Road	92.37	91.71	**92.59**	91.53	91.79
Sidewalk	**67.66**	65.01	67.25	66.42	57.58
Fence	**41.54**	37.92	40.79	39.84	23.62
Vegetation	87.13	**88.04**	84.48	**88.04**	66.56
Pole	**44.23**	42.11	43.77	41.94	33.28
Vehicle	**87.62**	86.20	87.02	86.54	77.92
Traffic Sign	51.29	51.37	48.33	**52.12**	18.85
Person	**64.90**	63.54	64.75	63.54	52.27
Bicycle	53.13	56.54	50.98	**56.57**	22.00

Fig. 3: Qualitative Examples from the Synthia Rand Cityscapes Fusion Experiments. **Left** Due to some error in the simulation, the pole right in Front of the camera is not visible to the depth expert. Because the class-scores from the RGB experts are very high, the fusion methods mostly recover from this failure. **Center** Lanemarkings are not visible to the depth expert, even tough it predicts them below pedestrians crossing the street. The correct classification from the RGB expert is fused into the output by all methods. **Right** Fusion can improve robustness for different weather conditions, as shown here for heavy rain occluding most of the RGB input.

this experiment are shown in table III. Qualitative examples can be found in figure 4. The Synthia Rand Cityscapes is especially designed to cover the same classes and urban street scenes present in Cityscapes, enabling a fair comparison of the results. The Cityscapes dataset does not provide depth images from a separate depth sensors, but disparity maps computed from stereo cameras. Due to considerable noise in the depth estimation, we observe in general much worse performance of the Depth expert, compared to the experiments on synthetic data. We also find that the fusion methods offer no significant improvement in segmentation, compared to the RGB expert. While the fusion improves the segmentation on classes such as *pole* or *vehicle*, it has lower performance than the RGB expert on classes like *sky*.

C. Modalities Trained on Different Datasets

An important advantage of our modular fusion methods is that unimodal experts do not require simultaneous training. They can be trained independently, even on different datasets, as not every modality may be available for every dataset.
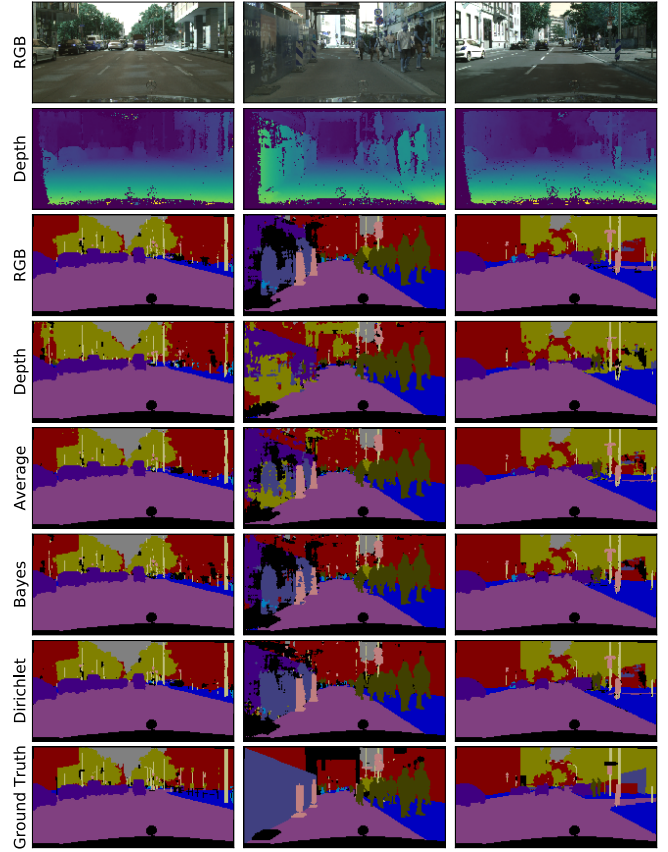

Fig. 4: Qualitative Examples from the Cityscapes Fusion Experiments. As the depth images are visibly worse than in the synthetic data, and there are no different weather conditions, fusion improvements are less obvious. The biggest improvements can be found for poles in the background, as well as the left wall in the center column.

We therefore test our fusion methods in the following scenario: There is very little labelled data for a new modality (depth) available. We take a Depth baseline that is trained on simulation data and just plug it into our system, calibrating it against a few labelled example images.

In this experiment, we take the Depth baseline trained on synthia data and fuse it together with the RGB baseline from experiment IV-B, validating the system on the cityscapes test set. The results are shown in Table IV. While averaging was a competitive method in the earlier experiments, it fails in this experiment as it is the only method without calibrating the experts. We can also see that the validation of the experts is working as expected. In fact, the simple Bayes Categorical fusion is mirroring the output of the RGB baseline.

For those classes where the depth expert carries some information, the Dirichlet fusion is always learning to improve the output of the RGB expert. The overall fusion results for Bayes and Dirichlet fusion are even better than with the depth baseline trained on Cityscapes. We attribute this to the observation that the Bayes and Dirichlet fusion perform well in a setting where they can choose between the experts, but as this experiment shows their model does not cope well with one expert that is always worse.
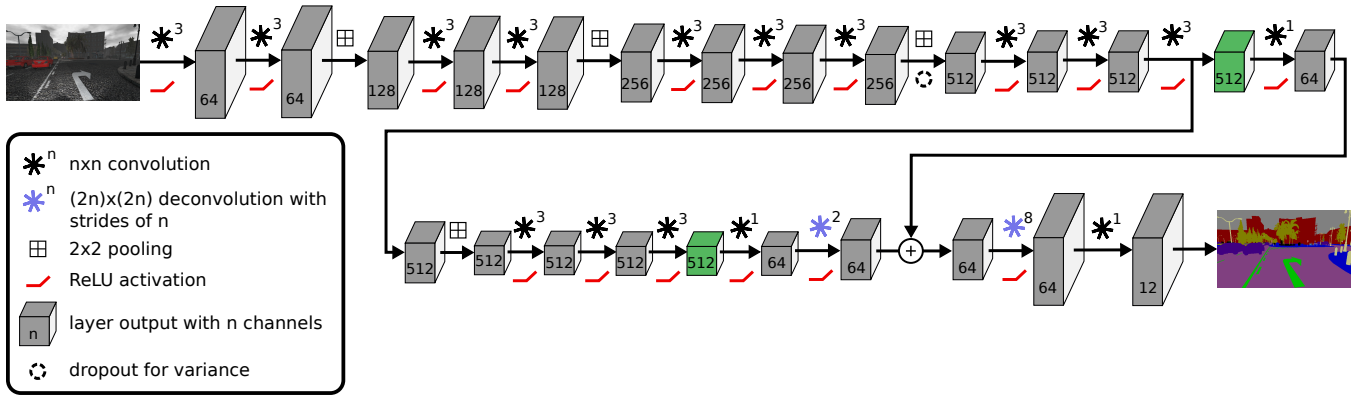
Fig. 5: The FCN architecture adapted from [8] and used in our experiments. It consists of the VGG-16 encoder [19] of stacked convolutions, and poolings. Instead of trainable deconvolutions, this architecture uses bilinear interpolations for the decoder, thus reducing the complexity of the network. Dropout is applied only for the variance fusion experiment. In case of multiple encoders, the marked green encoder outputs are stacked together before applying the 1×1 convolution.

TABLE IV: Fusion of AdapNet Baselines on Cityscapes. RGB is trained on Cityscapes, Depth is exclusively trained on Synthia Rand Cityscapes.

Class	Dirichlet	Bayes	Average	RGB	Depth
Mean	**69.28**	69.20	62.65	69.20	3.37
Sky	90.16	90.53	89.63	**90.54**	0.00
Building	**84.18**	84.09	68.99	84.09	14.09
Road	91.56	91.53	**91.63**	91.53	0.00
Sidewalk	**66.45**	66.42	62.91	66.42	0.03
Fence	39.75	**39.84**	35.76	**39.84**	0.00
Vegetation	**88.07**	88.04	79.85	88.04	22.60
Pole	**42.22**	41.94	32.34	41.94	0.36
Vehicle	**86.82**	86.54	86.15	86.54	0.00
Traffic Sign	**52.37**	52.11	43.21	52.12	0.00
Person	**63.89**	63.54	50.87	63.54	0.00
Bicycle	**56.60**	56.57	47.86	56.57	0.00

D. Further Benchmarks

In this section, we conduct additional tests with different FCN-based [17] architectures and further investigate possibilities of modular fusion and the performance of our proposed methods. The principle architecture we use is shown in Figure 5.

In the experiments before, we could compare the improvements of each method with the uni-modal baselines. As we now compare between fusion methods that are not based on any baselines, this is no longer possible. We therefore train all FCNs to their best possible performance and compare overall results.

Additionally, we evaluate the inference time per image of every method. The inference time is measured over 10000 trials and every method is evaluated on constant input of the same size as an image, to further reduce the influence of caching and data loading on the measurements. All computations are conducted on a single GPU. Dependent on the available hardware, the methods may benefit differently from different degrees of parallellisation.

a) Fusion by Variance: Using Dropout-Monte-Carlo [27], we can measure model uncertainties from each modality expert at inference time. In theory, this variance σ_j^2 should contain information about the uncertainties of the different classifiers, trained on different modalities. The

per-pixel certainty ω is then approximated by

$$\omega = 1/\left(\frac{1}{K} \sum_{j=0}^{K-1} \sigma_j^2 \right)$$

Where K is the number of classes and σ^2 is measured from the softmax output class scores $\mathbf{y}$ over a number of samples. We fuse the experts with a weighted sum.

$$\mathbf{y}_{\text{fused}} = \frac{\sum_{i=0}^{M-1} \omega_i \mathbf{y}_i}{\sum_{i=0}^{M-1} \omega_i} \qquad (3)$$

The class score vector $\mathbf{y}_i$ of every uni-modal expert $0 \leq i < M$ is weighted by the certainty of the given pixel and expert. Note that if the certainties of all the experts are assumed equal, this is reduced to the averaging fusion used before. During the development we found that this method is very sensitive to the type of dropout performed. In particular, any dropout close to the output layer leads to very noisy fusion outputs, requiring a large number of Monte-Carlo samples to compensate. Following the findings from [28], we choose to introduce a dropout layer after the third pooling, before the network branches into two parts and not immediately before a pooling layer. This is also indicated in Figure 5.

b) Full Fusion Network: As described in II-B, fusion is often conducted with a fully integrated network. As a benchmark we use the structure from [8] with two encoders fused together into one decoder, but without the recurrent layer. It is therefore following the architecture shown in figure 5. We train on aligned RGB and Depth images until convergence.

The results are shown in Table V. The experiment is again conducted with the standard set of 12 classes.

As expected, the fully integrated network expresses best performance. Contradictory to the experiments with AdapNet on the same dataset reported in section IV-A, the averaging fusion is performing significantly better than both the Bayes categorical and the Dirichlet fusion. We attribute this to the fact that both baselines have very similar performances.

We find that the variance fusion and the Dirichlet fusion express lower performance than the uni-modal experts. Both methods are based on mechanisms that attempt to measure

TABLE V: FCN-based Fusion Methods on Synthia Rand Cityscapes

	FuseFCN	Average	Bayes	Dirichlet	Variance	RGB	Depth
IoU	76.90	76.38	74.99	66.96	66.35	72.24	72.01
AP	83.79	83.54	82.91	73.82	73.83	80.35	81.15
Inference Time	72 ± 22 ms	43 ± 11 ms	46 ± 16 ms	52 ± 24 ms	310 ± 18 ms	22 ± 11 ms	22 ± 12 ms

uncertainties of individual experts. Further investigation for the Dirichlet fusion revealed that opposed to AdapNet, the FCN architecture often produces outputs that assign similar probabilities to more than two different classes. The AdapNet baselines from section IV-A, however, usually express higher certainty and assign high probability scores to one or two classes, also at the border of objects. We conclude that the Dirichlet fusion method, in particular the proposed EM fitting, is not able to capture the variability of the FCN output well.

While the Average Fusion is on average faster than the Bayes or Dirichlet fusion, as would be expected from the complexity of the calculation, the difference in inference time between the three methods stays within one standard deviation. We also note that the modular methods are all faster then the integrated FusionFCN, even tough this architecture is a 'late-fusion' method with independent encoders.

V. DISCUSSION

Our results indicate that statistical fusion of modalities is a promising avenue for semantic segmentation in cases where we cannot or do not want to retrain a fusion network on aligned data. Despite the good performance of the averaging on the synthetic data, we argue that it is only advisable to use this technique when the experts have comparably good performance. Especially in real-world applications, we usually have sensors expressing much better performance than the others. Here, the averaging fails to produce convincing results.

The statistical fusion is able to exploit the information even in cases of very strong performance differences and improve the segmentation result. Although not expressing superior performance in all tested cases, statistical fusion generally improves the segmentation performance over single modality systems without the need for cumbersome training.

We notice that the Dirichlet model is in general the best performing technique. However, the current EM algorithm to find the concentration parameters suffers under very flat, under-confident class probability outputs of the single experts. In these cases, the Dirichlet model cannot produce parameters that result in suitable decision functions for the fusion. Consequently, in cases of under-confident uni-modal experts, the Dirichlet fusion looses its expressiveness power to combine the modalities on every class.

A general finding from our experiments is that the Bayes Categorical fusion, and also the Dirichlet fusion, work best with experts that have complementary strengths. Here, these frameworks show their power to pick the best performing expert for every class or even learn on basis of their combination. In practice, we showed that this depends both on the

set of classes as well as the quality of the input data for the different modalities.

An analysis of the inference time for the different methods further shows that statistical fusion is a promising method of time-critical systems, as the inference time is significantly lower than a fully integrated fusion network architecture.

VI. CONCLUSIONS AND OUTLOOK

In this paper, we have presented and evaluated two modular approaches to fuse multiple modalities for semantic segmentation. The novel proposed Dirichlet fusion shows the best results of the statistical fusion methods, especially when using modalities with complementary strengths and weaknesses. Furthermore, the modularity in terms of the used segmentation experts allows for a seamless extension to new experts without re-training the already existing ones.

The performance of the proposed fusion scheme is close to the performance of specifically trained fusion networks, but requires no additional training on aligned data. It therefore gives wider access to datasets that do only contain a single modality.

The findings are consistent over different datasets from simulation to real world scenarios. The biggest problems with the statistical fusion were encountered in cases of low-confidence classifier outputs. In future work, we will therefore test whether measurements of input based uncertainties of the neural network classifiers can further improve the results of statistical fusion.

To conclude, the proposed statistical fusion promises to be a powerful basis of a modular framework for semantic segmentation. With this framework, we can produce semantic scene descriptions for a diverse set of robots, enabling collaboration and mutual scene understanding.

REFERENCES

[1] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," Apr. 2017. arXiv: 1704.06857 [cs.CV].

[2] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 77–85. DOI: 10.1109/CVPR.2017.16.

[3] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1912–1920. DOI: 10.1109/CVPR.2015.7298801.

[4] A. Ess, T. Mueller, H. Grabner, and L. v. Gool, "Segmentation-Based urban traffic scene understanding," in *Procedings of the British Machine Vision Conference 2009*, British Machine Vision Association, 2009, pp. 84.1–84.11. DOI: 10.5244/C.23.84.

[5] A. Gawel, C. D. Don, R. Siegwart, J. Nieto, and C. Cadena, "X-view: Graph-based semantic multi-view localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1687–1694, Jul. 2018. DOI: 10.1109/LRA.2018.2801879.

[6] M. Oberweger, P. Wohlhart, and V. Lepetit, "Hands deep in deep learning for hand pose estimation," Feb. 2015. arXiv: 1502.06807 [cs.CV].

[7] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "FuseNet: Incorporating depth into semantic segmentation via Fusion-Based CNN architecture," in *Asian Conference on Computer Vision*, vol. 10111 LNCS, Springer International Publishing, 2016, pp. 213–228. DOI: 10.1007/978-3-319-54181-5_14.

[8] Y. Xiang and D. Fox, "DA-RNN: Semantic mapping with data associated recurrent neural networks," in *Robotics: Science and Systems XIII*, Robotics: Science and Systems Foundation, Jul. 2017. DOI: 10.15607/RSS.2017.XIII.013.

[9] C. Cadena, A. Dick, and I. D. Reid, "Multi-modal Auto-Encoders as joint estimators for robotics scene understanding," in *Robotics: Science and Systems XII*, Robotics: Science and Systems Foundation, 2016. DOI: 10.15607/RSS.2016.XII.041.

[10] O. Mees, A. Eitel, and W. Burgard, "Choosing smartly: Adaptive multimodal fusion for object detection in changing environments," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2016, pp. 151–156. DOI: 10.1109/IROS.2016.7759048.

[11] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "AdapNet: Adaptive semantic segmentation in adverse environmental conditions," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 4644–4651. DOI: 10.1109/ICRA.2017.7989540.

[12] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 3234–3243. DOI: 10.1109/CVPR.2016.352.

[13] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, no. 3, pp. 418–435, May 1992. DOI: 10.1109/21.155943.

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010. DOI: 10.1007/s11263-009-0275-4.

[15] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1520–1528. DOI: 10.1109/ICCV.2015.178.

[16] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional Encoder-Decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017. DOI: 10.1109/TPAMI.2016.2644615.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3431–3440. DOI: 10.1109/CVPR.2015.7298965.

[18] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, Feb. 1991. DOI: 10.1162/neco.1991.3.1.79.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for Large-Scale image recognition," in *International Conference on Learning Representations*, 2015.

[20] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996. DOI: 10.1007/BF00058655.

[21] R. E. Schapire, "The strength of weak learnability," *Machine Learning*, vol. 5, no. 2, pp. 197–227, Jun. 1990. DOI: 10.1007/BF00116037.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[23] T. Minka, "Estimating a dirichlet distribution," MIT, Tech. Rep., 2000.

[24] M. Sklar, "Fast MLE computation for the dirichlet multinomial," May 2014. arXiv: 1405.0099.

[25] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop, coursera: Neural networks for machine learning," University of Toronto, Tech. Rep., 2012.

[26] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 3213–3223. DOI: 10.1109/CVPR.2016.350.

[27] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of The 33rd International Conference on Machine Learning*, M. F. Balcan and K. Q. Weinberger, Eds., ser. Proceedings of Machine Learning Research, vol. 48, New York, New York, USA: PMLR, 2016, pp. 1050–1059.

[28] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional Encoder-Decoder architectures for scene understanding," Nov. 2015. DOI: 10.13140/RG.2.1.2985.2407. arXiv: 1511.02680 [cs.CV].