Robustness Against Unknown Noise for Raw Data Fusing Neural Networks

Mario Bijelic[1,2], Christian Muench[1,3], Werner Ritter[1], Yuri Kalnishkan[3] and Klaus Dietmayer[2]

***Abstract*— Adverse weather conditions are extremely challenging for autonomous driving as most state-of-the-art sensors do not function reliably under such circumstances. One method to increase the detection performance is to fuse the raw data signal with neural networks that learn robust features from multiple inputs. Nevertheless noise due to adverse weather is complex, and in addition, automotive sensors fail asymmetrically. Neural networks that perform feature level sensor fusion can be particularly vulnerable if one sensor is corrupted by noise outside the training data distribution compared to decision level fusion. The reason for this is that no built-in mathematical mechanism prevents noise in one sensor channel from corrupting the overall network even though other sensor channels may provide high-quality data. We propose a simple data augmentation scheme that shows a neural network may be able to ignore data from underperforming sensors even though it has never seen that data during training. One can summarize this as a learned ”OR” operation at fusion stage. This learned operation is also generally applicable to other noise-types not present during training.**

I. Introduction

Humans experience the world through multiple modalities which enhance their understanding compared to relying on a single modality (e.g. audio). Algorithms may benefit from multiple modalities as well. Consider the multi-sensor systems used in autonomous car prototypes. One sensor may perform best in broad daylight (e.g. RGB camera) whereas another sensor shows its strength in heavy fog (gated camera) as can be seen in Fig. 2.

In the automotive sector and other research areas, such systems often utilize so-called late fusion, where detections from different sensors are fused on a late decision level, usually with a high-level decision logic. Such methods are found in occupancy grid maps and different types of tracking algorithms.

Recently, the fusion of raw data signals has been applied at different processing steps. For low-resolution lidar sensors and radar sensors, the raw signals have been directly incorporated into extended object tracking algorithms [1]. For high-resolution sensor data, e.g. cameras or high-resolution lidar scanners, additional algorithms such as neural networks are necessary to extract possible objects. Current best-performing methods are based on neural networks and ranked according to the KITTI 2d/3d [2] object detection benchmarks. Several object detection algorithms emerged fusing camera and lidar data such as the MV3D [3] network and the AVOD [4] network. In [3] the networks are based on a mixture of a lidar top view, lidar front view and camera image data. Especially the lidar top view is important. Based on the lidar top view regions of interest (ROI) are extracted and projected into other sensor frames. The fusion is performed by pooling the ROI and averaging the feature maps. Several processing layers are stacked on top and the bounding box is regressed for each frame. The authors called this process deep fusion. In [4] the ROI extraction relies on a common learned region proposal network (RPN) in-between camera and lidar. Such neural networks can be particularly vulnerable if one sensor is corrupted with noise outside the training data distribution. This danger arises from a missing mathematical mechanism that may prevent a corrupted sensor stream from affecting the overall neural network and its output, as will be shown experimentally. Typically two lines of defense against noisy real world data can be formulated. The first is increasing the number of training examples through additional data collection campaigns and the second is the use of data augmentation schemes which address specific disturbances. Typical noise patterns in adverse weather are a degeneration of contrast due to haze, blurring due to small water droplets on the windshield or artefacts such as sunglare. However, it might not be possible to address the complexities of adverse weather conditions and hardware

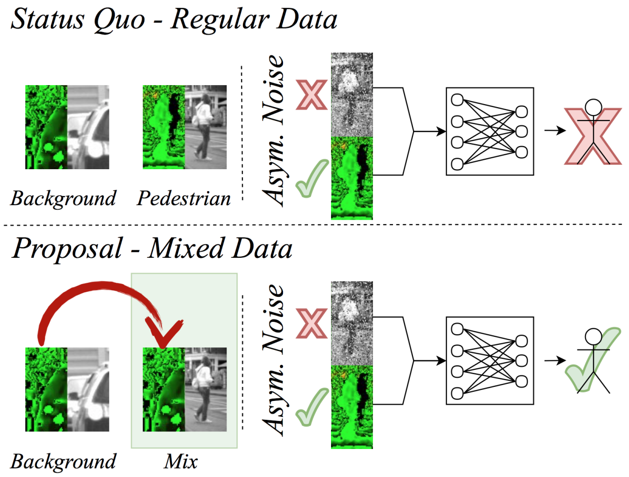

Fig. 1: Proposed learning technique: Training a fusion network with mixed examples out of ’background’ and ’object class’. This enables the network to learn a ’or’ operation and to retrieve information from one channel even if the second channel is disturbed with unknown noise.

[1] The authors are with Daimler AG, Wilhelm-Runge-Str. 11, 89081 Ulm, Germany. {mario.bijelic, christian.muench, werner.r.ritter}@daimler.com. [2] The authors are with University Ulm, Albert-Einstein-Allee 41, 89081 Ulm, Germany. [3] The authors are with Royal Holloway, Egham Hill, TW20 0EX Egham, United Kingdom.

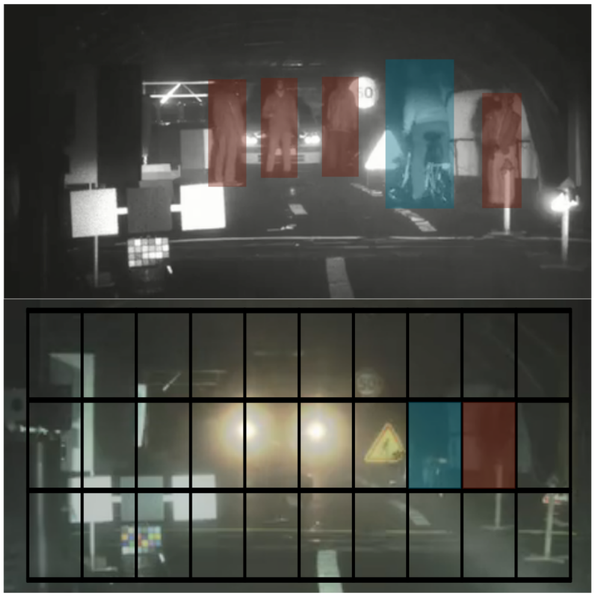

Fig. 2: Image discretization into Anchor boxes [6]/pooled ROI translates to a discretization into smaller classification and bounding box regression tasks. Here demonstrated for a gated imager (top) and a standard imager (bottom) in a fog chamber. Taken from [7]. Notice that the pedesrtrians (red boxes) are clearly visible in the gated imager but not so for the standard imager.

malfunctions through these two lines of defense as they can easily overlook disturbances and combinations thereof which are rare but may profoundly alter the performance of a neural network based fusion algorithm.

Therefore, a third line of defense for neural networks is needed which generalizes to noise patterns that were not introduced during training. In this work, an ”OR” learning method is proposed which neglects a disturbed sensor stream and retrieves the correct class information from the undisturbed sensor stream for the case of two unknown noise patterns (located outside training data distribution). To simplify this further: When a neural network is presented with a background image in one channel and a car (object) image in the other channel, then it should predict the car (object) label. Hence, the operation ”Background OR Object = Object” is important where ”Background” is replaced with ”Unknown Noise(Object)” without changing the outcome. This is not the default behaviour of feature level sensor fusion in neural networks, i.e. often the result is ”Unknown Noise(Object) OR Object = Background”. This shortcoming is addressed by a data augmentation scheme illustrated in Fig. 1. Experiments on Kitti and the Gavrila pedestrian dataset [5] are performed to demonstrate the effectiveness of the data augmentation scheme.

A. Related Work

To the best of the author’s knowledge, there does not appear to be any published work on increasing the robustness of fusion algorithms against unknown noise patterns in one sensor channel. Usually, publications focus on specific noise patterns, commonly encountered in their particular line of research or as data augmentation schemes [6], [8] to prevent overfitting. In general multimodal sensor fusion is applied in different fields of research. [9] points out that algorithms that use multiple modalities have been applied to problems such as audio-visual speech recognition, affect/emotion recognition and gesture recognition. In particular, fusing multiple modalities (e.g. audio, video) is a problem that researchers have engaged in for at least 25 years. Neural networks are one of many models that are used with the fusion of modalities being performed at the feature level. A meta-analysis on multimodal algorithms in the area of affect/ emotion recognition done by [10] shows a significant improvement of the algorithms if multiple modalities are used compared to unimodal algorithms (e.g. audio only). Another recent review by [11] reveals that multimodal algorithms are at least as widely used in the affect/emotion recognition community as unimodal ones. With the modality combination audio, video and text being by far the most favoured lately. When it comes to the type of fusion, both, decision-based fusion and feature based fusion are applied.

Publications dealing with noisy input data seem to focus on improving the accuracies of the algorithm through signal enhancement. This can be done by data augmentation (i.e. train on the noise) or preprocessing of the data such as Principal Component Analysis that can deal with specific noise types (e.g. Gaussian noise) or different signal enhancement networks that increase the signal further [12]–[14]. Robustness against unknown samples outside of the training data distribution affecting the input channel asymmetrically does not seem to be a major concern. Usually, existing work concentrates on domain-specific noise patterns.

There is also related work on leveraging multiple modalities in the intelligent vehicles community using neural networks. For RGB-D data, a color image combined with depth information captured using a stereo camera, has been included in recent publications [15]–[19]. RGB combined with infrared fusion using a convolutional neural network was first performed by [20]. Additionally, there are at least three publications by [3], [4], [21] that fuse RGB with Lidar data. Generally, researchers make use of data augmentation to enhance the robustness of their detectors. [16] create artificial noise that resembles real noise in the depth data from a stereo camera. Again there does not seem to be a focus on noise outside the training data distribution that may degrade the overall performance of the fusion algorithm.

This is, of course, different in the adversarial attacks research community [22]. The goals they try to achieve are usually two-fold. First, they want to trick a neural network into producing false detections. Second, they want to be able to defend against such attacks. Our research does not consider adversarial attacks for now. We focus purely on general noise patterns that suppress the detection rate of objects (i.e. the sensor is blind) by affecting one sensor only. Additionally, we do not aim to improve the prediction of noisy inputs but enable the network to ignore those inputs if it fails to process them properly.

B. Contribution

The following experiments demonstrate that a neural network that performs sensor fusion may be robust against unknown noise patterns in one sensor channel given the proposed data augmentation scheme (see Fig. 1). The analysis is based on two simplifications:

First: A convolutional neural network with two channels (i.e. two sensors) is considered. Instead of detecting objects a classification task is examined where the neural network is presented with an image pair (background or object) and needs to decide which class it belongs to. Though the authors believe that the method can be generalized to two-stage detectors [4] where the second stage performs a classification task on features extracted from the first stage. Basically, the region proposals corresponding to objects can be replaced with background proposals for one sensor channel (the other channel is untouched). As with the data augmentation scheme presented here the second stage would need to learn the operation "Background OR Object = Object" which should increase the performance against asymmetrical noise.

Second: Additionally, it is assumed that it is not known during test time if any neural network channel is disturbed. Therefore, it is not possible to adapt the prediction thresholds, which in turn translates to fixed thresholds at training and test time. Thus, if noise in one sensor channel lowers the prediction scores, the overall performance of the neural network is diminished (low recall).

II. Robust learning technique

The robust learning technique (RLT) is formulated as follows: A two-channel neural network is presented during training with either two object images, two background images or a mixed example with one object and one background image in either channel. If at least one object image is presented, the target classification task is to obtain an object classification otherwise the training example should be classified as background. In total, a two-channel neural network learns to handle the "OR" operation and to predict the object label if the network is presented with at least one undisturbed object image. The technique generalizes to different noise types applied to an object sample (e.g. image of a car). Even though the noise type was not presented during training. The learning technique was applied to four classification experiments on CIFAR [23], MNIST [24], KITTI [2] and to the Gavrila pedestrian dataset (GavData) [5] using two different neural network architectures SqueezeNet [25] and LeNet style convolutional neural networks [26]. The generalization is shown by disturbing one sensor channel by random Gaussian pixel noise (RGPN) or Gaussian blur (GBLUR) during test time. Due to space constraints only experiments on KITTI and GavData are explicitly mentioned and shown. Similar results can be achieved on MNIST and CIFAR with LeNet style convolutional neural networks. The authors provide the corresponding hyperparameters and network architectures upon request.

III. Evaluation Metrics

The performance of the network and the shift in the probability distribution of the prediction scores are evaluated. The latter is done with the Brier score

$$Br = \sum_{i}^{N} (1 - p_i)^2 / N \tag{1}$$

which measures the probability offset per class. N denotes the number of class examples and p_i the probability per example i. Usually, the Brier score goes up for a class if the uncertainty rises.

Measuring network performance is done with the recall and precision metrics that are best summarized in the prediction threshold agnostic precision recall curves (PRC). As different PRC implementations exist, this work follows the implementation in [27], [28]. Upon deployment of the network, it is assumed that the prediction threshold is fixed at some pre-defined combination of recall and precision values on the PRC, hereafter referred to as the optimal performance point. To achieve a threshold independent analysis the mean average precision (mAP) can also be considered, which is the PRC area under the curve. The mAP will be used to select the best performing networks as pre-trained models, though it is not useful for exploring the network performance when it is disturbed by noise as only fixed prediction thresholds are considered at deployment. The reason for this is that the occurrence of noise is assumed to be unknown (i.e. cannot change threshold dynamically as a counter measure).

Furthermore, it will be shown that the RLT does not change in certainty if one sensor channel is disturbed by unknown noise and a threshold adaptation is unnecessary to achieve good results.

The empirical reasoning of the observed effects is given through an activation analysis with t-SNE [29]. Throughout the evaluation, the results are compared to traditional learning technique (TLT), where no "OR" operation was learned and only symmetrical training examples where shown. E.g. just object or background pairs.

A. Noise

Two different kinds of noise, namely random Gaussian pixel noise (RGPN) and Gaussian blur (GBLUR), are applied. RGPN is sampled per colour channel and pixel from a Gaussian normal distribution with a standard variation of σ. Afterwards, the noise is added to the clean image. Values below 0 and above 255 are clipped.

GBLUR convolves the image I with a Gaussian kernel $g(x)$ with width σ,

$$g(x) = \exp\left(-\frac{x^2}{2\sigma^2}\right) / \sqrt{2\pi\sigma^2}, \tag{2}$$

$$I(x) = \int I(\tau) g(x - \tau) d\tau. \tag{3}$$

As the networks could be trained asymmetrically each channel is disturbed independently during evaluation. For KITTI the results for the worst performing channels are given. For

TABLE I: Training hyperparameter ranges for KITTI and GavData experiments.

Training Regime	Dataset	GavData	KITTI
Single Channel - Base Network	base lr/ weight decay	0.04 - 0.001/ 0.0005	0.01/ 0.0004
	batch size/ total training epochs	30-500 / 30-100	32/ 18.2
	decay policy	fixed and step down	fixed and step down
	decay epochs/ decay rate	10/ 0.1	7.3/ 0.1
Two Channel - Fusion Network	base lr/ weight decay	0.04 - 0.001/0.0005	0.0002-0.02/ $4 \cdot 10^{-8} - 4 \cdot 10^{-3}$
	batch size/ training epochs	30-500/ 30-100	32/ 6,12, 24
	decay policy	fixed and step down	fixed and step down
	decay epochs stpsize/ decay rate	10/ 0.1	2, 4, 8/ 0.1
	RLT examples in % of object example	0-20	50
Overall	Data augmentation	vertical image flip	vertical image flip
	Optimization method	SGD, momentum 0.9	SGD, momentum 0.9
	bottom layers lr = 0 during training	no	yes
	Target class	pedestrians	cars
	image size [pixel]	48x96	112x112

GavData the results are averaged over both channels as the two modalities are inherently distinct.

IV. Experiments

Experiments on the KITTI dataset and GavData will show that the RLT is applicable to a variety of hyperparameters, learned by the downstream classification layers after feature extraction and ensures robustness if one sensor channel is disturbed with an unknown noise.

A. KITTI Experiments

The KITTI experiments are designed to show that the network can learn the RLT for entangled input information, where input data in both channels are different but represent the same class. Since the input images are not the same the network is not merely learning simple rules such as: ”The images are not identical. Hence, the image stream is disturbed”. Therefore the network must learn more complex mechanisms to achieve robustness if one channel is disturbed. To prove this argument, the feature extraction layers are fixed (e.g. learning rate = 0) while only the top layers following the feature fusion layer (i.e. ”fire8” module) are trained.

1) Dataset creation: Approx. 14.000 images of cars (as target object) and 340.000 images of the general background were extracted from the KITTI dataset [2]. The background patches were selected at random in a range from 40 x 40 to 150 x 150 pixels (sampled uniformly) with a maximum intersection over union (IoU) of 0.15 with any car bounding boxes in the image frame. An IoU > 0 ensures that the network only classifies car patches that fit the patch properly as cars, which helps to suppress the false positive rate by reducing overfitting [8].

The car patches were extracted via the corresponding bounding boxes where either the height or the width was increased to get square patches. Cars that are heavily occluded or partially outside the image frame are neglected. Finally, all background and car patches are resized to 112 x 112 pixels by linear interpolation. No additional preprocessing steps were performed. Training and validation data is split 50/50 with training and validation images originating from different sequences to reduce correlation between training and validation data. Image pairs are selected to match the same label but represent different class realizations. Therefore, both images always belong to the object or background class but are usually different images from those classes.

2) Training Procedure: The overall two-channel network is built upon the SqueezeNet v1.1 [25] architecture. SqueezeNet is chosen as its fast training time below one hour per experiment (Titan XP and Intel i7-6950x) allows multiple experiments. The weights for an one-channel SqueezeNet, pre-trained on ImageNet [30], were fine-tuned on the KITTI data. The hyperparameters were picked randomly from table I to show that the RLT applies for a wide range of hyperparameter settings. For both the TLT and RLT same hyperparameter ranges were used.
The number of mixed training examples in RLT was not finetuned to a specific number of car/background and background/car pairs in the training data to increase the performance.

The two-channel neural network is obtained by concatenating the ”fire8” layers of two identical pre-trained single channel SqueezeNets. To make sure that the ”OR” Operation is learned by the downstream classification layers only, all weights up to and including layer ”fire8” will be fixed for the KITTI experiments (learning rate = 0), whereas the layers following layer ”fire8” are randomly initialized by Xavier and Gaussian initialization, similar to the original implementation [25]. As both inputs in both channels are from the same training distribution, the weights point to the same feature space. The fused network is trained with RLT and TLT. Both training approaches will be analyzed.

B. GavData Experiments

Experiments on the Gavrila pedestrian dataset (GavData) [5] show the RLT and TLT performance on multiple modalities. The dataset contains cropped background and pedestrian images from stereo intensity measures, stereo depth and stereo flow. To simulate a fusion system each input channel corresponds to one modality, e.g. stereo depth or intensity.

1) Dataset creation: The dataset is already preprocessed (e.g. background and pedestrian images are extracted) and split into a training and validation sub-dataset by [5]. The fusion will be based on stereo depth and intensity images.

2) Training Procedure: The network is trained similar to the KITTI experiments with an architecture based on the SqueezeNet v1.1 [25]. For each channel several single-channel base networks have been trained with parameters given in table I. Based on the pre-trained weights from [30] the networks were fine-tuned on either stereo depth or intensity data. The best performing networks (w.r.t. mAP) on both modalities were selected as base networks in the fusion network. Single networks reached a mAP of about 97% for intensity images and 95% for stereo depth respectively.

The depth was fed as a three-channel 8bit image with the depth information being split into the three channels metres, centimetres and millimetres. The values have been scaled to use the full 0-255 range. As the stereo depth accuracy is in the centimetre region, the third channel contains mainly noise which the network learns to ignore. Hence, the input in the third channel is set to zero without any observable decrease in mAP. The main advantage of this method is to enable the network to use the pre-trained weights from [30] to accelerate the training process. The same results were achieved by providing depth as a single channel float value input, though this works only if the first layer is trained from scratch, which slows down the convergence. A further increase in the mAP on stereo data may be achieved by applying different input modalities as in [31]. As the networks already achieved a high mAP of 95% and the optimization of stereo data was not a primary objective, no further optimizations were applied.

The best performing depth and intensity models have been chosen, and the weights were transferred to the fused neural network where the modalities are concatenated at the ”fire8” module. Downstream layers where initialised randomly similar to KITTI experiments.
Fixing the learning rate to 0 for layers below the concatenation is detrimental to the performance as features were not extracted based on a common feature map. Therefore the feature distribution is different, which limits the TLT and RLT to a mAP of 82% on the undisturbed validation set. Fine-tuning the upstream layers as well recovers the single network performance with a slightly higher mAP in the range of 96% to 98%.

V. Results

The results on KITTI and GavData are split into three parts. The first deals with the activation pattern evaluation through t-SNE leading to the RLT. The second considers the best performing neural network for the TLT and the RLT respectively and shows their uncertainty changes through the Brier score. The third demonstrates the performance of a number of neural networks trained with a wide range of hyperparameters given in table I.

A. Activation evaluation on KITTI

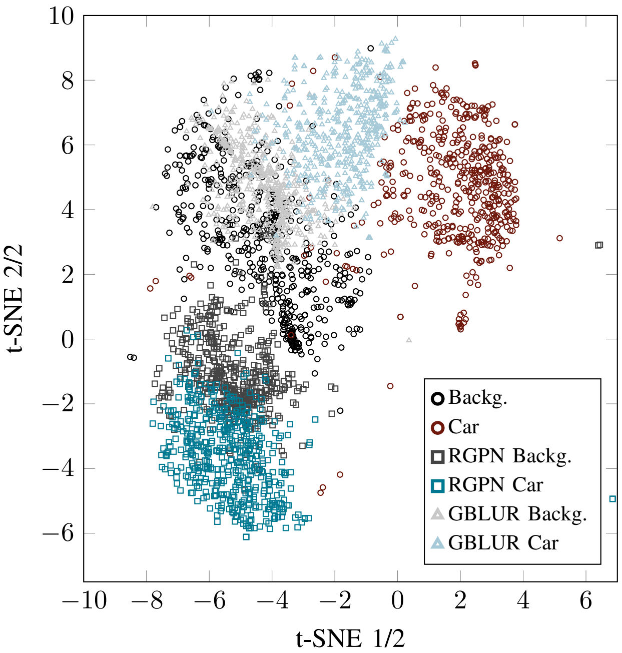

Fig. 3: t-SNE [29] clustering results for a 2d-output space, showing undisturbed and noisy car and background image activations at the fusion layer (”fire8”) on KITTI.

TABLE II: Recall Statistics for 45 models trained on GavData and 63 models trained on KITTI. Hard numbers from RLT and TLT corresponding to fig. 5. The reference (Ref.) shows the recall for undisturbed data.

Noise RLT/TLT	Ref. TLT	Ref. **RLT**	GBLUR TLT	GBLUR **RLT**	RGPN TLT	RGPN **RLT**
GavData						
median	99.55	99.41	91.45	97.08	46.13	92.20
u-quart.	99.64	99.55	92.44	97.88	46.61	93.50
l-quart.	99.48	99.33	90.69	96.36	45.64	90.86
u-whisk.	99.73	99.73	96.61	98.70	47.54	95.10
l-whisk.	99.17	98.78	88.73	95.46	40.54	87.63
KITTI						
median	97.1	97.3	83.5	93.0	42.7	86.2
u-quart.	97.3	97.5	84.7	94.4	48.6	89.6
l-quart.	96.9	97.2	81.1	91.1	30.3	78.4
u-whisk.	97.7	97.7	86.8	95.8	54.8	95.4
l-whisk.	95.0	96.6	72.7	89.7	8.5	68.4

The RLT is based on two key observations. First, if a network is trained with a ”background” class, containing high variance samples, the network learns to draw close decision boundaries around the low variance ”car” class, where images outside the ”car” class are classified as background. Second, if a class is disturbed by noise, the feature activations are shifted to the background class.

This can be understood by clustering the activations at fusion level (i.e. ”fire8” module) by applying the t-SNE [29] algorithm. Results can be seen in Fig. 3 with a total of 3000 processed images from the background and the car classes (KITTI data). Additionally, both classes have been disturbed by RGPN (σ = 0.9) and GBLUR (σ = 9).

The activation patterns of disturbed input images are clearly different from those of undisturbed images (see Fig. 3). Therefore, these patterns can be learned by the layers following the feature level fusion during training with all layers up to the ”fire8” module being fixed (learning rate = 0). This already gives an intuition as to why training on

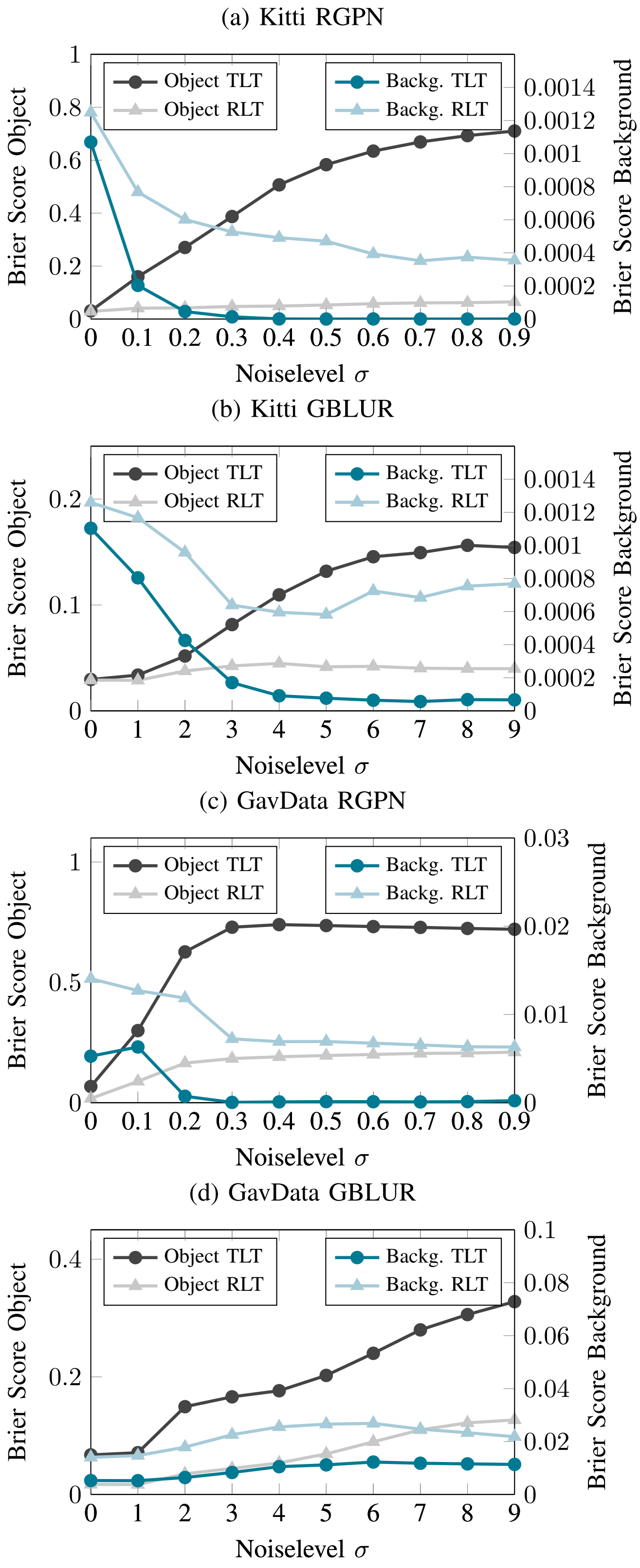

Fig. 4: Brier Score at different noise intensities for different unkown noise (a,c) RGPN/(b,d) GBLUR (not presented during trained) and datasets (a,b) KITTI and (c,d) GavData. Both learning techniques TLT and RLT are compared. Only one channel is disturbed while the other one is kept undisturbed.

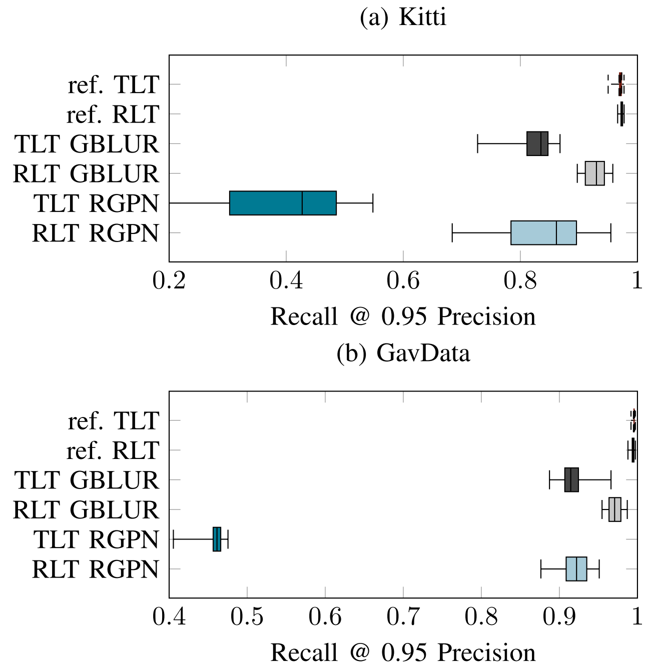

Fig. 5: Recall values for (a) 63 models trained KITTI data and (b) 45 trained on GavData. Hyperparameters can be seen in table I. The optimal perfomance point has been fixed for each model at precission of 95% the corisponding recall is shown as reference. As the noise is assumed to be unkown the threshold can't be optimized and therefore the optimal threshold on undisturbed data is applied to retrieve the corosponding recall on disturbed data. One channel was disturbed with either GBLUR $\sigma = 9$ and RGPN $\sigma = 0.9$.

one specific noise type does not seem to transfer to another noise type. The network may simply redraw the decision boundary to include disturbed car images (e.g. red circles and blue triangles for GBLUR) and since each noise type has a very specific activation pattern, this new decision boundary does not necessarily include other never before seen noise types (e.g. blue squares for RGPN still outside). Whereas for the training regime where the two-channel network is explicitly fed with mixed classes, it has to learn to recognize car features from the object channel no matter what kind of background image is present in the other channel.

Why is the network forced to generalize this way? The interpretation is that the variance of the background class is high compared to the variance of e.g. RGPN. It is easier for the top layers to learn to extract the object features from the undisturbed sensor channel irrespective of the type of noise in the disturbed channel compared to learning a representation of the whole background class (excluding RGPN). Specific noise patterns such as "if it is dark, then camera underperforms, but lidar works well" are easy to learn, but do not force the neural network to generalize to other scenarios such as "Features extracted from camera are strange (because of unknown noise), what now?". The standard training procedure does not necessarily force the

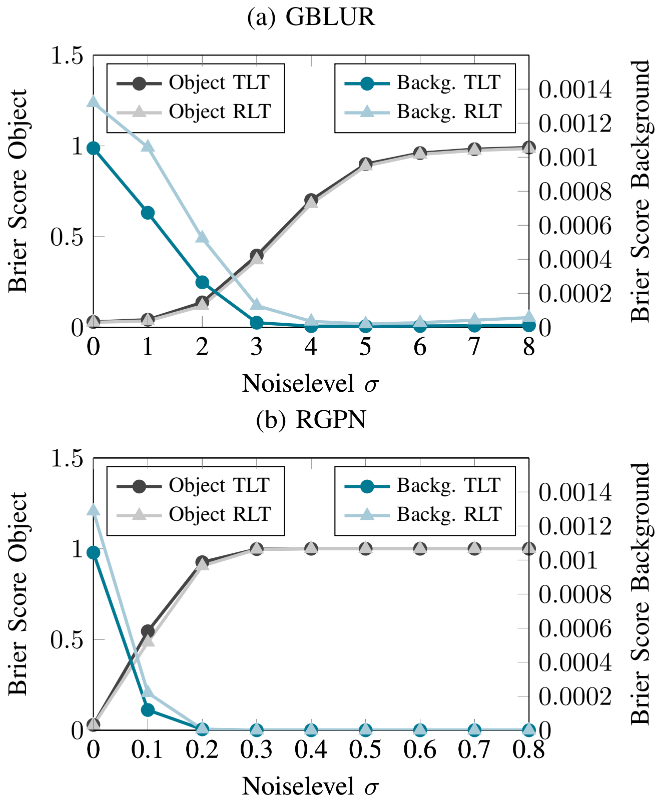

Fig. 6: Brier Score at different unkown noise intensities for different noiselevels (a) GBLUR/(b) RGPN. Trained on Kitti with TLT and RLT. Both channels are disturbed equally.

neural network to learn a general rule on how to handle this scenario. Though, the data augmentation scheme presented here does exactly that as can be seen in the following evaluation.

B. *Uncertainty Changes*

Second let us discuss the Brier scores given in Fig. 4 and 6. Here the object and the background class is plotted against the noise level of RGPN and GBLUR, which has been feed to the network. The networks themselves have not been trained on such kind of noisepatterns and therefore GBLUR and RGPN resemble unknown noise.

In Fig. 4 only one channel is disturbed. Here the RLT appears to benefit the robustness of the network, because when the network is disturbed the increase in the Brier score on the object class is much less pronounced compared to the network that was trained with the TLT. This is true for KITTI and GavData.

Though, disturbing the background class increases the confidence of the networks that it is indeed a background example (score decreases) which is consistent with our interpretation that the noise pushes the feature activation's into the background class. The effect is less significant for the RLT. If a background example contains object features even though it is not an object, then these features are recognized by the upper layers more easily when the RLT is performed. This is not a bad thing as our aim is not to decrease the false positive rate of the individual sub-networks by fusing them together. The apparent superiority of the TLT on the noisy background class only stems from the fact that the network is unable to recognize object features (even false positives) when it is disturbed (i.e. objects are classified as background). Ideally, the expected Brier score of the background class stays at the $\sigma = 0$ value, which means that the certainty should be not affected trough noisy input. Hence, the RLT also outperforms the TLT on the background class.

Fig. 6 illustrates the behaviour if both sensor channels are disturbed. Both TLT and RLT show the same performance and are almost indistinguishable which is to be expected as there is no undisturbed sensor channel left. The RLT is not a training scheme to extract information from noise but to enable the network to ignore sensor channels that it cannot properly process. Therefore, this result is to be expected, i.e. w.r.t. the Brier score the TLT and the RLT give rise to neural networks that are almost identical on undisturbed test data and disturbed test data where all sensor channels receive noisy inputs.

C. *Performance Drop*

Fig. 5 and table II provide the performance of the TLT and the RLT w.r.t the recall and precision metrics. As mentioned before the thresholds of the neural networks are fixed at test time with a corresponding precision on the undisturbed test data of 95%. When the test data is disturbed the performance of the neural network shifts to a higher precision (i.e. more than 95%) whereas the recall drops (empirical observation). Disturbing one sensor channel affects the recall for TLT networks significantly but much less so for the RLT. Again this is true for the Kitti and the GavData experiments. As with the Brier scores the two training schemes are almost indistinguishable on the undisturbed test data, however the RLT clearly outperforms the TLT if one channel receives noisy inputs.

VI. Conclusion

Sensor fusion networks based on neural networks like in [3], [4] are particularly vulnerable if one sensor channel is disturbed, because no mathematical mechanism prevents noise corrupting downstream classification layers.

Therefore a third line of defense, along with more data and data augmentation, has been introduced. The introduced method (RLT) specifically tackles the problem if one sensor channel is disturbed while another sensor stream is undisturbed. Such asymmetries can be observed quite often in adverse weather as seen in Fig. 2 or in [7] [32].

The problem has been broken down into classification tasks because usually each object detection discretizes an image trough anchor boxes or ROI into multiple regions for which several classification and bounding box regression tasks are solved (Fig. 2). Usually training an object detector can take more than an entire day, while a SqueezeNet classification task is trained within one hour. Therefore enabling the training of multiple models in a short time, thus demonstrating the increase in robustness of the RLT across a whole range of hyperparameters.

Implementing the RLT for a fused object detector like in [4] should be straight forward. During training pooled background image pairs, which are also necessary for hard example mining, can be queued and loaded into an image pool. The image pool is updated during training. At a given rate, which is a hyperparameter, background images can be taken to replace pooled objects pairs and to create mixed examples out of object and image pairs. This process has to be applied at two stages: once at the region proposal network (RPN) and once at the fusion classification stage.

The method has been experimentally validated on four different datasets and two different networks. Thanks to the Gavrila pedestrian dataset (GavData) it also has been tested for multiple modalities and a learned common feature map. Furthermore an experimental reasoning was provided in Fig. 3 by clustering the activation with t-SNE [29] at the corresponding fusion layer. Application of the robust learning technique (RLT) outperforms traditional learning with no mixed examples significantly in case of unknown noise patterns. The learned ”OR” operation helps to neglect the disturbed sensor stream and retrieves information from the undisturbed network channel.

The key idea is that a noisy object image ends up in the background feature space because the object feature space is small and specific. In other words, the network needs to learn the ”(object or background) = object” mapping which transfers to ”(object or noisy object) = object”. The inverse approach does not seem to work, i.e. training on a specific noise pattern does not necessarily transfer to other noise patterns. The reason for this is probably that the network can learn a decent representation of, e.g. Gaussian Noise and Gaussian Blur. It does not learn the ”OR” mapping. Instead, it learns that ”Gaussian Noise(object) = object”.

Acknowledgment

The research leading to these results has received funding from the European Union under the H2020 ECSEL Programme as part of the DENSE Project, contract number 692449.

References

[1] K. Granström, M. Baum, and S. Reuter, “Extended Object Tracking: Introduction, Overview and Applications,” *ISIF Journal of Advances in Information Fusion*, 2017.

[2] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The kitti dataset,” *IJRR*, 2013.

[3] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-View 3D Object Detection Network for Autonomous Driving,” *CVPR*, 2017.

[4] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. Waslander, “Joint 3d proposal generation and object detection from view aggregation,” *arXiv preprint arXiv:1712.02294*, 2017.

[5] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. Gavrila, “Multi-cue pedestrian classification with partial occlusion handling,” *CVPR*, 2010.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single Shot MultiBox Detector,” *ECCV*, 2016.

[7] M. Bijelic, T. Gruber, and W. Ritter, “Benchmarking Image Sensors Under Adverse Weather Conditions for Autonomous Driving,” in *IV*, 2018.

[8] A. Shrivastava, A. Gupta, and R. Girshick, “Training region-based object detectors with online hard example mining,” *CVPR*, 2016.

[9] T. Baltruaitis, C. Ahuja, and L.-P. Morency, “Multimodal machine learning: A survey and taxonomy,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[10] S. Dmello and J. Kory, “A review and meta-analysis of multimodal affect detection systems,” *ACM Computing Surveys*, 2015.

[11] S. Poria, E. Cambria, R. Bajpai, and A. Hussain, “A review of affective computing: From unimodal analysis to multimodal fusion,” *Information Fusion*, 2017.

[12] S. Diamond, V. Sitzmann, S. P. Boyd, G. Wetzstein, and F. Heide, “Dirty pixels: Optimizing image classification architectures for raw sensor data,” *CoRR*, 2017.

[13] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, “Deblurgan: Blind motion deblurring using conditional adversarial networks,” *CVPR*, 2018.

[14] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, “An all-in-one network for dehazing and beyond,” *CoRR*, 2017.

[15] O. Mees, A. Eitel, and W. Burgard, “Choosing Smartly: Adaptive Multimodal Fusion for Object Detection in Changing Environments,” *IROS*, 2016.

[16] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, “Multimodal Deep Learning for Robust RGB-D Object Recognition,” *IROS*, jul 2015.

[17] M. Schwarz and S. Behnke, “Data-efficient Deep Learning for RGB-D Object Perception in Cluttered Bin Picking,” *ICRA*, 2017.

[18] M. Schwarz, H. Schulz, and S. Behnke, “RGB-D Object Recognition and Pose Estimation based on Pre-trained Convolutional Neural Network Features,” *ICRA*, 2015.

[19] D. O. Pop, A. Rogozan, F. Nashashibi, and A. Bensrhair, “Incremental Cross-Modality Deep Learning for Pedestrian Recognition,” *IV*, 2017.

[20] J. Wagner, V. Fischer, M. Herman, and S. Behnke, “Multispectral Pedestrian Detection using Deep Fusion Convolutional Neural Networksicl,” *ESANN*, 2016.

[21] J. Schlosser, C. K. Chow, and Z. Kira, “Fusing LIDAR and Images for Pedestrian Detection using Convolutional Neural Networks*,” *ICRA*, 2016.

[22] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and Harnessing Adversarial Examples,” *ICLR*, 2015.

[23] A. Krizhevsky, “Learning multiple layers of features from tiny images,” in *None*, 2009.

[24] Y. LeCun and C. Cortes, “MNIST handwritten digit database,” *None*, 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[25] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size,” *arXiv:1602.07360*, 2016.

[26] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, 1998.

[27] F. Pedregosa, G. Varoquaux, and et. al., “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, 2011.

[28] 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*, 2010.

[29] L. Van Der Maaten and G. Hinton, “Visualizing Data using t-SNE,” *Journal of Machine Learning Research*, 2008.

[30] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A Large-Scale Hierarchical Image Database,” in *CVPRI*, 2009.

[31] S. Gupta, R. Girshick, P. Arbelaez, and J. Malik, “Learning rich features from RGB-D images for object detection and segmentation,” 2014.

[32] M. Bijelic, T. Gruber, and W. Ritter, “A Benchmark for Lidar Sensors in Fog: Is Detection Breaking Down?” in *IV*, 2018.