IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 22, 2021, accepted November 19, 2021, date of publication December 28, 2021, date of current version January 20, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3138978

One Versus All for Deep Neural Network for Uncertainty (OVNNI) Quantification

GIANNI FRANCHI[1], ANDREI BURSUC[2], EMANUEL ALDEA[3], (Member, IEEE), SÉVERINE DUBUISSON[4], AND ISABELLE BLOCH[5]

[1]ENSTA Paris, Institut Polytechnique de Paris, 91120 Palaiseau, France
[2]Valeo.AI, 75017 Paris, France
[3]SATIE, Université Paris-Saclay, 91190 Gif sur Yvette, France
[4]CNRS, LIS, Aix Marseille University, 13007 Marseille, France
[5]Sorbonne Université, CNRS, LIP6, 75005 Paris, France

Corresponding author: Gianni Franchi (gianni.franchi@ensta-paris.fr)

This work was supported in part by the Agency (ANR) Project MOHICANS under Grant ANR-15-CE39-0005, "Towards Modelling High-density Crowds for Assisting Planning and Safety," and in part by the Saclay-IA Cluster and CNRS Jean-Zay Supercomputer for the computation resources under Grant 2020-AD011011970.

ABSTRACT Deep neural networks (DNNs) are powerful learning models, yet their results are not always reliable. This drawback results from the fact that modern DNNs are usually overconfident, and consequently their epistemic uncertainty cannot be straightforwardly characterized. In this work, we propose a new technique to quantify easily the epistemic uncertainty of data. This method consists in mixing the predictions of an ensemble of DNNs trained to classify *One class versus All* the other classes (OVA) with predictions from a standard DNN trained to perform *All versus All* (AVA) classification. First of all, the adjustment provided by the AVA DNN to the score of the base classifiers allows for a more fine-grained inter-class separation. Moreover, the two types of classifiers enforce mutually their detection of out-of-distribution (OOD) samples, circumventing entirely the requirement of using such samples during training. The additional cost involved by the construction of the ensemble is offset by the ease of use of our proposed strategy and by its enhanced generalization potential, as it does not bind its performance in a given context to specific OOD datasets. The extensive experiments confirm the wide applicability of our approach, and our method achieves state of the art performance in quantifying OOD data across multiple datasets and architectures while requiring little hyper-parameter tuning.

INDEX TERMS Uncertainty estimation, DNN ensembles, one vs all classification, all vs all classification.

I. INTRODUCTION

Anomaly detection is the task of detecting data that deviate from the training distribution. Deep neural networks (DNNs) have reached state-of-the-art performance on machine learning [20], [45], and computer vision tasks [40], [70]. Significant progress has raised interest in adopting them in a wide range of decision-making systems, including safety-critical ones. Yet, one of the main weaknesses of these techniques is that they tend to be overconfident [22] in their decisions, even when they are wrong [22], [25], [63]. This leads to DNNs that might miss detecting anomalies. This issue is difficult to tackle, as the high inner complexity of DNNs results in a poor output explainability.

Anomaly or outlier detection is a wide thematic of research [66]. The objective is to detect rare or corrupted data, that are different from what we consider to be normal data. This research topic has multiple practical applications, such as risk management [81], safety [69] or automatic inspection and non destructive control [38]. Anomalies can also be linked to the knowledge uncertainty [27] of the DNNs. The precise identification of anomalies in DNN predictions is crucial for improving the reliability of such models, and a key step towards their deployment in practical settings.

In order to address this important problem, we propose to leverage a finer quantification of the uncertainty of DNNs. In contrast to most Bayesian DNN techniques [4], [17], [18], [35], [56], or to frequentist techniques such as Deep Ensembles [43], our approach relies on *One versus All* (OVA) training. In the statistical learning community, ensembles of OVA or *One versus One* (OVO) base classifiers for multi-class prediction have been particularly popular in association with Support Vector Machines (SVM), due to SVM being essentially a binary classifier, and to the simplicity of

The associate editor coordinating the review of this manuscript and approving it for publication was Mingbo Zhao.

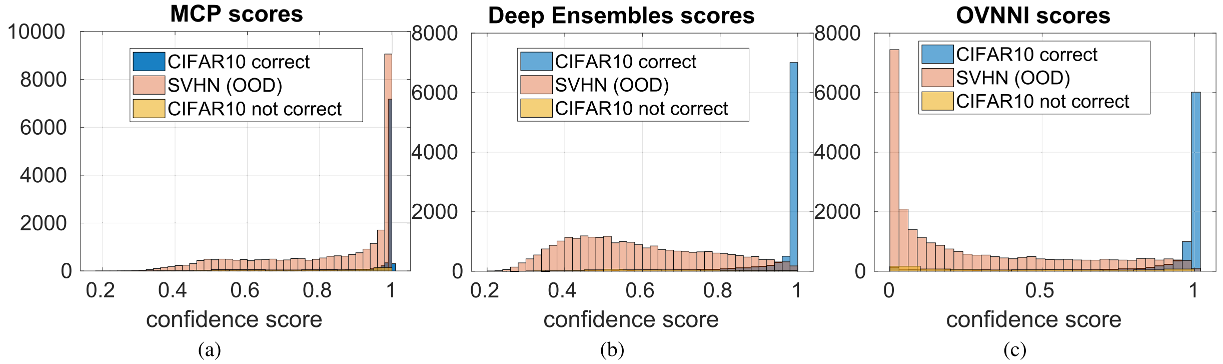


FIGURE 1. Distribution of values of prediction score on correct (in blue) or incorrect (in orange) predictions and OOD (in yellow) samples for different approaches. (a) Maximum Class Probability (MCP) [28], (b) Deep Ensembles [43], (c) OVNNI. All runs use Resnet50 [24] trained on CIFAR-10 and tested on CIFAR-10 and SVHN. Our proposed algorithm OVNNI generates very low prediction scores for OOD data, outperforming Deep Ensembles (current state-of-the-art) and MCP (baseline) on detecting OOD data.

the aggregation rules supported by fundamental theoretical results [37], [41], [79]. The most popular rule in case of OVA ensembles, *winner-takes-all* (WTA), assigns the test sample to the class for which the membership score is the highest. For a binary output, the WTA rule creates in the input space multiple unclassifiable regions, for which the class assignment is not unique, and the standard solution is to rely on continuous membership scores. In contrast to SVM-based learning, nowadays the OVA approach has been mostly discarded when training deep classifiers, in favor of *All vs All* (AVA) learning.

The predictive uncertainty of DNNs is commonly categorized into *aleatoric uncertainty* and *epistemic uncertainty* [30]. The former is related to randomness, typically due to the noise in the data. The latter concerns finite size training datasets. The epistemic uncertainty captures the uncertainty in the DNN parameters and their lack of knowledge on the model that generated the training data. In this paper, we propose to use OVA learning in order to improve the quantification of the epistemic uncertainty of the DNN. The underlying idea of our approach is that the score of a base classifier should be adjusted by a factor which approximates its local reliability in the input space from which the test sample originated. Initially for SVM learning, the reliability has been linked to the average value of the *local* objective function [53], which is approximated using the closest training samples belonging to the respective class. Here we propose to adjust the OVA scores by the score provided by an AVA DNN which will play thus the role of approximating the local class-specific objective function. This strategy allows for a particularly effective detection of out-of-distribution (OOD) samples in the test data, as we can discriminate between samples belonging to unclassifiable regions equally close to *some* classifiable regions, and samples belonging to unclassifiable regions far from *all* classifiable regions.

Figure 1 presents the distribution of the scores provided by the baseline, Deep Ensembles (the current state-of-the-art) and our method, respectively. The baseline is the single AVA classifier, for which the class assignment is performed based on the Maximum Class Probability (MCP) [28]. The baseline is unable to discriminate among in- and out-of-distribution samples, illustrated in blue/yellow and orange in the histograms, respectively. Deep Ensembles produces lower scores for OOD samples, but the in-distribution membership is still overestimated. Finally, OVNNI successfully assigns low scores to the OOD samples, while keeping at the same time the in-distribution scores high.

Our main contributions are the following: **(1)** We propose an effective non-Bayesian technique for uncertainty quantification in OOD data classification, that reaches state of the art results on calibration and on OOD data detection on a variety of datasets, and on all typical metrics. **(2)** We conduct extensive evaluation experiments on multiple computer vision tasks (image classification, semantic segmentation) and datasets (MNIST [46]/Not MNIST [1], CIFAR-10 [39]/SVHN [62], Camvid [8], StreetHazards [27], BDD Anomaly [27]) and compare with strong and recent related methods. We show that OVNNI excels at detecting OOD images and objects. **(3)** We shed a fresh light on *One vs All* classifiers that have been so far rather ignored in the context of DNNs and hope to rehabilitate them for such approaches. Our conclusions are in line with less recent findings, e.g. OVA classifier aggregation [71].

II. RELATED WORK

OOD detection is not a novel problem and has been studied before the deep learning revival in various branches of machine learning under slightly different taks: anomaly [51], outlier [7] or novelty detection [74]. In the last few years, this task has seen increased attention from different communities and has been addressed with: *predictive uncertainty estimation*, *ensemble methods*, *image reconstruction*, etc. In the following we review briefly some of the methods related to our approach.

A. ANOMALY DETECTION

Anomalies are linked to abnormal data detection. As building datasets for anomaly detection is a difficult task, we can make different assumptions. *Assumption 1*: we consider that we can collect data of anomalies; in this case, we assume that normal and abnormal data are available. This case was studied in [21], [54]. We relate to this case as supervised anomaly detection. Since it is hard to collect such data, lately research has focused on unsupervised anomaly detection that does not require any labeled training data [72], [73]. Then, semi-supervised anomaly detection approaches make *Assumption 2*: training data contain no anomalies, and, during inference, tests are performed on normal and abnormal data. Finally, weakly supervised learning considers a set of labeled normal training data, and also has no abnormal training data. This is studied in [17], [28]. In this case, the anomalies represent the lack of knowledge of the DNN and are related to epistemic uncertainty (*Assumption 1*). Our paper focuses on this kind of anomaly detection problem.

B. CLASSIFICATION WITH A BACKGROUND CLASS

In multiple computer vision tasks, *e.g.*, object detection [52], [70], it is common to use a *background* class in addition to the known classes to classify. This leads to a better separation of the classification space and a more discriminative classifier. While this seems to be a reasonable and straightforward approach, for OOD detection, it is likely to suffer from negative dataset bias [78] and thus not generalize to other background objects not seen during training. In our approach, we also use a part of the classes as background when training the individual classifiers, however the overlap of their decision boundaries, coupled with the AVA model, better distinguishes in- from out-of-distributions samples.

C. ANOMALY DETECTION BY RECONSTRUCTION

Anomalies can be detected by training an autoencoder [2], [12] or generative model [50], [73] on in-distribution data, and use the quality of the reconstruction as a proxy OOD, as the autoencoder is unlikely to decode accurately patterns not seen during training. Training such models for accurate and robust reconstruction requires large amounts of data.

D. BAYESIAN APPROACHES

Bayesian Neural Networks (BNN) [61] are elegant, intuitive and easy to reason models, that can capture the epistemic uncertainty through the exploitation of the distributions of their weights. In spite of recent progress that makes them more tractable [4], they are still limited to small or medium-size networks, while most DNNs usually enclose millions of parameters. Gal and Ghahramani [18] aimed for a method to imitate BNNs. To this end they proposed Monte Carlo Dropout (MC Dropout) to estimate the posterior predictive network distribution by sampling different subsets of neurons at each forward pass during test time and aggregate their predictions. In computer vision, MC Dropout is the most popular instance of BNNs due to its speed and simplicity. It has been extended to other tasks, *e.g.*, semantic segmentation [33], pose estimation [34]. However, the benefits of Dropout are more limited for convolutional layers, where specific architectural design choices must be made [33], [60]. Recent OOD benchmarks for semantic segmentation [26], [50] show that MC Dropout still induces many false positives.

E. ENSEMBLES

Ensemble methods are prominent techniques for measuring epistemic uncertainty. They have the potential to encapsulate a true diversity in the weights of the composing models, contrarily to the dispersion introduced by MC Dropout [16], which ultimately focuses on a single mode. Lakshminarayan *et al.* [43] propose training an ensemble of DNNs with different initialization seeds. Vyas *et al.* [80] train an ensemble of classifiers in a self-supervised way on different subsets of the training data, using the left-out data as OOD. Izmailov *et al.* [32] collect weight checkpoints from local minima and average them or fit a distribution over them and sample networks [56]. Franchi *et al.* [17] track weights trajectories across training and compute their distributions, further used for sampling an ensemble of networks. Our approach also exploits ensembles, however each network is specialized on a different classification task. We exploit the complementarity in this ensemble for better OOD predictions.

F. OVA/OVO ENSEMBLES

These aggregation techniques are popular for performing multi-label classification based on an ensemble of binary base classifiers. For OVO, instead of the baseline max-voting aggregation strategy, pairwise coupling [83] or ECOC [15] have been widely used, but the quadratically increasing number of base classifiers may limit significantly OVO applicability in the case of large sets of labels. In contrast, OVA fusion uses a linearly increasing number of base classifiers, and relies in most works on a Winner-Takes-All class assignment based on the maximum class response. To the best of our knowledge, these ensembling methods have not been used for estimating the epistemic uncertainty of DNNs. One-vs-all formulations have also been studied in a more recent publication [65] where an ensemble of one-vs-all DNNs is trained, with a new distance-based loss that can encode the distance of a point from the training manifold, maximizing the binary log-likelihood for the positive class and minimizing it for the negative classes. Despite the interesting results on image classification tasks, this approach does not seem scalable for computer vision tasks such as semantic segmentation.

G. DEEP OOD DETECTION

A recent line of approaches addresses OOD detection through DNNs specific heuristics. Hendrycks and Gimpel [28] established a standard baseline for OOD detection relying on the Maximum Class Probability from softmax. In [14] a confidence branch is attached to a classification network, which is trained to predict OOD samples, while ODIN [49] learns a temperature scaling for softmax values and adversarial perturbation to better distinguish OOD data. Lee *et al.* [48] get a class conditional Gaussian distribution with respect to features that they tune on a dataset with OOD data and in-distribution data. Lambert *et al.* [44] attenuate uncertainty by training on a large composite dataset leading to a more robust DNN. Zendel *et al.* [84] propose a semantic segmentation dataset for checking the confidence score of DNNs. The authors of [3] train a DNN to predict OOD confidence score. Lee *et al.* [47] train a GAN along with the classifier to produce near-distribution examples and enforce lower classifier confidence on GAN samples. Malinin and Gales [57] use Dirichlet networks to build a distribution over the prediction distributions for OOD detection. Most of these methods rely on a OOD dataset during training and are likely to specialize on specific anomalies from these data [29]. In contrast, in our approach we do not require OOD examples during training, as we leverage the multiple one-versus-all classifiers.

III. ONE VERSUS ALL FOR DEEP NEURAL NETWORK FOR UNCERTAINTY QUANTIFICATION (OVNNI)

This section focuses first on the necessary details on the traditional AVA training of a DNN. Then we describe our approach based on additional OVA training.

A. NOTATIONS

- The training and testing sets are denoted by $\mathcal{D}_l = (x_i, y_i)_{i=1}^{n_l}$ and $\mathcal{D}_\tau = (x_i, y_i)_{i=1}^{n_\tau}$, respectively, where x_i and y_i represent the observed sample and the corresponding label, respectively, with n_l and n_τ the size of the training and testing sets, and $i \in \{1 \dots n_l\}$ or $i \in \{1 \dots n_\tau\}$; x_i are input vectors and $y_i \in \{1, \dots, n_{\text{label}}\}$ are class labels. Unless otherwise specified, x_i and y_i, $i \in \{1 \dots n_l\}$, will refer to training data.
- X is the random variable associated with observed samples and Y the one associated with classes.
- The DNN is a function f of the observed data x_i, with $i \in \{1 \dots n_l\}$ or $i \in \{1 \dots n_\tau\}$, and vector $\boldsymbol{\omega}$ that contains the trainable weights. We call $f_{\boldsymbol{\omega}}(x_i)$ the output of the DNN associated with the weights $\boldsymbol{\omega}$ on the data x_i.
- $\mathcal{L}(\boldsymbol{\omega}, y_i)$ is the loss function used to measure the dissimilarity between the output $f_{\boldsymbol{\omega}}(x_i)$ of the DNN and the expected output y_i. Different loss functions can be considered according to the type of task. Here we will focus on the cross-entropy that will be introduced in the next section.

B. ALL VERSUS ALL TRAINING OF DEEP NEURAL NETWORKS

For image classification, the goal of a DNN is to map the input data to a probabilistic prediction that we denote $P(Y = y^* \mid X = x_i, \boldsymbol{\omega})$ with y^* a class label. During training, an optimization algorithm will improve the weights $\boldsymbol{\omega}$ in order to fit as much as possible the output to the ground truth vector of class labels. The loss is expected to measure the similarity between $f_{\boldsymbol{\omega}}(x_i)$ and y_i. Classically we use cross-entropy defined on a batch B of size $N \in \mathbb{N}$ by:

$$\mathcal{L}(\boldsymbol{\omega}, B) = -\frac{1}{N} \sum_{i=1}^{N} \log(P(Y = y_i \mid X = x_i, \boldsymbol{\omega})) \tag{1}$$

The minimization of this loss function is usually based on gradient methods. Computing the optimal value of each parameter involves a bin-to-bin measure of similarity, which may lead to overfitting issues.

A solution might be to use One Versus All training.

C. FROM ONE VERSUS ALL (OVA) TO OVNNI

The current state of the art on uncertainty estimation is Deep Ensembles [43]. This technique relies on ensembling multiple DNN models trained in parallel in order to optimize the same loss. In contrast to random forests [6], or Bagging [5] the diversity arises from the fact that different embodiments of the same model will converge towards different local optima during training. Conversely, in our approach the diversity is provided by the one-versus-all (OVA) models constructed using different labelings of the training set.

Learning to detect abnormal data means that the DNNs learn to point out: *"I do not know"*. The issue is that the cross-entropy loss, in collaboration with other heuristics for training DNNs, e.g., BatchNorm [31], many layers and residual connections [24], leads to highly overconfident DNNs [22]. Hence we must deal with an overconfident DNN that cannot handle unknown data. Our solution is to use a DNN for learning to classify each class vs all the other classes (OVA training); thanks to this training procedure, each DNN will learn to discriminate every class as well as their boundaries: this permits classifiers that it knows but also that it does not know.

The OVA strategy is conceptually simple, since at its core it involves training a binary classification DNN. One classifier is trained for each class, and prediction is then performed by running the obtained binary classifiers on the testing sample and choosing the prediction with the highest confidence score. Yet, the multiple classifiers involved will learn multiple probabilistic predictions, denoted by $P(Y_j = 1 \mid X = x_i, \boldsymbol{\omega}^j)$ with Y_j a binary random variable for each class j. We add a super script on $\boldsymbol{\omega}^j$, to inform that 1) weights are different from the ones trained to perform the AVA classification that we denote $\boldsymbol{\omega}$, and 2) they are also different from the weights of other classes (different from j).

By training one class versus all the other classes, the DNN learns in some sense the out of distribution classes, however with the significant advantage of not relying on explicitly provided OOD data, in contrast to other strategies [58], [67]. Thanks to this strategy, the DNN learns to better distinguish between objects from known classes and unknown objects from classes not seen during training. Yet, OVA might reduce the performances since the DNN focuses a lot on learning to discriminate every class. Therefore, in addition to the OVA base classifiers, we also perform an All versus all training that we aggregate with the probabilities of the OVA models in the following way, as shown in Figure 2.

Let us denote by Y the discrete random variable, that is taking its value in the list of all classes, and let us denote by Y_j a binary random variable that takes values 0 or 1, with $Y_j = 1$ meaning that the sample belongs to class j. Hence the OVA DNN of the class j provides $P(Y_j = 1 \mid X = x_i, \boldsymbol{\omega}^j)$, while the AVA DNN provides $P(Y = j \mid X = x_i, \boldsymbol{\omega})$ for all j

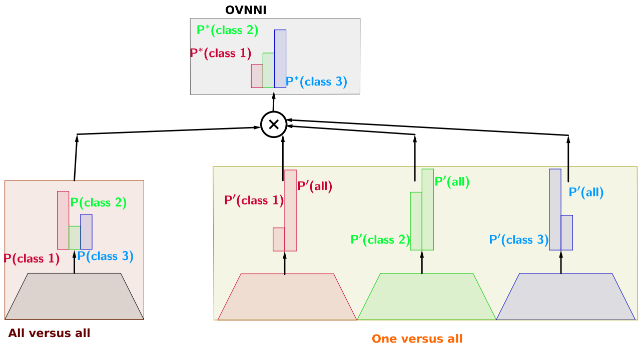


FIGURE 2. From AVA and OVA to OVNNI process in the case we deal with a database composed of just three classes.

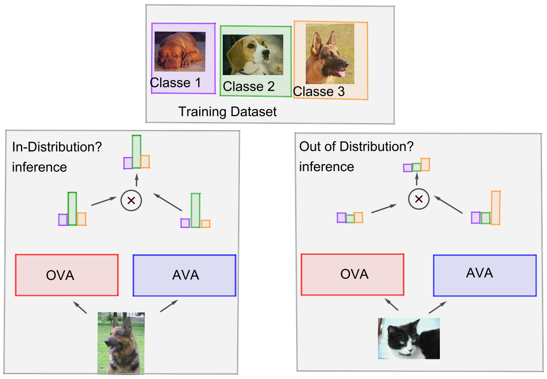


FIGURE 3. OVNNI inference process for the out of distribution case and the in distribution case.

in $\{1 \dots n_{\text{label}}\}$. We consider that the final confidence score for a sample x_i to belong to class j is:

$$p_j(x_i) = P(Y_j = 1 \mid X = x_i, \omega^j) \times P(Y = j \mid X = x_i, \omega)$$

This score is high if AVA and OVA are confident and low otherwise. Multiplying OVA and AVA scores also helps to increase the accuracy since AVA has lower accuracy than OVA (see Figure 3). Hence we propose to use this score as a way to quantify the confidence of the DNN.

D. UNCERTAINTY WITH OVNNI

When we optimize a DNN on a training set, it might suffer from mainly two kinds of uncertainty. The aleatoric uncertainty is linked to the data acquisition and in general it can be learned but not reduced. It can be reduced only if we have more information about this process and we can change it, *e.g.*, add more efficient camera sensors for low-light scenes. The epistemic uncertainty is related to the lack of knowledge of the DNN. Hence, by learning to say *"I do not know"*, our strategy can better model the epistemic uncertainty.

We consider that a measure of confidence must satisfy the following properties: (1) be bounded, (2) exhibit low values for OOD data, (3) have a confidence value that aligned to the accuracy of the algorithm, (4) get more confident if additional training samples are provided. The first point assures that we know what is the maximum and minimum of confidence. The second point is to ensure to detect OOD data, which is crucial since it provides information on the reliability of the DNN on one data. The third point is linked to the calibration [22], which is crucial to rely on the model predictions. The last point concerns the fact that we want to reduce the uncertainty when increasing the dataset.

We use as a measure of confidence for OVNNI the probability $\max_{j \in \{1 \dots n_{\text{label}}\}} \{p_j(x_i)\}$. This measure, bounded by 0 and 1, tells us how much we can rely on the DNN prediction and to which extent it can be used to model the epistemic uncertainty. Indeed in most approaches, the maximum class probability (MCP) [28] is used as a simple baseline to model uncertainty. In our case, we do not properly evaluate the MCP since we do not have a probability. Yet, our confidence score is directly inspired by the MCP. Other approaches make similar assumptions, for instance the evidential model from Sensoy *et al.* [75] where the uncertainty is quantified from belief measures.

IV. EXPERIMENTS

We continue by illustrating the performance of OVNNI for detecting OOD data by conducting five experiments. In the rest of this section we will describe the experimental protocol, followed by the five experiments. We implemented all approaches ourselves and used for all the same learning hyper-parameters per dataset, without particular tuning. Moreover, the number of ensembles is the same for all the techniques, and corresponds to the number of classes.

A. EXPERIMENTAL PROTOCOL

The detection of OOD data can be done either by techniques that measure the uncertainty, or by techniques that detect OOD data. We first have compared our OVNNI to three other uncertainty estimation techniques: MC Dropout [18], Deep Ensembles [43], and TRADI [17]. The major interest of these techniques comes from the fact that, since they estimate uncertainty, they also estimate the epistemic uncertainty and therefore the OOD data. We also have compared our approach to two other techniques: ODIN [49] and ConfidNET [11], that serve as references in unsupervised techniques for detecting OOD data. As a baseline algorithm, we use the maximum class probability (MCP) with AVA trained DNN. We denote this approach as MCP. As an additional baseline we consider one-class Support Vector Machine [59], [64], a classic method for outlier detection. We train it on AVA logits.

Note that we have not compared our OVNNI to techniques trained to *learn* OOD such as [58], [67], since in these cases the OOD data are in the training set, making this technique able to detect just with trained OOD data. To balance OVA training which typically has more samples available for the "All" class, we use weighted cross-entropy to train for each class, with weights for a given class based on $1 - \tau_{\text{class}}$, where τ_{class} is the proportion of data samples of this class in the training set. In addition, for a fair comparison in all experiments we use the same number of models for ensemble and Bayesian methods. We conducted several experiments in

TABLE 1. Comparative results obtained on the calibration task.

Dataset	OOD technique	Accuracy/mIoU	AUC	AUPR Error	AUPR Success	ECE
MNIST/Not MNIST 3 hidden layers	Baseline (MCP)	98.8	92.7	96.1	81.4	0.305
	MCP + One class SVM	98.8	97.4	98.4	95.9	0.072
	MC Dropout	98.2	88.1	89.8	81.7	0.494
	Deep Ensemble	**98.9**	97.7	98.4	95.8	0.462
	TRADI	98.6	97.1	98.4	94.6	0.407
	ODIN	98.8	94.2	96.8	85.6	0.500
	ConfidNET	98.2	97.4	98.8	94.1	0.461
	Ensemble OVA (ours)	97.2	**99.0**	**99.5**	**97.3**	0.179
	OVNNI (ours)	98.8	**99.1**	**99.6**	**97.9**	0.066
CIFAR10 ResNet50	Baseline (MCP)	93.1	83.9	92.9	67.5	0.606
	MCP +One class SVM	93.1	79.7	90.9	63.5	0.203
	MC Dropout	93.1	83.9	92.9	67.5	0.606
	Deep Ensemble	**95.0**	**95.8**	**97.7**	**92.1**	0.422
	ODIN	93.1	83.9	93.3	67.2	0.606
	ConfidNET	93.1	85.1	94.6	61.2	0.706
	Ensemble OVA (ours)	89.3	91.8	95.8	87.1	0.468
	OVNNI (ours)	93.3	94.3	**97.3**	**91.1**	**0.187**
Camvid ENET	Baseline (MCP)	85.8/52.9	79.7	52.1	92.6	0.146
	MC Dropout	80.3/48.6	80.2	56.1	89.3	0.168
	Deep Ensemble	88.0/58.2	83.2	54.3	94.0	0.112
	TRADI	83.4/51.4	83.2	55.9	93.8	0.110
	ConfidNET	83.4/52.8	81.3	58.3	92.6	0.121
	Ensemble OVA (ours)	87.9/52.8	91.7	69.6	98.4	0.060
	OVNNI (ours)	93.1/66.1	**94.0**	**75.7**	**99.0**	0.025
StreetHazards PSPNet (ResNet50)	Baseline (MCP)	90.0/54.6	91.6	50.8	98.9	0.055
	MC Dropout	88.0/47.9	88.8	51.8	97.8	0.092
	Deep Ensemble	90.2/55.0	92.2	52.0	99.0	0.051
	TRADI	90.2 /54.6	92.1	51.4	99.1	0.049
	ConfidNET	90.0/54.6	88.9	37.0	97.9	0.10
	Ensemble OVA (ours)	89.7/54.0	92.4	52.3	**99.1**	**0.048**
	OVNNI (ours)	90.0/54.6	**93.0**	**53.4**	**99.2**	**0.048**
BDD Anomaly PSPNet (ResNet50)	Baseline (MCP)	89.9/52.8	81.4	62.5	91.5	0.159
	MC Dropout	88.7/49.5	76.0	55.7	88.2	0.181
	Deep Ensemble	91.0/57.6	85.5	67.3	93.9	0.170
	TRADI	89.9/52.1	81.9	63.2	91.8	0.157
	ConfidNET	89.9/52.8	78.3	56.4	91.2	0.232
	Ensemble OVA (ours)	89.9/52.8	**91.2**	86.2	95.7	**0.072**
	OVNNI (ours)	90.7/55.4	**91.9**	**86.6**	**95.9**	**0.081**

two target applications: image classification (2 experiments) and semantic pixel segmentation (3 experiments). We considered 7 evaluation measures, in addition to accuracy. Details and results are given below.

1) EVALUATION MEASURES

The evaluation should focus on several points. The first one is the error/success on predicting if the DNN model has some knowledge about specific data. This involves detecting if the data is OOD or not. For that, we use three solutions proposed in [28]. We first only used the confidence score of the OOD data and of the in distribution test data. Based on these confidence scores, and as in [26], [28], we evaluated the AUC, AUPR and the FPR-95%-TPR, that are indicators of the accuracy of detecting OOD data.

However, these measures give no information about the number of good predictions (that should be high) and of bad predictions (that should be low).

This information is crucial since, although it is important to have a low score with the OOD data, the DNN should also reach a high confidence score for well-classified data, and low confidence scores elsewhere. In case the DNN does not reach this point then it might be unusable.

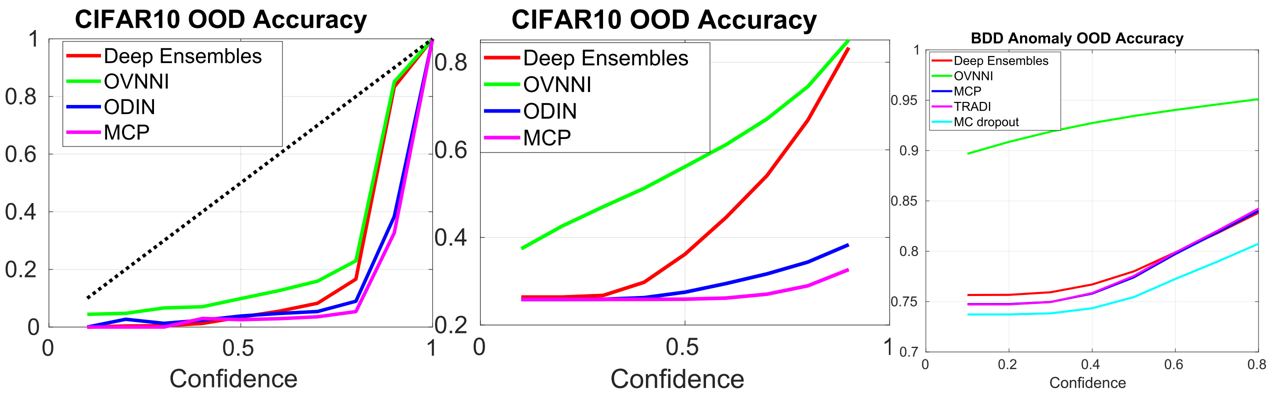
For that, the authors in [11] propose to use metrics similar to the one used by Hendrycks *et al.* [28] but rather than classifying into classes ''OOD'' or ''In distribution'', they classify as ''correctly classified'' or ''not correctly classified'' (this latter class contains both bad predictions and predictions on OOD data, see [11] for more details).

We also used the Expected Calibration Error (ECE) [22], which uses the M-bin histograms of confidence scores and accuracy. The ECE performs a bin-to-bin difference between the two histograms, then an average over the M bins. Similarly to [22] we set $M = 15$. This metric, by measuring the difference between the expected accuracy and confidence, is an indicator of the quality of the confidence, and should be close to 0.

To better understand the behavior of our DNNs when facing strong shifts in the input data distribution, we

TABLE 2. Comparative results obtained on the OOD detection task.

Dataset	OOD technique	AUC	AUPR	FPR-95%-TPR
MNIST/Not MNIST 3 hidden layers	Baseline (MCP)	94.0	96.0	24.6
	MCP + One class SVM	96.9	98.0	12.5
	MC Dropout	91.8	94.9	35.6
	Deep Ensemble	97.2	98.0	9.2
	TRADI	96.7	97.6	11.0
	ODIN	94.9	96.7	17.5
	ConfidNET	97.9	99.0	12.7
	Ensemble OVA (ours)	98.9	**99.4**	5.9
	OVNNI (ours)	**99.3**	**99.6**	**3.5**
CIFAR10 ResNet50	Baseline (MCP)	80.4	89.7	61.5
	MCP + One class SVM	78.8	89.6	61.5
	MC Dropout	80.4	89.7	62.6
	Deep Ensemble	**93.0**	**96.2**	**19.3**
	ODIN	80.3	89.9	61.3
	ConfidNET	84.8	94.0	68.3
	Ensemble OVA (ours)	88.5	93.0	30.9
	OVNNI (ours)	92.2	**95.8**	23.3
Camvid ENET	Baseline (MCP)	75.4	10.0	65.1
	MC Dropout	75.4	10.7	63.2
	Deep Ensemble	79.7	13.0	55.3
	TRADI	79.3	12.8	57.7
	ConfidNET	81.9	13.8	55.8
	Ensemble OVA (ours)	**97.1**	**71.1**	**13.5**
	OVNNI (ours)	**96.1**	**61.2**	**16.5**
StreetHazards PSPNet (ResNet50)	Baseline (MCP)	88.7	6.9	26.9
	MC Dropout	69.9	6.0	32.0
	Deep Ensemble	90.0	7.2	25.4
	TRADI	89.2	7.2	25.3
	ConfidNET	83.6	2.3	26.2
	Ensemble OVA (ours)	**91.6**	**12.7**	**21.9**
	OVNNI (ours)	**91.2**	**12.6**	**22.2**
BDD Anomaly PSPNet (ResNet50)	Baseline (MCP)	86.0	5.4	27.7
	MC Dropout	85.2	5.0	29.3
	Deep Ensemble	87.0	6.0	25.0
	TRADI	86.1	5.6	26.9
	ConfidNET	85.4	5.1	29.1
	Ensemble OVA (ours)	87.0	4.9	29.0
	OVNNI (ours)	**87.2**	**6.7**	**25.0**



(a)

(b)

(c)

FIGURE 4. (a) and (c) Accuracy vs confidence plot on the CIFAR10 \SVHN and BDD Anomaly experiments, respectively. (b) Calibration plot on the CIFAR10 \SVHN.

propose to evaluate the Corrupted Accuracy (cA), and Corrupted Expected Calibration Error (cE) for CIFAR10 [39]. For this scenario, we use CIFAR-10-C [27], where the author generated various noise with different levels of intensity.

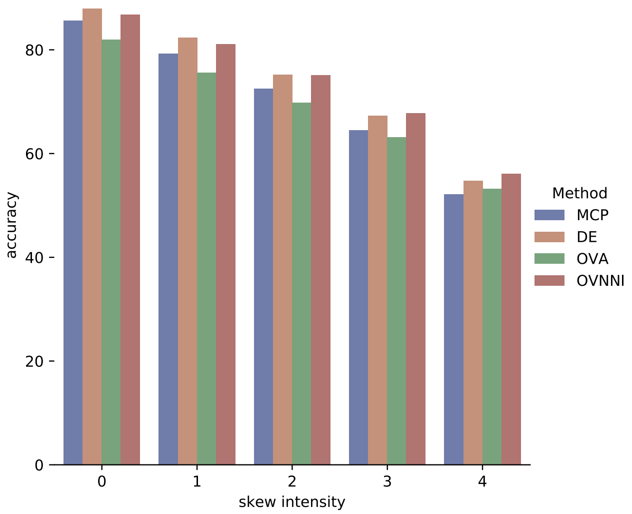


FIGURE 5. Test accuracy on CIFAR-10-C [27].

2) OOD CLASSIFICATION WITH MNIST [46]

Concerning the classification, we used in a first experiment MNIST [46] which is a dataset composed of digit images as training dataset and NotMnist [1] which contains letter images as OOD dataset. We first trained a classifier to learn to recognize the images of digits then tested it on the test set of MNIST and NotMnist hoping that the classifier would distinguish digits form letters. The DNN used for this experiment is fully connected and composed of 3 layers as in [17], [43]. Results are shown in Tables 1 and 2 (MNIST rows).

3) OOD CLASSIFICATION WITH CIFAR10 [39]

We also trained a network on CIFAR10 composed of classes airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships and trucks. We have considered as OOD SVHN dataset [62]. Many methods [56] train on CIFAR10 and test on the test set of CIFAR10 with noise or on STL-10 [10]. It turns out that the first test aims more at measuring random uncertainty and the second one the capability to adapt to the domain. We rather have preferred to consider as an OOD dataset SVHN which is a color image dataset of digits, that guarantees that the OOD data really comes from a distribution different from that of CIFAR10. The DNN we used on this experiment is Resnet50 [24], which has the advantage of being popular in the community. Results are shown in Tables 1 and 2 (CIFAR10 rows). One can see in Table 3 our results under dataset shift. Our approach has state-of-the-art results in terms of accuracy and almost regarding the ECE. Figure 5 shows that OVA is not adapted to aleatoric uncertainty while OVNNI maintains good performances. Hence, we have shown that our approach is resistant to aleatoric uncertainty.

4) OOD SEGMENTATION WITH CAMVID [8]

We used Camvid, a dataset conventionally used in works dealing with segmentation or uncertainty theory and deep learning [11], [17], [33]. This dataset is an ‘‘easy’’ dataset but allows quickly validating results. To test the ability of OVNNI to detect OOD pixels, we trained on all Camvid classes except 3 classes (pedestrian, bicycle, and car), that we deleted, by marking the corresponding pixels as unlabeled. These three classes correspond to OOD classes. Thus this experimental protocol, proposed in [17], makes it possible to validate that the trained DNN will detect the pixels on which it has not been trained as OOD. The DNN for this experiment is Enet [68]. Results are shown in Tables 1 and 2 (Camvid rows).

TABLE 3. Comparative results obtained on CIFAR-10-C [27].

Dataset	OOD technique	cA	cE
CIFAR10 ResNet50	Baseline (MCP)	70.81	0.286
	Deep Ensemble	73.11	**0.113**
	Ensemble OVA (ours)	69.74	91.8
	OVNNI (ours)	**73.35**	0.1267

5) OOD SEGMENTATION WITH StreetHazards [26]

StreetHazards is a large-scale dataset that contains different sets of synthetic images of street scenes. More precisely, this dataset is composed of 5125 images for training and 1500 test images. The training dataset contains 13 classes and the test dataset is composed of the 13 training classes and 250 OOD classes, making it possible to test the robustness of the algorithms with all possible scenarios. For this experiment we used PSPnet [85] with the experimental protocol in [26]. The architecture used for the PSPnet is ResNet50. Results are shown in Tables 1 and 2 (StreetHazards rows).

6) OOD SEGMENTATION WITH BDD ANOMALY [26]

BDD Anomaly dataset is a subset of BDD dataset, composed of 6688 street scenes for the training set and 361 for the testing set. The training set contains 17 classes, and the test dataset is composed of the 17 training classes and 2 OOD classes. For this experiment we used PSPnet [85] with the experimental protocol in [26]. The architecture used for the PSPnet is ResNet50. Results are shown in Tables 1 and 2 (BDD Anomaly rows).

B. VISUALIZING OVA AND AVA EMBEDDING

In this subsection, we perform two experiments to determine the behavior of the representations learned by the DNN with the different techniques. For both experiments we train a simple DNN composed of 3 hidden layers followed by a batch normalization on MNIST dataset [46].

In the first experiment, we have considered as training data only the images with the digits '0','1' and '2' images (the 3 first classes). Then we perform inference on the official test set composed of images with these classes and the OOD images which are composed of other classes. We represent in Figure 8 the softmax of a classical AVA training, a deep ensemble training and the OVNNI training. We can see that in contrast to other techniques, OVNNI results do not necessarily belong to the 2-dimensional simplex. In addition, OVNNI brings the OOD data far away from the simplex vertices which highlights its potential to detect OOD data.

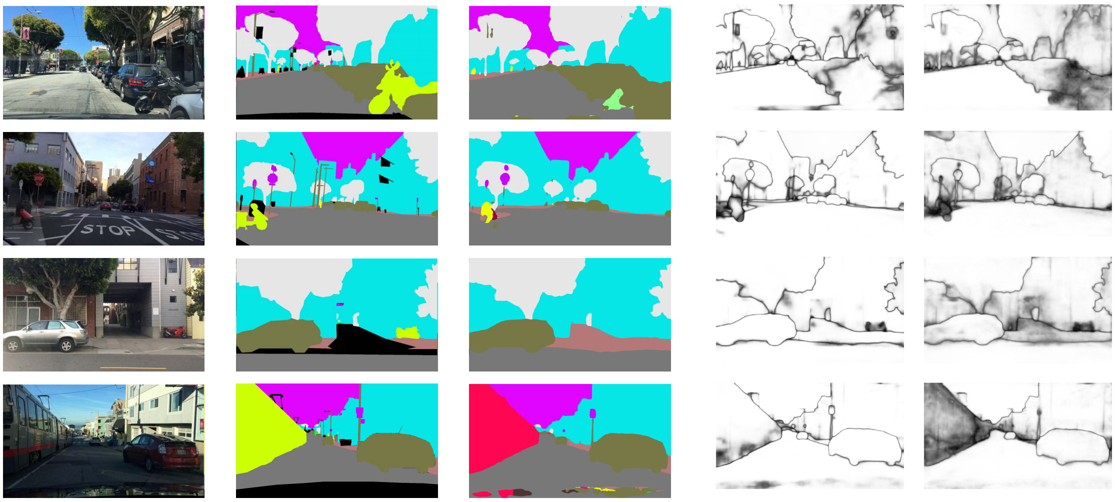


FIGURE 6. Results of OVNNI on BDD Anomaly. The first column is the input image, the second is the ground truth, the third is prediction and the fifth is the confidence score of OVNNI. For comparison, we add the MCP confidence score in the fourth column. We can see that OVNNI has a low score on the motorcycle on the three first rows and on the train on the last row which correspond to the OOD classes.

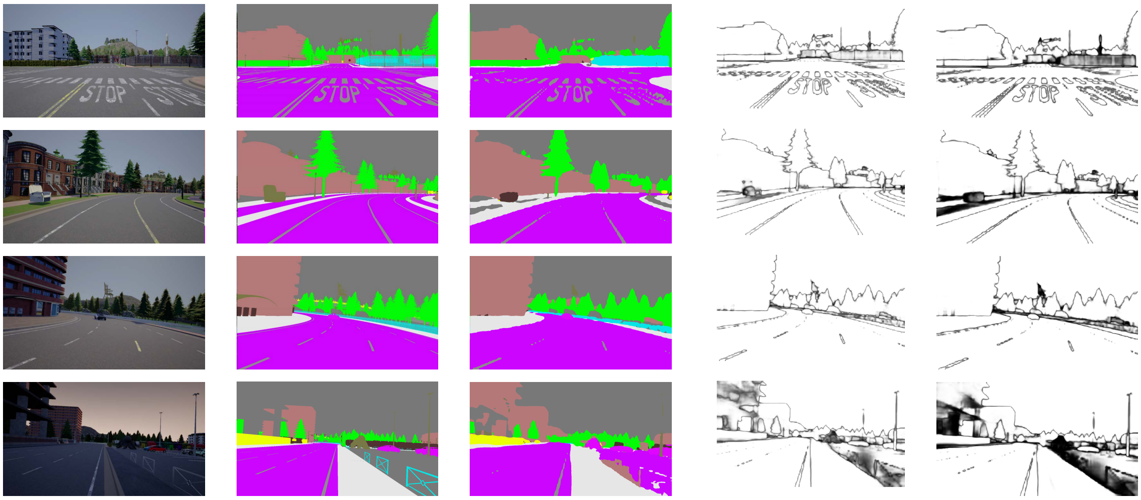


FIGURE 7. Results of OVNNI on StreetHazards. The first column is the input image, the second is the ground truth, the third is prediction and the last is the confidence score of OVNNI. For comparison, we add the MCP confidence score in the fourth column. We can see that OVNNI has a low score on the chair, the seat, the rocket and the spider which correspond to the OOD classes.

In the second experiment, we performed a classical AVA training, and we also performed the OVA training. Hence for the OVA training, we have 10 DNNs (since the dataset has 10 classes which are the 10 digits). The OOD class is composed of images of the NotMNIST dataset [1]. Hence, we apply the DNNs on this test dataset and on the AVA case, we collect for each data the feature space of the DNN just before the classification of each data. In the OVA case, we collect the same feature space but for the DNN of the predicted class.

We reduce the dimension of each of these feature spaces using t-SNE [55] and Principal Component Analysis (PCA) [82], and we plot the results in Figure 9. We can see that in the AVA case the OOD data are in the center of Figure 9 mixed with the other classes, and in the OVA case they are closer to the border whatever the dimensionality reduction algorithm we use. This is crucial because it shows that OVA learns a more interesting descriptor than AVA.

C. HYPER PARAMETERS

In Table 4, we summarize the hyper-parameters used in the Camvid [8], StreetHazards [26] and BDD-Anomaly [26] experiments. In Table 5, we summarize the hyper-parameters used in the MNIST [46] and the CIFAR10 [39] experiments.

D. DISCUSSIONS

On MNIST we can see in Tables 1 and 2 that OVNNI has competitive results for detecting OOD data; more specifically, its calibration score (ECE) is the best. With respect to the metrics proposed by Hendryck *et al.*, OVNNI is the most effective in detecting OOD images, improving the best AUC by 1.4% the best AUPR by 0.6% and the best FP by 62.0%.

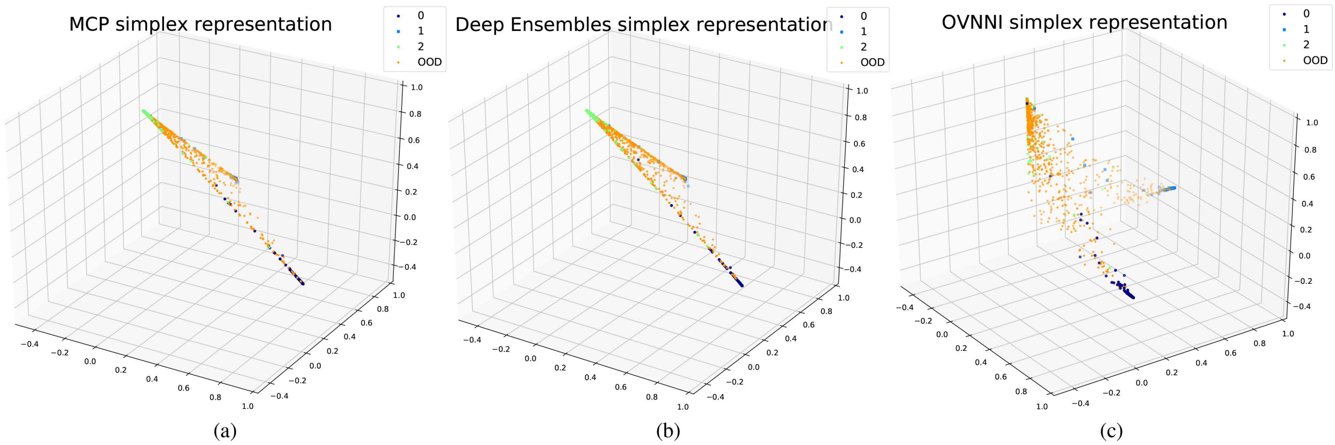
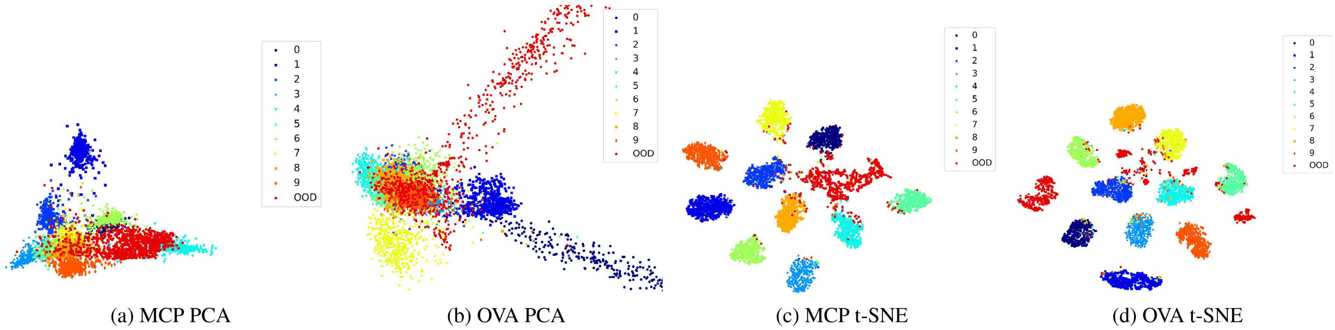


FIGURE 8. Results on MNIST - 3 classes experiments. We represent in these figures the softmax prediction outputs obtained by the baselines (a) MCP, (b) Deep Ensemble, and (c) by OVNNI, respectively.



(a) MCP PCA (b) OVA PCA (c) MCP t-SNE (d) OVA t-SNE

FIGURE 9. Results of the MNIST / NotMNIST experiment. We represent the projection on a 2D space of the feature space of the baseline MCP in (a) and (c), and of OVA in (b) and (d). We use PCA [82] and t-SNE [55] as dimensionality reduction algorithm.

Concerning the metrics proposed by Corbière *et al.*, OVNNI improves the AUC by 1.41%, the AUPR Error by 0.80% the AUPR success by 2.14% and the ECE by 70.6%.

On CIFAR10, although Deep Ensembles achieve good results on all the measurements as well, except on the ECE, note that OVNNI is better calibrated. This can also be seen in the histogram in Figure 4. The difference between OVNNI and Deep Ensembles is low and the crucial requirement of DNN is to have a good calibration. Hence, having a good calibration is more important than having a good AUC or AUPR. Also, we have represented the *accuracy vs confidence* curves in Figure 9. These curves are defined in [43] and are constructed by evaluating the accuracy of all data where the DNN has reached confidence thresholds. These curves show the performance of the OVNNI confidence index over CIFAR10. Finally, we have illustrated the OVNNI calibration on CIFAR10 in the calibration curve in Figure 9. The calibration plot is defined in [22] and is constructed by taking bins of data based on their confidence score. Then on each bin, we evaluate the accuracy, as it should ideally be comparable to the confidence score. These curves show once again the good performance of OVNNI in terms of calibration.

On Camvid we note that OVNNI improves the results of the state of the art by up to 77% with regard to the metrics proposed by Corbière *et al.* [11], and by up to 77% for calibration as well. Concerning the metrics proposed by Hendryck *et al.*, OVNNI improves the measurements by a maximum of 22%.

On StreetHazards we show in Table 2 that OVNNI has better results than the state of the art by improving the best results by up to 42.8%. In Table 1 OVNNI improves the result by a least 2.6% and improves state-of-the-art ECE by 2%. These results show the interest of using OVNNI for semantic segmentation.

Finally, on BDD Anomaly OVNNI improves the calibration by at least 48% which is highly relevant, given the importance of this metric. Furthermore concerning the other metrics, OVNNI improves the results by at least 22%. Furthermore, in Figure 4 we have illustrated the confidence accuracy curve of several algorithms. These curves underline again that OVNNI reaches the best performance in terms of calibration.

Overall, these results show that OVNNI improves the calibration of networks by rendering the confidence in their results more in line with their expected results. Making DNN models more reliable is crucial, especially in areas where the model should not be overconfident. In [22] the authors show that good accuracy of DNNs comes with a price, namely their reliability. In this work, we propose a solution that increases accuracy in most cases, while at the same time improving the calibration and the OOD detection performance.

TABLE 4. Hyper-parameter configuration used in the semantic segmentation experiments.

Hyper-parameter	Camvid	StreetHazards	BDD-Anomaly
Ensemble size	8	13	17
learning rate	5e-4	0.02	0.02
batch size	10	4	4
number of train epochs	100	20	20
weight decay	0.0001	0.0001	0.0001
Pretrained	True	True	True
Optimizer	Adam	SGD	SGD
SyncEnsemble BN	False	False	False

TABLE 5. Hyper-parameter configuration used in the classification experiments.

Hyper-parameter	MNIST/Not MNIST	CIFAR-10
Ensemble size J	10	10
initial learning rate	0.01	0.1
batch size	64	128
lr decay ratio	0.1	0.1
lr decay epochs	10	[60, 120, 160]
number of train epochs	20	200
weight decay	0.0001	0.0001
Optimizer	SGD	SGD
SyncEnsemble BN	False	False

The conceptual simplicity of this solution is a significant asset for its adoption, and the results also convey the message that *one vs all training* can still have an interest for a finer understanding of epistemic uncertainty in DNNs.

Ensemble of OVAs and OVNNI act like Deep Ensembles, i.e. discovering and exploring multiple modes [16]. Just like Deep Ensembles, they benefit from the multiple modes provided by each model leading to better calibrated predictions [16]. However, Deep Ensembles can still become overconfident as they follow modern training heuristics [20]. In OVNNI, weighting OVAs with AVA softmax leads to generally less confident predictions and improves calibration (plots in Figure 4 confirm this quantitatively). This effect is similar to temperature scaling for calibration [20]. However, we do not need an additional validation set to tune this factor. OVA acts as a single class classifier for OOD detection and also learns to perform classification.

E. LIMITATIONS OF THE APPROACH AND PERSPECTIVES

The OVA strategy is conceptually simple and straightforward to adopt and implement in most cases. The main limitation of OVA is related to cases with numerous classes. For popular datasets and tasks involving up to 10-15 classes the computational cost is on par with ensembling approaches, for which this is a typical size for the ensemble [43]. However beyond this number of classes, the approach is less appealing due to the increasing training cost. We note that in several practical settings, *e.g.*, perception for driving assistance [9], [23], [36], [77], the number of classes is often low (less than 10), in order to avoid ambiguity and class imbalance, which are frequent drawbacks of high granularity datasets. For tasks that do not allow for a low number of classes, we indicate a few potential strategies to render OVNNI more feasible for such cases. we can consider meta-classes that group multiple classes from the training set according to visual or semantic relatedness. The taxonomy of a number of datasets is derived from an ontology (usually hierarchical), *e.g.*, Imagenet [13], and we can use this criterion for grouping classes. Alternatively, such meta-classes can be learned [19], [86], as considered in other related approaches, *e.g.*, Error Correcting Output Codes rely on OVA and even on OVO classifiers [42], [76]. In this work, we did not need to adopt such strategies were not necessary as the computational cost is tractable on the considered datasets. However we plan to explore them in future works.

V. CONCLUSION

In this work, we presented an approach based on one versus all training and mixed with a modern approach based on deep learning. We show that the combination of these approaches reaches state of the art performance on all segmentation experiments. Regarding classification tasks, OVNNI exhibits the best calibration performance. Concurrent approaches suffer from a lack of performance in calibration in most datasets, hence the scores that they provide are overconfident, potentially leading to dangerous scenarios in critical applications. In addition to the reported performance, our approach needs little hyperparameter tuning and is easy to implement.

Future work involves first extending this strategy to new tasks such as medical image analysis. One could also use this framework for active learning since active learning algorithms require techniques that can detect OOD data.

ACKNOWLEDGMENT

The authors would like to thank Saclay-IA Cluster and CNRS Jean-Zay Supercomputer for the computation resources. This work was partly done while Isabelle Bloch was with LTCI, Télécom Paris, Institut Polytechnique de Paris.

REFERENCES

[1] *Notmnist Dataset*. Accessed: Oct. 1, 2022. [Online]. Available: http://yaroslavvb.blogspot.com/2011/09/notmnist-dataset.html

[2] C. Baur, B. Wiestler, S. Albarqouni, and N. Navab, "Deep autoencoding models for unsupervised anomaly segmentation in brain MR images," in *Proc. Int. MICCAI Brainlesion Workshop*. New York, NY, USA: Springer, 2018, pp. 161–169.

[3] P. Bevandić, I. Krešo, M. Oršić, and S. Šegvić, "Simultaneous semantic segmentation and outlier detection in presence of domain shift," in *Proc. German Conf. Pattern Recognit.* Springer, 2019, pp. 33–47.

[4] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 1613–1622.

[5] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[6] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[7] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.

[8] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *Proc. Eur. Conf. Comput. Vis.* New York, NY, USA: Springer, 2008, pp. 44–57.

[9] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2011, pp. 215–223.

[10] C. Corbière, N. Thome, A. Bar-Hen, M. Cord, and P. Pérez, "Addressing failure prediction by learning model confidence," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 2898–2909.

[11] C. Creusot and A. Munawar, "Real-time small obstacle detection on highways using compressive RBM road reconstruction," in *Proc. IEEE Intell. Veh. Symp. (IV)*, May 2015, pp. 162–167.

[12] T. DeVries and G. W. Taylor, "Learning confidence for out-of-distribution detection in neural networks," 2018, *arXiv:1802.04865*.

[13] T. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *J. Artif. Intell. Res.*, vol. 2, pp. 263–286, Apr. 1994.

[14] S. Fort, H. Hu, and B. Lakshminarayanan, "Deep ensembles: A loss landscape perspective," 2019, *arXiv:1912.02757*.

[15] G. Franchi, A. Bursuc, E. Aldea, S. Dubuisson, and I. Bloch, "TRADI: Tracking deep neural network weight distributions for uncertainty estimation," 2019, *arXiv:1912.11316*.

[16] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1050–1059.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[18] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1321–1330.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[20] D. Hendrycks, S. Basart, M. Mazeika, M. Mostajabi, J. Steinhardt, and D. Song, "A benchmark for anomaly segmentation," 2019, *arXiv:1911.11132*.

[21] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and Out-of-Distribution examples in neural networks," 2016, *arXiv:1610.02136*.

[22] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," 2018, *arXiv:1812.04606*.

[23] S. Hora, "Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management," *Rel. Eng. Syst. Saf.*, vol. 54, nos. 2–3, pp. 217–223, 1996.

[24] P. Izmailov, A. Wilson, D. Podoprikhin, D. Vetrov, and T. Garipov, "Averaging weights leads to wider optima and better generalization," 2018, *arXiv:1803.05407*.

[25] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," 2015, *arXiv:1511.02680*.

[26] A. Kendall and R. Cipolla, "Modelling uncertainty in deep learning for camera relocalization," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 4762–4769.

[27] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5574–5584.

[28] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Apr. 1998.

[29] A. Krizhevsky *et al.*, "Learning multiple layers of features from tiny images," Citeseer, Tech. Rep., 2009, doi: 10.1.1.222.9220.

[30] A. Krizhevsky and I. G. E. S. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[31] L. I. Kuncheva, "A theoretical study on six classifier fusion strategies," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 281–286, Apr. 2002.

[32] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. Adv. Neural Inf. Process. Systems.*, 2017, pp. 6402–6413.

[33] J. Lambert, Z. Liu, O. Sener, J. Hays, and V. Koltun, "MSeg: A composite dataset for multi-domain semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Oct. 2020, pp. 2879–2888.

[34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[35] Y. LeCun, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Apr. 1998.

[36] K. Lee, H. Lee, and K. J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," 2017, *arXiv:1711.09325*.

[37] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 7167–7177.

[38] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," 2017, *arXiv:1706.02690*.

[39] K. Lis, K. Nakka, P. Fua, and M. Salzmann, "Detecting the unexpected via image resynthesis," in *Proc. Int. Conf. Comput. Vis.*, Apr. 2019, pp. 2152–2161.

[40] F. T. Ting and K. M. Zhou, "Isolation forest," in *Proc. Int. Conf. Data Mining*, 2008, pp. 413–422.

[41] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* New York, NY, USA: Springer, 2016, pp. 21–37.

[42] Y. Zheng, "One-against-all multi-class SVM classification using reliability measures," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, vol. 2, Jun./Aug. 2005, pp. 849–854.

[43] L. V. D. Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[44] W. J. Maddox, P. Izmailov, T. Garipov, and D. P. Vetrov, "A simple baseline for Bayesian uncertainty in deep learning," 2019, *arXiv:1902.02476*.

[45] A. Malinin and M. Gales, "Predictive uncertainty estimation via prior networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 7047–7058.

[46] M. Masana, I. Ruiz, J. Serrat, J. van de Weijer, and A. Lopez, "Metric learning for novelty and anomaly detection," 2018, *arXiv:1808.05492*.

[47] M. Moya and D. Hush, "Network constraints and multi-objective optimization for one-class classification," *Neural Netw.*, vol. 9, no. 3, pp. 463–474, 1996.

[48] J. Mukhoti and Y. Gal, "Evaluating Bayesian deep learning methods for semantic segmentation," 2018, *arXiv:1811.12709*.

[49] R. Neal, *Bayesian Learning for Neural Networks*. Berlin, Germany: Springer-Verlag, 1996.

[50] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 1–9.

[51] P. Oliveri, "Class-modelling in food analytical chemistry: Development, sampling, optimisation and validation issues—A tutorial," *Anal. Chimica Acta*, vol. 982, pp. 9–19, Apr. 2017.

[52] S. Padhy, Z. Nado, J. Ren, J. Liu, J. Snoek, and B. Lakshminarayanan, "Revisiting one-vs-all classifiers for predictive uncertainty and out-of-distribution detection in neural networks," 2020, *arXiv:2007.05134*.

[53] A.-A. Papadopoulos, M. R. Rajati, N. Shaikh, and J. Wang, "Outlier exposure with confidence control for Out-of-Distribution detection," 2019, *arXiv:1906.03509*.

[54] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*.

[55] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[56] R. Rifkin and A. Klautau, "In defense of one-vs-all classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, Apr. 2004.

[57] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.* New York, NY, USA: Springer, 2017, pp. 146–157.

[58] B. Schölkopf, R. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 582–588.

[59] T. Tommasi, N. Patricia, B. Caputo, and T. Tuytelaars, "A deeper look at dataset bias," in *Domain Adaptation in Computer Vision Applications*. New York, NY, USA: Springer, 2017, pp. 37–55.

[60] K. Tumer and J. Ghosh, "Error correlation and error reduction in ensemble classifiers," *Connection Sci.*, vol. 8, nos. 3–4, pp. 385–404, 1996.

[61] A. Vyas, N. Jammalamadaka, X. Zhu, D. Das, B. Kaul, and T. L. Willke, "Out-of-distribution detection using an ensemble of self supervised leave-out classifiers," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 550–564.

[62] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 2, nos. 1–3, pp. 37–52, 1987.

[63] T. F. Wu, C. J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *J. Mach. Learn. Res.*, vol. 5, pp. 975–1005, Aug. 2004.

[64] O. Zendel, K. Honauer, M. Murschitz, D. Steininger, and D. G. Fernandez, "Wilddash-creating hazard-aware benchmarks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 402–416.

[65] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.

[66] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," 2019, *arXiv:1903.12261*.

[67] N. Görnitz, M. Kloft, K. Rieck, and U. Brefeld, "Toward supervised anomaly detection," *J. Artif. Intell. Res.*, vol. 46, pp. 235–262, Apr. 2013.

[68] J. Ma, L. Sun, H. Wang, Y. Zhang, and U. Aickelin, "Supervised anomaly detection in uncertain pseudo periodic data streams," *ACM Trans. Internet Technol.*, vol. 16, no. 1, pp. 1–20, 2016.

[69] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "F-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks," *Med. Image Anal.*, vol. 54, pp. 30–44, Apr. 2019.

[70] G. Pang, C. Shen, L. Cao, and A. Hengel, "Deep learning for anomaly detection: A review," 2020, *arXiv:2007.02500*.

[71] S. Westerlind, "Anomaly detection for portfolio risk management: An evaluation of econometric and machine learning based approaches to detecting anomalous behaviour in portfolio risk measures," M.S. thesis, School Ind. Eng. Manage., KTH, 2018.

[72] G. Rajbahadur, A. Malton, A. Walenstein, and A. Hassan, "A survey of anomaly detection for connected vehicle cybersecurity and safety," in *Proc. IEEE Intell. Veh. Symp.*, Apr. 2018, pp. 421–426.

[73] K. Kosmas and E. Hristoforou, "The effect of magnetic anomaly detection technique in eddy current non-destructive testing," *Int. J. Appl. Electromagn. Mech.*, vol. 25, nos. 1–4, pp. 319–324, May 2007.

[74] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. CVPR*, 2015, pp. 427–436.

[75] M. Hein, M. Andriushchenko, and J. Bitterwolf, "Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem," in *Proc. CVPR*, 2019, pp. 1–8.

[76] J. Deng, W. Dong, R. Socher, L. Li, and K. Li, "ImageNet: A large-scale hierarchical image database," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[77] T. Gao and D. Koller, "Discriminative learning of relaxed hierarchy for large-scale visual recognition," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2072–2079.

[78] J. Zhou, T. Tsang, S. Ho, and K. Müller, "N-ary decomposition for multi-class classification," *Mach. Learn.*, vol. 108, no. 5, pp. 809–830, 2019.

[79] M. Lachaize, H.-M. S. Le, E. Aldea, A. Maitrot, and R. Reynaud, "Evidential framework for error correcting output code classification," in *Engineering Applications of Artificial Intelligence*, vol. 73. Amsterdam, The Netherlands: Elsevier, 2018, pp. 10–21.

[80] Y. Song, Q. Kang, W. Tay, and Y. Tay, "Error-correcting output codes with ensemble diversity for robust learning in neural networks," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 9722–9729.

[81] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 3183–3193.

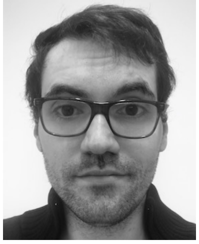
[82] C. Holger, V. Bankiti, A. H. Lang, S. Vora, and V. E. Liong, "NuScenes: A multimodal dataset for autonomous driving," 2019, *arXiv:1903.11027*.

[83] H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, and Y. Zhou, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2446–2454.

[84] R. Kesten, M. Usman, J. Houston, T. Pandya, K. Nadhamuni, A. Ferreira, M. Yuan, B. Low, A. Jain, P. Ondruska, S. Omari, S. Shah, A. Kulkarni, A. Kazakova, C. Tao, L. Platinsky, W. Jiang, Shet, and V. Lyft, "Level 5 perception dataset 2020," *arXiv*, 2020.

[85] E. Haussmann, M. Fenzi, K. Chitta, J. Ivanecky, H. Xu, D. Roy, A. Mittel, and N. Koumchatzky, "Scalable active learning for object detection," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Apr. 2020, pp. 1430–1435.

[86] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 448–456.



GIANNI FRANCHI received the M.Sc. degree from the Ecole Centrale Marseille, in 2013, and the Ph.D. degree in applied mathematics from PSL University (Mines ParisTech), in 2016. He held a postdoctoral position at Siegen University, from October 2016 to December 2017, and at Paris Saclay University. He is currently an Assistant Professor at ENSTA Paris. His research interests include computer vision, machine learning, statistical learning, and video understanding with a particular focus on the theoretical aspects of uncertainty characterization in deep neural networks.

ANDREI BURSUC received the Ph.D. degree from Mines ParisTech, in 2012. He was a Postdoctoral Researcher at Inria Rennes and Inria Paris. In 2016, he moved to industry to pursue research on autonomous systems. He is currently a Research Scientist at Valeo.AI, Paris, France. He is teaching undergraduate courses at the Ecole Normale Supérieure and the Ecole Polytechnique. His current research interests include computer vision and deep learning, in particular learning with limited supervision and predictive uncertainty quantification. He serves regularly as a reviewer for major computer vision and machine learning conferences and journals.

EMANUEL ALDEA (Member, IEEE) received the B.S. degree in computer science from the Ecole Polytechnique, in 2005, the M.S. degree in computer science from the Paris 6 University, in 2006, and the Ph.D. degree in image processing from the Télécom ParisTech, in 2009. He is currently an Associate Professor at Paris-Saclay University. His current research interests include image processing and computer vision for autonomous systems.

SÉVERINE DUBUISSON received the master's and Ph.D. degrees in system control from the University of Technology of Compiègne, in 1997 and 2000, respectively. She has been an Associate Professor with Sorbone Universtity. She is currently an Associate Professor with Aix-Marseille University, France. Her research interests include computer vision, visual tracking, probabilistic models for video sequence analysis, and human interaction.

ISABELLE BLOCH graduated from the Ecole des Mines de Paris, Paris, France, in 1986. She received the master's degree from the University Paris 12, Paris, in 1987, the Ph.D. degree from the Ecole Nationale Supérieure des Télécommunications (Télécom Paris), Paris, in 1990, and the Habilitation degree from the University Paris 5, Paris, in 1995. She has been a Professor at Télécom Paris, until 2020. She is currently a Professor at Sorbonne Université. Her current research interests include 3-D image understanding, computer vision, mathematical morphology, information fusion, fuzzy set theory, structural, graph-based, and knowledge-based object recognition, spatial reasoning, artificial intelligence, and medical imaging.