Neural Network Meta Classifier: Improving the Reliability of Anomaly Segmentation

Jurica Runtas[a] and Tomislav Petković[b]
University of Zagreb, Faculty of Electrical Engineering and Computing, Unska 3, 10000 Zagreb, Croatia

Keywords: Computer Vision, Semantic Segmentation, Anomaly Segmentation, Entropy Maximization, Meta Classification, Open-Set Environments.

Abstract: Deep neural networks (DNNs) are a contemporary solution for semantic segmentation and are usually trained to operate on a predefined closed set of classes. In open-set environments, it is possible to encounter semantically unknown objects or anomalies. Road driving is an example of such an environment in which, from a safety standpoint, it is important to ensure that a DNN indicates it is operating outside of its learned semantic domain. One possible approach to anomaly segmentation is entropy maximization, which is paired with a logistic regression based post-processing step called meta classification, which is in turn used to improve the reliability of detection of anomalous pixels. We propose to substitute the logistic regression meta classifier with a more expressive lightweight fully connected neural network. We analyze advantages and drawbacks of the proposed neural network meta classifier and demonstrate its better performance over logistic regression. We also introduce the concept of informative out-of-distribution examples which we show to improve training results when using entropy maximization in practice. Finally, we discuss the loss of interpretability and show that the behavior of logistic regression and neural network is strongly correlated. The code is publicly available at https://github.com/JuricaRuntas/meta-ood.

1 INTRODUCTION

Semantic segmentation is a computer vision task in which each pixel of an image is assigned into one of predefined classes. An example of a real-world application is an autonomous driving system where semantic segmentation is an important component for visual perception of a driving environment (Biase et al., 2021; Janai et al., 2020).

Deep neural networks (DNNs) are a contemporary solution to the semantic segmentation task. DNNs are usually trained to operate on a predefined closed set of classes. However, this is in a contradiction with the nature of an environment in which aforementioned autonomous driving systems are deployed. Such systems operate in a so-called open-set environment where DNNs will encounter anomalies, i.e., objects that do not belong to any class from the predefined closed set of classes used during training (Wong et al., 2019).

From a safety standpoint, it is very important that a DNN classifies pixels of any encountered anomaly as anomalous and not as one of the predefined classes. The presence of an anomaly indicates that a DNN is operating outside of its learned semantic domain so a corresponding action may be taken, e.g., there is an unknown object on the road and an emergency braking procedure is initiated.

One approach to anomaly segmentation is entropy maximization (Chan et al., 2020). It is usually paired with a logistic regression based post-processing step called meta classification, which is used to improve the reliability of detection of anomalous pixels in the image, driving subsequent anomaly detection.

In this paper, we explore entropy maximization approach to anomaly segmentation where we propose to substitute the logistic regression meta classifier with a lightweight fully connected neural network. Such a network is more expressive than the logistic regression meta classifier, so we expect an improvement in anomaly detection performance. Then, we provide additional analysis of the entropy maximization that shows that caution must be taken when using it in practice in order to ensure its effectiveness. To that end, we introduce the concept of informative out-of-distribution examples which we show to im-

[a] https://orcid.org/0009-0003-0505-2889
[b] https://orcid.org/0000-0002-3054-002X

Runtas, J. and Petković, T.
Neural Network Meta Classifier: Improving the Reliability of Anomaly Segmentation.
DOI: 10.5220/0013143000003912

In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 2: VISAPP*, pages 348-355
ISBN: 978-989-758-728-3; ISSN: 2184-4321

prove training results. Finally, we discuss the loss of interpretability and show that the behavior of logistic regression and neural network is strongly correlated, suggesting that the loss of interpretability may not be a significant drawback after all.

2 RELATED WORK

The task of identifying semantically anomalous regions in an image is called anomaly segmentation or, in the more general context, out-of-distribution (OoD) detection. Regardless of a specific method used for anomaly segmentation, the main objective is to obtain an anomaly segmentation score map. The anomaly segmentation score map $\mathbf{a}$ indicates the possibility of the presence of an anomaly at each pixel location where higher score indicates more probable anomaly (Chan et al., 2022). Methods described in the literature differ in the ways how such a map is obtained.

The methods described in the early works are based on the observation that anomalies usually result in low confidence predictions allowing for their detection. These methods include thresholding the maximum softmax probability (Hendrycks and Gimpel, 2017), ODIN (Liang et al., 2018), uncertainty estimation through the usage of Bayesian methods such as Monte-Carlo dropout (Gal and Ghahramani, 2016; Kendall et al., 2015), ensembles (Lakshminarayanan et al., 2017) and distance based uncertainty estimation through Mahalanobis distance (Denouden et al., 2018; Lee et al., 2018) or Radial Basis Function Networks (RBFNs) (Li and Kosecka, 2021; van Amersfoort et al., 2020). These methods do not rely on the utilization of negative datasets containing images with anomalies so they are classified as anomaly segmentation methods without outlier supervision.

However, methods such as entropy maximization (Chan et al., 2020) use entire images sampled from a negative dataset. Some methods cut and paste anomalies from images in the chosen negative dataset on the in-distribution images (Bevandić et al., 2019; Bevandić et al., 2021; Grcić et al., 2022). The negative images are used to allow the model to learn a representation of the unknown; therefore, such methods belong to the category of anomaly segmentation methods with outlier supervision.

Finally, there are methods that use generative models for the purpose of anomaly segmentation (Biase et al., 2021; Blum et al., 2019; Grcić et al., 2021; Lis et al., 2019; Xia et al., 2020), usually through the means of reconstruction or normalizing flows, with or without outlier supervision. Current state-of-the-art anomaly segmentation methods (Ackermann et al., 2023; Rai et al., 2023; Nayal et al., 2023; Delić et al., 2024) utilize mask-based semantic segmentation (Cheng et al., 2021; Cheng et al., 2022).

3 METHODOLOGY

In this section, we describe a method for anomaly segmentation called entropy maximization. Then, we describe a post-processing step called meta classification, which is used for improving the reliability of anomaly segmentation. Finally, we describe our proposed improvement to the original meta classification approach (Chan et al., 2020). All methods described in the following two subsections are introduced and thoroughly described in (Chan et al., 2020; Chan et al., 2022; Oberdiek et al., 2020; Rottmann et al., 2018; Rottmann and Schubert, 2019).

3.1 Notation

Let $\mathbf{x} \in [0,1]^{H \times W \times 3}$ denote a normalized color image of spatial dimensions $H \times W$. Let $I = \{1,2,...,H\} \times \{1,2,...,W\}$ denote the set of pixel locations. Let $\mathcal{C} = \{1,2,...,C\}$ denote the set of $|\mathcal{C}|$ predefined classes. We define a set of training data used to train a semantic segmentation neural network in a supervised manner as $\mathcal{D}_{in}^{train} = \left\{(\mathbf{x}_j, \mathbf{m}_j)\right\}_{j=1}^{N_{in}^{train}}$, where N_{in}^{train} denotes the total number of in-distribution training samples and $\mathbf{m}_j = (m_i)_{i \in I} \in \mathcal{C}^{H \times W}$ is the corresponding ground truth segmentation mask of $\mathbf{x}_j$. Let $\mathbf{F}: [0,1]^{H \times W \times 3} \to [0,1]^{H \times W \times |\mathcal{C}|}$ be a semantic segmentation neural network that produces pixel-wise class probabilities for a given image $\mathbf{x}$.

3.2 Anomaly Segmentation via Entropy Maximization

Let $\mathbf{p}_i(\mathbf{x}) = \left(p_i(c|\mathbf{x})\right)_{i \in I, c \in \mathcal{C}} \in [0,1]^{|\mathcal{C}|}$ denote a vector of probabilities such that the $p_i(c|\mathbf{x})$ is a probability of a pixel location $i \in I$ of a given image $\mathbf{x} \in \mathcal{D}_{in}$ being a pixel that belongs to the class $c \in \mathcal{C}$. We define $\mathbf{p}(\mathbf{x}) = \left(\mathbf{p}_i(\mathbf{x})\right)_{i \in I} \in [0,1]^{H \times W \times |\mathcal{C}|}$, the probability distribution over images in $\mathcal{D}_{in}$. When using $\mathcal{D}_{in}^{train}$ to train a semantic segmentation neural network $\mathbf{F}$, one can interpret that the network is being trained to estimate $\mathbf{p}(\mathbf{x})$, denoted by $\hat{\mathbf{p}}(\mathbf{x})$. For a semantic segmentation network in the context of anomaly segmentation, it would be a desirable property if such network could output a high prediction uncertainty for OoD pixels which can in turn be quantified with a per-pixel entropy. For a given image $\mathbf{x} \in [0,1]^{H \times W \times 3}$ and

a pixel location $i \in I$, the per-pixel prediction entropy is defined as

$$E_i\big(\hat{\mathbf{p}}_i(\mathbf{x})\big) = -\sum_{c \in C} \hat{p}_i(c|\mathbf{x}) \log\big(\hat{p}_i(c|\mathbf{x})\big), \quad (1)$$

where $E_i\big(\hat{\mathbf{p}}_i(\mathbf{x})\big)$ is maximized by the uniform (non-informative) probability distribution $\hat{\mathbf{p}}_i(\mathbf{x})$ which makes it an intuitive uncertainty measure.

We define a set of OoD training samples as $\mathcal{D}_{out}^{train} = \big\{(\mathbf{x}_j, \mathbf{m}_j)\big\}_{j=1}^{N_{out}^{train}}$ where N_{out}^{train} denotes the total number of such samples. In practice, $\mathcal{D}_{out}^{train}$ is a general-purpose dataset that contains diverse taxonomy exceeding the one found in the chosen domain-specific dataset $\mathcal{D}_{in}^{train}$ and it serves as a proxy for images containing anomalies.

It has been shown (Chan et al., 2020) that one can make the output of a semantic segmentation neural network $\mathbf{F}$ have a high entropy on OoD pixel locations by employing a multi-criteria training objective defined as

$$\begin{aligned} \mathcal{L} = (1-\lambda) \cdot \mathbb{E}_{(\mathbf{x},\mathbf{m}) \in \mathcal{D}_{in}^{train}} \Big[l_{in}\big(\mathbf{F}(\mathbf{x}), \mathbf{m}\big) \Big] + \\ \lambda \cdot \mathbb{E}_{(\mathbf{x},\mathbf{m}) \in \mathcal{D}_{out}^{train}} \Big[l_{out}\big(\mathbf{F}(\mathbf{x}), \mathbf{m}\big) \Big], \end{aligned} \quad (2)$$

where $\lambda \in [0,1]$ is used for controlling the impact of each part of the overall objective.

When minimizing the overall objective defined by Eq. (2), a commonly used cross-entropy loss is applied for in-distribution training samples defined as

$$l_{in}\big(\mathbf{F}(\mathbf{x}), \mathbf{m}\big) = -\sum_{i \in I} \sum_{c \in C} \mathbb{1}_{m_i = c} \cdot \log\big(\hat{p}_i(c|\mathbf{x})\big), \quad (3)$$

where $\mathbb{1}_{c=m_i} \in \{0,1\}$ is the indicator function being equal to one if the class index $c \in C$ is, for a given pixel location $i \in I$, equal to the class index m_i defined by the ground truth segmentation mask $\mathbf{m}$ and zero otherwise. For OoD training samples, a slightly modified cross-entropy loss defined as

$$l_{out}\big(\mathbf{F}(\mathbf{x}), \mathbf{m}\big) = -\sum_{i \in I_{out}} \sum_{c \in C} \frac{1}{|C|} \log\big(\hat{p}_i(c|\mathbf{x})\big) \quad (4)$$

is applied for pixel locations $i \in I$ labeled as OoD in the ground truth segmentation mask $\mathbf{m}$. It can be shown (Chan et al., 2020) that minimizing l_{out} defined by Eq. (4) is equivalent to maximizing per-pixel prediction entropy $E_i\big(\hat{\mathbf{p}}_i(\mathbf{x})\big)$ defined by Eq. (1), hence the name entropy maximization. The anomaly segmentation score map $\mathbf{a}$ can then be obtained by normalizing the per-pixel prediction entropy, i.e.,

$$\mathbf{a} = (a_i)_{i \in I} \in [0,1]^{H \times W}, \; a_i = \frac{E_i\big(\hat{\mathbf{p}}_i(\mathbf{x})\big)}{\log(|C|)}. \quad (5)$$

3.3 Meta Classification

Meta classification is the task of discriminating between a false positive prediction and a true positive prediction. Training a network with a modified entropy maximization training objective increases the network's sensitivity towards predicting OoD objects and can result in a substantial number of false positive predictions (Chan et al., 2019; Chan et al., 2020). Applying meta classification in order to post-process the network's prediction has been shown to significantly improve the network's ability to reliably detect OoD objects. For a given image $\mathbf{x}$, we define a set of pixel locations being predicted as OoD as

$$\hat{I}_{out}(\mathbf{x}, \mathbf{a}) = \big\{ i \in I \mid a_i \geq t, t \in [0,1] \big\} \quad (6)$$

where t represents a fixed threshold and $\mathbf{a}$ is computed using Eq. (5). Based on $\hat{I}_{out}(\mathbf{x}, \mathbf{a})$, a set of connected components representing OoD object predictions defined as $\hat{\mathcal{K}}(\mathbf{x}, \mathbf{a}) \subseteq \mathcal{P}\big(\hat{I}_{out}(\mathbf{x}, \mathbf{a})\big)$ is constructed. Note that $\mathcal{P}\big(\hat{I}_{out}(\mathbf{x}, \mathbf{a})\big)$ denotes the power set of $\hat{I}_{out}(\mathbf{x}, \mathbf{a})$.

Meta classifier is a lightweight model added on top of a semantic segmentation network $\mathbf{F}$. After training $\mathbf{F}$ for entropy maximization on the pixels of known OoD objects, a structured dataset of hand-crafted metrics is constructed. For every OoD object prediction $\hat{k} \in \hat{\mathcal{K}}(\mathbf{x}, \mathbf{a})$, different pixel-wise uncertainty measures are derived solely from $\hat{\mathbf{p}}(\mathbf{x})$ such as normalized per-pixel prediction entropy of Eq. (1), maximum softmax probability, etc. In addition to metrics derived from $\hat{\mathbf{p}}(\mathbf{x})$, metrics based on the OoD object prediction geometry features are also included such as the number of pixels contained in $\hat{k}$, various ratios regarding interior and boundary pixels, geometric center, geometric features regarding the neighborhood of $\hat{k}$, etc. (Chan et al., 2020; Rottmann et al., 2018).

After a dataset with the hand-crafted metrics is constructed, a meta classifier is trained to classify OoD object predictions in one of the following two sets,

$$\begin{aligned} C_{TP}(\mathbf{x}, \mathbf{a}) &= \big\{ \hat{k} \in \hat{\mathcal{K}}(\mathbf{x}, \mathbf{a}) \mid IoU(\hat{k}, \mathbf{m}) > 0 \big\} \text{ and} \\ C_{FP}(\mathbf{x}, \mathbf{a}) &= \big\{ \hat{k} \in \hat{\mathcal{K}}(\mathbf{x}, \mathbf{a}) \mid IoU(\hat{k}, \mathbf{m}) = 0 \big\}, \end{aligned} \quad (7)$$

where C_{TP} represents a set of true positive OoD object predictions, C_{FP} a set of false positive OoD object predictions and IoU represents the intersection over union of a OoD object prediction $\hat{k}$ with the corresponding ground truth segmentation mask $\mathbf{m}$. Each $(\mathbf{x}, \mathbf{m}) \in \mathcal{D}_{out}^{meta}$ is an element of a dataset containing known OoD objects used to train a meta classifier.

During inference, a meta classifier predicts whether an OoD object predictions obtained from $\mathbf{F}$ are false positive. Certainly, the prediction is done

without the access to the ground truth segmentation mask $\mathbf{m}$ and is based on learned statistical and geometrical properties of the OoD object predictions obtained from the known unknowns. OoD object predictions classified as false positive are then removed and the final prediction is obtained.

3.4 Neural Network Meta Classifier

In (Chan et al., 2020), authors use logistic regression for the purpose of meta classifying predicted OoD objects. Their main argument for the use of logistic regression is that since it is a linear model, it is possible to analyze the impact of each hand-crafted metric used as an input to the model with an algorithm such as Least Angle Regression (LARS) (Efron et al., 2004). However, we argue that even though it is desirable to have an interpretable model in order to analyze the relevance and the impact of its input, it is possible to achieve a significantly greater performance by employing a more expressive type of model such as a neural network.

Let $\hat{K}$ be a set containing OoD object predictions for every $(\mathbf{x},\mathbf{m}) \in \mathcal{D}_{out}^{meta}$ defined as $\hat{K} = \bigcup_{(\mathbf{x},\mathbf{m})\in\mathcal{D}_{out}^{meta}} \hat{\mathcal{K}}(\mathbf{x},\mathbf{a})$. We formally define the aforementioned hand-crafted metrics dataset as $\mu \subset \mathbb{R}^{|\hat{K}|\times N_m}$, where N_m is the total number of hand-crafted metrics derived from each OoD object prediction.

We propose that instead of logistic regression as a meta classifier, a lightweight fully connected neural network is employed. Let $\mathbf{F}^{meta} : \mu \rightarrow [0,1]$ denote such a neural network. We can interpret that $\mathbf{F}^{meta}$ outputs the probability of a given OoD object prediction being false positive based on the corresponding derived hand-crafted metrics according to Eq. (7). Let p^F denote such probability. Since $\mathbf{F}^{meta}$ is essentially a binary classifier, we can train it using the binary cross-entropy loss defined as

$$\mathcal{L}^{meta} = -\sum_{i=1}^{N} y_i \log\left(p_i^F\right) + (1-y_i)\log\left(1-p_i^F\right), \quad (8)$$

where N represents the number of OoD object predictions included in a mini-batch and y_i represents the ground truth label of a given OoD object prediction and is equal to one if given OoD object prediction is false positive and zero otherwise.

4 EXPERIMENTS

In this section, we briefly describe the experimental setup and evaluate our proposed neural network meta classifier.

4.1 Experimental Setup

For the purpose of the entropy maximization, we use DeepLabv3+ semantic segmentation model (Chen et al., 2018) with a WideResNet38 backbone (Wu et al., 2016) trained by Nvidia (Zhu et al., 2018). The model is pretrained on Cityscapes dataset (Cordts et al., 2016). The pretrained model is fine-tuned according to Eq. (2). We use Cityscapes dataset (Cordts et al., 2016) as $\mathcal{D}_{in}^{train}$ containing 2,975 images while for $\mathcal{D}_{out}^{train}$ we use a subset of COCO dataset (Lin et al., 2014) which we denote as COCO-OoD. For the purpose of $\mathcal{D}_{out}^{train}$, we exclude images containing class instances that are also found in Cityscapes dataset. After filtering, 46,751 images remain. The model is trained for 4 epochs on random square crops of height and width of 480 pixels. Images that have height or width smaller than 480 pixels are resized. Before each epoch, we randomly shuffle 2,975 images from Cityscapes dataset with 297 images randomly sampled from the remaining 46,751 COCO images. Hyperparameters are set according to the baseline (Chan et al., 2020): loss weight $\lambda = 0.9$, entropy threshold $t = 0.7$. Adam optimizer (Kingma and Ba, 2017) is used with learning rate $\eta = 1 \times 10^{-5}$.

4.2 Evaluation of Neural Network Meta Classifier

We use (Chan et al., 2020) as a baseline. We substitute the logistic regression with a lightweight fully connected neural network whose architecture is shown in Table 1. The proposed meta classifier is trained on the hand-crafted metrics derived from OoD object predictions of the images in LostAndFound Test (Pinggera et al., 2016). Derived hand-crafted metrics, i.e., corresponding OoD object predictions are leave-one-out cross validated according to Eq. (7). The meta classifier is trained using Adam optimizer with learning rate $\eta = 1 \times 10^{-3}$ and weight decay $\gamma = 5 \times 10^{-3}$ for 50 epochs with a mini-batch size $N = 128$. Note that in our case, the total number of hand-crafted metrics $N_m = 75$. Also note that the logistic regression meta classifier has 76 parameters. The results are shown in Table 2 and Fig. 1. In our experiments, the improved performance is especially noticeable when considering OoD object predictions consisting of a very small number of pixels.

5 DISCUSSION

In this section, we introduce the notion of high and low informative OoD proxy images, and we show

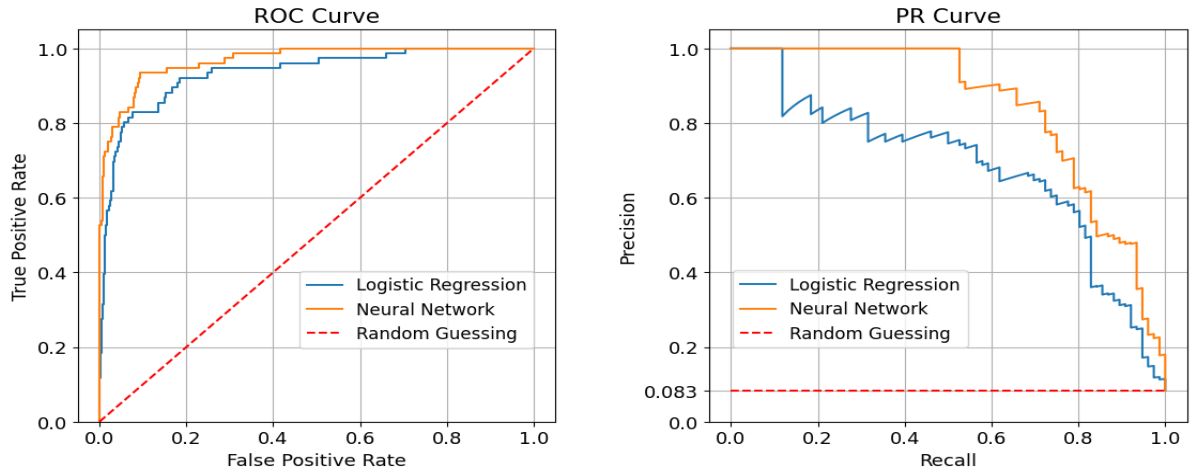

Figure 1: ROC and PR meta classifier curves for OoD object predictions of LostAndFound Test images. On the PR curve, random guessing is represented as a constant dashed red line whose value is equal to the ratio of the number of OoD objects and the total number of predicted OoD objects.

Table 1: Architecture of the neural network meta classifier. All layers are fully connected and a sigmoid activation is applied after the last layer. The total number of parameters is 17,176.

Layer	# of neurons	# of parameters
Input layer	75	5,700
1. layer	75	5,700
2. layer	75	5,700
Output layer	1	76

Table 2: Performance comparison of meta classifiers. Note that the given results are based on OoD object predictions obtained with entropy threshold $t = 0.7$ of Eq. (6).

Model Type	Logistic Regression		Neural Network
Source	Baseline	Reproduced	Ours
AUROC	0.9444	0.9342	**0.9680**
AUPRC	0.7185	0.6819	**0.8418**

that the high informative proxy OoD images are the ones from which the semantic segmentation network can learn to reliably output high entropy on OoD pixels of images seen during inference. Then, we discuss the loss of interpretability, a drawback of using the proposed neural network meta classifier instead of the interpretable logistic regression meta classifier and show that it may not be a significant drawback after all.

5.1 On Outlier Supervision of the Entropy Maximization

We introduce the notion of high informative and low informative proxy OoD images. What we mean by high and low informative is illustrated with Fig. 2. We have noticed empirically that high informative proxy OoD images have two important characteristics that differentiate them from the low informative proxy OoD images: spatially clear separation between objects and clear object boundaries.

Our conjecture is that the low informative proxy OoD images have little to no impact on the entropy maximization training or can even negatively impact the training procedure. On the other hand, high informative proxy OoD images are the ones from which the semantic segmentation network can learn to reliably output high entropy on OoD pixels of images seen during inference, denoted by $\mathcal{D}_{out} \setminus \mathcal{D}_{out}^{train}$, where $\setminus$ represents the set difference.

To investigate our conjecture, we perform the entropy maximization training on subsets of COCO-OoD. We consider it difficult to universally quantify the mentioned characteristics of high informative proxy OoD images, however, we notice a significant correlation between the percentage of the labeled OoD pixels and the desirable properties found in high informative OoD proxy images. We use COCO-OoD proxy for the creation of the two disjoint sets such that the first contains images from COCO-OoD that have at most 20% of pixels labeled as OoD (denoted as L-20%-OoD) and the second that contains images from COCO-OoD that have at least 80% of pixels labeled as OoD (denoted as M-80%-OoD). Table 3 shows that performing the entropy maximization training using M-80%-OoD results in a little to no improvement in comparison to the model trained exclusively on the in-distribution images. On the other hand, using L-20%-OoD produced even better results than the ones obtained with the usage of COCO-OoD.

(a) Examples of high informative proxy OoD images.

(b) Examples of low informative proxy OoD images.

Figure 2: Examples of high and low informative proxy OoD images. The first row contains the proxy OoD images while the second row contains ground truth segmentation masks such that the white regions represent pixels labeled as OoD for which Eq. (4) is applied.

Table 3: Results for the entropy maximization training using COCO-OoD subsets. Column DLV3+W38 contains the results obtained from the model used for fine-tuning (Zhu et al., 2018) which was trained exclusively on the in-distribution images. Other columns contain results obtained from the best model after performing the entropy maximization training numerous times with a given subset.

Metric	FPR_{95}				AUPRC			
Source	DLV3+W38	COCO-OoD	L-20%-OoD	M-80%-OoD	DLV3+W38	COCO-OoD	L-20%-OoD	M-80%-OoD
LostAndFound Test	0.35	0.15	**0.09**	0.13	0.46	0.75	**0.78**	0.48
Fishyscapes Static	0.19	0.17	**0.12**	0.31	0.25	0.64	**0.73**	0.25

5.2 Interpretability of Neural Network Meta Classifier

A drawback of using a neural network as a meta classifier is the loss of interpretability. However, we attempt to further understand the performance of our proposed meta classifier. Fig. 3 shows LARS path for ten hand-crafted metrics most correlated with the response of logistic regression, i.e., the ones which contribute the most in classifying OoD object predictions. One can interpret LARS as a way of sorting the hand-crafted metrics based on the impact on the response of logistic regression. Algorithm 1 offers a way to leverage this kind of reasoning in order to gain a further insight in how neural network meta classifier behaves in comparison to logistic regression. Note that we assume that LARS sorts hand-crafted metrics in descending order with respect to the correlation. Fig. 4 shows results of executing Algorithm 1 for both meta classifiers.

For the logistic regression meta classifier, after we take a subset of μ containing 21 most correlated hand-crafted metrics according to LARS, adding remaining hand-crafted metrics results in little to no improvement in performance. We can see that the neural network meta classifier exhibits a similar behavior, although in a more unstable manner. The obvious difference in performance can be most likely attributed to the fact that neural network meta classifier is more expressive and better aggregates the hand-crafted metrics. We argue that the hand-crafted metrics having the most impact on the performance of logistic regression meta classifier also do so in the case of neural network meta classifier. Such insight could alleviate presumably the most significant drawback of using neural network meta classifier instead of logistic regression meta classifier - the loss of interpretability.

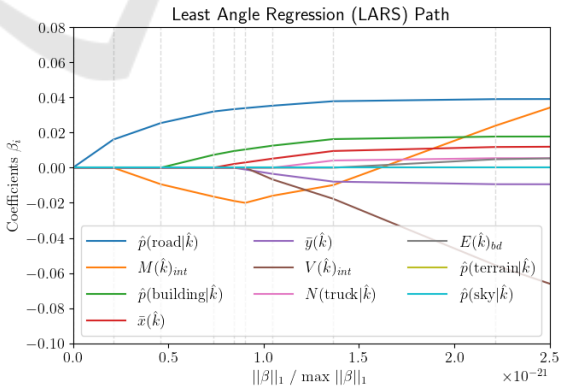

Figure 3: LARS path for the hand-crafted metrics at $t = 0.7$. A detailed description of the hand-crafted metrics can be found in (Chan et al., 2020).

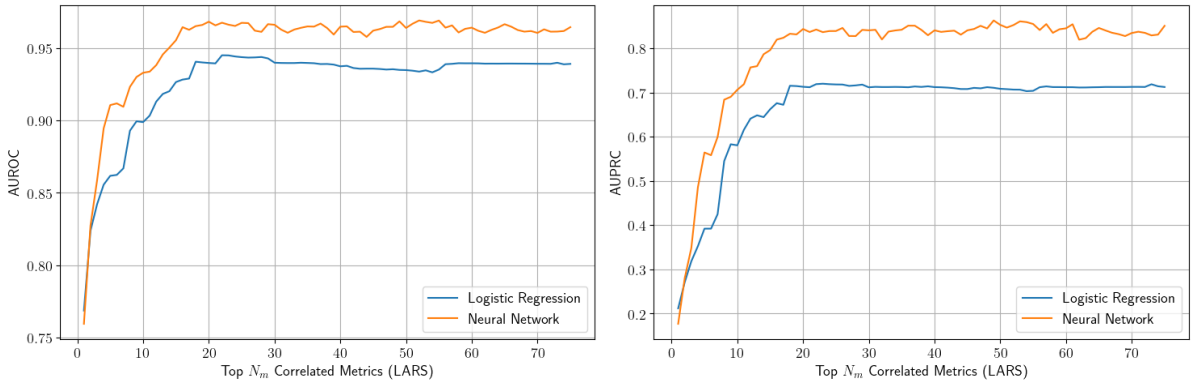

Figure 4: Performance comparison of logistic regression meta classifier and neural network meta classifier when trained on subsets of the hand-crafted metrics dataset μ. For each value N_m on the x-axis, we train the meta classifiers on the subset of μ such that we take the first N_m metrics having the most correlation with the response according to LARS.

Algorithm 1: Incremental meta classifier evaluation.

Input : $\mathbf{F}^{meta}$, μ, N_m
Output: lists of AUROC and AUPRC metrics
AUROC $\longleftarrow$ [];
AUPRC $\longleftarrow$ [];
MetricsSortedByCorrelation $\longleftarrow$ LARS(μ);
for $i = 1$ **to** N_m **do**
  $\xi \longleftarrow$ MetricsSortedByCorrelation[:i];
  initializeModel($\mathbf{F}^{meta}$);
  trainModel($\mathbf{F}^{meta}$, ξ);
  $(m_1, m_2) \longleftarrow$ evaluateModel($\mathbf{F}^{meta}$, ξ);
  AUROC.append(m_1);
  AUPRC.append(m_2);
end

6 CONCLUSION

In this paper, we explored the anomaly segmentation method called entropy maximization which can increase the network's sensitivity towards predicting OoD objects, but which can also result in a substantial number of false positive predictions. Hence, the meta classification post-processing step is applied in order to improve the network's ability to reliably detect OoD objects. Our experimental results showed that employing the proposed neural network meta classifier results in a significantly greater performance in comparison to the logistic regression meta classifier.

Furthermore, we provided additional analysis of the entropy maximization training which showed that in order to ensure its effectiveness, caution must be taken when choosing which images are going to be used as proxy OoD images. Our experimental results demonstrated that high informative proxy OoD images are the ones from which the semantic segmentation network can learn to reliably output high entropy on OoD pixels of images seen during inference and are therefore more beneficial to the entropy maximization training in terms of how well a semantic segmentation neural network can detect OoD objects afterwards.

Finally, a drawback of using the neural network meta classifier is the loss of interpretability. In our attempt to further analyze the performance of the proposed neural network meta classifier, we found that the behavior of logistic regression and neural network is strongly correlated, suggesting that the loss of interpretability may not be a significant drawback after all.

REFERENCES

Ackermann, J., Sakaridis, C., and Yu, F. (2023). Maskomaly:zero-shot mask anomaly segmentation.

Bevandić, P., Krešo, I., Oršić, M., and Šegvić, S. (2019). Simultaneous semantic segmentation and outlier detection in presence of domain shift. *CoRR*, abs/1908.01098.

Bevandić, P., Krešo, I., Oršić, M., and Šegvić, S. (2021). Dense outlier detection and open-set recognition based on training with noisy negative images. *CoRR*, abs/2101.09193.

Biase, G. D., Blum, H., Siegwart, R., and Cadena, C. (2021). Pixel-wise anomaly detection in complex driving scenes. *CoRR*, abs/2103.05445.

Blum, H., Sarlin, P., Nieto, J. I., Siegwart, R., and Cadena, C. (2019). The fishyscapes benchmark: Measuring blind spots in semantic segmentation. *CoRR*, abs/1904.03215.

Chan, R., Rottmann, M., and Gottschalk, H. (2020). Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation. *CoRR*, abs/2012.06575.

Chan, R., Rottmann, M., Hüger, F., Schlicht, P., and Gottschalk, H. (2019). Metafusion: Controlled false-

negative reduction of minority classes in semantic segmentation.

Chan, R., Uhlemeyer, S., Rottmann, M., and Gottschalk, H. (2022). Detecting and learning the unknown in semantic segmentation.

Chen, L., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *CoRR*, abs/1802.02611.

Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. (2022). Masked-attention mask transformer for universal image segmentation.

Cheng, B., Schwing, A. G., and Kirillov, A. (2021). Per-pixel classification is not all you need for semantic segmentation.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. *CoRR*, abs/1604.01685.

Delić, A., Grcić, M., and Šegvić, S. (2024). Outlier detection by ensembling uncertainty with negative objectness.

Denouden, T., Salay, R., Czarnecki, K., Abdelzad, V., Phan, B., and Vernekar, S. (2018). Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance. *CoRR*, abs/1812.02765.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, 32(2).

Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning.

Grcić, M., Bevandić, P., and Šegvić, S. (2021). Dense anomaly detection by robust learning on synthetic negative data. *ArXiv*, abs/2112.12833.

Grcić, M., Bevandić, P., and Šegvić, S. (2022). Dense-hybrid: Hybrid anomaly detection for dense open-set recognition.

Hendrycks, D. and Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Janai, J., Güney, F., Behl, A., and Geiger, A. (2020). Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Found. Trends. Comput. Graph. Vis.*, 12(1–3):1–308.

Kendall, A., Badrinarayanan, V., and Cipolla, R. (2015). Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *CoRR*, abs/1511.02680.

Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles.

Lee, K., Lee, K., Lee, H., and Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks.

Li, Y. and Kosecka, J. (2021). Uncertainty aware proposal segmentation for unknown object detection. *CoRR*, abs/2111.12866.

Liang, S., Li, Y., and Srikant, R. (2018). Enhancing the reliability of out-of-distribution image detection in neural networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

Lis, K., Nakka, K. K., Fua, P., and Salzmann, M. (2019). Detecting the unexpected via image resynthesis. *CoRR*, abs/1904.07595.

Nayal, N., Yavuz, M., Henriques, J. F., and Güney, F. (2023). Rba: Segmenting unknown regions rejected by all.

Oberdiek, P., Rottmann, M., and Fink, G. A. (2020). Detection and retrieval of out-of-distribution objects in semantic segmentation. *CoRR*, abs/2005.06831.

Pinggera, P., Ramos, S., Gehrig, S., Franke, U., Rother, C., and Mester, R. (2016). Lost and found: Detecting small road hazards for self-driving vehicles. *CoRR*, abs/1609.04653.

Rai, S. N., Cermelli, F., Fontanel, D., Masone, C., and Caputo, B. (2023). Unmasking anomalies in road-scene segmentation.

Rottmann, M., Colling, P., Hack, T., Hüger, F., Schlicht, P., and Gottschalk, H. (2018). Prediction error meta classification in semantic segmentation: Detection via aggregated dispersion measures of softmax probabilities. *CoRR*, abs/1811.00648.

Rottmann, M. and Schubert, M. (2019). Uncertainty measures and prediction quality rating for the semantic segmentation of nested multi resolution street scene images. *CoRR*, abs/1904.04516.

van Amersfoort, J., Smith, L., Teh, Y. W., and Gal, Y. (2020). Simple and scalable epistemic uncertainty estimation using a single deep deterministic neural network. *CoRR*, abs/2003.02037.

Wong, K., Wang, S., Ren, M., Liang, M., and Urtasun, R. (2019). Identifying unknown instances for autonomous driving. *CoRR*, abs/1910.11296.

Wu, Z., Shen, C., and van den Hengel, A. (2016). Wider or deeper: Revisiting the resnet model for visual recognition. *CoRR*, abs/1611.10080.

Xia, Y., Zhang, Y., Liu, F., Shen, W., and Yuille, A. L. (2020). Synthesize then compare: Detecting failures and anomalies for semantic segmentation. *CoRR*, abs/2003.08440.

Zhu, Y., Sapra, K., Reda, F. A., Shih, K. J., Newsam, S. D., Tao, A., and Catanzaro, B. (2018). Improving semantic segmentation via video propagation and label relaxation. *CoRR*, abs/1812.01593.